

# **Инструкция по развертыванию Системы и Сервисов взаимодействия**

# Содержание

<b>1</b>	<b>Mongo db(ReplicaSet)</b> .....	<b>7</b>
1.1	Описание .....	7
1.2	Зависимости проекта.....	7
1.3	Развертывание с использованием docker-образа .....	7
1.3.1	Необходимое ПО.....	7
1.3.2	Структура папок монтируемых на host-машину .....	7
<b>2</b>	<b>Node портал прокси</b> .....	<b>10</b>
2.1	Описание сервиса:.....	10
2.2	Репозиторий в Git: .....	10
2.3	Примечание:.....	10
2.4	Развертывание с использованием docker-образа .....	10
2.4.1	Необходимое ПО.....	10
2.4.2	Структура папок монтируемых на host-машину .....	10
2.4.3	Для развертывания необходимо.....	11
<b>3</b>	<b>PostgreSQL на CentOS 8</b> .....	<b>13</b>
3.1	PostgreSQL 11 .....	13
3.2	PostgreSQL 12 .....	24
<b>4</b>	<b>Балансировщик nginx</b> .....	<b>33</b>
4.1	Структура папок .....	33
4.2	Описание файлов конфигураций .....	35
4.3	Настройка ssl для сайта .....	39
4.3.1	Настройка RSA сертификатов.....	39
4.3.2	Настройка GOST сертификатов.....	39
<b>5</b>	<b>Важные настройки ЕЦП.МИС</b> .....	<b>41</b>
<b>6</b>	<b>Настройка видеосвязи</b> .....	<b>64</b>
6.1	Разворачивание сервиса на nodejs. docker-compose.yml .....	64
6.2	Настройки ЕЦП.МИС .....	64
6.2.1	promed.php .....	64
6.3	Настройки портала .....	65
6.3.1	site.php -работа web портала .....	65
6.3.2	regions.php - работа через мобильное приложение .....	65
6.4	Настройка балансировщика .....	65
6.4.1	nodejs.conf.....	66
6.4.2	nginx.conf.....	66
6.5	Примечания.....	66
<b>7</b>	<b>Настройка региональных параметров при разворачивании "коробочной" версии ЕЦП.МИС</b> .....	<b>67</b>
<b>8</b>	<b>Настройки БД после развертывания.</b> .....	<b>68</b>
<b>9</b>	<b>Памятка - установка "коробочной" версии в новые регионы ЕЦП.МИС</b> .....	<b>74</b>
<b>10</b>	<b>Получение токена авторизации.</b> .....	<b>77</b>
<b>11</b>	<b>Порядок развертывания БД и действия до загрузки дампа</b> .....	<b>79</b>
<b>12</b>	<b>Региональный портал мед. услуг</b> .....	<b>80</b>
12.1	Зависимости проекта .....	80
12.2	Развертывание с использованием docker-образа .....	80
12.2.1	Необходимое ПО.....	80
12.2.2	Структура папок монтируемых на host-машину .....	80
12.2.3	Для развертывания необходимо.....	80
12.3	Общие файлы конфигурации.....	85
12.4	Региональные файлы конфигурации .....	91
<b>13</b>	<b>Сервис LDAP</b> .....	<b>108</b>
<b>14</b>	<b>Сервис ЕРМП</b> .....	<b>110</b>

14.1	Программное обеспечение: .....	110
14.2	Настройка авторизации в ЕРМП.....	112
<b>15</b>	<b>Сервис формирования отчетов birt-viewer .....</b>	<b>114</b>
<b>16</b>	<b>Установка веб сервера на linux .....</b>	<b>122</b>
<b>17</b>	<b>Установка сервера хранения справочников, логов, кэша, сессий .....</b>	<b>128</b>
<b>18</b>	<b>Фоновые процессы системы .....</b>	<b>130</b>
18.1	Фоновые процессы базы данных.....	130
18.1.1	Создание реестра счетов .....	130
18.1.2	Чистка лога действия пользователя.....	130
18.1.3	Формирование ЛВН.....	131
18.1.4	Vacum.....	131
18.1.5	Чистка логов PostgreSQL.....	131
18.1.6	Обновление мат представлений.....	131
18.1.7	Backup .....	131
18.1.8	Auto kill .....	131
18.1.9	Collect data iops for zabbix .....	131
18.2	Фоновые процессы WEB сервера .....	131

## **ЧАСТЬ 2 – Сервисы для взаимодействия с внешними федеральными системами**133

<b>1</b>	<b>swan-api.....</b>	<b>134</b>
1.1	Разворачивание .....	134
1.1.1	Установка с применением docker.....	134
1.1.1.1	Необходимое ПО.....	134
1.1.1.2	Структура папок.....	134
1.1.1.3	docker-compose.yml.....	134
1.1.2	Установка напрямую на ОС.....	134
1.1.2.1	Необходимое ПО.....	134
1.1.2.2	Установка TomEE.....	134
1.1.2.3	Настройка TomEE как сервиса .....	134
1.2	Описание файлов конфигурации. ....	134
1.3	Обновление .....	137
1.4	Действия после установки .....	137
<b>2</b>	<b>ЕСИА (ScanCodeService) .....</b>	<b>138</b>
<b>3</b>	<b>ЕСИА (ScanCodeService) Docker .....</b>	<b>152</b>
3.1	Описание сервиса:.....	152
3.2	Репозиторий в Git: .....	152
3.3	Примечание:.....	152
3.4	Развертывание с использованием docker-образа.....	152
3.4.1	Необходимое ПО.....	152
3.4.2	Структура папок монтируемых на host-машину .....	152
3.4.3	Для развертывания необходимо.....	152
<b>4</b>	<b>ЕСИА (ScanCodeService) OLD .....</b>	<b>157</b>
<b>5</b>	<b>ИЕМК .....</b>	<b>164</b>
5.1	Разворачивание .....	164
5.1.1	Установка с применением docker.....	164
5.1.1.1	Необходимое ПО.....	164
5.1.1.2	Структура папок и файлов.....	164
5.1.1.3	docker-compose.yml.....	165
5.1.2	Установка напрямую на ОС.....	167
5.1.2.1	Необходимое ПО.....	167
5.1.2.2	Установка TomEE .....	167
5.1.2.3	Настройка TomEE как сервиса .....	167
5.2	Описание конфигурации. ....	167
5.3	Описание БД log_service для использования сервиса ИЕМК.....	170
5.3.1	iemc.integrationservicesystemoption .....	170
5.3.2	iemc.integrationserviceeventype .....	172
5.3.3	iemc.integrationserviceeventlog .....	173

<b>6</b>	<b>Разворачивание ФЭР .....</b>	<b>174</b>
6.1	Установка с применением docker .....	174
6.1.1	Необходимое ПО.....	174
6.1.2	Структура папок и файлов.....	174
6.1.3	docker-compose.yml .....	175
6.2	Описание файлов конфигурации .....	176
6.2.1	job.xml.....	181
6.3	Действия после установки .....	190
<b>7</b>	<b>КУ ФЭР Профосмотры (Dispensary).....</b>	<b>191</b>
7.1	Разворачивание .....	191
7.1.1	Установка с применением docker.....	191
7.1.1.1	Необходимое ПО.....	191
7.1.1.2	Структура папок.....	191
7.1.1.3	docker-compose.yml .....	191
7.1.2	Установка напрямую на ОС.....	191
7.1.2.1	Необходимое ПО.....	191
7.1.2.2	Установка TomEE.....	192
7.1.2.3	Настройка TomEE как сервиса .....	192
7.2	Описание файлов конфигурации .....	192
7.3	Действия после установки .....	195
<b>8</b>	<b>РЭМД (Электронные медицинские документы) .....</b>	<b>197</b>
8.1	Разворачивание .....	197
8.1.1	Установка с применением docker.....	197
8.1.1.1	Необходимое ПО.....	197
8.1.1.2	Структура папок.....	197
8.1.1.3	docker-compose.yml .....	197
8.1.2	Установка напрямую на ОС.....	197
8.1.2.1	Необходимое ПО.....	197
8.1.2.2	Установка TomEE .....	197
8.1.2.3	Настройка TomEE как сервиса .....	197
8.2	Описание конфигурации. ....	197
8.2.1	Описание настроек сервиса .....	197
8.2.2	Написание настроек ЕЦП.МИС .....	200
8.2.3	Настройка балансировщика nginx.....	201
8.3	Действия после установки .....	201
<b>9</b>	<b>ТФОМС.....</b>	<b>202</b>
9.1	Настройка RabbitMQ.....	202
9.1.1	Устанавливаем erel и erlang .....	202
9.1.2	Далее устанавливаем сам RabbitMQ.....	202
9.1.3	Меняем ограничение на количество открытых файлов .....	202
9.2	Настройка ЭЦП .....	204
9.3	Настройка заданий ЭЦП .....	205
<b>10</b>	<b>ФРБТ / ФРВИЧ / ФР14Н .....</b>	<b>208</b>
10.1	Принцип работы сервиса .....	208
10.2	Развертывание .....	208
10.2.1	Необходимое ПО.....	208
10.2.2	Установка .....	208
10.2.3	Настройка .....	209
10.3	Описание конфигурации.....	209
10.4	Описание БД.....	209
<b>11</b>	<b>ФРМО .....</b>	<b>210</b>
11.1	Разворачивание .....	210
11.1.1	Установка напрямую на ОС.....	210
11.1.1.1	Установка Крипто Про JCP .....	210
11.1.1.2	Установка TomEE 1.7.4 .....	211
11.1.1.3	Настройка ActiveMQ .....	212
11.1.2	Установка с помощью docker .....	213
11.1.2.1	docker-compose.yml .....	213
11.1.3	Установка и настройка сервиса ФРМО.....	213

11.1.4	Описание файлов конфигурации .....	215
11.1.5	Проверка работоспособности .....	220
<b>12</b>	<b>ФРМР.....</b>	<b>222</b>
12.1	Разворачивание .....	222
12.1.1	Установка с применением docker.....	222
12.1.1.1	Необходимое ПО.....	222
12.1.1.2	Структура папок и файлов.....	222
12.1.1.3	docker-compose.yml.....	222
12.1.2	Установка напрямую на ОС.....	223
12.1.2.1	Необходимое ПО.....	223
12.1.2.2	Установка TomEE.....	223
12.1.2.3	Настройка TomEE как сервиса .....	223
12.2	Описание файлов конфигурации.....	223
<b>13</b>	<b>Инструкция по внедрению сервиса ФРМР / ФРМО в МИС (РМИС) .....</b>	<b>227</b>
1	Общая информация .....	229
2	Способы тестирования .....	229
2.1	Регулярный запуск .....	230
2.2	Режим первого запуска.....	230
2.3	Режим массовой выгрузки .....	230
2.4	Дополнительная информация и ссылки .....	231
3	Postgre ФРМО. Отчет о тестировании и массовая выгрузка.....	231
3.1	Общие сведения: .....	231
3.1.1	Для запуска единичной сессии ФРМО можно использовать скрипт FRMOSession_run.sql:.....	231
3.1.2	Ошибки по сессии можно посмотреть по скрипту FRMOSessionError.sql: .....	232
3.2	Запуск массовой выгрузки: .....	233
3.3	Запуск массовой выгрузки выполняется скриптом MassiveTestCycle0.sql:.....	233
4	Postgre ФРМР. Отчет о тестировании и массовая выгрузка. Подробная инструкция. 240	
4.1	Общие сведения .....	241
4.2	Подготовка.....	241
4.3	Проверка ФРМР.....	241
4.3.1	Создание врача.....	241
4.4	Составление отчета о тестировании.....	247
4.5	Массовая выгрузка.....	248
4.5.1	Проведение Массовой выгрузки.....	248
4.5.2	Составление отчета .....	248
4.6	SQL Scripts.....	249
4.6.1	Проверка и подготовка Сессии.....	249
4.6.2	Запуск методов ФРМР: .....	249
4.6.3	Работа с данными.....	251
4.6.4	Проверка данных Врача .....	253
5	ФРМО/ФРМР. Инструкция по внедрению. ....	261
5.1	Порядок подключения ИС к Системе.....	261
5.2	Подключение к тестовой версии сервисов ФРМР/ФРМО .....	261
5.3	Отчет о тестировании: .....	262
5.4	Массовая выгрузка: .....	263
5.5	Подключение к промышленной версии: .....	263
6	ФРМО. Отчет о тестировании и массовая выгрузка.....	265
6.1	Общие сведения: .....	265
6.1.1	Для запуска единичной сессии ФРМО можно использовать скрипт FRMOSession_run.sql (альтернатива способу запуска через Промед):.....	265
6.1.2	Описание скрипта:.....	266
6.1.3	Ошибки по сессии можно посмотреть по скрипту FRMOSessionError.sql (также доступны в Промеде): .....	266
6.1.4	Описание скрипта:.....	267
6.2	Запуск массовой выгрузки: .....	267
6.2.1	Для запуска массовой выгрузки необходимо создание контрольного примера структуры МО: одна МО и по две единицы структурных элементов каждого типа. Описание данных с указанием форм Системы изложены в инструкции:.....	267
6.2.2	Описание данных для массовой выгрузки ФРМО.....	267

6.2.3	Запуск массовой выгрузки выполняется скриптом MassiveTestCycle.sql. Этот скрипт также может использоваться для получения тестовых отчетов. Перед запуском скрипта должна быть выполнена правка ошибок форматно-логического контроля, полученных на стадии верификации сессии экспорта в ФРМО. Во время выполнения скрипта должно быть гарантировано, что данные используемых объектов структуры МО не будут изменяться каким-либо внешним процессом.....	267
6.2.4	Версия для MSSQL: .....	267
6.2.5	Версия для PostgreSQL: .....	271
6.2.6	Скрипт get_MO.sql для Медицинской организации, экспортируемой в ФРМО: .....	280
6.2.7	Скрипт get_Buildings.sql для списка зданий, экспортируемых в ФРМО: .....	283
6.2.8	Скрипт get_Departments.sql для списка подразделений и отделений, экспортируемых в ФРМО: .....	284
6.2.9	Скрипт get_Staff.sql для списка штатных расписаний, экспортируемых в ФРМО: .....	288
7	ФРМР. Отчет о тестировании и массовая выгрузка. Подробная инструкция.....	289
7.1	Общие сведения .....	290
7.2	Подготовка .....	290
7.3	Проверка ФРМР. ....	290
7.3.1	Создание врача .....	290
7.4	Составление отчета о тестировании .....	296
7.4.1	Порядок выполнения методов .....	297
7.5	Массовая выгрузка .....	297
7.5.1	Проведение Массовой выгрузки. ....	297
7.5.2	Составление отчета .....	297
7.6	SQL Scripts.....	298
7.6.1	Проверка и подготовка Сессии.....	298
7.6.2	Запуск методов ФРМР: .....	298
7.6.3	Работа с данными .....	299
7.6.4	Проверка данных Врача .....	301
<b>14</b>	<b>ФСС (ЛВН) .....</b>	<b>316</b>
14.1	Настройка сервиса .....	316
14.2	Настройка ЕЦП.МИС.....	316
14.3	Настройка cron .....	317
14.3.1	Получение токена авторизации .....	318
14.4	Настройка cron на БД.....	319
<b>15</b>	<b>Голосовой помощник.....</b>	<b>320</b>
15.1	Требования к полигону развертывания .....	320
15.1.1	2Требования к техническим и программным средствам.....	325
15.2	Core .....	330
15.3	Pool Unit.....	331
15.4	Требования по обеспечению безопасности .....	332
15.5	Инструкция по установке пользователя на виртуальную машину .....	333

# 1 Mongo db(ReplicaSet)

## 1.1 Описание

Создание кластера(реплика-сет) Mongo db. Развертывание на 3х машинах, с инициализацией кластера и созданием пользователей promed/zabbix.

## 1.2 Зависимости проекта

## 1.3 Развертывание с использованием docker-образа

### 1.3.1 Необходимое ПО

- docker v19.03
- docker-compose 1.25

### 1.3.2 Структура папок монтируемых на host-машину

/data

└─ db - папка с дата файлами бд.

/etc

└─ timezone-временная зона.

└─ localtime - текущее время.

/etc/mongo

└─ auth - файлы аутентификации.

/sys/kernel/mm/transparent\_hugepage/

└─ enabled - файл вкл/выкл использования "hugepage".

**Для развертывания необходимо**

Для каждой машины:

1. Создать папку развертывания
2. В корне папки развертывания создать файл **.env** ([пример](#) содержания ниже)
3. Создать в папке развертывания файл **docker-compose.yaml** ([пример](#) содержания ниже)
4. Проверить синтаксис конфигурации docker-compose командой: **docker-compose config**

Для одной из машин:

1. В файл **docker-compose.yaml** добавить содержание блока [extension](#) в секции "services:".

Для каждой машины:

1. Запустить сервис с помощью команды: **docker-compose up -d**

**Примечание:**

В соответствии с [рекомендациями](#) Mongo, файловая система для лучшей производительности на машинах должна быть xfs.

Для кастомизации параметров запуска нод кластера, в секции "command" контейнера-ноды можно добавить [ключи](#) запуска. Либо смонтировать готовые конфиги в контейнер, секцию "volumes:". Пример(- ./mongod.conf:/etc/mongod.conf:ro)

```
MONGO_INITDB_ROOT_USERNAME=<имя пользователя>
MONGO_INITDB_ROOT_PASSWORD=<пароль пользователя>
MONGO_KEY_FILE=<пароль для подключения нод к кластеру>
MONGO_HOST_1=<ip адрес хоста | dns имя хоста>:27017
MONGO_HOST_2=<ip адрес хоста | dns имя хоста>:27017
MONGO_HOST_3=<ip адрес хоста | dns имя хоста>:27017
```

#### Code Block 1 .env

```
version: "3.8"

services:
  # Контейнер инициализации ключевого файла для rs и установки hugepages
  на хост-машине в "never".
  pre-init:
    image: d-repo.rtmis.ru/promed/mongo:4.4.1
    env_file: .env
    command: sh -c "echo never >
/sys/kernel/mm/transparent_hugepage/enabled &&
    echo $MONGO_KEY_FILE > /etc/mongo/auth/mongo.keyfile &&
    chown mongod:mongod /etc/mongo/auth/mongo.keyfile &&
    chmod 400 /etc/mongo/auth/mongo.keyfile"
  privileged: true
  restart: "no"
  volumes:
  -
/sys/kernel/mm/transparent_hugepage/:/sys/kernel/mm/transparent_hugepage/:
rw
  - ./auth:/etc/mongo/auth/
  # Контейнер запуска дата-ноды
  mongo:
    image: d-repo.rtmis.ru/promed/mongo:4.4.1
    env_file: .env
    hostname: mongo
    depends_on:
    - pre-init
    command: mongod --replSet rs0 --port 27017 --keyFile
/etc/mongo/auth/mongo.keyfile --bind_ip_all
    ports:
    - 27017:27017
    volumes:
    # Папка дата-файлов. Создастся в текущем каталоге. Если диск с xfs
    смонтирован в ОС отдельно, указываем полный путь до точки монтирования.
    - ./mongo:/data/db
    - /etc/timezone:/etc/timezone:ro
    - /etc/localtime:/etc/localtime:ro
    - ./auth:/etc/mongo/auth:ro
  privileged: true
  ulimits:
    nproc: 64000
    nofile: 64000
```

#### Code Block 2 docker-compose.yaml



```
# Контейнер инициализации реплика-сет и создания пользователей.
mongo-init:
  image: d-repo.rtmis.ru/promed/mongo:4.4.1
  hostname: mongo-init
  env_file: .env
  depends_on:
    - mongo
  restart: "no"
  command: sh -c "/tmp/mongo_init.sh"
  volumes:
    - ./auth:/etc/mongo/auth/
```

### Code Block 3 extension

## 2 Node портал прокси

### 2.1 Описание сервиса:

Веб-сервер на **node.js** с подключенным модулем [socket.io](https://socket.io). Представляет собой сервис **доставки мгновенных сообщений** до клиентов подключенных к этому сервису.

Сервис предназначен для работы в связке с модулем электронной очереди (ЭО) и модулем Расписания врачей РМИС. Доступ к данному сервису прописывается в конфигах промеда и конфигах регионального портала медицинских услуг.

Доступ к данному сервису должны иметь все - не только вебы, но и сами клиенты (врачи армов, пользователи регионального портала мед. услуг). Сервис для конечного пользователя должен обязательно виден как защищенный сервис (<https://>), для возможности выполнения кросс-доменных запросов в нему.

### 2.2 Репозиторий в Git:

<https://git.promedweb.ru/rtmis/promed/-/tree/master/jscore/node-portal-proxy>

### 2.3 Примечание:

Для "Промеда`а" обязательная зависимость. Настройка через файл конфигурации promed.conf

```
// NODE JS прокси для портала
// 1. для имени хоста обязательно указывать протокол (http:// или
https://) в зависимости от того, в каком режиме запущен NODE-SERVER (можно
посмотреть при старте сервера или в логах)
// 2. номера портов дублируются - это нормально, т.к. сокет сервер теперь
инициализируется через порт веб-сервера
// 3. если блок не закомментирован. Обязателен к разворачиванию сервис
node-portal-
proxy(https://confluence.rtlabs.ru/pages/viewpage.action?pageId=253706886)
define('NODEJS_PORTAL_PROXY_HOSTNAME', 'http://127.0.0.1');
define('NODEJS_PORTAL_PROXY_HTTPPORT', '7070');
define('NODEJS_PORTAL_PROXY_SOCKETPORT', '7070');
```

Code Block 4 Пример promed.conf

## 2.4 Развертывание с использованием docker-образа

### 2.4.1 Необходимое ПО

- docker v19.03
- docker-compose 1.25

### 2.4.2 Структура папок монтируемых на host-машину

/srv/node-portal-proxy/

- |— certs - папка с сертификатами
- |— logs - папка с логами
- |— constants.js - файл настроек сервера

### 2.4.3 Для развертывания необходимо

1. Создать папку развертывания
2. В корне папки развертывания создать файл **constants.js** (пример содержания ниже)
3. Создать папки **logs**, **certs**
4. Поместить в папку **certs** действующий сертификат, ключ
5. Создать в папке развертывания файл **docker-compose.yaml** (пример содержания ниже)
6. Заменить значение внешнего порта в секции **ports** с **7070** на необходимое значение внешнего порта сервиса
7. Проверить синтаксис конфигурации docker-compose командой: **docker-compose config**
8. Запустить сервис с помощью команды: **docker-compose up -d**
9. Предоставить доступ к сервису ([пример](#) настройки балансировщика)

```
var constants = {
    httpServerPort : 7070,

    // для создания https сервера нужен сертификат сервера и его ключ
    // можно указать путь до файла или положить внутри папки нода
    certKey : 'certs/server.key',
    // сам сертификат
    cert: 'certs/server.crt',

    // путь до лога
    portal_proxy_log: 'logs/node_portal_proxy_info.log',
};

for (var key in constants) {
    exports[key] = constants[key];
};
```

**Code Block 5 constants.js**

```
version: '3.3'

services:
  node-portal-proxy:
    image: d-repo.rtmis.ru/promed/node-portal-proxy:1.0.0
    container_name: node-portal-proxy
    volumes:
      - ./constants.js:/srv/node-portal-proxy/constants.js
      - ./logs:/srv/node-portal-proxy/logs/
      - ./certs:/srv/node-portal-proxy/certs/
    ports:
      - 7070:7070
```

**Code Block 6 docker-compose.yaml**

```

map $http_upgrade $connection_upgrade {
    default upgrade;
    '' close;
}

server {
    listen                7070;
    server_name          promed.promedweb.ru;

    ssl                  on;
    ssl_certificate      /etc/nginx/ssl/promedweb_2020.crt;
    ssl_certificate_key  /etc/nginx/ssl/promedweb_2020.key;
    ssl_session_timeout 5m;
    ssl_protocols        TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers          kEECDH+AES128:kEECDH:kEDH:-
3DES:kRSA+AES128:kEDH+3DES:DES-CBC3-SHA:!RC4:!aNULL:!eNULL:!MD5:!EXPORT$
    ssl_prefer_server_ciphers on;

    error_log            /var/log/nginx/nodejs_error.log;
    open_file_cache_errors off;

    location / {
        proxy_pass                https://node-portal-proxy;
        proxy_set_header          Access-Control-Allow-
Origin                *;
        proxy_connect_timeout     10;
        proxy_send_timeout        60s;
        proxy_read_timeout        600s;
        proxy_http_version        1.1;
        proxy_set_header          Upgrade
$http_upgrade;
        proxy_set_header          Connection
$connection_upgrade;
    }
}

```

**Code Block 7** Примере настройки Nginx

## 3 PostgreSQL на CentOS 8

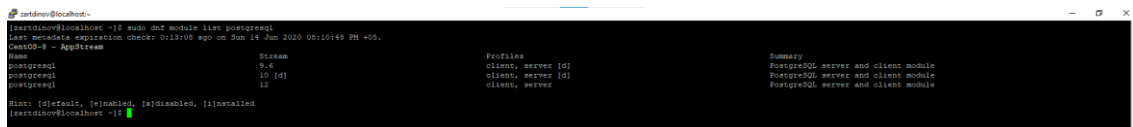
- [PostgreSQL 11](#)
- [PostgreSQL 12](#)

### 3.1 PostgreSQL 11

```
OS: CentOS 8  
DB: PostgreSQL 11
```

Смотрим доступные версии PostgreSQL:

```
$ sudo dnf module list postgresql
```



```
zartdinov@localhost:~$ sudo dnf module list postgresql  
Last metadata expiration check: 0:13:08 ago on Sun 14 Jun 2020 08:20:48 PM +05.  
CentOS-8 - AppStream  
Name           Stream           Profiles           Summary  
postgresql     9.6              client, server [d] PostgreSQL server and client module  
postgresql     10 (d)           client, server [d] PostgreSQL server and client module  
postgresql     12              client, server    PostgreSQL server and client module  
Hint: [d]default, [e]enabled, [s]disabled, [i]installed  
zartdinov@localhost:~$
```

По умолчанию в AppStream доступно 3 версии PostgreSQL:

- 9.6
- 10
- 12

Нам нужна 11 версия PostgreSQL.

Актуальную ссылку на RPM repository PostgreSQL можно взять с официального сайта компании PostgreSQL: [PostgreSQL: Repository](https://www.postgresql.org/ftp/repofiles/)

Добавляем RPM repository PostgreSQL:

```
$ sudo dnf -y install  
https://download.postgresql.org/pub/repos/yum/repofiles/EL-8-x86_64/pgdg-  
redhat-repo-latest.noarch.rpm
```

```

sardine@localhost:~$ sudo dnf -y install https://download.postgresql.org/pub/repos/yum/repos/EL-8-x86_64/pgdg-redhat-repo-latest.noarch.rpm
Last metadata expiration check: 0:00:53 ago on Mon 15 Jun 2020 02:59:50 PM +05.
pgdg-redhat-repo-latest.noarch.rpm
Dependencies resolved.
Package Architecture Version Repository Size
Installing:
pgdg-redhat-repo noarch 42.0-11 @commandline 11 k
Transaction Summary
Install 1 Package
Total size: 11 k
Installed size: 21 k
Downloading Packages:
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing :
Installing : pgdg-redhat-repo-42.0-11.noarch 1/1
Verifying : pgdg-redhat-repo-42.0-11.noarch 1/1
Installed:
pgdg-redhat-repo-42.0-11.noarch
Complete!
sardine@localhost:~$

```

Смотрим дополнительную информацию по добавленному RPM repository PostgreSQL:

```

$ sudo rpm -qi pgdg-redhat-repo

```

```

sardine@localhost:~$ sudo rpm -qi pgdg-redhat-repo
Name : pgdg-redhat-repo
Version : 42.0
Release : 11
Architecture: noarch
Install Date: Mon 15 Jun 2020 03:00:45 PM +05
Group : Unspecified
License : LGPL
Signature : DSA/SHA1, Fri 01 May 2020 07:21:12 PM +05, Key ID ff1642e143df0f3
Source RPM : pgdg-redhat-repo-42.0-11.src.rpm
Build Date : Fri 01 May 2020 07:21:12 PM +05
Build Root : /tmp/roots/42-44/pgdg11a
Relocations : (not relocatable)
Vendor : PostgreSQL Global Development Group
URL : https://yum.postgresql.org
Summary : PostgreSQL PGDG RPMs - Yum Repository Configuration for Red Hat / CentOS
Description
This package contains yum configuration for Red Hat Enterprise Linux, CentOS,
and also the GPG key for PGDG RPMs.
sardine@localhost:~$

```

Отключаем стандартный AppStream PostgreSQL:

```

$ sudo dnf -y module disable postgresql

```

```

sardine@localhost:~$ sudo dnf -y module disable postgresql
Last metadata expiration check: 0:03:49 ago on Mon 15 Jun 2020 03:05:36 PM +05.
Dependencies resolved.
Package Architecture Version Repository Size
Disabling modules:
postgresql
Transaction Summary
Complete!
sardine@localhost:~$

```

Проверяем отключение стандартного AppStream PostgreSQL:

```

$ sudo dnf module list postgresql

```

```

sardine@localhost:~$ sudo dnf module list postgresql
Last metadata expiration check: 0:05:19 ago on Mon 15 Jun 2020 03:09:36 PM +05.
CentOS-8 - AppStream
Name Stream Profile Summary
postgresql 9.6 [s] client, server [d] PostgreSQL server and client module
postgresql 10 [d] [s] client, server [d] PostgreSQL server and client module
postgresql 12 [s] client, server PostgreSQL server and client module
Hint: [default], [enabled], [disabled], [installed]
sardine@localhost:~$

```

Очищаем историю транзакций:

```
$ sudo dnf -y clean all
```

## Устанавливаем PostgreSQL:

```
$ sudo dnf -y install postgresql11-server postgresql11
```

```
satinov@localhost:~$ sudo dnf -y install postgresql11-server postgresql11
Last metadata expiration check: 2020/07/05 09:15:57 ago on Mon 19 Jul 2020 09:12:18 PM +03.
Dependencies resolved.
=====================================================================================================================================
Package                               Architecture      Version           Repository        Size
=====================================================================================================================================
Installing:
postgresql                             x86_64            11.8-2PGDG.rhel8 postgresql        1.7 M
postgresql-server                       x86_64            11.8-2PGDG.rhel8 postgresql        4.9 M
Installing dependencies:
libicu                                   x86_64            60.3-2.el8_1     BaseOS           8.8 M
postgresql-libs                         x86_64            11.8-2PGDG.rhel8 postgresql        390 K
Transaction Summary
-----
Install 4 Packages

Total download size: 16 M
Installed size: 42 M
Downloading Packages:
(1/4): libicu-60.3-2.el8_1.x86_64.rpm                                                    19 MB/s | 5.9 MB  00:00
(2/4): postgresql-libs-11.8-2PGDG.rhel8.x86_64.rpm                                    218 KB/s | 390 KB  00:01
(3/4): postgresql-11.8-2PGDG.rhel8.x86_64.rpm                                         288 KB/s | 1.7 MB  00:02
(4/4): postgresql-server-11.8-2PGDG.rhel8.x86_64.rpm                                  2.1 MB/s | 4.9 MB  00:02
Total                                                                                   5.2 MB/s | 16 MB  00:03
Warning: /var/cache/dnf/pgdg11-b4f717a855a476d/packages/postgresql-11.8-2PGDG.rhel8.x86_64.rpm: Header V4 DSA/SHA1 Signature, key ID 442df093: NOKEY
PostgreSQL 11 for RHEL/CentOS 8 - x86_64
Importing GPG key 442df093:
  Userid  : "PostgreSQL RPM Building Project <pg@rpm-software-project.org>"
  Fingerprint: 4009 2208 1A37 0336 7274 0376 1814 02D2 44D0 F09E
  From     : /etc/pki/rpm-gpg/RPM-GPG-KEY-PGQG
Key imported successfully.
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction:
  Preparing                | 1/1
  Installing postgresql-libs-11.8-2PGDG.rhel8.x86_64 | 1/4
  Running scriptlet: postgresql-libs-11.8-2PGDG.rhel8.x86_64 | 1/4
  Installing libicu-60.3-2.el8_1.x86_64 | 2/4
  Running scriptlet: libicu-60.3-2.el8_1.x86_64 | 2/4
  Installing postgresql-libs-11.8-2PGDG.rhel8.x86_64 | 3/4
  Running scriptlet: postgresql-libs-11.8-2PGDG.rhel8.x86_64 | 3/4
  Installing postgresql-server-11.8-2PGDG.rhel8.x86_64 | 4/4
  Running scriptlet: postgresql-server-11.8-2PGDG.rhel8.x86_64 | 4/4
  Verifying libicu-60.3-2.el8_1.x86_64 | 1/4
  Verifying postgresql-libs-11.8-2PGDG.rhel8.x86_64 | 2/4
  Verifying postgresql-libs-11.8-2PGDG.rhel8.x86_64 | 3/4
  Verifying postgresql-server-11.8-2PGDG.rhel8.x86_64 | 4/4
Installed:
  postgresql-11.8-2PGDG.rhel8.x86_64                postgresql-server-11.8-2PGDG.rhel8.x86_64
  libicu-60.3-2.el8_1.x86_64                        postgresql-libs-11.8-2PGDG.rhel8.x86_64
Complete!
satinov@localhost ~$
```

## Создаем кластер DB PostgreSQL:

```
$ sudo /usr/pgsql-11/bin/postgresql-11-setup initdb
```

Ключом "-D" можно указать любую, другую директорию для кластера DB PostgreSQL:

```
$ sudo /usr/pgsql-11/bin/postgresql-11-setup initdb -D /dbpostgres
```

Активируем запуск службы PostgreSQL при запуске сервера и запустим службу PostgreSQL:

```
$ sudo systemctl enable --now postgresql-11
```

```
satinov@localhost ~$ sudo systemctl enable --now postgresql-11
Created symlink /etc/systemd/system/multi-user.target.wants/postgresql-11.service → /usr/lib/systemd/system/postgresql-11.service.
satinov@localhost bin$
```









```
# IPv4 local connections:
host    all         all         0.0.0.0/0    md5
```



```
zartdinov@localhost:~$ cat /etc/postgresql/11/pg_hba.conf
# PostgreSQL configuration files.
# You can find the configuration files for the following programs in the
# corresponding subdirectories:
#
# postgresql.conf, postgresql.auto.conf
#
# Do not edit this file directly.
# For more details, see the PostgreSQL documentation.

# IPv4 local connections:
local all all 0.0.0.0/0 md5
# IPv6 local connections:
#local all all :::0/0 md5
# Replication connections:
local replication all peer
host replication all 127.0.0.1/32 ident
host replication all :::1/128 ident
--TEXT--
```

Перезапускаем службу PostgreSQL:

```
$ sudo systemctl restart postgresql
```

Запускаем пользователем postgres command line psql:

```
$ sudo -u postgres psql
```



```
zartdinov@localhost:~/pgsql-11/bin$ sudo -u postgres psql
psql (11.9)
Type "help" for help.

postgres=#
```

Создаем пользователя:

```
CREATE USER "zartdinov" WITH SUPERUSER ENCRYPTED PASSWORD 'QWERTY';
```



```
postgres=# CREATE USER "zartdinov" WITH SUPERUSER ENCRYPTED PASSWORD 'QWERTY';
CREATE ROLE
postgres=#
```

Открываем port 5432 на firewall:

```
$ sudo firewall-cmd --add-port=5432/tcp --permanent
```



```
zartdinov@localhost-  
[zartdinov@localhost ~]$ sudo firewall-cmd --add-port=5432/tcp --permanent  
#success#  
[zartdinov@localhost ~]$
```

Перезапускаем firewall:


```
$ sudo firewall-cmd --reload
```



```
zartdinov@localhost-  
[zartdinov@localhost ~]$ sudo firewall-cmd --reload  
#success#  
[zartdinov@localhost ~]$
```

Проверяем подключение к серверу DB PostgreSQL с другого персонального компьютера с помощью cmd telnet:

```
$ telnet 192.168.100.103 5432
```



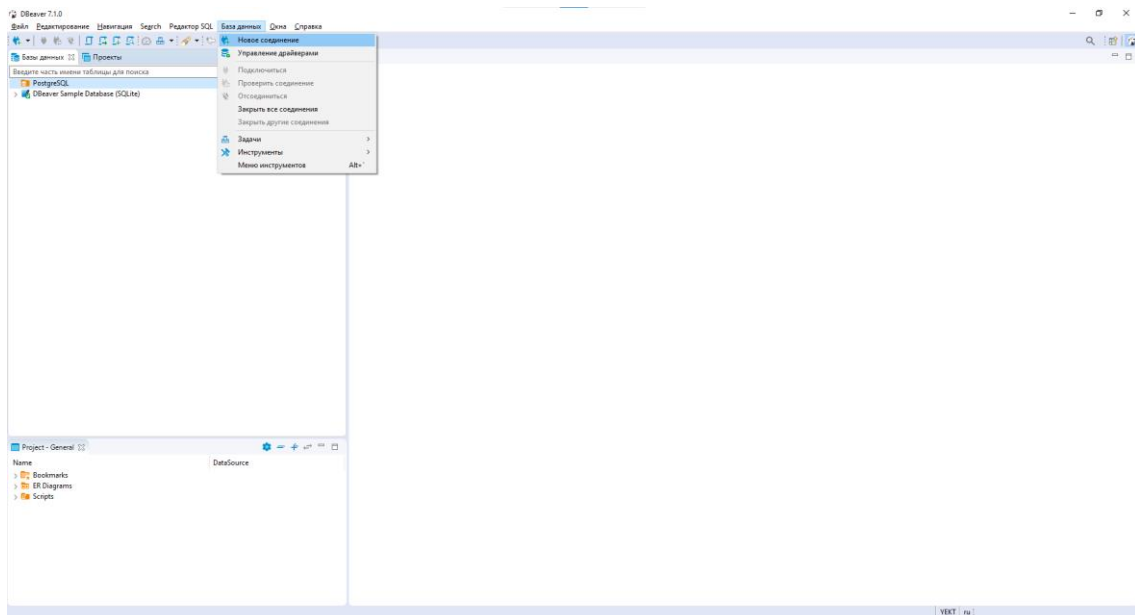
```
C:\Windows\system32\cmd.exe  
[Microsoft Windows [version 10.0.19041.2144]  
(c) Корпорация Майкрософт (Microsoft Corporation), 2020. Все права защищены.  
C:\Users\Ismayil>telnet 192.168.100.103 5432
```

Если появится черный экран, port 5432 был успешно открыт:

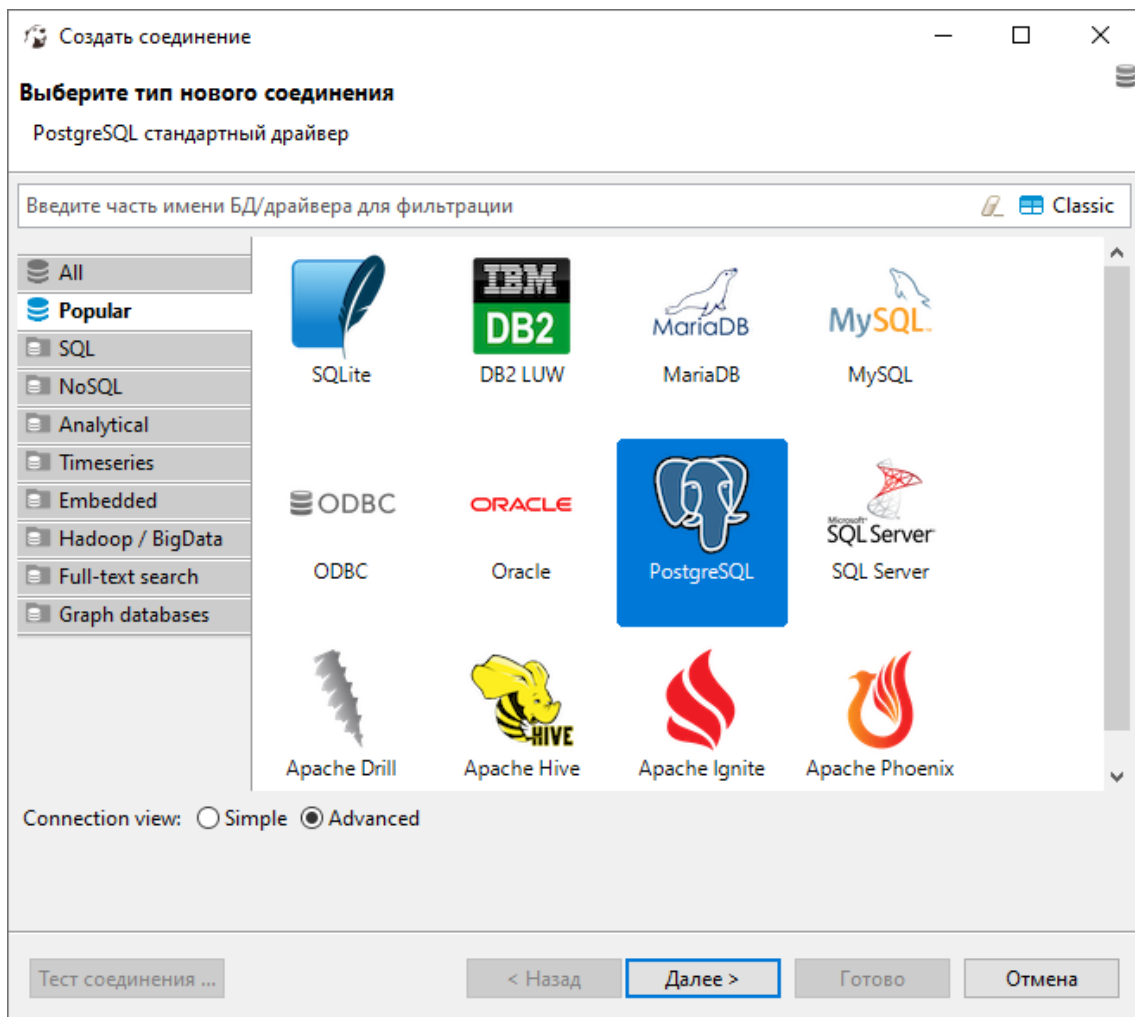


```
Telnet 192.168.100.103
```

В Dbeaver, в меню выбираем пункт "Базы данных", "Новое соединение":



В открывшемся окне выбираем базу данных "PostgreSQL":



Создать соединение

### Настройки соединения

Свойства соединения с PostgreSQL

Главное PostgreSQL Свойства драйвера SSH Proxy SSL

Server

Хост: 192.168.100.103 Порт: 5432

База данных: postgres

Аутентификация

Аутентификация: Database Native Database native authentication

Пользователь: zartdinov

Пароль: ●●●●●●  Сохранять пароль локально

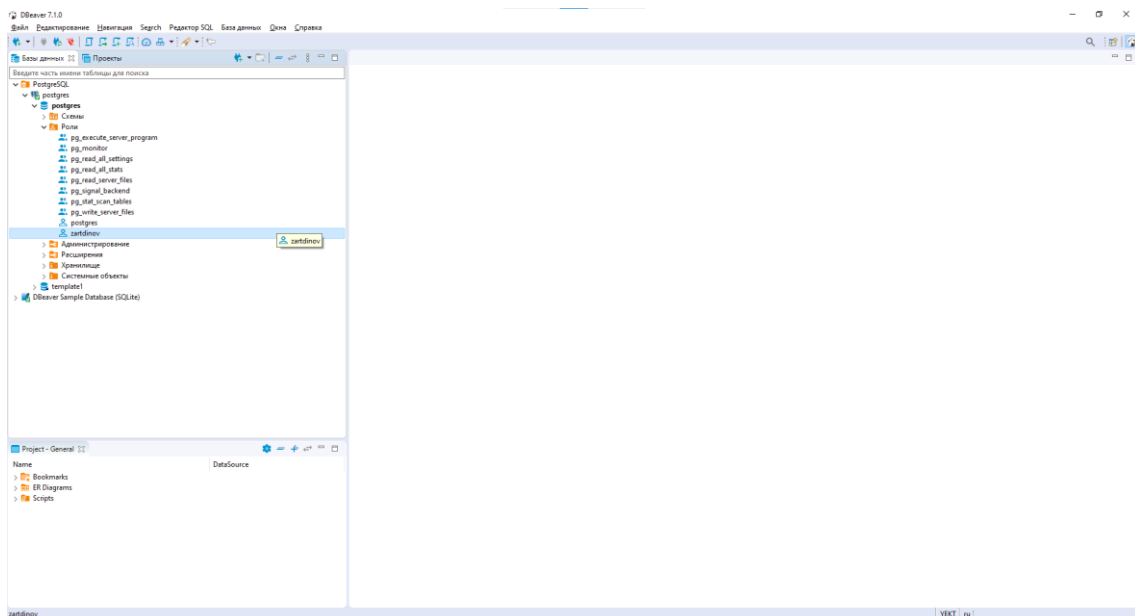
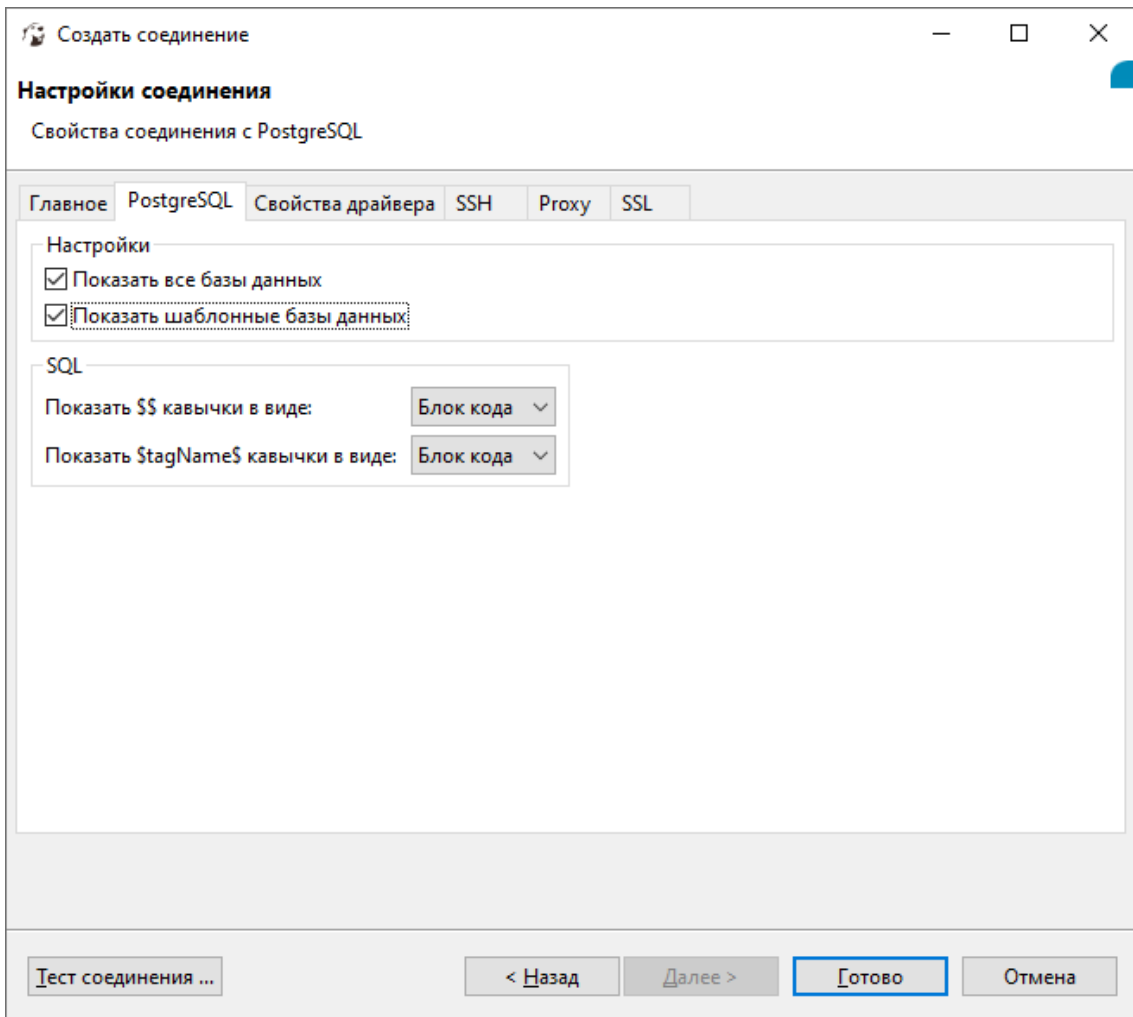
Advanced

Локальный клиент: PostgreSQL Binaries

**i** Вы можете использовать системные переменные в параметрах. Описание соединения (название, тип, ...)

Драйвер: PostgreSQL [Настройки драйвера](#)

Тест соединения ... < Назад Далее > **Готово** Отмена



## 3.2 PostgreSQL 12

OS: CentOS 8  
DB: PostgreSQL 12

Смотрим доступные версии PostgreSQL:

```
$ sudo dnf module list postgresql
```

В ответ должны получить:

```
zardnov@localhost-
[zardnov@localhost ~]$ sudo dnf module list postgresql
Last metadata expiration check: 0:12:08 ago on Sun 14 Jun 2020 08:10:48 PM +05.
Name                               Stream          Profiles                               Summary
postgresql                          9.4             client, server (d)                    PostgreSQL server and client module
postgresql                          10 (d)         client, server (d)                    PostgreSQL server and client module
postgresql                          12             client, server                          PostgreSQL server and client module

Name: [d]default, [e]enabled, [w]disabled, [i]installed
[zardnov@localhost ~]$
```

По умолчанию поток модуля установки рассматривает PostgreSQL 10, чтобы переключить на PostgreSQL 12:

```
$ sudo dnf -y module enable postgresql:12
```

Установка PostgreSQL 12:

```
$ sudo dnf -y install postgresql-server
```

```
zardnov@localhost-
[zardnov@localhost ~]$ sudo dnf -y install postgresql-server
Last metadata expiration check: 0:00:16 ago on Sun 14 Jun 2020 10:49:58 PM +05.
Dependencies resolved.
=====================================================================================================================================
 Package                               Architecture      Version                               Repository      Size
=====================================================================================================================================
Installing:
 postgresql-server                     x86_64            12.1-2.module_el8.1.0+273+979c16e6.x86_64  AppStream      5.5 M
Installing dependencies:
 libpq                                  x86_64            12.1-3.el8                               AppStream      195 k
 postgresql                             x86_64            12.1-2.module_el8.1.0+273+979c16e6.x86_64  AppStream      1.4 M
 libicu                                  x86_64            60.3-2.el8_1                              baseos         8.3 M
Transaction Summary
-----
Install 4 Packages

Total download size: 16 M
Installed size: 62 M
Downloading Packages:
(1/4): libpq-12.1-3.el8.x86_64.rpm                                                    809 kB/s | 195 kB  00:00
(2/4): postgresql-12.1-2.module_el8.1.0+273+979c16e6.x86_64.rpm                    2.4 MB/s | 1.4 MB  00:00
(3/4): libicu-60.3-2.el8_1.x86_64.rpm                                                 20 MB/s | 8.8 MB  00:00
(4/4): postgresql-server-12.1-2.module_el8.1.0+273+979c16e6.x86_64.rpm             5.0 MB/s | 5.5 MB  00:01
-----
Total: 9.9 MB/s | 16 MB  00:01
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing                | 1/4
  Installing postgresql-12.1-3.el8.x86_64                                           1/4
  Installing postgresql-12.1-2.module_el8.1.0+273+979c16e6.x86_64                  2/4
  Installing libicu-60.3-2.el8_1.x86_64                                             3/4
  Running scriptlet: libicu-60.3-2.el8_1.x86_64                                     3/4
  Installing postgresql-server-12.1-2.module_el8.1.0+273+979c16e6.x86_64          4/4
  Running scriptlet: postgresql-server-12.1-2.module_el8.1.0+273+979c16e6.x86_64  4/4
  Verifying                 | 1/4
  Verifying postgresql-12.1-2.module_el8.1.0+273+979c16e6.x86_64                  2/4
  Verifying postgresql-server-12.1-2.module_el8.1.0+273+979c16e6.x86_64          3/4
  Verifying libicu-60.3-2.el8_1.x86_64                                             4/4
Installed:
 postgresql-server-12.1-2.module_el8.1.0+273+979c16e6.x86_64      libpq-12.1-3.el8.x86_64      postgresql-12.1-2.module_el8.1.0+273+979c16e6.x86_64      libicu-60.3-2.el8_1.x86_64
Complete!
[zardnov@localhost ~]$
```

Создаем кластер DB PostgreSQL:

```
$ sudo postgresql-setup --initdb
```



```
zartimov@localhost:~$ sudo systemctl start postgresql
[sartimov@localhost ~]$ sudo systemctl status postgresql
* Initializing database in '/var/lib/pgsql/data'.
* Initialized, logs are in /var/lib/pgsql/initdb_postgresql.log
[sartimov@localhost ~]$
```

Запускаем службу PostgreSQL:

```
$ sudo systemctl start postgresql
```

Активируем запуск службы PostgreSQL при запуске сервера:

```
$ sudo systemctl enable postgresql
```

Проверяем запуск службы PostgreSQL:

```
$ sudo systemctl status postgresql
```

```
zartimov@localhost:~$ sudo systemctl start postgresql
[sartimov@localhost ~]$ sudo systemctl status postgresql
Created symlink /etc/systemd/system/multi-user.target.wants/postgresql.service → /usr/lib/systemd/system/postgresql.service.
[sartimov@localhost ~]$ sudo systemctl status postgresql
● postgresql.service - PostgreSQL database server
   Loaded: loaded (/usr/lib/systemd/system/postgresql.service; enabled; vendor preset: disabled)
   Active: active (running) since Sun 2020-06-14 22:59:46 +03; 49s ago
     Main PID: 3104 (postmaster)
       Tasks: 6 (limit: 18495)
      Memory: 17.4M
     CGroup: /systemd/system/postgresql.service
            └─3104 /usr/bin/postmaster -D /var/lib/pgsql/data
                  └─3105 postgres: logger
                        └─3106 postgres: checkpointer
                              └─3108 postgres: background writer
                                    └─3109 postgres: walwriter
                                          └─3110 postgres: autovacuum launcher
                                                └─3111 postgres: stats collector
                                                      └─3112 postgres: logical replication launcher

Jun 14 22:58:46 localhost.localdomain systemd[1]: Starting PostgreSQL database server...
Jun 14 22:58:46 localhost.localdomain postmaster[3104]: 2020-06-14 22:58:46.635 +03 [3104] LOG: starting PostgreSQL 12.1 on x86_64-redhat-linux-gnu, compiled by gcc (GCC) 8.3.1.20190507 (Red Hat 8.3.1-4), 64-bit
Jun 14 22:58:46 localhost.localdomain postmaster[3104]: 2020-06-14 22:58:46.642 +03 [3104] LOG: listening on IPv4 address "111", port 5432
Jun 14 22:58:46 localhost.localdomain postmaster[3104]: 2020-06-14 22:58:46.652 +03 [3104] LOG: listening on IPv6 address "::111", port 5432
Jun 14 22:58:46 localhost.localdomain postmaster[3104]: 2020-06-14 22:58:46.669 +03 [3104] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
Jun 14 22:58:46 localhost.localdomain postmaster[3104]: 2020-06-14 22:58:46.682 +03 [3104] LOG: listening on Unix socket "/tmp/.s.PGSQL.5432"
Jun 14 22:58:46 localhost.localdomain postmaster[3104]: 2020-06-14 22:58:46.679 +03 [3104] LOG: redirecting log output to logging collector process
Jun 14 22:58:46 localhost.localdomain postmaster[3104]: 2020-06-14 22:58:46.679 +03 [3104] WARNING: Future log output will appear in directory "log".
[sartimov@localhost ~]$
```

```
$ sudo vim /var/lib/pgsql/data/postgresql.conf
```

Строку:

```
listen_addresses = 'localhost'
```





```
# IPv4 local connections:
host    all         all        0.0.0.0/0    md5
```



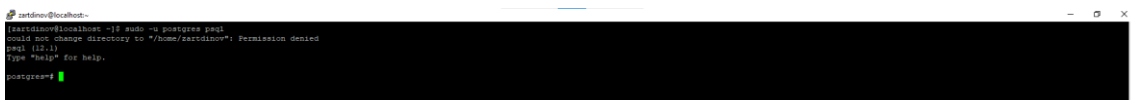
```
zartdinov@localhost:~$ nano /etc/postgresql/12/pg_hba.conf
# IPv4 local connections:
host    all         all        0.0.0.0/0    md5
# IPv6 local connections:
host    :::         :::        :::          md5
# Replication connections:
local   replication all         all        peer
host    replication all         127.0.0.1/32 ident
host    replication all         :::        ident
-- INSERT --
zartdinov@localhost:~$
```

Перезапускаем службу PostgreSQL:

```
$ sudo systemctl restart postgresql
```

Запускаем пользователем postgres command line psql:

```
$ sudo -u postgres psql
```



```
zartdinov@localhost:~$ sudo -u postgres psql
psql (12.1)
Type "help" for help.

postgres@localhost:~$
```

Создаем пользователя:

```
CREATE USER "zartdinov" WITH SUPERUSER ENCRYPTED PASSWORD 'QWERTY';
```



```
postgres@localhost:~$ CREATE USER 'zartdinov' WITH SUPERUSER ENCRYPTED PASSWORD 'QWERTY';
CREATE ROLE
postgres@localhost:~$
```

Открываем port 5432 на firewall:

```
$ sudo firewall-cmd --add-port=5432/tcp --permanent
```

```
cardinov@localhost:~$ sudo firewall-cmd --add-port=5432/tcp --permanent
Получено
cardinov@localhost:~$
```

Перезапускаем firewall:

```
sudo firewall-cmd --reload
```

```
cardinov@localhost:~$ sudo firewall-cmd --reload
Получено
cardinov@localhost:~$
```

Проверяем подключение к серверу DB PostgreSQL:

В cmd с помощью telnet:

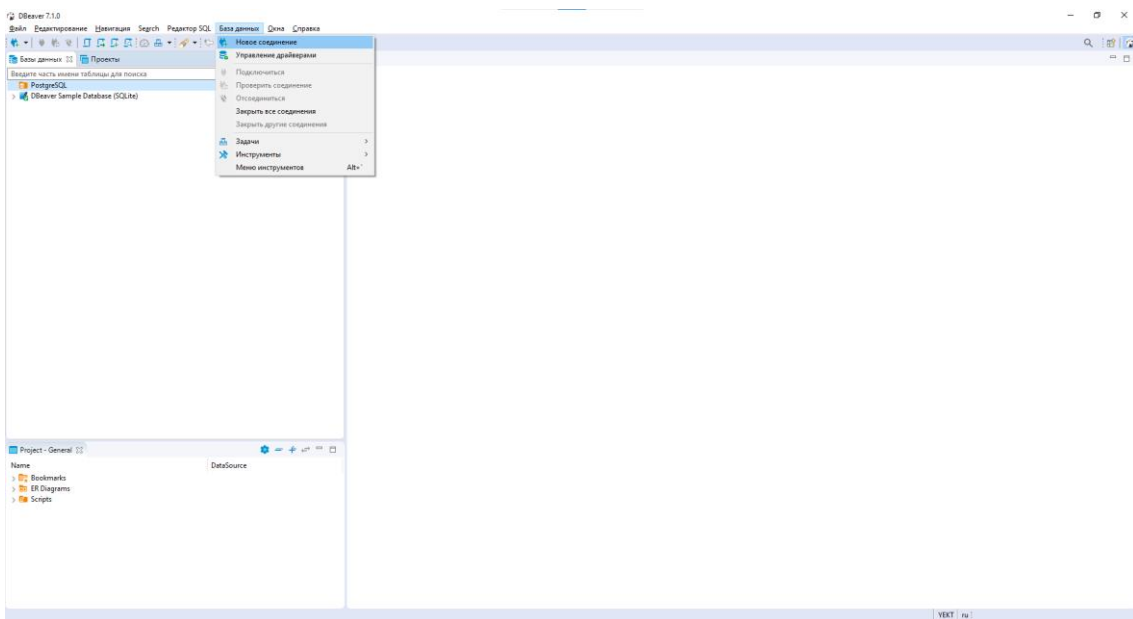
```
$ telnet 192.168.100.103 5432
```

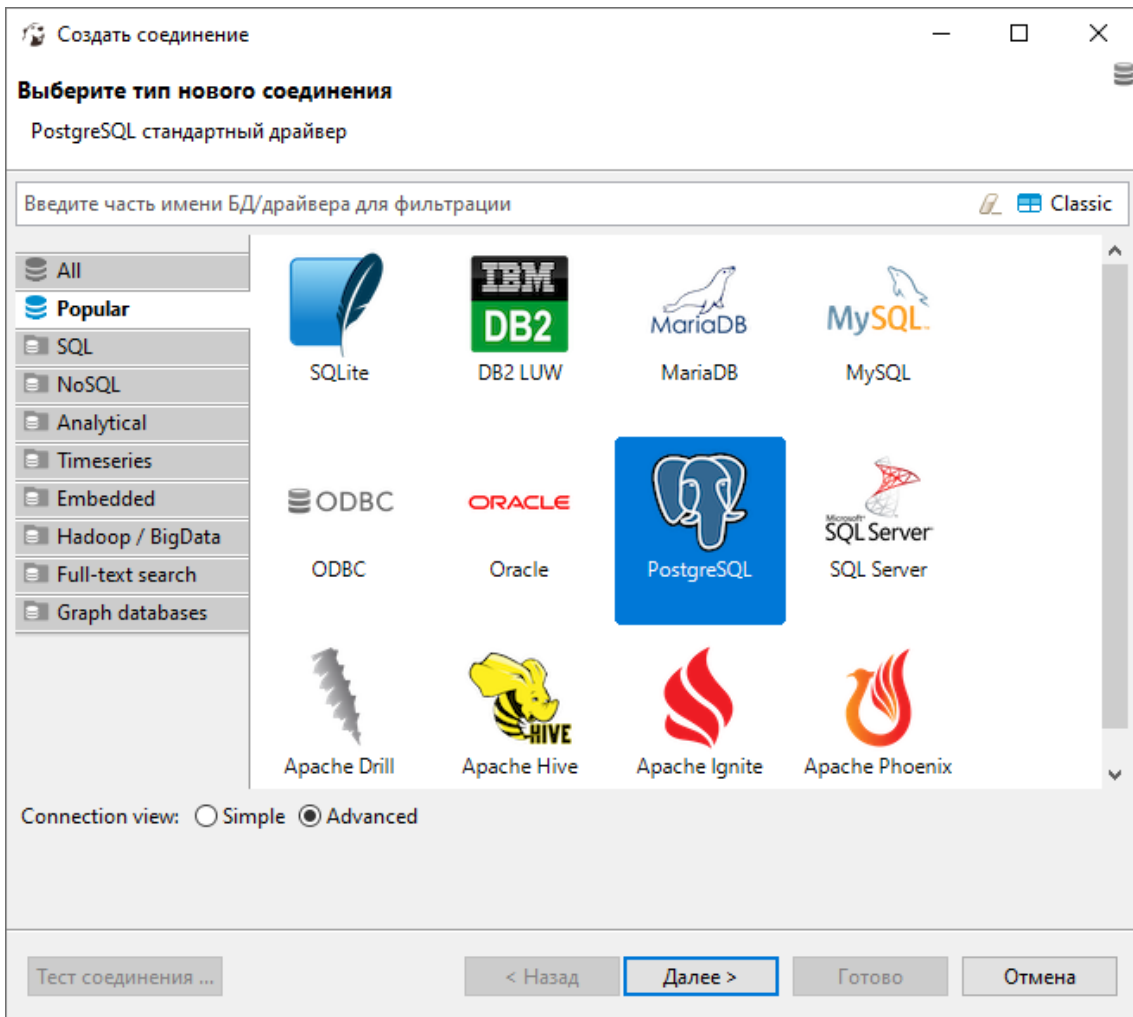
```
C:\Windows\system32\cmd.exe
Microsoft Windows [version 10.0.19041.704]
(c) Корпорация Майкрософт (Microsoft Corporation), 2020. Все права защищены.
C:\Users\kamo>telnet 192.168.100.103 5432_
```

Если появится черный экран, port 5432 был успешно открыт:

```
Telnet 192.168.100.103
```

В Dbeaver:





Создать соединение

### Настройки соединения

Свойства соединения с PostgreSQL

Главное PostgreSQL Свойства драйвера SSH Proxy SSL

Server

Хост: 192.168.100.103 Порт: 5432

База данных: postgres

Аутентификация

Аутентификация: Database Native Database native authentication

Пользователь: zartdinov

Пароль: ●●●●●●  Сохранять пароль локально

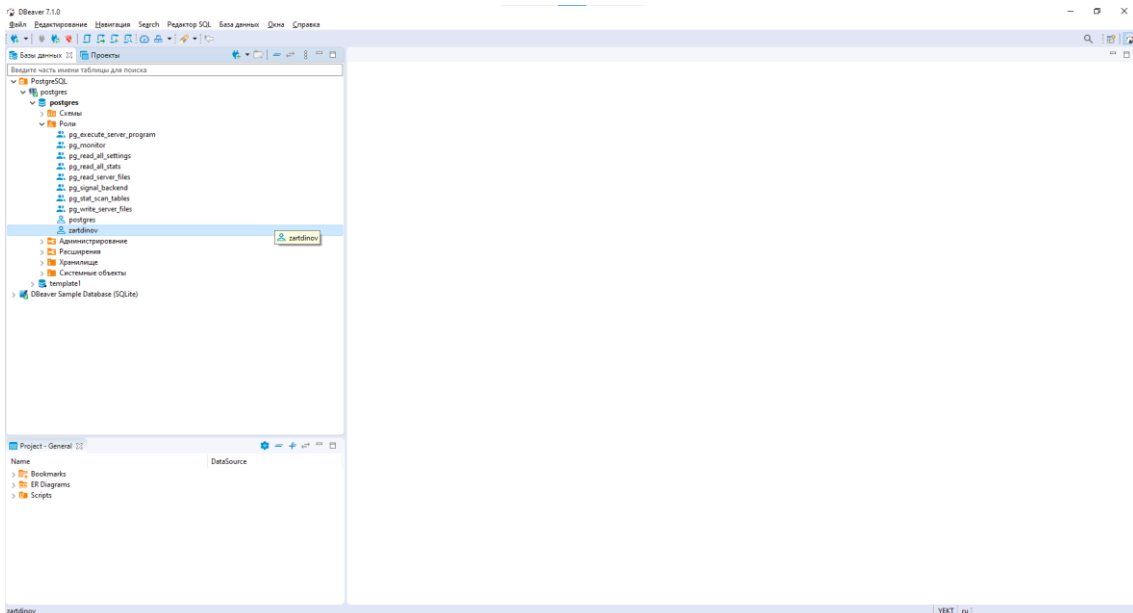
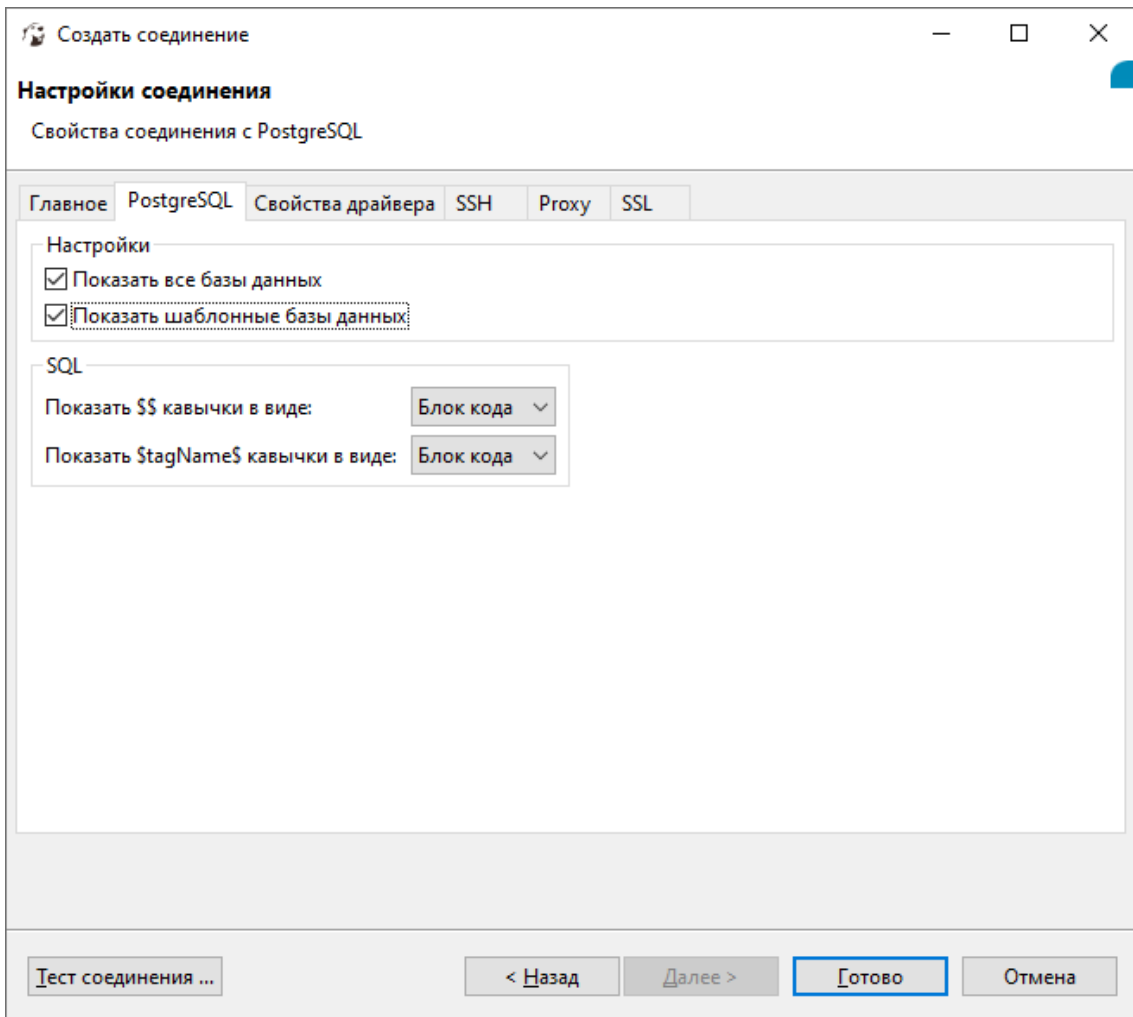
Advanced

Локальный клиент: PostgreSQL Binaries

**i** Вы можете использовать системные переменные в параметрах. Описание соединения (название, тип, ...)

Драйвер: PostgreSQL [Настройки драйвера](#)

Тест соединения ...





## 4 Балансировщик nginx

### 4.1 Структура папок

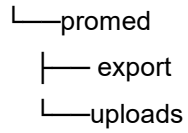
/etc/nginx

```
|— conf.d
| |— html
| |   └─ index.html
| |— main
| |   └─ promed.conf
| |— nojob
| |   └─ promed.conf
| |— params
| |   └─ birt_param.conf
| |   └─ ermp_param.conf
| |   └─ other_max.conf
| |   └─ other_min.conf
| |   └─ promed_param.conf
| |— ssl
| |   └─ promed.ssl
| |   └─ certificate.crt
| |       └─ certificate.key
| └─ upstream
|     └─ upstream.conf
|— fastcgi.conf
|— fastcgi_params
|— koi-utf
|— koi-win
|— mime.types
|— nginx.conf
|— proxy_params
|— scgi_params
|— uwsgi_params
└─ win-utf
```

html	находятся html файлы, для показа объявления в момент остановки системы
main	основная папка для хранения конфигурации ЕЦП.МИС
nojob	папка с копией конфигов из main но вместо проксирования на рабочие сервера показывает html заглушку из папки html
params	папка с параметрами для location внутри конфига

ssl	лежат сертификаты SSL/TLS для шифрованного трафика и файлы ssl в которых прописываются сертификаты и ключи и параметры их подключения. сами файлы инклюдятся в необходимые конфиги
upstream	файл с upstream до backend серверов

/srv/hosts



promed	папка - копия системы Промед. так же обновляется как и остальные веб сервер
export	папка export подключается в nfs/samba хранилища
uploads	папка uploads подключается в nfs/samba хранилища

## 4.2 Описание файлов конфигураций

```

user nginx; # Пользователь под которым будет
производиться работа сервиса
worker_processes auto; # Автоматически создастся
столько worker процессов сколько выделено машине
worker_rlimit_nofile 65536; # максимальное количество
открытых файлов на ядро. по умолчанию в системе 1024
error_log /var/log/nginx/error.log crit; # отключить error log
невозможно. он всё равно будет писать файлы, чтобы писал на диск меньше
ставим crit
pid /var/run/nginx.pid; # pid файл главного процесса
nginx

events {
    worker_connections 10240; #Реальное количество коннектов больше.
Просто умножай на количество процессов
    use epoll; #Наиболее эффективно для Linux так что
лучше оставить
}

http {
    include /etc/nginx/mime.types; # описание типов файлов
    например он узнает что такое текст и т.д.
    default_type application/octet-stream; # тип по умолчанию, если
не понятно какой тип

    #Это понятно. формат логов
    log_format main '$remote_addr - $remote_user [$time_local]
"$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';
    access_log access_log off; # отключаем
    access логи.
    error_log /var/log/nginx/nginx_error.log crit; # error log для
системы только критичесике
    sendfile on; #экономия
    ресурсов при отдаче файлов
    tcp_nopush on;
    tcp_nodelay off;

    keepalive_timeout 180; #keepalive
соединения. которые зависли - выгоняем. тут расчет в секундах
    gzip on; #Включаем сжатие
простых данных, для уменьшение трафика, в основном текстовых файлов
    gzip_proxied any; #разрешает
сжатие для всех проксированных запросо
    gzip_min_length 1100; #длина ответа.
после которой начинается сжатие, если меньше - сжатие не происходит
    gzip_http_version 1.0; #Минимальная
версия протокола, при каком включается gzip
    gzip_buffers 4 8k; # Задаёт число и
размер буферов, в которые будет сжиматься ответ.
    gzip_comp_level 6; # самое
эффективное сжатие 9, но очень сильно съедает процессор, хотя сильного
уменьшения с 6 небольшое
    gzip_types text/plain text/css application/x-javascript text/xml
application/xml application/xml+rss text/javascript application/json;
# типы сжимаемых данных
# include /etc/nginx/include/nojob/*.conf; # Заглушки
Промед по умолчанию выключены
    include /etc/nginx/include/upstream/*.conf; # подключаем
upstream конфиги
    include /etc/nginx/include/main/*.conf; # подключаем
папку с основными конфигами
}

```

**Code Block 8 nginx.conf**

```

# 192.168.0.{1,2,3,4,5} по порту 2080 нежно заменить на IP и порты ваших
# веб серверов
upstream promed_backend {
    server 192.168.0.1:2080 weight=1 max_fails=1 fail_timeout=5s;
    server 192.168.0.2:2080 weight=1 max_fails=1 fail_timeout=5s;
    server 192.168.0.3:2080 weight=1 max_fails=1 fail_timeout=5s;
    server 192.168.0.4:2080 weight=1 max_fails=1 fail_timeout=5s;
    server 192.168.0.5:2080 weight=1 max_fails=1 fail_timeout=5s;
    keepalive 40;
}
upstream birt_backend {
    server 192.168.0.1:8080 weight=1 max_fails=1
fail_timeout=5s;
    server 192.168.0.2:8080 weight=1 max_fails=1
fail_timeout=5s;
    keepalive 40;
}
upstream ermp_backend {
    server 192.168.0.1:8080 weight=1 max_fails=1
fail_timeout=5s;
    server 192.168.0.2:8080 weight=1 max_fails=1
fail_timeout=5s;
    keepalive 40;
}
upstream wiki_backend {
    server 192.168.0.1:2022 weight=1 max_fails=1
fail_timeout=5s;
    keepalive 40;
}

```

**Code Block 9 upstream/upstream.conf**

```

# Делаем редирект со всеми параметрами на ssl
server {
    listen                80;
    server_name          promed.ru;           # Заменить на
необходимый адрес
    access_log           off;
    error_log            /dev/null crit;
    open_file_cache_errors off;

    rewrite ^(.*) https://$host$1 permanent;
}

server {
    listen                443 ssl http2;     # Включаем для
этого адреса ssl и http2 протокол
    server_name          promed.ru;         # Заменить на
необходимый адрес
    include              /etc/nginx/ssl/promed.ssl; # Добавляем
настройки ssl которые я описан ниже
    access_log           /var/log/nginx/promed_access.log main
buffer=64k;
    error_log            /var/log/nginx/promed_error.log warn;
    open_file_cache_errors off;

    location / {
        proxy_pass      http://promed_backend/;
        proxy_next_upstream error timeout http_500 http_502
http_503 http_504;
        include
/etc/nginx/params/promed_param.conf;

        location /birt-viewer/run {
            proxy_pass      http://birt_backend/birt-
viewer/preview;
            include
/etc/nginx/conf.d/params/birt_param.conf;
        }

        location /birt-viewer/ {
            proxy_pass      http://birt_backend/birt-
viewer/;
            include
/etc/nginx/conf.d/params/birt_param.conf;
        }

        location /ermp/servlets/ {
            proxy_pass      http://ermp_backend/ermp/;
            include
/etc/nginx/conf.d/params/ermp_param.conf;
        }
        location /ermp/reports/ {
            proxy_pass      http://ermp_backend/ermp/;
            include
/etc/nginx/conf.d/params/ermp_param.conf;
        }
        location /wiki/ {
            proxy_pass
http://wiki_backend/wiki/;
            include
/etc/nginx/conf.d/params/other_min.conf;
        }

        #Добавляем если статика находится у нас на nginx
        location ~*
^(?!/wiki/|/export/|/uploads/).+.(jpg|jpeg|gif|png|ico|zip|tgz|gz|rar|bz2|

```



```

#на Ubuntu 18.04 и выше уже стоит openssl версии 1.1.1f и установка не
требуется

#ставим библиотеку для подключения gost
apt update
apt install libengine-gost-openssl1.1
#Правим /etc/ssl/openssl.cnf
#в начало файла вставляем
openssl_conf = openssl_def
#в конец файла вставляем
[openssl_def]
engines = engine_section

[engine_section]
gost = gost_section

[gost_section]
engine_id = gost
dynamic_path = /usr/lib/x86_64-linux-gnu/engines-1.1/gost.so
default_algorithms = ALL
CRYPT_PARAMS = id-Gost28147-89-CryptoPro-A-ParamSet

```

#### Code Block 12 Ubuntu: Установка Openssl +GOST

```

# Ставим Nginx +openssl(с поддержкой ГОСТ). На данный момент собраны
только для centos 7
cat << EOF > /etc/yum.repos.d/rtmis.repo
[rtmis]
name=rtmis repo
baseurl=https://yum-repo.rtmis.ru/centos/$releasever/$basearch/
gpgcheck=0
enabled=1
EOF

yum remove openssl
yum install -y nginx

```

#### Code Block 13 Centos: Установка Openssl +GOST

```

#Ключ gost_2020 показан для примера необходимо изменить в соответствии с
названием файла вашего ключа.
ssl_certificate /etc/nginx/ssl/gost_2020.crt;
ssl_certificate_key /etc/nginx/ssl/gost_2020.key;
ssl_session_timeout 5m;
ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
ssl_ciphers GOST2012-GOST8912-GOST8912:HIGH:MEDIUM;
ssl_prefer_server_ciphers on;

```

#### Code Block 14 ssl/promed.ssl

Для тестирования nginx с GOST 2012 можно использовать следующие сертификаты <https://paste.is-mis.ru/AZjP9tNz> (GOST2012\_256)  
 При использовании этих сертификатов сайт должен открываться только в браузерах chromiun-gost или sputnik



## 5 Важные настройки ЕЦП.МИС

Все файлы находятся в ЕЦП.МИС в папке promed/config.

В папке config создаем папку с названием региона, например, krasnoyarsk. Там можем повторять файлы конфигов из основной папки. Обязательно нужно повторить файл database.php. Если конфиг в основной и региональной папке совпадают, первоначально конфиг берется из региональной папки.

Пример конфигов можно взять по ссылке <https://git.promedweb.ru/dev/configs/postgre>

### config.php

```
// Список регионов
// По примеру добавляем имя региона латиницей (такова будет папка с именем
// региона в конфигах)
$config['regions'] = [
    0 => ['nick'=>'undefined', 'name'=>'Неопределенный', 'schema'=>'dbo'],
    2 => ['nick'=>'ufa', 'name'=>'Уфа', 'schema'=>'r2'],
];

// Параметр develop влияет на загрузку JS и CSS файлов, формирование
// ссылок на заглавной странице ЕЦП.МИС, а также на добавление текста ошибок
// при выполнении запросов к БД
// Если значение TRUE, то все JS и CSS файлы грузятся "как есть" (отдельно
// и без минимизации), ошибки БД выдаются на клиент вместе с запросами.
// Если значение FALSE, практически все JS и CSS файлы собираются в
// соответствующие engine-файлы. При наличии ошибок БД пользователь не видит
// тексты запросов.
$config['develop'] = TRUE;

// Режим API ЕЦП.МИС
// Допустимые значения: common, lis, rish
// 1) common - режим API для функционала, который еще не вынесен в
// микросервисы
// 2) lis - режим микросервиса ЛИС
// 3) rish - режим "монолита", должен использоваться для веб-серверов
// ЕЦП.МИС
$config['PROMED_MODULE'] = 'rish';

// Путь до логов
$config['log_path'] = '/srv/promed/promed/logs/';
```

### constants.php

```
// Включить пейджинг с помощью OFFSET / FETCH NEXT в запросах (не работает
// в старых SQL SERVER)
define("USE_NEW_SQL_PAGING", true);
```

### database.php

```

// настройки распределяются по группам
структура групп показана а примере группы default
// Default database
$db['default']['hostname'] = '';
$db['default']['username'] = '';
$db['default']['password'] = '';
$db['default']['database'] = '';
$db['default']['dbdriver'] = "postgre";
//$db['default']['port'] = '6432'; -----данное значение необходимо если
нужно изменить стандартный порт postgresql "5432" на другой
$db['default']['dbprefix'] = "";
$db['default']['pconnect'] = FALSE;
$db['default']['db_debug'] = TRUE;
$db['default']['cache_on'] = FALSE;
$db['default']['cachedir'] = "";
$db['default']['char_set'] = "utf-8";
$db['default']['dbcollat'] = "utf8_general_ci";
$db['default']['query_timeout'] = 6000;
$db['default']['schema'] = 'dbo';
$db['default']['context_info'] = true; // true - включить заполнение
контекста при каждом запросе в БД, false - выключить

// наряду с Default главная группа при использовании базы postgre
$db['postgres'] = $db['default'];

// реестровая База данных
$db['registry'] = $db['default'];

// База данных для формирования дерева отчетов
$db['reports'] = $db['default'];

// База данных для отчетов
$db['bdreports'] = $db['default'];

// База данных для поиска например поиска человека
$db['search'] = $db['default'];

// Архивная БД
$db['archive'] = $db['default'];

// База данных UserPortal , которая также используется у портала К-
врачу.
// то есть логин и пароль на данный момент берется все те же настройк из
default, заменяя толко имя базы данных.
$db['UserPortal'] = $db['default'];
$db['UserPortal']['database'] = "userportaltest";

// База данных портала. Первой страницы в ЕЦП.МИС. где хранятся новости,
объявления и т.д.
$db['portal'] = $db['default'];

// База данных портала. Первой страницы в ЕЦП.МИС. где хранятся новости,
объявления и т.д. , Но специально для Postgre
$db['portalPg'] = array_merge($db['postgres'], [
    'database' => 'portaltest',
]);

// база данных для хранения сессии. Да сессии хранятся всё равно в mongo,
но без нее не работает
$db['php_session'] = $db['default'];
$db['php_session'] = $db['postgres'];
$db['php_session']['database'] = "php_session_test";

// База для хранения Логов действий пользователей
$db['phplog'] = $db['default'];
$db['phplog']['database'] = "php_log";

```

```
$db['phplog']['schema'] = "dbo";

// База СМП версии 1
$db['smp'] = $db['default'];
$db['smp']['hostname'] = '';
$db['smp']['database'] = '';
$db['smp']['username'] = '';
$db['smp']['password'] = '';

//Электронные медицинские документы
$db['emd'] = $db['default'];
$db['emd']['database'] = 'emd';

//Лабораторные информационные системы (ЛИС)
$db['lis'] = $db['default'];
$db['lis']['database'] = 'promedlistest2';

// база для формирования реестров ЛВН
$db['registry_es'] = $db['registry'];
```

database.php должен быть в общей и региональной папке. Узнавать параметры БД у Администраторов БД.  
При использовании PostgreSQL у групп default, postgres, phplog необходимо прописать схему работы - **dbo**

**esia.php**

```

/**
 * Настройки для авторизации через ЕСИА
 */
$config['esia'] = [
    // Включить/отключить авторизацию через ЕСИА
    'enabled' => true,

    // Тип авторизации (esia - через портал госуслуг или egisz - через ИА
    ЕГИСЗ)
    'type' => 'egisz',

    // Путь к SAML обработчику ИА ЕГИСЗ
    'egisz_path' => 'https://ia-
test.egisz.rosminzdrav.ru/realms/master/protocol/saml',

    // Идентификатор ИС в ЕСИА
    'client_id' => '98089a68-972c-916e-c2c5-5ec939c71b5c',

    // Путь к РПМ для перенаправления с ЕСИА
    'redirect_uri' => 'https://prm.promedweb.ru/?c=IaEgisz&m=login', //
для ИА ЕГИСЗ
    // 'redirect_uri' => 'https://prm.promedweb.ru/?c=Esia&m=login', //
для портала госуслуг

    // Путь к РПМ для получения данных пользователя
    'redirect_uri_info' =>
'https://prm.promedweb.ru/?c=Esia&m=user_info&user_id=:user_id',

    // Путь к контроллеру ЕСИА для получения авторизационного кода
    'ac_path' => 'https://esia.gosuslugi.ru/aas/oauth2/ac',

    // Путь к контроллеру ЕСИА для получения маркера идентификации кода
    'te_path' => 'https://esia.gosuslugi.ru/aas/oauth2/te',

    // Область доступа для получения всех данных по пользователю
    'usr_scope' => 'fullname birthdate snils medical_doc',

    // Путь к REST контроллеру для получения данных по пользователю
    'usr_data_path' => 'https://esia.gosuslugi.ru/rs/prns/:user_id',

    // Путь к REST контроллеру для получения контактных данных по
    пользователю
    'usr_contacts_path' =>
'https://esia.gosuslugi.ru/rs/prns/:user_id/ctts?embed=(elements)',

    // Сертификат ИА ЕГИСЗ
    'egisz_cert' =>
'MIICmzCCAYMCBgFSPzj0bTANBgkqhkiG9w0BAQsFADARMQ8wDQYDVQQDDAZtYXN0ZXIwHhcNM
TYwMTE0MDgxOTAzWhcNMjYwMTE0MDgyMDQzWjARMQ8wDQYDVQQDDAZtYXN0ZXIwggEiMA0GCSq
GSIB3DQEBAQUAA4IBDwAwggEKAoIBAQDI1leonyVTyqVM/R+4+6RDPN110LaEPqNRC0pPWDEmA
ATuzsuDDjHaEzArbXsej30Yk+j1h6jVh3pLtfwYJyYnN926ZcQtSjpUHN+iEcNqIB+d3NHe+41
HPLvb8U19Y4TsndcWTj9tG1KpGDOZrtGBu3TYtSGOGaeLdEB5V1+ApOiKt1HeMwsEwdCr5YKc2
2KTQpicVDDHMOiQ7+B1Eje7o+wwCkQFpc6t103Lezhdt0azZH8V8DJ5NscVRxHcv/i7jZIFrY/
7ulBmhPfsIjxx2GtC/BEaG+MNw918a4JsToP2sCSA6LNGh+Bngp/nlAtH9zIz4xWsSsQQ2P+CI
6fPAGMBAAEwDQYJKoZIhvcNAQELBQADggEBAF4Ycgy/AvkTO11KtTjCcgoyCnMSUFQqnglwNVJ
vbln1WYGqkbAt+1Zl8GZk28HFbdAyOzIp0QXGksXTAPRHcGblOV5faMM8/YnrGDdH47HSkRWuB
ykZgEwW9wZ6Gqj2eLV/NNv030AiJriEWLXtdAQDffFbBmrJ3bGjyOj1KkkExz7mrFSRr2eLXwC
/PT3648RN6ay837+Q9ZL1UA1eWfcbxsRzgnzYHm4+iC6D7ulM3RlhY7Woem+RQE9+7nA3m4lrq
0df4u3Yc10SCQ2bV97kCAWv1btfsuo2JPeTFDhdhK8ALUZywz1c/mcLA2/+bdiMEQXw2yeOCjS
8oD8zBNw=',

    // Сертификат
    'cert' => '-----BEGIN CERTIFICATE-----
...
-----END CERTIFICATE----- ',

```

```
// Закрытый ключ
'key' => '-----BEGIN PRIVATE KEY-----
...
-----END PRIVATE KEY----- ',

// Пароль для закрытого ключа, если ключ не зашифрован, то параметр
удаляем
// 'key_pass' => '12345678',
);
```

**hooks.php**

```

// Начало логирования
$hook['pre_system'][] = [
    // Для логирования в БД SQL указать DBLog. Используется БД, указанная
    // в группе соединения phplog файла database.php
    // Для логирования в MongoDB указать DBMongoLog, но тогда нужно
    // настроить задание на запуск /?c=PhpLogService&m=transferDataFromMongoDB
    'class' => 'DBMongoLog',

    // Метод, начинающий выполнение логирования
    'function' => 'StartRequest',

    // Имя библиотеки для обработки запросов
    // Для логирования в БД SQL указать DBLog.php
    // Для логирования в MongoDB указать DBMongoLog.php
    'filename' => 'DBMongoLog.php',

    // Путь до библиотеки
    'filepath' => 'hooks',

    // Параметры (пусты для старта логирования)
    'params' => [],
];

// Конец логирования
$hook['post_system'] = [
    'class' => 'DBMongoLog',
    'function' => 'FinishRequest',
    'filename' => 'DBMongoLog.php',
    'filepath' => 'hooks',

    /**
     * В params могут быть переданы правила логирования. Если они пусты,
     * то логируются все обращения
     * Правила передаются массивом, в каждом правиле может быть задано
     * регулярной выражение для имени контроллера и имени метода (оба вместе или
     * каждый по отдельности)
     * В лог попадают только запросы, удовлетворяющие какому-либо из
     * правил
     * Пустое правило аналогично разрешению всего
     *
     * Пример логирования всех запросов к контроллеру EvnPL
     * [
     *     'controller' => '/^EvnPL$/i'
     * ]
     *
     * Пример логирования всех запросов, в которых метод содержит слово
     * person
     * [
     *     'method' => '/person/i'
     * ]
     */
    'params' => []
];

```

**mongodb.php**

**mongodblog.php**

**mongodbsessions.php**

```

// Адрес сервера MongoDB
$config['mongo_host'] = "127.0.0.1";

// Порт
$config['mongo_port'] = 27017;

// Наименование БД
$config['mongo_db'] = "db";

// Данные учетной записи для подключения к MongoDB
// Оставить пустыми, если MongoDB запущена не в режиме auth
$config['mongo_user'] = "";
$config['mongo_pass'] = "";

// Использование постоянного соединения
$config['mongo_persist'] = TRUE;
$config['mongo_persist_key'] = 'ci_mongo_persist';

// Формат результатов запроса
// 1) array - массив
// 2) object - объект
$config['mongo_return'] = 'array';

// When you run an insert/update/delete how sure do you want to be that
the database has received the query?
// safe = the database has received and executed the query
// fync = as above + the change has been committed to harddisk <- NOTE:
will introduce a performance penalty
$config['mongo_query_safety'] = 'w';

// Suppress connection error password display
$config['mongo_suppress_connect_error'] = TRUE;

// If you are having problems connecting try changing this to TRUE
$config['host_db_flag'] = FALSE;

// Поля в нижнем регистре
$config['fields_underscore'] = TRUE;

```

portal.php

```

/**
 * Конфигурация портала
 */

// Пользователи портала
$config['users'] = [
    [
        'username' => 'admin',
        'password' => '*****',
    ]
];

// Телефоны службы техподдержки
$config['phones'] = [];

// Разные ссылки
$config['links'] = [

    // Справочная система
    'promedhelp' => '',

    // Форум поддержки
    'forum' => '',

    // Ссылка на статью с описанием последних изменений
    'lastupdates' => '',
];

// Название разворачиваемого продукта
$config['titles'] = [
    'main_title' => 'ЕЦП',
    'main_page_1' => 'ЕЦП',
    'main_page_2' => 'Единая цифровая платформа',
    'RIAMS' => 'ЕЦП',
    'auth_page_enter' => 'Вход в ЕЦП',
];

// Разные настройки
$config['settings'] = [

    // Количество новостей на главной странице
    'news_on_main_page' => 0,

    // Количество новостей на страницах архива новостей
    'news_on_page' => 20,

    // Максимальное количество строк в новости, после которого новость
    // надо обрезать
    'news_body_cut_lines' => 15,

    // Время хранения в PHP-кэше (в секундах)
    'cache_time' => 300,
];

// Основные сервисы
$config['products'] = [
    'promed' => [
        'title' => 'ЕЦП',
        'description' => 'Единая цифровая платформа',
        'icon' => '/img/portal/icon-promed.png',
        'url' => '?c=portal&m=promed',
    ],
];

```



**promed.php**

```

// Путь к папке с log-файлами ЕЦП.МИС
define("PROMED_LOGS", APPPATH . '/logs');

// Java нужна для плагина добавляющего PDF для нанесения метки что
документ подписан
$config['JAVA_PATH'] = '/usr/bin/java';

// Адрес LDAP
define("LDAP_SERVER", "192.168.37.3");
define("LDAP_SERVER_PORT", 389);
define("LDAP_DOMAIN", "dc=swan,dc=perm,dc=ru");
define("LDAP_GROUP_PATH", "ou=Groups,LDAP_DOMAIN");
define("LDAP_USER_PATH", "ou=Users,LDAP_DOMAIN");
define("LDAP_ADDRBOOK_PATH", "ou=Books,LDAP_DOMAIN");
define("LDAP_USER", "cn=admin,LDAP_DOMAIN");
define("LDAP_PASS", "amokkamokk");

// SMTP хост
// Если SSL или TLS соединение, то следует писать ssl://сервер или
tls://сервер
define("SMTP_HOST", "mx.swan-it.ru");
define("SMTP_PORT", "25");
define("SMTP_TIMEOUT", "5");
define("SMTP_EMAIL", "info@swan-it.ru");
define("SMTP_EMAIL_PASS", "");
define("SMTP_EMAIL_DESCR", "ЕЦП");

// Включение/выключение оповещения через СМС пользователей портала при
записи/отмене записи в ЕЦП.МИС
$config['USER_PORTAL_IS_ALLOW_NOTIFY_ABOUT_RECORD_CANCEL'] = TRUE;

// Отправлять уведомления пользователям портала самозаписи
$config['USER_PORTAL_NOTIFICATION'] = true;

// Адрес Промед для использования в BIRT при вставке в печатные формы
штрих-кодов и иных изображений, генерируемых ЕЦП.МИС
$config['PromedURL'] = 'http://192.168.37.3:2080';

// Отладочный режим
// 1 - включен, 0 - выключен
$config['IS_DEBUG'] = "0";

// Отображать статистику на портале
$config['SHOW_PORTAL_STATS'] = FALSE;

// Параметры подключения к сервису онлайн идентификации
$config['IDENTIFY_SERVICE_URI'] =
"http://11.0.0.4/LPUWebServices/LPU_WebService.asmx";
$config['IDENTIFY_SERVICE_PORT'] = 3455;
$config['IDENTIFY_SERVICE_LOGIN'] = "";
$config['IDENTIFY_SERVICE_PASS'] = "";

// Признак необходимости проверять прикрепление, если с момента последнего
прикрепления не прошел год
$config['CHECK_ATTACH_IF_YEAR_EXPIRE'] = false;

// Признак необходимости производить проверку на двойников
$config['CHECK_PERSON_DOUBLES'] = true;

// Список кодов гинекологических специальностей (пример для Уфы, для
конкретного региона вместо ufa подставить соответствующее наименование)
$config['ufa']['GIN_LSP_CODE_LIST'] = [ 522, 540, 622, 640, 822, 840 ];

// Список кодов стоматологических специальностей (пример для Уфы, для
конкретного региона вместо ufa подставить соответствующее наименование)

```

```

$config['ufa']['STOM_LSP_CODE_LIST'] = [ 526, 527, 528, 529, 530, 559,
560, 561, 562, 626, 627, 628, 629, 630, 659, 660, 661, 662, 826, 827, 828,
829, 830, 859, 860, 861, 862 ];

// Максимальное время ожидания отчёта
$config['ReportMaxExecutionTime'] = 1800;

// Время кэширования отчёта, мин.
// В течение этого времени, сформированный тем же пользователем, с теми же
параметрами отчёт не будет формироваться заново, а будет браться готовый с
сервера
$config['ReportCacheTime'] = 5;

// Массив дополнительных армов на СМП сервере
$config['ADDITIONAL_SMP_ARMS'] = ['mstat'];

// Возможность добавлять карты СМП из АРМ мед. статистика
$config['MedStatAddCards'] = true;

// Настройки NODE JS
define('NODE_ENABLED', false);
define('NODEJS_SERVER_HOSTNAME', '127.0.0.1');
define('NODEJS_HTTPSERVER_PORT', '9900');
define('NODEJS_SOCKETSERVER_PORT', '9999');

// NODE JS прокси для портала
// 1. для имени хоста обязательно указывать протокол (http:// или
https://) в зависимости от того, в каком режиме запущен NODE-SERVER (можно
посмотреть при старте сервера или в логах)
// 2. номера портов дублируются - это нормально, т.к. сокет сервер теперь
инициализируется через порт веб-сервера
// 3. если блок не закомментирован. Обязателен к разворачиванию сервис
node-portal-
proxy(https://confluence.rtlabs.ru/pages/viewpage.action?pageId=253706886)
define('NODEJS_PORTAL_PROXY_HOSTNAME', 'http://127.0.0.1');
define('NODEJS_PORTAL_PROXY_HTTPPORT', '7070');
define('NODEJS_PORTAL_PROXY_SOCKETPORT', '7070');
// включить\выключить подключение к ноду-портала из клиента
define('NODEJS_PORTAL_ENABLE', true);

define('NODEJS_SMP_HTTPSERVER_PORT', '8800');
define('NODEJS_SMP_SOCKETSERVER_PORT', '8888');
define('NODEJS_PROMED_SOCKET_PORT', '9995');
define('NODEJS_CONTROL_ENABLE', true);

define('NODEJS_VIDEOCHAT_SOCKET_HOST', 'https://promed.promedweb.ru');
define('NODEJS_VIDEOCHAT_SOCKET_PORT', '9991');
define('NODEJS_VIDEOCHAT_ENABLE', true);
define('NODEJS_VIDEOCHAT_ICE_SERVERS', ' [{
    "urls": "stun:stun.l.google.com:19302"
} ]');

// Путь к справке
define("CONFLUENCE_PATH", "https://confluence.rtlabs.ru/display/main/");
define("CONFLUENCE_AUTH_PATH",
"https://confluence.rtlabs.ru/?os_authType=basic&os_username=users_perm&os
_password=q1W@e3R$");
define("WIKI_PATH", "https://192.168.36.64/wiki/perm/wiki/");

// Путь к бирту
define("BIRT_SERVLET_PATH", "http://perm.swn.local/birt-viewer/");
define("BIRT_SERVLET_PATH_ABS", "http://192.168.37.44:8080/birt_perm/");

// Включить поддержку картридера
define("CARDREADER_IS_ENABLE", true);

```

```

// Включить логин по ЭЦП
define("ECP_IS_ENABLE", true);

// Путь до локального вебсервиса
define("LOCALPHP", "http://perm.promed/");

// Экспорт данных
// Определение путей для экспорта всех данных в ЕЦП.МИС

// Корневой каталог для экспорта
define("EXPORTPATH_ROOT", "export/");
// Каталог для экспорта штатного расписания ФРМП
define("EXPORTPATH_STAFF", EXPORTPATH_ROOT."staff/");
// Каталог для экспорта регистра по орфанным заболеваниям
define("EXPORTPATH_ORPHAN", EXPORTPATH_ROOT."orphan_register/");
// Каталог для экспорта реестров
define("EXPORTPATH_REGISTRY", EXPORTPATH_ROOT."register_files/");
// Каталог для экспорта прикрепленного населения
define("EXPORTPATH_ATTACHED_LIST", EXPORTPATH_ROOT."attached_list/");
// Каталог для экспорта реестра медработников
define("EXPORTPATH_MEDPERSONAL_LIST",
EXPORTPATH_ROOT."medpersonal_list/");
// Каталог для экспорта МЭС
define("EXPORTPATH_MES", EXPORTPATH_ROOT."mes_files/");
// Каталог для экспорта регистра по ДД
define("EXPORTPATH_DD", EXPORTPATH_ROOT."dd_files/");
// Каталог для экспорта регистра по детям сиротам
define("EXPORTPATH_DO", EXPORTPATH_ROOT."do_files/");
// Каталог для экспорта ТАП
define("EXPORTPATH_PL", EXPORTPATH_ROOT."pl_files/");
// Каталог для экспорта КВС
define("EXPORTPATH_PS", EXPORTPATH_ROOT."ps_files/");
// Каталог для экспорта картотеки
define("EXPORTPATH_PC", EXPORTPATH_ROOT."pc_files/");
// Каталог для экспорта подписанных документов
define("EXPORTPATH_SIGNED", EXPORTPATH_ROOT."signed_files/");
// Каталог для экспорта для qwerty и реестра фонда из штатного расписания
ЛПУ
define("EXPORTPATH_LPU_STAFF_QWERTY_REG_FOND",
EXPORTPATH_ROOT."lpu_staff_qwerty_reg_fond_files/");
// Каталог для экспорта DBF файлов для министерства труда
define("EXPORTPATH_LABOR", EXPORTPATH_ROOT."labordep_files/");
// Каталог для экспорта DBF файлов для ВСМЭ
define("EXPORTPATH_BSME", EXPORTPATH_ROOT."bsme_files/");
// Каталог для хранения электронных медицинских документов
define("EXPORTPATH_EMD", EXPORTPATH_ROOT."emd_files/");
// Каталог для экспорта PDF файлов
define("EXPORTPATH_PDF_PRINT", EXPORTPATH_ROOT."pdf_print/");
// Каталог для экспорта коечного фонда (для Самары)
define("EXPORTPATH_HOSPITAL_BED_FOND",
EXPORTPATH_ROOT."hospital_bed_fond_files/");
// Каталог для экспорта лотов (unit of trading)
define("EXPORTPATH_UOT", EXPORTPATH_ROOT."uot_files/");
// Каталог для экспорта цен на ЖНВЛП
define("EXPORTPATH_JNVLP_PRICE", EXPORTPATH_ROOT."jnvlp_price_files/");
// Каталог для экспорта сводных заявок
define("EXPORTPATH_CONSOLIDATED_REQUEST",
EXPORTPATH_ROOT."consolidated_request_files/");
// Каталог для экспорта содержимого регистра остатков
define("EXPORTPATH_OSTAT_REGISTRY",
EXPORTPATH_ROOT."ostat_registry_files/");

// Корневой каталог для импорта файлов
define("IMPORTPATH_ROOT", "uploads/");

```

```

// Корневой каталог для формирования файлов отчётов ФЛК
define("DIRECTORYPATH", IMPORTPATH_ROOT . "Directory_files/");
// Корневой каталог для файлов пользователей в рамках профилей
define("USERSPATH", IMPORTPATH_ROOT . "users/");
// Корневой каталог для хранения файлов организаций (в т.ч. МО)
define("ORGSPATH", IMPORTPATH_ROOT . "orgs/");
define("ORGSPHOTO_PATH", ORGSPATH . "photos/");
// Корневой каталог для прикрепленных к сообщениям файлов
define("FILESPATH", IMPORTPATH_ROOT . "messages/");
// Корневой каталог для изображений препаратов
define("DRUGSPATH", IMPORTPATH_ROOT . "drugs/");
// Корневой каталог для документов EvnMediaFiles
define("EVNMEDIAPATH", IMPORTPATH_ROOT . "evnmedia/");
// Корневой каталог для документов pmMediaData
define("PMMEDIAPATH", IMPORTPATH_ROOT . "pmmmedia/");

// ?
define("VALIDATE_ROOT", "validate");

// Корневой каталог сохранения файлов отчётов
define("REPORTPATH_ROOT", "export/reports/");

// Путь до проксирующего контроллера отчётов
define("REPORT_CONTROLLER", '/?c=ReportRun&m=Run');

// Параметры БД для хранения сессий
define('USE_DBSESSIONS', FALSE);
define('SESSIONDB_HOSTNAME', "192.168.36.61");
define('SESSIONDB_USER', NULL);
define('SESSIONDB_PASS', NULL);
define('SESSIONDB_DBNAME', "PHP_session");
define('SESSIONDB_TABLE', "PHPSessions");

// Портал "К врачу" (1 - старый, 2 - новый)
define('KVRACHU_TYPE', "2");

// База с пользователями ЭР
define('ACCDATABASE', "UserPortal");

// E-mail, с которого отправляются уведомления
define('KVRACHU_MAIL', "reg@k-vrachu.ru");
define('KVRACHU_MAIL_PASS', "****");
define('KVRACHU_MAIL_DESCR', "Система уведомлений");

// Путь до сообщения об отсутствии Gears (устарело)
define('GEARS_WARNING', "/?c=portal&m=article&id=3");

// Путь к шрифтам для FPDF
define('FPDF_FONTPATH', APPPATH . 'libraries/fpdf16font/');

// Включить логирование в textlog
define('DOLOG', TRUE);

// Путь к PACS Промед
define('PROMED_PACS_AETITLE', 'DCM4CHEE');
define('PROMED_PACS_IP', '192.168.36.159');
define('PROMED_PACS_PORT', '11112');
define('PROMED_PACS_WADOPORT', '8080');
define('PROMED_PACS_HTTPS_WADOPORT', '8443');

// PACS сервис
define('PACS_SERVICE_IP', '192.168.36.226');
define('PACS_SERVICE_PORT', '8080');
define('PACS_SERVICE_NAME', 'DCMWebService');

// STOMP-сообщения

```

```

// Требуется подробное описание
define('STOMP_MESSAGE_ENABLE', FALSE);
define('STOMP_MESSAGE_SERVER_URL', 'tcp://192.168.36.208:61613');
define('STOMP_MESSAGE_DESTINATION_RULE', '/queue/ru.swan.fer.Rule');
define('STOMP_MESSAGE_DESTINATION_SLOT', '/queue/ru.swan.fer.Slot');
define('STOMP_MESSAGE_TIMEOUT', 2);

// Очередь сообщений
define('STOMPMQ_MESSAGE_ENABLE', TRUE);
define('STOMPMQ_MESSAGE_SERVER_URL', 'tcp://192.168.37.3:61613');
define('STOMPMQ_MESSAGE_DESTINATION_RULE', '/queue/ru.swan.directory');
define('STOMPMQ_MESSAGE_DESTINATION_SLOT', '/queue/ru.swan.person');
define('STOMPMQ_MESSAGE_TIMEOUT', 2);

// Служба проверки подписи JAX-RS
define('SIGN_SERVICE_URL', 'http://192.168.37.3:8081/service/api');

// Настройки подключения (логин и пароль) к ЛИС для получения справочников
define('LIS_LOGIN', "admin");
define('LIS_PASSWORD', '***');
define('LIS_CLIENTID', '***');

// Путь до портала К-врачу
define('KVRACHU_URL', 'https://swantest.k-vrachu.ru');

// Настройка сервиса подписания ЭЛН для ФСС
$config['EvnStickServiceEncryption'] = true;
$config['EvnStickServiceEncryptionCert'] =
'documents/FSS_TEST_CERT_2020_GOST2012.pem';
if ($config['EvnStickServiceEncryption']) {
    $config['EncryptionServiceUrl'] =
'http://192.168.37.32:8180/jcpCipher/CipherServiceImplService?WSDL';
    $config['EvnStickServiceUrl'] = 'https://docs-
test.fss.ru/WSLnCryptoV11/FileOperationsLnPort?WSDL';
} else {
    $config['EvnStickServiceUrl'] =
'http://193.148.44.24/FSSWSLn/FileOperationsLnPort?WSDL';
}

// Путь до openssl с поддержкой GOST-2012-256 нужен дя проверки хэша и
// проверки подписания
$config['OPENSSL_PATH'] = '/usr/bin/openssl';
$config['OPENSSL_CONF'] = '/etc/ssl/openssl.cnf';
$config['RHASH_PATH'] = '/usr/bin/rhash';

// Идентификатор пациента из ФЭР
$config['FER_PERSON_ID'] = 3615838;

// Настройки Wialon
define('WIALON_API_URL', 'http://wialon.03.perm.ru:8026/ajax.html');
define('WIALON_IMAGE_HOST', 'http://wialon.03.perm.ru:8022');

// Настройки сервиса ЕГИССО
define('EGISSO_REST_URL',
'https://promed.promedweb.ru/PortalGateService/webresources/rest/execJson'
);

// Настройки СППР
$config['DSS_API_URL'] = 'http://192.168.37.3:2071';

// Список МО, которым доступны новые АРМ врача поликлиники и ЭМК
$config['LPU_LIST_WITH_ALLOWED_EXTJS6_ARMS'] = [
    10010833 => [
        'LpuSection' => [ 99560017734 ] // список отделений, доступны
        новые АРМ врача поликлиники и ЭМК
    ],
];

```

```

10010833 => [],
13002730 => [],
13002737 => [],
10010817 => [],
9990000183 => [],
150185 => [],
9990000166 => [],
13002362 => [],
9990010223 => [],
];

// Загрузка ExtJS 6 без ExtJS 2.
$config['USE_EXTJS6_ONLY'] = false;

// Настройка подключения к БД для проверки учетных документов на вхождение
в реестры
$config['RegistryChecksDBConnection'] = 'default';

// Список электронных очередей, для которых должна использоваться
нелинейная схема обслуживания его ПО
$config['NON_LINEAR_ELECTRONIC_QUEUE_LIST'] = [ 92 ];

// Минимальное время предложения места в ЭО
$config['MinTimeQueue'] = 2;

// Базовая часть для ссылок в гридах Справочник инфоматов и Справочник
электронных табло
$config['infomat_base'] = 'https://swantest.k-vrachu.ru';

define('USE_POSTGRESQL', 'TRUE');

define('USE_POSTGRESQL_LIS', 'TRUE');

// Используем ли БД PostgreSQL
$usePostgre = false;
// Используем ли БД PostgreSQL для ЛИС
$usePostgreLis = false;

// Для тестовых серверов можем включать/выключать использование БД
PostgreSQL через COOKIE
if (isset($config['IS_DEBUG']) && $config['IS_DEBUG']) {
    if (isset($_COOKIE['usePostgre'])) $usePostgre =
    $_COOKIE['usePostgre'];
    if (isset($_COOKIE['usePostgreLis'])) $usePostgreLis =
    $_COOKIE['usePostgreLis'];
}

define('USE_POSTGRESQL', filter_var($usePostgre,
FILTER_VALIDATE_BOOLEAN));
define('USE_POSTGRESQL_LIS', filter_var($usePostgreLis,
FILTER_VALIDATE_BOOLEAN));

// Библиотека для аудита
defined('USERAUDIT_LIBPATH') || define("USERAUDIT_LIBPATH", APPPATH .
(USE_POSTGRESQL ? 'libraries/pgsql/UserAudit.php' :
'libraries/UserAudit.php'));

// Включить использование базы emd для хранения электронных медицинских
документов
$config['EMD_ENABLE'] = true;

// Отключение проверки сертификата
$config['EMD_DISABLE_CERT_CENTER_CHECK'] = true;

// Настройка сервиса интеграции с ТФОМС
$config['TFOMSAutoInteract'] = [

```

```

    'host' => '192.168.37.43',
    'port' => '5672',
    'user' => 'admin',
    'password' => 'admin',
    'publisher_queues' => [
        'common' => 'test_input',
        'answer' => 'test_answer',
    ],
    'consumer_queues' => [
        'common' => 'test_output',
        'answer' => 'tfomsToPromed.answer',
    ],
    'init_date' => '2019-01-01',
    'read_only' => false,
    'allowed_lpus' => [],
];

// Данные RabbitMQ для взаимодействия с ИС "Профилактика"
$config['ServiceISProf'] = [
    'host' => '45.139.16.125', // хост
    'port' => '55672', // порт
    'user' => 'promed_svc', // логин
    'password' => '***', // пароль
    'vhost' => 'Promed', // VirtualHost
    'request_queue' => 'Income' // название очереди
];

// ?
$config['ExchInspectPlan'] = $config['TFOMSAutoInteract'];

// Доступ к ПАК НИЦ МБУ
$config['MBU'] = [
    'apiurl' => 'http://194.0.219.199/pdkdl/api/fhir/', // адрес api
    'timeout' => 600
];

// ?
$config['enableSmsTalonCode'] = 'true';

// Список идентификаторов МО (справочник Lpu), являющихся психиатрическими
$config['PsychoLpuList'] = [
    13004374, // ПЕРМЬ ККПБ (новая)
    13002453,
    10010868,
    150225,
    150229,
    150301,
    150209,
    150221,
    150059,
    10010987,
    13002446,
    13002528,
    10010884,
    10011058,
    13002676,
    13002690,
    13002684,
    13002683,
    13002692,
    13002697,
    13002687,
    13002686,
];

// Ключ API для Yandex Maps API

```



```

define('YANDEX_API_KEY', '');

// Тип используемого кэша
$config['cache']['use'] = 'mongo';

// Настройка жизни кэша в секундах для некоторых кэшей (чтобы не лазить в
код при необходимости изменения)
$config['cache']['ttl'] = [
    //'LpuBuildingList' => 3600, // на час
    //'LpuSectionList' => 1800,
    //'LpuUnitList' => 3600,
    //'LpuSectionWardList' => 3600,
    //'MedServiceList' => 1800,
    //'MedStaffFactList' => 1800
];

// ?
$config['minimal'] = true;

// Настройки для АС МЛО
$config['asmlo_server'] = '192.168.37.3:8880/interface'; // сервер
$config['asmlo_timeout'] = 100000; // таймаут в секундах
$config['asmlo_login'] = 'admin'; // логин
$config['asmlo_password'] = '***'; // пароль

// Настройки использования архивной БД
$config['archive_database_enable'] = false;
$config['archive_database_date'] = '2014-01-01';

// Экспорт MSEExp
$config['MSEExp'] = [
    'host' => '192.168.36.58',
    'port' => '21',
    'user' => 'test',
    'password' => 'test',
    'export_folder' => 'export',
    'import_folder' => 'import',
];

// Блокировка медленных функций
$config['blockSlowDownFunctions'] = false;
$config['blockedSlowDownFunctions'] = [
    'search' => ['exportsearchresultstodbf'],
    'personcard' => ['exportpctodbf', 'loadattachedlist'],
    'vaccinectl' => [], // если не указаны методы блокируется весь
контроллер
    'vaccine' => [],
    'vaccine_list' => [],
];

// Для контроллера Search можно заблокировать некоторые виды поиска,
например, РПН: Поиск (SearchFormType = 'PersonCard')
$config['blockedSlowDownSearchFormTypes'] = [
    'personcard'
];

// ?
$config['PMU'] = [
    'xmlurl' => 'http://logs.promedweb.ru/web04/web/passport/passport-
xml',
    'replicatorurl' => 'http://passport.promedweb.ru/PassportReplicator',
];

// ?
$config['SwServiceLis'] = [
    'apiurl' => 'http://127.0.0.1:2080/api/lis',
];

```

```

        'timeout' => 30,
        'name' => 'ЛИС',
        'nick' => 'lis',
    ];

$config['SwServiceCommon'] = [
    'apiurl' => 'http://127.0.0.1:2080/api/common',
    'timeout' => 30,
    'name' => 'Общий функционал',
    'nick' => 'common',
];

// ?
$config['dbReplicatorQueue'] = [
    'enable' => false,
    'host' => '10.159.8.44',
    'port' => '61613',
    'login' => 'sabirov',
    'password' => '***',
    'destPrefix' => '/topic/dbReplicator.VirtualTopic.',
    'timeout' => 2,
];

// Признак сервера СМП
$config['IsSMPServer'] = false;

// FCM_KEY для push уведомлений
define('PUSH_NOTIFICATION_FCM_KEY', '***');

// Добавление test_id атрибута для авто-тестов.
$config['TEST_ID_ENABLED'] = 1;

// Количество записей обрабатываемых за один запуск задания по переносу
записей лога. По умолчанию 2000000
$config['PhpLogCountRun'] = 2000000;
// Количество записей, обрабатываемых за один шаг задания. По умолчанию
5000
$config['PhpLogCountStep'] = 5000;
// Время, через которое допустимо повторное выполнение переноса записей в
сек. По умолчанию 1 час в секундах
$config['PhpLogTimeToLive'] = 3600;

```

### Code Block 15 promed.php

### proxy\_queries.php

```
// Конфигурационные данные для хука ProxyQueries
$config['proxy_queries'] = [
    'enable' => false,

    'settings' => [
        'server' => '192.168.37.7', // https://perm.swn.local
        'http_port' => '80',
        'https_port' => '443',
        'timeout' => '5',
        'http_auth_user' => null,
        'http_auth_passwd' => null,
        'session_key' => 'proxyq',
    ],

    // Порядок проксирования
    'order' => 'allow,deny',

    // Запросы, которые не надо проксировать
    'deny' => [
        [ 'c' => 'promed', 'm' => 'getJSFile' ],
        [ 'c' => 'perm_promed', 'm' => 'getJSFile' ],
    ],

    // Запросы, которые всегда проксируются
    'allow' => [],
];
```

**session.php**

```

/**
 * Использовать дополнительный драйвер для работы с сессиями
 * Варианты:
 * false - хранение сессий как задано в настройках PHP
 * 'db' - хранение сессий в базе данных
 * 'mongodb' - хранение сессий в MongoDB
 */

$config['session_driver'] = 'mongodb';

// Вести журнал авторизаций в системе (true - вкл, false - выкл)
define('USERAUDIT_LOGINS', true);

// Обработчик хранения сессий в MONGODB
if ( isset($config['session_driver']) && $config['session_driver'] ==
'mongodb' ) {
    // Параметры БД возьмется из конфига, а название таблицы уточним здесь
    $config['mongodb_session_settings'] = array('table'=>'Session');

    // Подключаем библиотеку для работы с сессиями в МонгоДБ
    require_once(APPPATH.'libraries/MONGODB_SESSIONS.php');
    $mongodb_sessions = new MONGODB_SESSIONS($config); # создаем объект
}

// Стартуем!
if (!ini_get("session.auto_start"))
{
    if (!isset($_SESSION)) { // Проверяем не была ли уже создана
сессия, нужно если мы подключаем index.php из стороннего скрипта, например
генерим документацию
        // достаём параметр из REQUEST_URI, как это делается в
URI.php (который еще не загружен)
        if (!empty($_SERVER['REQUEST_URI'])) {
            $uri = parse_url('http://dummy' .
$config['REQUEST_URI']);
            $query = isset($uri['query']) ? $uri['query'] :
'';

            parse_str($query, $get);
            if (!empty($get['sess_id'])) {
                // устанавливаем текущий идентификатор
сессии
                session_id($get['sess_id']);
            }
        }
        session_start();
    }
} else {
    if ( isset($config['session_driver']) && $config['session_driver']
== 'db' ) {
        die('Нельзя использовать автозагрузку сессий при хранении
сессий в БД!');
    }
}

```

sms.php

```
/**
 * Настройки для отправки SMS
 */
$config['sms'] = [
    // Библиотека, через которую отправляем СМС
    'lib' => 'SMSTraffic',

    // Логин учетной записи для отправки СМС
    'login' => 'swan:test',

    // Пароль учетной записи для отправки СМС
    'password' => '***',

    // Отправитель
    'sender' => 'k-vrachu.ru'
];
```

**timetabletypes.php**

```

// Настройка отображения бирок в зависимости от типа
$config['types'] = [
    [
        'id' => 1,
        'name' => 'Обычная',
        'style' => 'background-color: #ddffdd;',
        'style_person' => 'background-color: #ffdddd;',
        'sources' => [ 1, 2, 3, 4, 5, 6, 7, 8, 9 ],
        'places' => [ 1, 2, 3 ],
    ],
    [
        'id' => 2,
        'name' => 'Резервная',
        'style' => 'background-color: yellow;',
        'style_person' => 'background-color: orange;',
        'sources' => [ 1, 2, 3, 4, 5 ],
        'places' => [ 1, 2, 3 ],
    ],
    [
        'id' => 3,
        'name' => 'Платная',
        'style' => 'background-color: #ff99ff;',
        'style_person' => 'background-color: #ff9999;',
        'sources' => [ 1, 2, 3, 4, 5 ],
        'places' => [ 1, 3 ],
    ],
    [
        'id' => 4,
        'name' => 'Бирка для ЦЗ',
        'style' => 'background-color: #66cccc;',
        'style_person' => 'background-color: #227777;',
        'sources' => [ 1, 2, 3, 4, 5, 7 ],
        'places' => [ 1 ],
    ],
    [
        'id' => 5,
        'name' => 'По направлению',
        'style' => 'background-color: #eebb88;',
        'style_person' => 'background-color: #aa9966;',
        'sources' => [ 1, 2, 3, 4, 5 ],
        'places' => [ 1, 3 ],
    ],
    [
        'id' => 6,
        'name' => 'Экстренная',
        'style' => 'background-color: #66ff00;',
        'style_person' => 'background-color: orange;',
        'sources' => [ 1, 2, 3, 4, 5 ],
        'places' => [ 2 ],
    ],
    [
        'id' => 7,
        'name' => 'Для врачей своей ЛПУ',
        'style' => 'background-color: #66ff00;',
        'style_person' => 'background-color: orange;',
        'sources' => [ 3 ],
        'places' => [ 1, 2, 3 ],
    ],
];

```

rest.php

```
// список, которые определяет какие данные нужно логировать в RESTLog_Y-m-  
d.log для каждого запроса  
$config['rest_log_params'] = ['uri', 'method', 'params', 'api_key',  
'ip_address', 'time', 'authorized'];
```

### **promed.conf (в nginx)**

Регион указываем в настройках nginx, например, krasnoyarsk:

```
fastcgi_param REGION krasnoyarsk;
```

## 6 Настройка видеосвязи

### 6.1 Разворачивание сервиса на nodejs. docker-compose.yml

```
version: '2'
services:
  nodevideochat:
    image: docker.promedweb.ru/node-video-chat:8
    container_name: node-video-chat
    restart: unless-stopped
    ports:
      - "9991:9991"
```

Code Block 16 docker-compose.yml

### 6.2 Настройки ЕЦП.МИС

#### 6.2.1 promed.php

```
define('NODEJS_VIDEOCHAT_SOCKET_HOST', 'https://ADDRESS');
define('NODEJS_VIDEOCHAT_SOCKET_PORT', '9991');
define('NODEJS_VIDEOCHAT_ENABLE', true);
define('NODEJS_VIDEOCHAT_ICE_SERVERS', ' [{
  "urls": "stun:stun.l.google.com:19302"
}, {
  "urls": "turn:turn.promedweb.ru:5349?transport=udp",
  "username": "admin",
  "credential": "admin999"
}]');
```

Code Block 17 promed.php

**ADDRESS** - необходимо поменять на тот адрес вашего ЕЦП.МИС по которому работает ваш видеочат сервер, в идеале по тому же адресу, что и работает сам ЕЦП.МИС.



## 6.3 Настройки портала

### 6.3.1 site.php - работа web портала

```
'nodejs_videochat_socket_host' => 'https://ADDRESS',
'nodejs_videochat_socket_port' => '9991',
'nodejs_videochat_enable' => true,
'nodejs_videochat_ice_servers' =>
' [{"urls": "stun:stun.l.google.com:19302"}, {"urls":
"turn:turn.promedweb.ru:5349?transport=udp",
"username": "admin", "credential": "admin999"} ]',

// примечание для эксплуатации: в конфиге два массива 'record' и это
НЕ вложенный в 'service' => array(...'record'), а отдельный массив
'record'
'record' => array(
    'videoChatIsOn' => true, // признак, что видеочат включен
    'timeForVideoChat' => 15, // время отведенное на консультацию в
минутах
    ...
    ...
);
```

Code Block 18 site.php

надо поменять ADDRESS

### 6.3.2 regions.php - работа через мобильное приложение

```
'perm' => array(
    'nodejs_videochat_socket_host' => 'https://ADDRESS',
    'nodejs_videochat_socket_port' => '9991',
    'nodejs_videochat_enable' => true,
    'nodejs_videochat_ice_servers' => array(
        array('urls' => 'stun:stun.l.google.com:19302'),
        array(
            'urls' => 'turn:turn.promedweb.ru:5349?transport=udp',
            'username' => 'admin',
            'credential' => 'admin999'
        )
    )
);
```

Code Block 19 regions.php

Данные настройки на пермском k-vrachi.ru делаются. первоначально все мобильные устройства обращаются именно к нему.  
надо поменять ADDRESS

## 6.4 Настройка балансировщика

video.conf

```

server {
    listen                9991 ssl;
    server_name          ADDRESS;
    include               ssl/ecp.ssl;
    access_log           /var/log/nginx/node_video_access.log;
    error_log            /var/log/nginx/node_video_error.log;
    open_file_cache      off;
    open_file_cache_errors

        location / {
            proxy_pass    http://ADDRESS_NODE:9991;
            include        params/nodejs.conf;
            proxy_set_header
                Allow-Origin *;
            Access-Control-
        }
    }
}

```

**Code Block 20 video.conf**

ADDRESS\_NODE - Адрес развернутого nodejs контейнера

### 6.4.1 nodejs.conf

```

proxy_connect_timeout 15;
proxy_send_timeout    60s;
proxy_read_timeout    600s;
proxy_http_version    1.1;
proxy_set_header      Upgrade
$http_upgrade;
proxy_set_header      Connection
$connection_upgrade;

```

**Code Block 21 params/nodejs.conf**

### 6.4.2 nginx.conf

```

map $http_upgrade $connection_upgrade {
    default upgrade;
    '' close;
}

```

**Code Block 22 nginx.conf**

## 6.5 Примечания

## 7 Настройка региональных параметров при разворачивании "коробочной" версии ЕЦП.МИС

Адыгея считается эталонным регионом для коробочной версии системы. При разворачивании нового региона нужно скопировать конфиги с региональными параметрами `parameters_client.php` и `parameters_server.php` из `promed/parameters/adygeya` в аналогичную папку для этого региона.

Дальнейшую настройку параметров региона можно выполнять следующим образом:

1. Найти системное название параметра в конфигурационном файле с параметрами по умолчанию `promed/parameters/parameters_client.php` и `promed/parameters/parameters_server.php`. По комментариям к параметрам определить нужный параметр.
2. Найти такой же параметр в региональном конфигурационном файле `promed/parameters/{регион}/parameters_client.php` и `promed/parameters/{регион}/parameters_server.php`. Если параметра в региональном файле нет, то скопировать его из файла с параметрами по умолчанию.
3. Изменить значение параметра нужным образом. Параметр может иметь значение одного из типов: `null`, булевый, числовой, строка или массив значений перечисленных типов.

Конфиги правятся эксплуатацией. Конфиги не добавляются в `.gitignore`.

Разработка после добавления новых параметров в задаче указывает их в поле "Действия при обновлении". Пример по КСГ/КСЛП/группировке в задаче <https://jira.is-mis.ru/browse/PROMEDWEB-21882>:

Добавить параметры в `promed/parameters/parameters_server.php`:

```
/* Расчет КСГ выполняется хранимой функцией */
$config['EvnSection_model_CalcKSGByStoredFunction'] = false;
/* Расчет КСЛП выполняется хранимой функцией */
$config['EvnSection_model_CalcKSLPByStoredFunction'] = false;
/* Расчет группировки выполняется хранимой функцией */
$config['EvnSection_model_CalcGroupByStoredFunction'] = false;
```

Добавить параметры в `promed/parameters/yanao/parameters_server.php`:

```
/* Расчет КСГ выполняется хранимой функцией */
$config['EvnSection_model_CalcKSGByStoredFunction'] = true;
/* Расчет КСЛП выполняется хранимой функцией */
$config['EvnSection_model_CalcKSLPByStoredFunction'] = true;
/* Расчет группировки выполняется хранимой функцией */
$config['EvnSection_model_CalcGroupByStoredFunction'] = true;
```

## 8 Настройки БД после развертывания.

После развертывания БД необходимо вручную провести следующие действия:

### 1. Установка значения региона.

Значение региона определяется процедурой GetRegion().

Для первой версии dbo.getregion необходимо установить возвращаемое значение кода региона.

```
CREATE OR REPLACE FUNCTION dbo.getregion()  
RETURNS integer  
LANGUAGE plpgsql  
AS $function$  
declare r integer;  
User_Name varchar;  
begin  
r:=1; --Установить значение кода региона.  
return r;  
end;  
$function$;
```

В случае если установлена вторая версия процедуры

```

CREATE OR REPLACE FUNCTION dbo.getregion()
  RETURNS integer
  LANGUAGE plpgsql
AS $function$
declare r integer;
      User_Name varchar;
begin
  --dbo.GetRegion версия 2.0
  r:=0;
  User_Name:=current_user;
  begin
    SELECT to_number(current_setting('Session.Region')) into r;
  EXCEPTION
  when others then r:=0;
  END;
  --RAISE notice 'Session.Region %', r;
  if r=0 then
    if User_Name like 'web_promed%' then
      User_Name:=regexp_replace(User_Name,'[^0-9]+',' ');
      if ISNUMERIC(User_Name) = 1 then
        r:=to_number(User_Name);

        end if;
      end if;
    --RAISE notice 'web_promed %', r;
    if r=0 then
      select region_id into r from pmuser_region where
pmuser_name=lower(User_Name);
      end if;
      --RAISE notice 'pmuser_region %', r;
      r:=coalesce(r,0);
      --RAISE notice 'coalesce %', r;
      if r=0 then
        r:=COALESCE(to_number(dbo.get_sysparametr('region')),0);
      end if;
      if r=0 then
        r:=59;
      end if;
      SELECT to_number(set_config('Session.Region', r::varchar, False)) into
r;
      --RAISE notice 'set Session.Region %', r;
      return r;
    end;
  $function$
;

```

то необходимо в таблице параметров настроить регион и сдвиг по времени для текущего региона

```

--Ввести [Код_региона]
INSERT INTO dbo.sysparametr
(sysparametr_id,sysparametr_tag,sysparametr_name,sysparametr_begdate,sysparametr_value,pmuser_insid,pmuser_updid)
values (2,'region','Код региона','1900-01-01 00:00:00'::timestamp,[Код_региона],1,1,'2020-09-07 10:56:42.511380'::timestamp)
ON CONFLICT ( sysparametr_id ) DO UPDATE SET
sysparametr_id=2,sysparametr_tag='region',sysparametr_name='Код региона',sysparametr_begdate='1900-01-01 00:00:00'::timestamp,sysparametr_value=' [Код_региона]',
pmuser_insid=1,pmuser_updid=1,sysparametr_insdt='2020-09-07 09:43:14.404389'::timestamp,

-- Ввести '[сдвиг_времени]' в формате '+HH:MI'
INSERT INTO dbo.sysparametr
(sysparametr_id,sysparametr_tag,sysparametr_name,sysparametr_begdate,sysparametr_value,pmuser_insid,pmuser_updid)
values (3,'timezone','Сдвиг времени','1900-01-01 00:00:00'::timestamp,' [сдвиг_времени]',1,1,'2020-09-07 10:56:49.774212'::timestamp)
ON CONFLICT ( sysparametr_id ) DO UPDATE SET
sysparametr_id=3,sysparametr_tag='timezone',sysparametr_name='Сдвиг времени',sysparametr_begdate='1900-01-01 00:00:00'::timestamp,sysparametr_value=' [сдвиг_времени]',
pmuser_insid=1,pmuser_updid=1,sysparametr_insdt='2020-09-07 10:36:30.535680'::timestamp,

```

### 1. Установка диапазона сиквенсов.

Для обеспечения глобальной не пересекаемости значений ID устанавливаем правила для вновь создаваемых БД

Начальные значения всех ID определяются следующим образом

0 000 00 00 XXX XXX XXX

| | | | | ----- Девять знаков. Приращиваемое значение  
| | | | ----- Два знака. Номер сервера по порядку (10 -PROD,20 -DEV,30-TEST,40-Release)  
| | |----- Два знака. Код группы/функциональной принадлежности (Человек - 1, ЛИС - 2, Поликлиника - 3, Стационар - 4, СМП - 5, Отчетный - 6, Реестры - 7, Резервный СМП - 8, Рабочая Промед - 9)  
| |-----Три знака. Код региона (Пермский Край- 059, Башкортостан -002, Крым 082 и тд. Для Казахстана кода вида 101,102 )  
|-----Один знак. Не заполняем

Для изменения диапазона сиквенсов используется данный скрипт.

```

do
$$
declare
sqltext varchar(1000);
rec record;
min_value varchar(20);
max_value varchar(20);
min_value_int varchar(10);
max_value_int varchar(10);
begin
    /* https://redmine.swan-
it.ru/projects/knowledgebase/wiki/Правила_формирования_ID

0 000 00 00  XXX XXX XXX
| | | | |
| | | | | ----- Девять знаков. Приращиваемое значение
значение
| | | ----- Два знака. Номер сервера по порядку
| | ----- Два знака. Код группы/функциональной
принадлежности ( Человек - 1, ЛИС - 2, Поликлиника - 3, Стационар - 4, СМП
- 5, Отчетный - 6, Реестры - 7 )
| ----- Три знака. Код региона (Пермский Край- 059,
Башкортостан -002, Крым 082 и тд. Для Казахстана кода вида 101,102 )
-----Один знак. Не заполняем
*/

    min_value:='0' || lpad(dbo.getregion()::text,3,'0') || '0101000000001';
    max_value:='0' || lpad(dbo.getregion()::text,3,'0') || '0101999999999';
    if dbo.getregion()<100 then
        min_value_int:= lpad(dbo.getregion()::text,3,'0') || '1010001';
    else
        min_value_int:= lpad(dbo.getregion()::text,3,'0') || '101001';
    end if;
    max_value_int:='2147483647';
    for rec in (select nspname,relname,pg_sequence.* from
pg_catalog.pg_sequence pg_sequence
    join pg_class on pg_class.oid=pg_sequence.seqrelid
    join pg_namespace on pg_namespace.oid=pg_class.relnamespace
    --and relname='descriptions_descriptions_id_seq'
    and (relname not like 'pmgen_%' or relname= 'pmgen_pmgen_id_seq')
    where seqmin<>cast(min_value as bigint) and seqmin<>cast(min_value_int
as bigint)
    --and seqmax>cast(min_value as bigint)
    order by 1,2) loop
        if rec.seqmax>2147483647 then
            if rec.seqmax<cast(min_value as bigint) then
                sqltext:= 'alter SEQUENCE
||rec.nspname||'.'||rec.relname|| ' MAXVALUE '||max_value||';';
                execute(sqltext);
            end if;
            if rec.seqmin>max_value::bigint then
                sqltext:= 'alter SEQUENCE
||rec.nspname||'.'||rec.relname|| ' MINVALUE 1 RESTART WITH 1';
                execute(sqltext);
            end if;
            sqltext:= 'alter SEQUENCE
||rec.nspname||'.'||rec.relname|| ' RESTART WITH '||min_value||';';
            sqltext:= sqltext||chr(13)||chr(10)||'alter SEQUENCE
||rec.nspname||'.'||rec.relname|| ' INCREMENT BY 1 MINVALUE
||min_value||' MAXVALUE '||max_value||' START '||min_value||' CACHE 1 NO
CYCLE;';
            execute(sqltext);
        else
            if rec.seqmax<cast(min_value_int as bigint) then

```

```

        sqltext:= 'alter SEQUENCE
'||rec.nspname||'.'||rec.relname|| ' MAXVALUE '||max_value_int||';';
        execute(sqltext);
    end if;
    if rec.seqmin>min_value_int::bigint then
        sqltext:= 'alter SEQUENCE
'||rec.nspname||'.'||rec.relname|| ' MINVALUE 1 RESTART WITH 1';
        execute(sqltext);
    end if;
    sqltext:= 'alter SEQUENCE
'||rec.nspname||'.'||rec.relname|| ' RESTART WITH '||min_value_int||';';
    sqltext:= sqltext||chr(13)||chr(10)||'alter SEQUENCE
'||rec.nspname||'.'||rec.relname|| ' INCREMENT BY 1 MINVALUE
'||min_value_int||' MAXVALUE '||max_value_int||' START '||min_value_int||'
CACHE 1 NO CYCLE;';
    execute(sqltext);
    end if;
end loop;
end;
$$

```

### 3. Чистка региональных значений.

- Очистка версий справочников:

```
delete from stg.localdbversion ;
```

### 4. Чистка региональных схем

В БД для каждого региона создаются отдельные схемы

- rN - для расчета реестров
- rptN - для хранения структур БД уникальных отчетов

где N - номер региона.

После развертывания БД необходимо удалить схемы чужих регионов, и при необходимости создать схемы для своего региона.



```

DO
$$
DECLARE

    r record;
    sql_text text;

BEGIN

    sql_text := 'CREATE SCHEMA IF NOT EXISTS ' ||
quote_ident('r' || dbo.getregion()::text) || ';';
    RAISE NOTICE '%', sql_text;
    EXECUTE sql_text;

    sql_text := 'CREATE SCHEMA IF NOT EXISTS ' ||
quote_ident('rpt' || dbo.getregion()::text) || ';';
    RAISE NOTICE '%', sql_text;
    EXECUTE sql_text;

    FOR r IN
        SELECT pn.nspname AS schema_name FROM pg_catalog.pg_namespace pn
        WHERE pn.nspname NOT IN
('r' || dbo.getregion()::text, 'rpt' || dbo.getregion()::text)
        AND ( pn.nspname ~ '^ (r|rpt) (\d) ' )
    LOOP

        sql_text := 'DROP SCHEMA ' || quote_ident(r.schema_name) || '
CASCADE;';
        RAISE NOTICE '%', sql_text;
        EXECUTE sql_text;

    END LOOP;

END;
$$;

```

Структура региональных схем разрабатывается отдельно для каждого региона и не входит в общий дамп.

## 5. Обновление БД

При необходимости провести обновления БД выполнив **Выгрузка diff в БД** [Liquibase](#)

## 6. Удаление из локального справочника региональных схем

```

delete from stg.regionallocaldblist where localdblist_id in (select
localdblist_id FROM stg.localdblist
where localdblist_schema like 'r%' and localdblist_schema!='rls');

delete FROM stg.localdblist
where localdblist_schema like 'r%' and localdblist_schema!='rls';

```

## 9 Памятка - установка "коробочной" версии в новые регионы ЕЦП.МИС

№	Задача	исполнитель	ожидаемый результат
1	выделение мощностей	РП + СМ + ОЭИР	
2	закупка лицензий КриптоПро - много процессорная по 1 шт на каждый сервис интеграции + 1 портал к врачу + 1 на тест	РП	
3	Завести Эпик в Жире для привязывания всех задач и проблем по тестированию стенда	РП, СМ	
4	подготовка серверов, без приклада	ОЭИР	настроены ВМ, даны доступы для ОЭИР + ДБА + ОТС + внедрение?
5	настройка СУБД postgres и mongoDB + предоставление прав доступа ОТС, ОФС, Внедрение, ЗЛП	ДБА	СУБД готовы к развертыванию эталонной БД
6	получить копию БД "коробки"	ДБА	Копия БД 192.168.36.24/mnt/backup/ (Если нет доступа обсудить с <a href="#">Климачев Алексей Леонидович</a> ) как забрать
7	Развернуть БД ЕЦП.МИС, emd	ДБА	должны быть базы: - основная Promed (Установить diffы до <b>diff20201002_0001.sql</b> ) - emd,php_log,usrer_portal,log_service
8	Переименовать схемы под текущий регион. и выполнить <a href="#">Настройки БД после развертывания.</a>	ДБА	должны быть схемы: - основная - региона - реестровая r<код_региона> - отчетная rpt<код региона>  Коды регионов <a href="#">Таблица утвержденных имен регионов и их кодов</a>
9	Настроить MongoDB <a href="#">Установка сервера хранения справочников, логов, кэша, сессий</a>	ДБА + ОЭИР	отсутствие ошибок в прикладе по обращению к БД

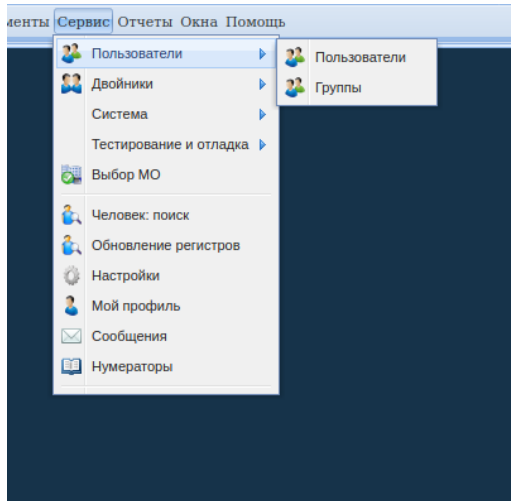
№	Задача	исполнитель	ожидаемый результат
1 0	Проверка наличия всех артефактов в БД. Загрузка недостающих данных в справочники, необходимых для начала ввода случаев.	ОТС + Биллинг	нужные схемы и справочники на месте можно воспользоваться скриптами для разливки данных <a href="#">Коробка</a> . <a href="#">Пакетная миграция данных Адыгеи на другие регионы</a> .
1 1	получить код для установки "коробки" и провести <a href="#">Настройка региональных параметров при разворачивании "коробочной" версии ЕЦП.МИС</a>	ЗЛП + ОЭИР	Актуальная код тут <a href="#">Промед Релиз Адыгея - еср-рр.1.0.25</a>
1 2	развернуть сервисы на продлайке	ОЭИР	код развернут на серверах и запускается без ошибок.
1 3	Настроить ПродЛайк	ОЭИР	в стенд можно зайти под УЗ администратора
1 4	установить и настроить отчеты из <a href="http://git.promedweb.ru/rtmis/report_pg">git.promedweb.ru/rtmis/report_pg</a>	ОЭИР	отчеты есть на стенде и запускаются  при необходимости создание папки для региона обратится к <a href="#">Лупандин Роман Владимирович</a>
1 5	Настроить загрузку отчетов на стенд	ОЭИР	
1 6	Завести УЗ в прод-лайке	Внедрение	
1 7	Подготовить чек лист для проверки по вводу случаев в систему, аналогично Адыгее <a href="https://confluence.rtlabs.ru/pages/viewpage.action?pageId=236076865">https://confluence.rtlabs.ru/pages/viewpage.action?pageId=236076865</a>	Биллинг	
1 8	Протестировать ввод случаев по чек-листу из предыдущего пункта и отметить проверенное.	Биллинг + Внедрение + ОФС	
1 9	Проверить отчёты	Внедрение + ОФС	
2 0	Проверить работу биллинга, формирование счетов реестров	Биллинг	
2 1	Развернуть БД на продуктиве	ДБА	
2 2	Развернуть сервисы на продуктиве	ОЭИР	

№	Задача	исполнитель	ожидаемый результат
2 3	Настроить Продуктивный контур	ОЭИР	
2 4	Проверить продуктивный контур	Внедрение + ОК	

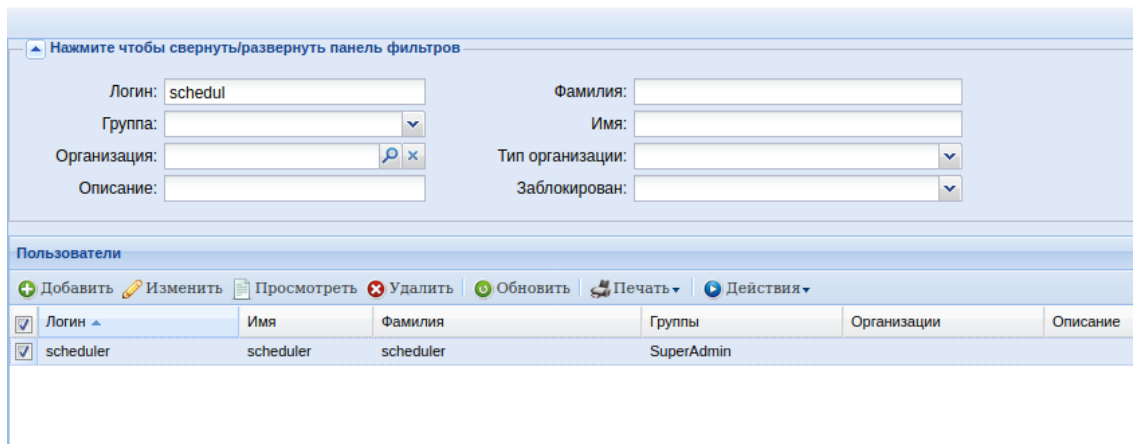
## 10 Получение токена авторизации.

Для создания пользователя необходима учетная запись с правами Супер Администратора СВАН.

Переходим к пользователям.



Проверяем есть ли пользователь scheduler, если пользователя нет то нажимаем добавить. Если пользователь существует, заходим в него.

A screenshot of the user management interface. At the top, there is a search form with fields for 'Логин' (Login), 'Группа' (Group), 'Организация' (Organization), 'Описание' (Description), 'Фамилия' (Surname), 'Имя' (Name), 'Тип организации' (Organization Type), and 'Заблокирован' (Blocked). Below the form, there is a table of users. The table has columns for 'Логин', 'Имя', 'Фамилия', 'Группы', 'Организации', and 'Описание'. A single user named 'scheduler' is listed with the group 'SuperAdmin'.

Логин	Имя	Фамилия	Группы	Организации	Описание
scheduler	scheduler	scheduler	SuperAdmin		

Для добавления пользователя вводим логин и пароль для этого пользователя. Добавляем в пользователю группа SuperAdmin (СуперАдминистратор СВАН)

Для генерации токена нажимаем кнопку "Сгенерировать", вводим дату меньше 19.01.2038 года. О проблеме можно почитать [тут](#)

Пользователь: Редактирование

1. Основное 2. Доступ к АРМ

Организация:

Наименование	Тип
Название не определено	

Логин:   Заблокирован Идент. МАРШа:

Временный пароль:

Токен:  До:

Сотрудник

Сотрудник:

Фамилия:  Полное имя:

Имя:  Эл. почта:

Отчество:  Описание:

Группы:

Группа	Описание
SuperAdmin	Супер Администратор СВАН

Количество параллельных сеансов:

Сохраняем пользователя.

Сгенерированный токен необходим для запуска сервисов

# 11 Порядок развертывания БД и действия до загрузки дампа

## 1. Порядок развертывания БД

Базы данных необходимо развертывать в следующем порядке:

- userportal
- promed
- php\_log
- portal
- emd

## 2. Внешние сервера

Необходимо прописать параметры внешних серверов

Для основной БД Promed внешний сервер на БД UserPortal

```
CREATE SERVER userportal
OPTIONS (
SET host 'IP сервера',
SET dbname 'userportal');
```

Для БД PHP\_Log внешний сервер на основную БД

```
CREATE SERVER promed
OPTIONS (
SET host 'IP сервера',
SET dbname 'promed');
```

наименование параметра dbname должно точно соответствовать наименованию сервера подключения, включая регистры.

Так же в случае смены паролей необходимо изменить обертки для внешних серверов

```
alter USER MAPPING FOR public
SERVER userportal
OPTIONS (
set password '*****',
set "user" 'locomotive');
```

Подставив соответствующих пользователя и пароль.

## 12 Региональный портал мед. услуг

Репозиторий РПМУ в Git: <https://git.promedweb.ru/rtmis/medservice>

На портале поддерживается старая и новая версия. Новые регионы работают только с новой версией.

Поддерживается стеки:

mssql + php 5.6 (текущие регионы)

postgre + php 7.2 (Адыгея)

Общие файлы находятся в папке application/config.

Региональная специфика учитывается в файлах, расположенных в папках, наименование которых соответствует региону. Например, buryatia\_new для новой версии портала и buryatia для старой версии.

### 12.1 Зависимости проекта

- [Промед\(ЕЦП.МИС\)](#)
- [Node Portal Proxy](#)

### 12.2 Развертывание с использованием docker-образа

#### 12.2.1 Необходимое ПО

- docker v19.03
- docker-compose 1.25

#### 12.2.2 Структура папок монтируемых на host-машину

/srv/promed/applicaion

├─ logs - папка с логами

├─ web.conf - файл настроек веб-сервера nginx

├─ config - папка с конфигами

└─ config/secrets - папка с файлом .env содержащим учетные данные для подключения к внешним системам

#### 12.2.3 Для развертывания необходимо

1. Создать папку развертывания
2. В корне папки развертывания создать файл .env ([пример](#) содержания ниже)
3. Создать папки **logs/nginx**, **logs/php**, **config**, **secrets**
4. Поместить в папку **config** необходимые конфигурационные файлы (<https://git.promedweb.ru/rtmis/medservice/-/tree/master/application/config>)
5. В файлах конфигурации пароли приведены к виду переменных. Пример:



```
'hostname' => '192.168.36.30',
    'database' => 'promedtest',
    'username' => "user_r59",
    // Значение из переменной если предустановленно иначе
    'password' => isset($_SERVER['DATABASE_DEFAULT_PASSWORD']) ?
$_SERVER['DATABASE_DEFAULT_PASSWORD'] : 'ПАРОЛЬ',
    'pooling' => TRUE,
```

#### Code Block 23 database.php

Переменные подставляются из файла `./secrets.env`. Предопределить пароли можно:

- через файл `./secrets.env`. (пример содержания ниже)
- изменением значения конфигурационного файла `isset($_SERVER['DATABASE_DEFAULT_PASSWORD']) ? $_SERVER['DATABASE_DEFAULT_PASSWORD'] : 'ПАРОЛЬ ПО УМОЛЧАНИЮ'` на `'ТЕКУЩИЙ ПАРОЛЬ'`. Пример:

```
'hostname' => '192.168.36.30',
    'database' => 'promedtest',
    'username' => "user_r59",
    /**
     * Значение предопределенное
     * старая строка 'password' =>
isset($_SERVER['DATABASE_DEFAULT_PASSWORD']) ?
$_SERVER['DATABASE_DEFAULT_PASSWORD'] : 'ПАРОЛЬ ПО УМОЛЧАНИЮ',
     */
    'password' => 'ТЕКУЩИЙ ПАРОЛЬ',
    'pooling' => TRUE,
```

#### Code Block 24 database.php

- либо изменением значения конфигурационного файла `$_SERVER['DATABASE_DEFAULT_PASSWORD'] : 'ПАРОЛЬ ПО УМОЛЧАНИЮ'` на `$_SERVER['DATABASE_DEFAULT_PASSWORD'] : 'ТЕКУЩИЙ ПАРОЛЬ'`. Пример:

```
'hostname' => '192.168.36.30',
    'database' => 'promedtest',
    'username' => "user_r59",
    /**
     * Значение предопределенное
     * старая строка 'password' =>
isset($_SERVER['DATABASE_DEFAULT_PASSWORD']) ?
$_SERVER['DATABASE_DEFAULT_PASSWORD'] : 'ПАРОЛЬ ПО УМОЛЧАНИЮ',
     */
    'password' => isset($_SERVER['DATABASE_DEFAULT_PASSWORD']) ?
$_SERVER['DATABASE_DEFAULT_PASSWORD'] : 'ТЕКУЩИЙ ПАРОЛЬ',
    'pooling' => TRUE,
```

#### Code Block 25 database.php

- При выборе предопределения паролей через `.env`. Поместить в папку `secrets` файл `.env` (<https://git.promedweb.ru/rtmis/medservice/-/tree/master/application/config/.env.example>) и заполнить ([пример](#) содержания ниже).
- Создать файл `web.conf` ([пример](#) содержания ниже)
- Установить на папки и файлы созданные в п.п. 3, 4, 5 владельца `www-data` (UID: 33, GID: 33)
- Создать в папке развертывания файл `docker-compose.yaml` ([пример](#) содержания ниже)

10. Заменить значение внешнего порта в секции **ports** с **2080** на необходимое значение внешнего порта сервиса
11. Проверить синтаксис конфигурации docker-compose командой: **docker-compose config**
12. Запустить сервис с помощью команды: **docker-compose up -d**
13. Настроить балансировщик ([пример](#) содержания ниже)

```
SYSPATH=/srv/promed/application
```

#### Code Block 26 .env

```
server {
    listen 80;
    root /srv/promed/;
    # Логирование в stdout контейнера
    #error_log stderr warn;
    #access_log /dev/stdout;
    access_log /srv/promed/application/logs/nginx/access.log;
    error_log /srv/promed/application/logs/nginx/error.log;

    location ~ /\.php$ {
        charset utf-8;
        try_files $uri = 404;
        include fastcgi_params;
        fastcgi_pass unix:/var/run/php/php7.2-fpm.sock;
        fastcgi_read_timeout 3600;
        fastcgi_send_timeout 3600;
        fastcgi_index index.php;

        # задать переменную окружения региона
        # для отладки - development
        # для прод - production
        fastcgi_param PORTAL_REGION perm;

        # Задать переменную окружения
        # для отладки - "development"
        # для прод - "production"
        fastcgi_param KOHANA_ENV development;

        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;

        #Для запуска без балансировщика с сертификатом комментируем
        параметры ниже
        fastcgi_param HTTPS on;
        fastcgi_param X-Forwarded-Proto https;
    }

    location / {
        charset utf-8;
        index index.php index.html index.htm;
        try_files $uri $uri/ /index.php$is_args$args;
    }
}
```

#### Code Block 27 web.conf

```

[www]
user = www-data
group = www-data

listen = /run/php/php7.2-fpm.sock
listen.owner = www-data
listen.group = www-data
pm = dynamic
pm.max_children = 460
pm.start_servers = 8
pm.min_spare_servers = 6
pm.max_spare_servers = 24
pm.process_idle_timeout = 60s;
pm.max_requests = 1536
env[TMP] = /tmp
env[TMPDIR] = /tmp
env[TEMP] = /tmp
php_flag[display_errors] = off
php_admin_value[error_log] = /srv/promed/application/logs/php/error.log
; Логгирование в stdout контейнера
;php_admin_value[error_log] = /proc/self/fd/2
php_admin_flag[log_errors] = on

php_admin_value[memory_limit] = 2048M
php_admin_value[date.timezone] = Europe/Moscow
php_admin_value[post_max_size] = 400M
php_admin_value[upload_max_filesize] = 400M
php_admin_value[max_execution_time] = 3600
php_admin_value[session.gc_maxlifetime] = 7200

```

**Code Block 28 php-fpm.conf**

```

# В комментариях название конфигурационного файла, для которого
предопределяется переменная.

#api.php
API_APIKEY=""

#auth.php
AUTH_LDAP_PASS=""

#captha.php
CAPTCHA_RECAPTCHA_SECRET=""
CAPTCHA_RECAPTCHA_SITEKEY=""

#database.php
DATABASE_DEFAULT_PASSWORD=""
DATABASE_ACCOUNTS_PASSWORD=""
DATABASE_DEFAULT_POSTGRES_PASSWORD=""
DATABASE_ACCOUNTS_POSTGRES_PASSWORD=""
DATABASE_EMD_PASSWORD=""
#DATABASE_MONGO_PASSWORD=""

#esia.php
ESIA_CRT=""
ESIA_KEY=""

#mail.php
MAIL_PASSWORD=""

#site.php
SITE_FCM_KEY=""
GENERAL_PURPOSE_PASSWORD=""
SWTOKEN=""

#sms.php
SMS_PASSWORD=""

#uec_auth.php
UEC_AUTH_MASTER_SECRET=""

```

**Code Block 29** ./secrets/.env

```

version: '3.0'

services:
  medservice:
    container_name: medservice
    image: d-repo.rtmis.ru/medservice/medservice:rpms-1-0-0
    environment:
      SYSPATH: ${SYSPATH}
    volumes:
      - ./logs:${SYSPATH}/logs
      - ./config:${SYSPATH}/config
      - ./secrets:${SYSPATH}/config/secrets
      - ./web.conf:/etc/nginx/conf.d/web.conf
    ports:
      - 2080:80

```

**Code Block 30** docker-compose.yaml

```

server {
    listen                80;
    server_name          promed.local;
    open_file_cache_errors off;
    access_log           /var/log/nginx/access.log;
    error_log            /var/log/nginx/error.log;

    location / {
        proxy_pass          http://medservice/;
        proxy_redirect      off;
        proxy_set_header    Host                $host;
        proxy_set_header    X-Real-IP
$remote_addr;
        proxy_set_header    X-Forwarded-For
$proxy_add_x_forwarded_for;
        proxy_set_header    X-Forwarded-Proto $scheme;
        proxy_set_header    Access-Control-Allow-Origin
*;
    }

    upstream medservice {
        least_conn;
        server medservice:80 weight=2 max_fails=3
    }

    fail_timeout=10s;
}

```

**Code Block 31** Пример конфигурации для LB Nginx.

## 12.3 Общие файлы конфигурации

### api.php

Настройки API

```

return array(
    // Ключ приложения, передаваемый при обращении к API
    'apiKey' => '44c8a6356a3e67ed38c9d9df0e6f8e57',
);

```

### auth.php

Настройки LDAP для авторизации пользователей в АРМ администратора холла, АРМ сотрудника пункта забора, АРМ сотрудника обр. учерждения

```

return array(
    'driver' => 'ORM',
    'hash_method' => 'md5',
    'hash_key' => NULL,
    'lifetime' => 1209600,
    'session_key' => 'auth_user',
    'LDAP_ROOT' => 'dc=swan,dc=perm,dc=ru',
    'LDAP_USER_PATH' => 'ou=Users,dc=swan,dc=perm,dc=ru',
    'LDAP_SERVER' => '192.168.37.3',
    'LDAP_SERVER_PORT' => 389,
    'LDAP_USER' => 'cn=admin,dc=swan,dc=perm,dc=ru',
    'LDAP_PASS' => 'amokkamokk',
);

```

## cache.php

### Настройки кэширования

```

return array(
    'file' => array(
        'driver' => 'file',
        'cache_dir' => APPPATH.'cache',
        'default_expire' => 3600,
        'ignore_on_delete' => array(
            '.gitignore',
            '.git',
            '.svn'
        )
    )
);

```

## captcha.php

### Настройки Captcha

```

return array(
    'default' => array(
        'style' => 'basic',
        'width' => 200,
        'height' => 50,
        'complexity' => 5,
        'background' => '',
        'fontpath' => MODPATH.'captcha/fonts/',
        'fonts' => array('DejaVuSerif.ttf'),
        'promote' => FALSE,
    ),
    'recaptcha' => array(
        'style' => 'recaptcha',
        'url' => 'https://www.google.com/recaptcha/api/siteverify',
        'secret' => '***',
        'sitekey' => '***',
        'DontShowCaptcha' => true,
    )
);

```

## demo.php

Конфигурация демонстрационных аккаунтов

```
return array(  
    'users' => array(  
        // Логин демонстрационной учетной записи  
        'demo',  
    ),  
);
```

## record\_restriction.php

Настройки механизма ограничения записи по больнице прикрепления человека

```

// ПОРЯДОК КЛЮЧЕЙ (0, 1, 2) ВАЖЕН

return array(
    'default' => array(
        /**
         * 3. Список профилей, к которым запрещена запись (безусловно)
         */
        0 => array(
            'code' => 'banned_profiles_list',
            'value' => FALSE,
            'profiles_array' => array()
        ),
        /**
         * 2. Запись по территории обслуживания МО
         */
        1 => array(
            'code' => 'mo_service_area',
            'value' => FALSE
        ),
        /**
         * 1. Запись по месту прикрепления
         */
        2 => array(
            'code' => 'attaching_point',
            'value' => FALSE,
            /**
             * 1.1 Профили, к которым разрешена запись независимо от
прикрепления.
             */
            'profiles' => array(
                'value' => TRUE,
                'profiles_array' => array()
            ),
            /**
             * 1.2 Запись для неприкрепленного населения
             */
            'patient_without_attaching' => array(
                'value' => FALSE
            ),
            /**
             * 1.4 Запись в МО без прикрепленного населения
             */
            'mo_without_attaching' => array(
                'value' => FALSE
            )
        )
    ),
    /** Соответствие кода профиля названию
    // Ключ массива - идентификатор записи в справочнике
    dbo.LpuSectionProfile
    'default_profiles_list' => array(
        35217 => 'Терапевт',
        35255 => 'Терапевт/Педиатр',
        35188 => 'Педиатрия',
        35232 => 'Хирургия',
        35184 => 'Отоларингология',
        35185 => 'Офтальмология',
        35250 => 'Травматология',
        35220 => 'Дерматология',
        35136 => 'Гинекология',
        35122 => 'Гинекология',
        35123 => 'Гинекология',
        35247 => 'Гинекология',
        35248 => 'Гинекология',
        35249 => 'Гинекология',
        35140 => 'Стоматология',
    )
)

```



```
35205 => 'Стоматология',  
35209 => 'Стоматология',  
35210 => 'Стоматология',  
35206 => 'Стоматология'  
    )  
);
```

### **regions.php**

Список регионов с настройками для мобильного приложения

```

return array(
    'list' => array(
        'astrakhan' => array(
            // Наименование для печати
            'print_name' => 'Астраханская область',
            'geo_name' => 'Астраханская область',
            // Адрес РПМУ
            'link' => 'https://doctor30.ru/',
            // Системное наименование региона
            'nick' => 'astrakhan',
            // Код региона
            'code' => '30',
            // Использование мобильного приложения
            'mobile' => true,
            // Адрес API
            'apiaddress' => 'https://doctor30.ru/api/v1/',
            // Использование платных услуг
            'payservices' => false,
            // В мобильном приложении включена авторизация через
            госуслуги
            'esiaLoginEnabled' => true,
            // Авторизация через email
            'emailLoginEnabled' => true,
            // В мобильном приложении включена ЭМК
            'mobileEmkEnabled' => true,
            // В мобильном приложении включен вызов врача на дом
            'mobileDoctorhomeEnabled' => true
        ),
        'penza' => array(
            'print_name' => 'Пензенская область',
            'geo_name' => 'Пензенская область',
            'link' => 'https://пенза-доктор.рф',
            'punylink' => 'https://xn----7sbnbkvqmbbuw.xn--plai',
            'nick' => 'penza',
            'code' => '58',
            'mobile' => true,
            'apiaddress' => 'https://xn----7sbnbkvqmbbuw.xn--
            plai/api/v1/',
            'payservices' => true,
            'esiaLoginEnabled' => true,
            'emailLoginEnabled' => true,
            'mobileEmkEnabled' => true,
            'mobileDoctorhomeEnabled' => true
        ),
        'perm' => array(
            'print_name' => 'Пермский край',
            'geo_name' => 'Пермский край',
            'link' => 'https://swantest.k-vrachu.ru/',
            'nick' => 'perm',
            'code' => '59',
            'mobile' => true,
            'apiaddress' => 'https://swantest.k-vrachu.ru/api/v1/',
            'payservices' => true,
            'esiaLoginEnabled' => true,
            'emailLoginEnabled' => true,
            'mobileEmkEnabled' => true,
            'mobileDoctorhomeEnabled' => true,
            'mobileDiaryEnabled' => true,
            // Отключение функционала анкетирования в мобильном устройстве
            'mobilePollEnabled' => true,
            // Отключение возможности регистрации в мобильном устройстве
            'mobileRegisterEnabled' => false,
            // если нет параметра или NULL - отключены, 0 - только
            рейтинг, 1 - текст + рейтинг
            'doc_feedbacks' => 1,

```

```

        // если нет параметра или NULL - отключены, 0 - только
рейтинг, 1 - текст + рейтинг
        'lpu_feedbacks' => 1,
        // включить/отключить СМС уведомления// Для сайта параметр
находится в config/имя_региона/site.php
        'sms_notifications' => false,
    ),
)
);

```

## session.php

Настройки для работы с сессиями

```

return array(
    'native' => array(
        'name' => 'session_name',
        'lifetime' => 43200,
    ),
    'cookie' => array(
        'name' => 'cookie_name',
        'encrypted' => FALSE,
        'lifetime' => 43200,
    ),
    'database' => array(
        'name' => 'cookie_name',
        'encrypted' => FALSE,
        'lifetime' => 43200,
        'group' => 'accounts',
        'table' => 'sessions',
        'columns' => array(
            'session_id' => 'session_id',
            'last_active' => 'last_active',
            'contents' => 'contents'
        ),
        'gc' => 500,
    ),
);

```

## ues\_auth.php

Настройки интеграции с сервисом авторизации УЭК

```

return array(
    // Мастер пароль сервиса авторизации
    'master_secret' => 'password'
);

```

## 12.4 Региональные файлы конфигурации

### database.php

Настройки соединения с БД

```

return array(
    // БД ЕЦП
    'default' => array(
        // Тип драйвера
        'type' => 'sqlsrv',
        // Настройки соединения
        'connection' => array(
            // Адрес сервера
            'hostname' => '10.255.0.62',
            // Наименование БД
            'database' => 'PromedWebBuryatiya',
            // Учетная запись
            'username' => 'web_user',
            // Пароль
            'password' => '***',
            // ?
            'pooling' => TRUE,
        ),
        'table_prefix' => '',
        'charset' => 'utf8',
        'caching' => TRUE,
        'profiling' => TRUE
    ),
    // БД РПМУ
    'accounts' => array(
        'type' => 'sqlsrv',
        'connection' => array(
            'hostname' => '10.255.0.62',
            'database' => 'UserPortal',
            'username' => 'web_user',
            'password' => '***',
            'pooling' => TRUE,
        ),
        'table_prefix' => '',
        'charset' => 'utf8',
        'caching' => FALSE,
        'profiling' => TRUE
    ),
    //Для БД PG
    'default_postgres' => array
    (
        'type' => 'PostgreSQL',
        'connection' => array(
            'hostname' => 'host',
            'database' => 'db_name',
            'username' => 'user',
            'password' => '***',
            'persistent' => FALSE,
        ),
        'primary_key' => '', // Column to return from INSERT queries,
        see #2188 and #2273
        'schema' => '',
        'table_prefix' => '',
        'charset' => 'utf8',
        'caching' => FALSE,
    ),
    'accounts_postgres' => array
    (
        'type' => 'PostgreSQL',
        'connection' => array(
            'hostname' => 'host',
            'database' => 'db_name',
            'username' => 'user',
            'password' => '***',
            'persistent' => FALSE,
        ),
    ),

```

```
        'primary_key' => '', // Column to return from INSERT queries,  
see #2188 and #2273  
        'schema' => '',  
        'table_prefix' => '',  
        'charset' => 'utf8',  
        'caching' => FALSE,  
    ),  
);
```

## **esia.php**

Настройки для авторизации через ЕСИА

```

return array(

    // использовать регистрацию на портале только через ЕСИА
    'registration' => false,

    // использовать авторизацию на портале только через ЕСИА
    'auth' => false,

    // использовать расширенный ЕСИА контроллер (с новым функционалом)
    'use_extended_esia' => true,

    // Включить/отключить логирование
    'log' => false,

    // Включить/отключить авторизацию через ЕСИА
    'enabled' => true,

    // Идентификатор ИС в ЕСИА
    'client_id' => 'MZPK015901',

    // Путь к РПМ для перенаправления с ЕСИА
    'redirect_uri' => 'https://k-vrachu.ru/login/esia',

    // Путь к РПМ для получения данных пользователя
    'redirect_uri_info' => 'https://k-vrachu.ru/esia/user_info/:user_id',

    // Путь к контроллеру ЕСИА для получения авторизационного кода
    'ac_path' => 'https://esia.gosuslugi.ru/aas/oauth2/ac',

    // Путь к контроллеру ЕСИА для получения маркера идентификации кода
    'te_path' => 'https://esia.gosuslugi.ru/aas/oauth2/te',

    // Область доступа для получения всех данных по пользователю
    'usr_scope' => 'http://esia.gosuslugi.ru/usr_inf?oid=:user_id',

    // Путь к REST контроллеру для получения данных по пользователю
    'usr_data_path' => 'https://esia.gosuslugi.ru/rs/prns/:user_id',

    // Путь к REST контроллеру для получения контактных данных по
    // пользователю

    'usr_contacts_path' =>
'https://esia.gosuslugi.ru/rs/prns/:user_id/ctts?embed=(elements)',

    // Сертификат
    'crt' => '-----BEGIN CERTIFICATE-----
...
-----END CERTIFICATE-----',

    // Закрытый ключ
    'key' => '-----BEGIN ENCRYPTED PRIVATE KEY-----
...
-----END ENCRYPTED PRIVATE KEY-----',

    'key_pass' => '***'
);

```

## hospitals.php

Не используется

```
return array(
    'menu' => array(
        'main' => array(
            'page' => 'main',
            'title' => 'Главная',
            'url' => ''
        ),
        'terms' => array(
            'page' => 'terms',
            'title' => 'Условия оказания помощи',
            'url' => 'terms'
        ),
        'care_types' => array(
            'page' => 'care_types',
            'title' => 'Виды помощи',
            'url' => 'care_types'
        ),
        'criteria' => array(
            'page' => 'criteria',
            'title' => 'Показатели',
            'url' => 'criteria'
        ),
        'drugs' => array(
            'page' => 'drugs',
            'title' => 'Лекарства',
            'url' => 'drugs'
        ),
    ),
);
```

## kiosk.php

Настройки информационного киоска (для старой версии портала)

```

return array(
    /**
     * Заголовок главного экрана
     */
    'title' => 'Медицинские услуги',

    /**
     * Время бездействия для возврата на начальную страницу
     */
    'idle_timeout' => 900000,

    /**
     * Услуги
     */
    'services' => array(
        'record' => array(
            'title' => 'Запись на прием к врачу',
            'title_html' => 'Запись<br> на прием к врачу',
            'description' => 'Система предназначена для записи на прием к
медицинским специалистам',
            'link_title' => 'Записаться',
            'url' => 'record',
            'icon' => '/design/common/img/main_items_01.png',
            'type' => 'service',
            'enabled' => TRUE
        ),
        'fake_service1' => array(
            'title' => 'Подача заявления на прикрепление к медицинской
организации',
            'title_html' => 'Подача заявления на прикрепление к
медицинской организации',
            'description' => 'Сервис приема заявлений на прикрепление к
медицинской организации',
            'link_title' => 'Подать заявление',
            'url' => 'notimplemented',
            'icon' => '/design/common/img/main_items_02.png',
            'type' => 'service',
            'enabled' => TRUE
        ),
        'fake_service2' => array(
            'title' => 'Запись на медицинскую услугу',
            'title_html' => 'Запись на медицинскую услугу',
            'description' => 'Система записи на медицинскую услугу',
            'link_title' => 'Записаться',
            'url' => 'notimplemented',
            'icon' => '/design/common/img/main_items_03.png',
            'type' => 'service',
            'enabled' => TRUE
        ),
        'fake_service3' => array(
            'title' => 'Подача заявления на страхование в системе ОМС',
            'title_html' => 'Подача заявления на страхование в системе
ОМС',
            'description' => 'Сервис приема заявлений на страхование в
системе ОМС',
            'link_title' => 'Подать заявление',
            'url' => 'notimplemented',
            'icon' => '/design/common/img/main_items_04.png',
            'type' => 'service',
            'enabled' => TRUE
        ),
        'fake_service4' => array(
            'title' => 'Вызов участкового врача на дом',
            'title_html' => 'Вызов участкового врача на дом',
            'description' => 'Сервис предназначен для вызова участкового
врача',

```



```

        'link_title' => 'Вызвать врача',
        'url' => 'notimplemented',
        'icon' => '/design/common/img/main_items_05.png',
        'type' => 'service',
        'enabled' => TRUE
    ),
    'pharm' => array(
        'title' => 'Поиск льготных лекарств',
        'title_html' => 'Поиск льготных лекарств',
        'description' => 'Наличие медикаментов в аптеках системы
обеспечения необходимыми лекарственными средствами (дополнительное
лекарственное обеспечение)',
        'link_title' => 'Найти',
        'url' => 'notimplemented',
        'icon' => '/design/common/img/main_items_06.png',
        'type' => 'info',
        'enabled' => TRUE
    ),
    'hospitals' => array(
        'title' => 'Перечень медицинских организаций',
        'title_html' => 'Перечень медицинских организаций',
        'description' => 'Страницы медицинских организаций',
        'link_title' => 'Найти',
        'url' => 'notimplemented',
        'icon' => '/design/common/img/main_items_07.png',
        'type' => 'info',
        'enabled' => TRUE
    ),
    'fake_service5' => array(
        'title' => 'Перечень страховых медицинских организаций',
        'title_html' => 'Перечень страховых медицинских организаций',
        'description' => 'Страницы страховых медицинских организаций',
        'link_title' => 'Найти',
        'url' => 'notimplemented',
        'icon' => '/design/common/img/main_items_08.png',
        'type' => 'info',
        'enabled' => TRUE
    )
),

/**
 * Настройки сканера штрихкодов
 */
'barcode' => array(
    /**
     * Включить/отключить работу со сканером штрихкодов
     */
    'enabled' => true,

    /**
     * Название COM порта, по которому подключен сканер штрихкодов
     */
    'barcodePort' => 'COM18',

    /**
     * Интервал опроса порта в миллисекундах
     */
    'barcodeInterval' => 1000
),

/**
 * Настройки считывателя карт
 */
'ues' => array(
    /**
     * Включить/отключить работу со считывателем карт

```

```
        */
        'enabled' => true,

        /**
         * Интервал опроса считывателя в миллисекундах
         */
        'interval' => 1000
    )
);
```

## mail.php

Настройки для отправки электронной почты

```
return array(
    'server' => '192.168.36.17',
    'port' => 25,
    'login' => 'reg@k-vrachu.ru',
    'password' => '***',
    'from' => 'noreply@k-vrachu.ru'
);
```

## site.php

Общие настройки для региона

```

return array(
    /**
     * Название сайта
     */
    'title' => 'Региональный портал медицинских услуг',

    /**
     * Региональные установки
     */
    'region' => array(
        'name' => 'Пермь',
        'nick' => 'perm',
        'code' => 59,
        'default_city' => array(
            'city_name' => 'Пермь',
            'city_id' => 3310, // Идентификатор города из таблицы
dbo.KLArea
            'area_id' => 59 // Идентификатор региона из таблицы dbo.KLArea
        )
    ),

    /**
     * Медицинский услуги
     */
    'services' => array(
        'record' => array(
            'title' => 'Запись на приём к врачу',
            'title_html' => 'Запись на приём к врачу',
            'description' => 'Запись на плановый прием к врачу по месту
жительствa. Требуется регистрация.<br>Государственная услуга.',
            'link_title' => 'Записаться',
            'url' => '/service/record',
            'icon' => '/design/common/img/main_items_01.png',
            'type' => 'service',
            'enabled' => TRUE,
            'need_auth' => TRUE,
            'class' => 'kvrachu',
            'banner' => 'record'
        ),
        'schedule' => array(
            'title' => 'Расписание работы врачей',
            'title_html' => 'Расписание работы врачей',
            'description' => 'Информация о наличии врачей-специалистов в
медицинских организациях, режиме их работы и расписании приемов',
            'link_title' => 'Расписание',
            'url' => '/service/schedule',
            'icon' => '/design/common/img/main_items_01.png',
            'type' => 'service',
            'enabled' => TRUE,
            'need_auth' => FALSE,
            'class' => 'rasp',
            'banner' => 'schedule'
        ),
        'emk' => array(
            'title' => 'Медицинская карта',
            'title_html' => 'Медицинская карта',
            'description' => 'Просмотр Вашей Электронной Медицинской
Карты',
            'link_title' => 'Записаться',
            'url' => '/service/emk',
            'icon' => '/design/common/img/main_items_01.png',
            'type' => 'service',
            'date' => '2016-10-01',
            'enabled' => TRUE,
            'need_auth_esia' => TRUE,
            'class' => 'emk',

```

```

        'banner' => 'emk',
        'allow_user_access_esia' => true
    ),
    'diary' => array(
        'title' => 'Дневник здоровья',
        'title_html' => 'Дневник здоровья',
        'description' => 'Описание для дневника здоровья',
        'link_title' => 'Дневник здоровья',
        'url' => '/service/diary',
        'icon' => '',
        'type' => 'service',
        'enabled' => true,
        'need_auth' => true,
        'class' => 'diary',
        'banner' => 'diary',
        'version' => 2
    ),
    'doctorhome' => array(
        'title' => 'Вызов участкового врача на дом',
        'title_html' => 'Вызов участкового врача на дом',
        'description' => 'Опишите симптомы и вызовите участкового
врача на дом',
        'link_title' => 'Вызвать врача',
        'url' => '/service/doctorhome',
        'icon' => '/design/common/img/main_items_05.png',
        'type' => 'service',
        'enabled' => TRUE,
        'need_auth' => TRUE,
        'class' => 'nadam',
        'banner' => 'nadam'
    ),
    'pharm' => array(
        'title' => 'Поиск льготных лекарств',
        'title_html' => 'Поиск льготных лекарств',
        'description' => 'Наличие медикаментов в аптеках системы
обеспечения необходимыми лекарственными средствами (дополнительное
лекарственное обеспечение)',
        'link_title' => 'Найти',
        'url' => '/service/pharm',
        'icon' => '/design/common/img/main_items_06.png',
        'type' => 'info',
        'enabled' => true,
        'class' => 'drugs'
    ),
    'hospitals' => array(
        'title' => 'Медицинские организации',
        'title_html' => 'Медицинские организации',
        'description' => 'Список медицинских организаций и врачей',
        'link_title' => 'Найти',
        'url' => '/service/hospitals',
        'icon' => '/design/common/img/main_items_07.png',
        'type' => 'info',
        'enabled' => TRUE,
        'class' => 'medorg',
        'banner' => 'hospitals'
    ),
    'find_regions' => array(
        'title' => 'Поиск участка прикрепления',
        'title_html' => 'Поиск участка прикрепления',
        'description' => 'Поиск медицинской организации, врачебного
участка, и врача, обслуживающего указанный адрес',
        'link_title' => 'Найти',
        'url' => '/service/regions',
        'icon' => '/design/common/img/main_items_10.png',
        'type' => 'info',
        'enabled' => TRUE,

```

```

        'class' => 'uchastok',
        'banner' => 'regions'
    ),
    'payservices' => array(
        'title' => 'Платные медицинские услуги',
        'title_html' => 'Платные услуги',
        'description' => 'Услуги, предоставляемые дополнительно к
бесплатным медицинским услугам',
        'link_title' => 'Записаться',
        'url' => '/service/record/paid',
        'icon' => '/design/common/img/main_items_01.png',
        'type' => 'service',
        'enabled' => true,
        'need_auth' => TRUE,
        'class' => 'payservices',
        'banner' => 'payservices',
    ),
    'attachment' => array(
        'title' => 'Прикрепление',
        'title_html' => 'Прикрепление',
        'description' => 'Добавить или изменить прикрепление к
медицинской организации',
        'link_title' => 'Изменить',
        'url' => '/service/emk/attachment',
        'icon' => '/design/common/img/main_items_01.png',
        'type' => 'service',
        'enabled' => true,
        'need_auth' => TRUE,
        'class' => 'attachment',
        'banner' => 'attachment',
    ),
),

/**
 * Профили врачей
 */
'profiles' => array(
    // Список id главных специальностей для вывода в начале списка
    'main_profiles' => array(
        // терапевт, стоматолог, гинеколог, педиатр
        //6, 10, 11, 29, 31, 36, 37, 38, 56, 63, 100, 10000, 12006,
12013, 401455, 401387, 401468, 401444
        // хирургия, дерматология, травматология, гинекология,
стоматология, отоларингология, офтальмология
    ),
    // Список id главных специальностей для вывода в начале списка
    'main_profiles_auth' => array(
        //6, 10, 11, 29, 31, 56, 63, 100, 401455, 401387, 401468,
401444
        // хирургия, отоларингология, офтальмология, травматология,
дерматология, гинекология, стоматология,
    ),
    // Список специальностей по которым есть прикрепление
    'attach' => array(2, 21, 12), // Терапевт, педиатр или гинеколог
    'attach_adult' => array(2, 5080), // по терапевту или по врачу
общей практики,
    'attach_child' => array(21), // для детей по педиатру
    'attach_women' => array(12), // гинеколог
    // Стоматолог
    'stomatologist' => array(35, 36, 37, 38, 12006),
    // Терапевт
    'therapist' => array(2, 33020),
    // Педиатр
    'pediatrician' => array(21, 914),
    // Гинеколог
    'gynecologist' => array(12, 12013, 10000)
)

```

```

),
/**
 * Настройки записи на прием
 */
'record' => array(
    // Время, после которого запрещена запись на следующий день (Н:i)
    'max_time' => '17:00',
    // Возраст взрослого пациента
    'adult_age' => 18,
    // Количество дней, на которые доступна запись
    'max_day' => 14,
    'allow_record_today' => true,
    // Включить автоматическую модерацию
    'enable_automatic_moderation' => true,
    'enable_automatic_moderation_always' => true,

    /**
     * Проверки при записи
     */
    'validations' => array(
        // Разрешать только одно предстоящее посещение по профилю
        'only_one_record_allowed' => true,
        // Разрешать только одно посещение к одному специалисту в одно
        ЛПУ на день
        'only_one_record_on_day_allowed' => true,
    ),

    /**
     * Разрешить платный прием
     */
    'allow_pay' => true,
    'allow_disp' => true,

    /**
     * Разрешить настройку разрешения оповещений из Промеда
     */
    'set_promed_notifications' => false,

    /**
     * Разрешить отменять записи в любое время
     */
    'allow_cancel_any_time' => true,

    /**
     * Ограничение записи пациентов с закрытым полисом ОМС
     */
    'restrict_record_with_closed_polis' => true
),

/**
 * Список официальных документов для вывода на страницах
 */
'documents' => array(
    array(
        'link' => '/docs/pgg2015.pdf',
        'title' => 'Программа государственных гарантий оказания
гражданам Российской Федерации бесплатной медицинской помощи на 2015 год и
на плановый период 2016 и 2017 годов'
    ),
    array(
        'link' => '/docs/tpgg2015_perm.pdf',
        'title' => 'Закон № 427-ПК от 25 декабря 2014 «О
территориальной программе государственных гарантий бесплатного оказания
гражданам медицинской помощи на 2015 год и на плановый период 2016 и 2017
годов»'
    )
)

```

```

    ),
    array(
        'link' => '/docs/sed-34-01-06-346.pdf',
        'title' => 'Приказ Министерства Здравоохранения Пермского края
№ сэд-34-01-06-346 от 21 сентября 2011 г. «Об утверждении порядка
прикрепления населения к медицинским организациям для получения
амбулаторно-поликлинической помощи»'
    )
),

/**
 * Часовой пояс
 */
'timezone' => 'Asia/Yekaterinburg',

/**
 * Ключ для яндекс-карт
 */
'map_key' => '****',

/**
 * Объявление для главной страницы
 */
'notice' => array(
    // Текст объявления (поддерживается Textile)
    'text' => '
        Сайт работает в ознакомительном режиме. Вскоре состоится
переход в рабочий режим. <a href="/about/new">Что нового?</a>
    ',
    // Дата и время (Y-m-d H:i) срока действия объявления
    'expires' => '2016-09-28 12:00'
),

/**
 * Тип добавления человека в картотеку
 * 1 - Добавление без модерации, если человек не найден в БД
 * 2 - Отправка человека на модерацию, если человек не найден в БД
 * 3 - Запрет добавления человека если он не найден в БД
 */
'add_person_type' => 1,

/**
 * Настройки авторизации
 */
'auth' => array(
    /**
     * Количество неверных попыток авторизации до момента, когда
учетная запись будет временно заблокирована
     */
    'max_attempts_count' => 5,

    /**
     * Количество минут, на которые блокируется учетная запись после
определенного выше числа неверных попыток авторизации.
     */
    'ban_time' => 10,

    'swToken' => '****',
    'scan_code_service' =>
'ws://127.0.0.1:8080/ScanCodeService/wsscancode'
),
'homevisit' => array(
    /**
     * Временной промежуток, когда доступен сервис
     */
    'from_time' => '08:00',

```

```

        'to_time' => '23:00'
    ),
    'MinTimeQueue' => '2',

    /**
     * E-mail технической поддержки, используется для формы обратной связи
     */
    'support_email' => 'support@k-vrachu.ru',

    /**
     * Яндекс метрика
     */
    'metrika' => 41455854,

    /**
     * Регионы
     */
    'regions' => array(
        'astrakhan' => array(
            'print_name' => 'Астраханская область',
            'geo_name' => 'Астраханская область',
            'link' => 'http://doctor30.ru'
        ),
        'ufa' => array(
            'print_name' => 'Республика Башкортостан',
            'geo_name' => 'Республика Башкортостан',
            'link' => 'https://rb.k-vrachu.ru'
        ),
        'buryatia' => array(
            'print_name' => 'Республика Бурятия',
            'geo_name' => 'Республика Бурятия',
            'link' => 'https://reg03.k-vrachu.ru/'
        ),
        'karelia' => array(
            'print_name' => 'Республика Карелия',
            'geo_name' => 'Республика Карелия',
            'link' => 'https://reg.zdrav10.ru/'
        ),
        'perm' => array(
            'print_name' => 'Пермский край',
            'geo_name' => 'Пермский область',
            'link' => 'https://demo.k-vrachu.ru/'
        ),
        'pskov' => array(
            'print_name' => 'Псковская область',
            'geo_name' => 'Псковская область',
            'link' => 'https://k-vrachu.pskov.ru/'
        ),
        'ekb' => array(
            'print_name' => 'Свердловская область',
            'geo_name' => 'Свердловская область',
            'link' => 'https://so.k-vrachu.ru/'
        ),
        'khakassia' => array(
            'print_name' => 'Республика Хакасия',
            'geo_name' => 'Республика Хакасия',
            'link' => 'https://k-vrachu.mz19.ru/'
        ),
    ),

    /**
     * Настройки связи с ЕЦП
     */
    'promed' => array(
        /**
         * Путь к ЕЦП

```



```

    */
    'path' => 'http://pro.local/rish',

    /**
     * Путь к Birt
     */
    'birt_path' => 'http://192.168.136.40:812/birt-viewer/preview'
  ),
  'external' => array(
    'promed_path' => 'http://pro.local'
  ),

  /**
   * Настройки отзывов
   */
  'feedbacks' => array(
    'doctor' => array(
      'enabled' => true // включить/отключить отзывы о врачах
      , 'display_type' => 0 // 0 или не указано - только звездочки, 1
- звездочки + текст
      , 'posting_expired_time' => 10
    ),
    'lpu' => array(
      'enabled' => true // включить/отключить отзывы о больницах
      , 'display_type' => 0 // 0 или не указано - только звездочки, 1
- звездочки + текст
    )
    // возможность модерации отзывов через панель модерации
    // (1 - модерация разрешена | 0 или НЕ УКАЗАНО - модерация не
разрешена)
    // для типа отображения "только звездочки", признак модерации не
учитывается
    , 'canModerate' => 1
    // кто может оставлять отзывы
    // 1 - только пользователи ЕСИА
    // 0 - все пользователи
    , 'only_esia_can_post' => 0
  ),

  /**
   * Настройки жалоб
   */
  'complaints' => array(
    'enabled' => false // включить/отключить жалобы
  ),

  'logspath' => APPPATH . 'logs',

  /**
   * NODE JS прокси для портала
   * 1. для имени хоста обязательно указывать протокол (http:// или
https://) в зависимости от того в каком режиме запущен NODE-SERVER (можно
посмотреть при старте сервера или в логах)
   * 2. номера портов дублируются - это нормально, т.к. сокет сервер
теперь инициализируется через порт веб-сервера
   */
  'nodejs_portal_proxy_hostname' => 'https://perm.swn.local',
  'nodejs_portal_proxy_httpport' => '7070',
  'nodejs_portal_proxy_socketport' => '7070',

  'workplaces' => array(
    // Название точки входа в ЕЦП
    'endpoint_name' => 'ЕИСЗ ПК',

    // настройки для разных типов АРМ

```

```

        // АРМ СОТРУДНИКА ОБР. УЧ.
        'edu' => array(
            // Список доступных ЛПУ для сотр. обр. учреждений
            'lpu_list' => array(10010833, 150185)
        )
    ),

    // Медицинский опросник - сбор структурированной медицинской
    информации и поддержка принятия решений
    'dss' => array(
        'api_path' => 'http://192.168.37.3:2071/patient/API.php', //
        тестовый сервер АПИ СППР
        'api_version' => '4-1-0'
    ),

    // Количество строк на экране ТВ
    'scoreboard_timetable_max_rows' => 5,

    // Включить/отключить СМС уведомления
    // Для API находится в config/regions.php
    'sms_notifications' => false,
    'fcm_key' => '***',

    // Анкеты в ЭМК
    'polls' => array(
        'enabled' => true
    ),

    // Путь до общих сервисов
    'SwServiceCommon' => array(
        'default' => array(
            'url' => 'http://192.168.37.3:2080/api/common',
            'timeout' => 20,
            'name' => 'Общий функционал',
            'nick' => 'common',
            'enabled' => true
        )
    ),

    // Путь до РИШ
    'SwServiceRish' => array(
        'default' => array(
            'url' => 'http://192.168.37.3:2080/api/rish',
            'timeout' => 20,
            'name' => 'РИШ',
            'nick' => 'rish',
            'enabled' => true
        )
    ),

    // Дополнительное логирование
    'logging' => array(
        // включено или отключено
        'enabled' => true,

        // здесь можно исключить какие-либо логи
        // (ElectronicQueue, ProffService, SwService, Infomat)
        'excluded_logs' => array()
    ),

    // использование моделей запросами на PG
    'use_postgres' => true,
    // разрешить использование функционала заявок
    'enable_record_request' => false,
    // перенаправлять на сайт ЕПГУ при записи к врачу
    'use_epgu_record' => false

```

```
);
```

## **sms.php**

### Настройки для отправки SMS

```
return array(  
    'login' => 'svan',  
    'password' => '***',  
    'prefix' => 'karelia',  
    'sender' => 'reg.zdrav10',  
    'proxy' => array(  
        /*  
        'address' => '192.168.36.167',  
        'port' => '3128',  
        'login' => NULL,  
        'password' => NULL,  
        */  
    )  
);
```

## 13 Сервис LDAP

Для установки на centos нужен доступ к интернету и выполнить следующую команду

```
curl https://cloud.swan-it.ru/index.php/s/Xp4xk9XN7XSpsgz/download |  
sudo bash
```

После установки регистрируется сервис в systemd `opendj.service`

Сервис уже преднастроен и залита пустая база с Base DN `dc=ecp,dc=ru`

Администрирование осуществляется по порту 4444 пароль `gjnjhjncnefkybt`

В базе находится один пользователь для входа в ЕЦП.МИС `admin/AmoKK`

Данный пользователь может быть не привязан ни к одной МО. в этом случае входим в ЕЦП.МИС. Видим выбор МО, но выбрать якобы невозможно. Но если начать вводить Тест, то можно перейти в ЛПУ тест. после чего создать пользователя с привязкой к МО и необходимыми правами. и в последующем входить уже под созданным пользователем.

Добавляем в настройки ЕЦП.МИС в `promed.php`, заменив адрес сервера на необходимый, следующие настройки:

```
// сервер LDAP  
define("LDAP_SERVER", "10.1.1.1");  
define("LDAP_SERVER_PORT", 389);  
  
// настройки дерева  
define("LDAP_DOMAIN", "dc=ecp,dc=ru");  
define("LDAP_GROUP_PATH", "ou=Groups, ".LDAP_DOMAIN);  
define("LDAP_USER_PATH", "ou=Users, ".LDAP_DOMAIN);  
define("LDAP_ADDRBOOK_PATH", "ou=Books, ".LDAP_DOMAIN);  
  
// учетная запись  
define("LDAP_USER", "cn=admin, ".LDAP_DOMAIN);  
define("LDAP_PASS", "amokkamokk");
```

### Code Block 32 promed.php

Установка ограничения по памяти:

Правим службу и оставляем

в файле `/opt/opendj/config/java.properties`

выставляем необходимые границы памяти.

```
overwrite-env-java-home=true
```

```
start-ds.java-args="-server" "-Xms2g" "-Xmx6g" "-XX:+UseCompressedOops"
```

запускаем

```
/opt/opendj/bin/dsjavaproperties
```

перезапускаем службу.

---

Рестроирование индексов

```
rebuild-index -h localhost -p 4444 -D "cn=directory manager" --bindPassword ***** --baseDN  
dc=ufa,dc=local --rebuildAll --trustall
```

## 14 Сервис ЕРМП

### 14.1 Программное обеспечение:

1. Java 1.8
2. Tomcat 9
3. Драйвер Postgresql (Положить в lib томката)  
(42.2.6) <https://mvnrepository.com/artifact/org.postgresql/postgresql/42.2.6>

На данный момент можно взять сам ЕРМП <https://cloud.swan-it.ru/index.php/s/PHxrRok6DoxStBE>

js файлы из папки ermp\_promed по ссылке выше надо положить в папку ermp в корневой папке ЕЦП.МИС. (Если ЕЦП.МИС в папке /srv/promed, то положить в папку /srv/promed/ermp)

Старые файлы js лучше удалить.

.gwt.rc файл положить рядом с папкой WEB-INF (лежит так же в ermp\_promed)

Настройки находятся в папке webapps/ermp/WEB-INF/settings.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<servers>
  <server>
    <active>true</active>
    <!-- Можно не править на данный момент не используется -->
    <ermp-export>
      <see-in-browser>/export/ermp_files</see-in-browser>
      <url>file:///d:\ermp\<</url>
      <port></port>
      <user></user>
      <password></password>
      <ermp_temp></ermp_temp>
    </ermp-export>
    <!-- Указываем подключени к БД mongodb в которой хранятся сессии, т.е. к
    той которая указана в конфигурации промеда в файле mongodbsessions.php -->
    <noSQLAuthorizationDBConnection active="true">
      <username></username>
      <password></password>
      <host></host>
      <port></port>
      <db_name></db_name>
      <table_name>Session</table_name>
    </noSQLAuthorizationDBConnection>
    <!-- Указываем подключени к БД postgresql основная. Группа в database.php
    default -->
    <authorizationDBconnection>
      <username></username>
      <password></password>
      <url>jdbc:postgresql:///</url>
    </authorizationDBconnection>
    <!-- Указываем подключени к БД postgresql основная. Группа в database.php
    default -->
    <workDBconnection>
      <username></username>
      <password></password>
      <url>jdbc:postgresql:///</url>
    </workDBconnection>
    <service-auth>
      <url>http://mz.nsser.prognoz.ru/MedStaff/MedStaff.svc/basic</u
    rl>
      <!--
    url>https://service.rosminzdrav.ru/MedStaffIntegration/medstaff.svc/basic<
    /url-->
      <username></username>
      <password></password>
      <!--hdStorePassword></hdStorePassword-->
    </service-auth>
    <app-env>
    <!-- Кроме кода ДЛО (описание и регулярное выражение) пока ничего не
    правила по данным настройкам желательно проконсультироваться у
    разработчиков (Разорвин Виктор, Фролов Антон) -->
      <lpuSearchConstraint>true</lpuSearchConstraint>
      <kodDloMask>[0-9]{6}</kodDloMask>
      <kodDloMaskMessage>Код ДЛО должен состоять из 6
    цифр</kodDloMaskMessage>
      <useKodDLOWithLPU>false</useKodDLOWithLPU>
      <generateDloButtonVisible>true</generateDloButtonVisible>
      <needINN>true</needINN>
      <needSNILS>true</needSNILS>
      <needAddress>true</needAddress>
      <needDocument>true</needDocument>
      <exportAllPost>true</exportAllPost>
      <!-- контроль на обязательность полей -->
      <useFRMR>false</useFRMR>
      <!-- запрет на редактирование сотрудника-->

```

```

        <preventEditableWorker>true</preventEditableWorker>
        <!--количество ставок-->
        <medPersonalRate>1.5</medPersonalRate>
    </app-env>
</server>
</servers>
<!--
Для возможности редактирования Строки штатного расписания на месте работы
в таблице БД persis.ClientSettings должен присутствовать параметр
isEditStaff со значением true (таблица региональная)
Для возможности редактирования Должности в строке штатного расписания в
таблице БД persis.ClientSettings должен присутствовать параметр isEditPost
со значением true (таблица региональная)
-->

```

**Code Block 33 settings.xml**

Выше в комментариях написано какие подключения прописывать. Читайте внимательно.

[Настройка авторизации в ЕРМП](#)

## 14.2 Настройка авторизации в ЕРМП

Настройка авторизации в ЕРМП производится в файле настроек WEB-INF/settings.xml

Блоки отвечающие за настройку

Авторизация через БД (на данный момент уже не используется) :

```

...
<authorizationDBconnection>
    <username></username>
    <password></password>
    <url>jdbc:postgresql:///</url>
</authorizationDBconnection>
...

```

**Code Block 34 settings.xml**

Авторизация через MongoDB:

```

<noSQLAuthorizationDBConnection active="true">
    <uri></uri>
    <username></username>
    <password></password>
    <host></host>
    <port></port>
    <db_name></db_name>
    <table_name>Session</table_name>
</noSQLAuthorizationDBConnection>

```

**Code Block 35 settings.xml**



Выбор способа авторизации производится следующим образом: по умолчанию используется авторизация через бд, если же в теге `noSQLAuthorizationDBConnection` параметр `active` установлен в "true" то производится авторизация через MongoDB

Настройка авторизации через бд (на данный момент уже не используется):

- указать имя пользователя (`<username>someLogin</username>`)
- пароль (`<password>****</password>`)
- строку соединения jdbc с сервером базы данных (`<url>jdbc:postgresql://192.168.36.24:5432/promedtest</url>`)

Настройка авторизации через Mongo:

1) Настройка соединения с одиночным сервером MongoDB

- строка соединения оставляется пустой (`<uri></uri>`)
- указать если требуется авторизация имя пользователя (`<username>someLogin</username>`)
- пароль (`<password>****</password>`)
- хост (`<host>192.168.36.84</host>`)
- порт (`<port>27017</port>`)
- имя базы (`<db_name>db_perm</db_name>`)
- имя коллекции (`<table_name>Session</table_name>`)

Доступно с версии v1.0.2

2) Настройка соединения с кластером MongoDB

- указывается строка соединения с кластером (`<uri>mongodb://mongors1n1:27017,mongors1n2:27017,mongors1n3:27017/?replicaSet=mongors1</uri>`)
- имя базы (`<db_name>db_perm</db_name>`)
- имя коллекции (`<table_name>Session</table_name>`)  
при указании строки соединения (uri) соединение создается из него значения хоста и порта игнорируются даже если заполнены

[Проверка правильности настроек](#)

## 15 Сервис формирования отчетов birt-viewer

1. Java 1.8 (1.11)
2. Tomcat 9
3. Драйвер Postgresql (Положить в lib томката) (42.2.6) <https://mvnrepository.com/artifact/org.postgresql/postgresql/42.2.6>
4. birt-viewer 4.8

Скачать birt-viewer можно по ссылке <https://download.eclipse.org/birt/downloads/>

Стандартная инструкция по установке находится по ссылке - <https://www.eclipse.org/birt/documentation/integrating/viewer-setup.php>

После установки надо скачать шрифты, которые используются в отчетах <https://cloud.swan-it.ru/index.php/s/sLtpxTcb3HefPap>

положить в одну из следующих папок:

```
/usr/X/lib/X11/fonts/TrueType
/usr/share/fonts/default/TrueType
/usr/openwin/lib/X11/fonts/TrueType
/usr/openwin/lib/locale/euro_fonts/X11/fonts/TrueType
/usr/openwin/lib/locale/iso_8859_2/X11/fonts/TrueType
/usr/openwin/lib/locale/iso_8859_5/X11/fonts/TrueType
/usr/openwin/lib/locale/iso_8859_7/X11/fonts/TrueType
/usr/openwin/lib/locale/iso_8859_8/X11/fonts/TrueType
/usr/openwin/lib/locale/iso_8859_9/X11/fonts/TrueType
/usr/openwin/lib/locale/iso_8859_13/X11/fonts/TrueType
/usr/openwin/lib/locale/iso_8859_15/X11/fonts/TrueType
/usr/openwin/lib/locale/ar/X11/fonts/TrueType
/usr/openwin/lib/locale/hi_IN.UTF-8/X11/fonts/TrueType
/usr/openwin/lib/locale/ja/X11/fonts/TT
/usr/openwin/lib/locale/ko/X11/fonts/TrueType
/usr/openwin/lib/locale/ko.UTF-8/X11/fonts/TrueType
/usr/openwin/lib/locale/KOI8-R/X11/fonts/TrueType
/usr/openwin/lib/locale/ru.ansi-1251/X11/fonts/TrueType
/usr/openwin/lib/locale/th_TH/X11/fonts/TrueType
/usr/openwin/lib/locale/zh_TW/X11/fonts/TrueType
/usr/openwin/lib/locale/zh_TW.BIG5/X11/fonts/TT
/usr/openwin/lib/locale/zh_HK.BIG5HK/X11/fonts/TT
/usr/openwin/lib/locale/zh_CN.GB18030/X11/fonts/TrueType
/usr/openwin/lib/locale/zh/X11/fonts/TrueType
/usr/openwin/lib/locale/zh.GBK/X11/fonts/TrueType
/usr/X11R6/lib/X11/fonts/TrueType
/usr/X11R6/lib/X11/fonts/truetype
/usr/X11R6/lib/X11/fonts/tt
/usr/X11R6/lib/X11/fonts/TTF
/usr/X11R6/lib/X11/fonts/OTF
```

```
/usr/share/fonts/ja/TrueType  
/usr/share/fonts/truetype  
/usr/share/fonts/ko/TrueType  
/usr/share/fonts/zh_CN/TrueType  
/usr/share/fonts/zh_TW/TrueType  
/var/lib/defoma/x-ttcidfont-conf.d/dirs/TrueType
```

Перезапускаем сервис.

С папке report создаем файл ConnectLib.rptlibrary

```

<?xml version="1.0" encoding="UTF-8"?>
<library xmlns="http://www.eclipse.org/birt/2005/design" version="3.2.23"
id="1">
  <property name="createdBy">Eclipse BIRT Designer Version
4.2.2.v201301221637 Build &lt;4.2.2.v20130206-1509</property>
  <property name="units">in</property>
  <property name="theme">defaultTheme</property>
  <parameters>
    <scalar-parameter name="CurrentRegion" id="374">
      <property name="hidden">true</property>
      <text-property name="promptText">Текущий регион</text-property>
      <property name="valueType">static</property>
      <property name="isRequired">>false</property>
      <property name="dataType">integer</property>
      <property name="distinct">>true</property>
      <simple-property-list name="defaultValue">
        <value type="constant">1</value>
      </simple-property-list>
      <list-property name="selectionList"/>
      <property name="paramType">simple</property>
      <property name="controlType">text-box</property>
      <structure name="format">
        <property name="category">Unformatted</property>
      </structure>
    </scalar-parameter>
  </parameters>
  <data-sources>
    <oda-data-source extensionID="org.eclipse.birt.report.data.oda.jdbc"
name="DataSourceReportPgSQL" id="365">
      <list-property name="privateDriverProperties">
        <ex-property>
          <name>metadataBidiFormatStr</name>
          <value>ILYNN</value>
        </ex-property>
        <ex-property>
          <name>disabledMetadataBidiFormatStr</name>
        </ex-property>
        <ex-property>
          <name>contentBidiFormatStr</name>
          <value>ILYNN</value>
        </ex-property>
        <ex-property>
          <name>disabledContentBidiFormatStr</name>
        </ex-property>
      </list-property>
      <property
name="odaDriverClass">org.postgresql.Driver</property>
      <property name="odaURL"></property>
      <property name="odaUser"></property>
      <encrypted-property name="odaPassword"
encryptionID="base64"></encrypted-property>
    </oda-data-source>
    <oda-data-source extensionID="org.eclipse.birt.report.data.oda.jdbc"
name="DataSourceReportPgSQLStac" id="367">
      <list-property name="privateDriverProperties">
        <ex-property>
          <name>metadataBidiFormatStr</name>
          <value>ILYNN</value>
        </ex-property>
        <ex-property>
          <name>disabledMetadataBidiFormatStr</name>
        </ex-property>
      </list-property>
    </oda-data-source>
  </data-sources>
</library>

```

```

        <name>contentBidiFormatStr</name>
        <value>ILYNN</value>
    </ex-property>
    <ex-property>
        <name>disabledContentBidiFormatStr</name>
    </ex-property>
</list-property>
<property
name="odaDriverClass">org.postgresql.Driver</property>
    <property name="odaURL"></property>
    <property name="odaUser"></property>
    <encrypted-property name="odaPassword"
encryptionID="base64"></encrypted-property>
</oda-data-source>
    <oda-data-source extensionID="org.eclipse.birt.report.data.oda.jdbc"
name="DataSourceReportPgSQLLis" id="368">
        <list-property name="privateDriverProperties">
            <ex-property>
                <name>metadataBidiFormatStr</name>
                <value>ILYNN</value>
            </ex-property>
            <ex-property>
                <name>disabledMetadataBidiFormatStr</name>
            </ex-property>
            <ex-property>
                <name>contentBidiFormatStr</name>
                <value>ILYNN</value>
            </ex-property>
            <ex-property>
                <name>disabledContentBidiFormatStr</name>
            </ex-property>
        </list-property>
    </property>
name="odaDriverClass">org.postgresql.Driver</property>
    <property name="odaURL">jdbc:postgresql:///</property>
    <property name="odaUser"></property>
    <encrypted-property name="odaPassword"
encryptionID="base64"></encrypted-property>
</oda-data-source>
    <oda-data-source extensionID="org.eclipse.birt.report.data.oda.jdbc"
name="DataSourceReportPgSQLRegistry" id="369">
        <list-property name="privateDriverProperties">
            <ex-property>
                <name>metadataBidiFormatStr</name>
                <value>ILYNN</value>
            </ex-property>
            <ex-property>
                <name>disabledMetadataBidiFormatStr</name>
            </ex-property>
            <ex-property>
                <name>contentBidiFormatStr</name>
                <value>ILYNN</value>
            </ex-property>
            <ex-property>
                <name>disabledContentBidiFormatStr</name>
            </ex-property>
        </list-property>
    </property>
name="odaDriverClass">org.postgresql.Driver</property>
    <property name="odaURL"></property>
    <property name="odaUser"></property>
    <encrypted-property name="odaPassword"
encryptionID="base64"></encrypted-property>
</oda-data-source>

```

```

    <oda-data-source extensionID="org.eclipse.birt.report.data.oda.jdbc"
name="DataSourceReportPgSQLPolka" id="370">
    <list-property name="privateDriverProperties">
        <ex-property>
            <name>metadataBidiFormatStr</name>
            <value>ILYNN</value>
        </ex-property>
        <ex-property>
            <name>disabledMetadataBidiFormatStr</name>
        </ex-property>
        <ex-property>
            <name>contentBidiFormatStr</name>
            <value>ILYNN</value>
        </ex-property>
        <ex-property>
            <name>disabledContentBidiFormatStr</name>
        </ex-property>
    </list-property>
    <property
name="odaDriverClass">org.postgresql.Driver</property>
    <property name="odaURL"></property>
    <property name="odaUser"></property>
    <encrypted-property name="odaPassword"
encryptionID="base64"></encrypted-property>
    </oda-data-source>
    <oda-data-source extensionID="org.eclipse.birt.report.data.oda.jdbc"
name="DataSourceReportPgSQLSMP" id="371">
    <list-property name="privateDriverProperties">
        <ex-property>
            <name>metadataBidiFormatStr</name>
            <value>ILYNN</value>
        </ex-property>
        <ex-property>
            <name>disabledMetadataBidiFormatStr</name>
        </ex-property>
        <ex-property>
            <name>contentBidiFormatStr</name>
            <value>ILYNN</value>
        </ex-property>
        <ex-property>
            <name>disabledContentBidiFormatStr</name>
        </ex-property>
    </list-property>
    <property
name="odaDriverClass">org.postgresql.Driver</property>
    <property name="odaURL"></property>
    <property name="odaUser"></property>
    <encrypted-property name="odaPassword"
encryptionID="base64"></encrypted-property>
    </oda-data-source>
    <oda-data-source extensionID="org.eclipse.birt.report.data.oda.jdbc"
name="DataSourceReportPgSQLNSI" id="372">
    <list-property name="privateDriverProperties">
        <ex-property>
            <name>metadataBidiFormatStr</name>
            <value>ILYNN</value>
        </ex-property>
        <ex-property>
            <name>disabledMetadataBidiFormatStr</name>
        </ex-property>
        <ex-property>
            <name>contentBidiFormatStr</name>
            <value>ILYNN</value>
        </ex-property>
    </list-property>

```

```

        <ex-property>
            <name>disabledContentBidiFormatStr</name>
        </ex-property>
    </list-property>
    <property
name="odaDriverClass">org.postgresql.Driver</property>
    <property name="odaURL"></property>
    <property name="odaUser"></property>
    <encrypted-property name="odaPassword"
encryptionID="base64"></encrypted-property>
    </oda-data-source>
    <oda-data-source extensionID="org.eclipse.birt.report.data.oda.jdbc"
name="DataSourceReportPgSQLAll" id="373">
        <list-property name="privateDriverProperties">
            <ex-property>
                <name>metadataBidiFormatStr</name>
                <value>ILYNN</value>
            </ex-property>
            <ex-property>
                <name>disabledMetadataBidiFormatStr</name>
            </ex-property>
            <ex-property>
                <name>contentBidiFormatStr</name>
                <value>ILYNN</value>
            </ex-property>
            <ex-property>
                <name>disabledContentBidiFormatStr</name>
            </ex-property>
        </list-property>
    </property>
name="odaDriverClass">org.postgresql.Driver</property>
    <property name="odaURL"></property>
    <property name="odaUser"></property>
    <encrypted-property name="odaPassword"
encryptionID="base64"></encrypted-property>
    </oda-data-source>
    <oda-data-source extensionID="org.eclipse.birt.report.data.oda.jdbc"
name="DataSourceEMD" id="7">
        <list-property name="privateDriverProperties">
            <ex-property>
                <name>metadataBidiFormatStr</name>
                <value>ILYNN</value>
            </ex-property>
            <ex-property>
                <name>disabledMetadataBidiFormatStr</name>
            </ex-property>
            <ex-property>
                <name>contentBidiFormatStr</name>
                <value>ILYNN</value>
            </ex-property>
            <ex-property>
                <name>disabledContentBidiFormatStr</name>
            </ex-property>
        </list-property>
    </property>
name="odaDriverClass">org.postgresql.Driver</property>
    <property name="odaURL"></property>
    <property name="odaUser"></property>
    <encrypted-property name="odaPassword"
encryptionID="base64"></encrypted-property>
    </oda-data-source>
</data-sources>
<themes>
    <theme name="defaultTheme" id="4"/>

```

```
</themes>
<page-setup>
  <simple-master-page name="NewSimpleMasterPage" id="3"/>
</page-setup>
</library>
```

Необходимо заполнить следующее:

```
<property
name="odaURL">jdbc:postgresql://DB_ADDRESS/DB_NAME</property>
<property name="odaUser">USERNAME</property>
<encrypted-property name="odaPassword"
encryptionID="base64">PASSWORD_BASE64</encrypted-property>
```

В строках капслоком прописаны параметры, которые необходимо поменять

*DB\_ADDRESS* - поменять на адрес Базы данных и порт

*DB\_NAME* - имя БД

*USERNAME* - ползователь под которым будет подключаться

*PASSWORD\_BASE64* - Пароль зашифрованный base64

Источники данных:

*DataSourceReportPgSQL* - Указывается подключение к отчетной БД

*DataSourceReportPgSQLStac* - Указывается подключение к БД стационара

*DataSourceReportPgSQLLis* - Указывается подключение к БД ЛИС

*DataSourceReportPgSQLRegistry* - Указывается подключение к реестровой БД

*DataSourceReportPgSQLPolka* - Указывается подключение к БД поликилининики

*DataSourceReportPgSQLSMP* - Указывается подключение к БД СМП

*DataSourceReportPgSQLNSI* - НЕОБХОДИМО ПОЯСНИТЬ У РАЗРАБОТЧИКОВ

*DataSourceReportPgSQLAll* - Общей БД (Нужно для печати рецептов и т.д.)

```
<parameters>
<scalar-parameter name="CurrentRegion" id="374">
<property name="hidden">>true</property>
<text-property name="promptText">Текущий регион</text-property>
<property name="valueType">static</property>
<property name="isRequired">>false</property>
<property name="dataType">integer</property>
<property name="distinct">>true</property>
```



```
<simple-property-list name="defaultValue">  
<value type="constant">1</value>  
</simple-property-list>  
<list-property name="selectionList"/>  
<property name="paramType">simple</property>  
<property name="controlType">text-box</property>  
<structure name="format">  
<property name="category">Unformatted</property>  
</structure>  
</scalar-parameter>  
</parameters>
```

В разделе <parameters> необходимо заполнить значение <value type="constant"> кодом региона, для которого разворачивается сервер birt.

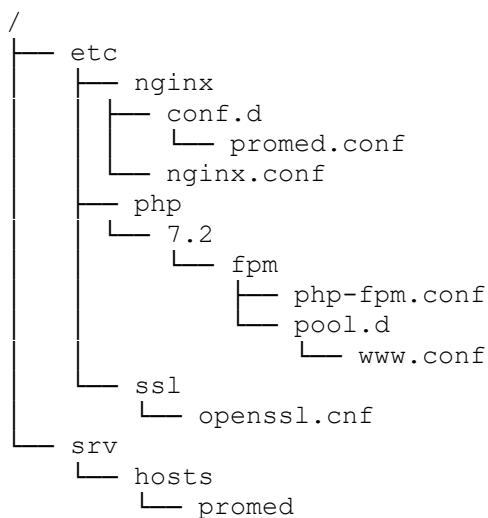
Отчеты положить в папку reports сервиса birt-viewer.

## 16 Установка веб сервера на linux

### Необходимое программное обеспечение:

- nginx последняя стабильная версия
- openssl версии 1.1.1
- java версии 1.8 и выше
- при использовании MSSQL необходим Microsoft ODBC драйвер версии 17
- php 7.0 и выше
- модули php
- bcmath
- fpm
- mbstring
- stomp
- ldap
- gd
- pgsql
- curl
- opcache
- soap
- zip
- xml
- mongodb
- dbase
- rar
- sqlsrv(при использовании MSSQL базы данных)

### Структура папок



```
└─ export
└─ uploads
```

### Описание файлов и папок

/etc/nginx/nginx.conf	Основной конфиг nginx
/etc/nginx/conf.d/promed.conf	Конфиг ЕЦП.МИС
/etc/php/7.2/fpm/php-fpm.conf	Основной конфиг PHP
/etc/php/7.2/fpm/pool.d/ <a href="#">www.conf</a>	конфиг php для ЕЦП.МИС
/etc/ssl/openssl.cnf	Openssl конфиг
/srv/hosts/promed	путь для приложения ЕЦП.МИС
/srv/hosts/promed/export	папка со сгенерированными документами, например ответы
/srv/hosts/promed/uploads	папка с документами, что загружают пользователи

в папке export необходимо создать самостоятельно следующие папки:

```
./register_files
./registry_es
./ostat_registry_files
./reports
./vzn_register
./dd_files
./attach_list
./medpersonal_list
./staff_files
./cryptcp_temp
./jnvlp_price_files
./pl_files
./miac_export
./registry_es_files
./person_card_inform
./stickfssdata_files
./template
./pdf_print
./export_morbus_onko_data
./evn_prescr_mse
./measures_rehab
./medsvid_blanks
./bsme_files
./mes_files
./person_disp_list
./medpersonal_data_frmf
./nolos_person_register
./egisso_receipt
./do_files
./hosp_data_for_tfoms
./vk_journals
./labordep_files
./rhash_temp
./signed_files
./person_polis_list
./QueryToDbf
./disp_list
./frl_import
```

- ./attached\_list
- ./emd\_files
- ./ps\_files
- ./pc\_files
- ./rrl\_files
- ./lpu\_staff\_qwerty\_reg\_fond\_files

В папке uploads необходимо самостоятельно создать следующие папки:

- ./messages
- ./geozones
- ./Directory\_files
- ./importRegistryFromTFOMS
- ./drugs
- ./mce
- ./doc\_normative\_files
- ./users
- ./importHospDataFromTfoms
- ./RgistryFields
- ./importPersonRPN
- ./pmmedia
- ./evnmedia
- ./personcardattaches
- ./kazakh\_project
- ./audioCalls
- ./persons
- ./mseattaches
- ./orgs

Папкам export и uploads необходимо дать права на запись, так же права на запись необходимо дать папкам logs, если используете логин папке внутри приложения и папке ./vendor/mpdf/mpdf/tmp

```

apt-get install software-properties-common curl tzdata -y ;\
ln -sf /usr/share/zoneinfo/$TZ /etc/localtime ;\
dpkg-reconfigure -f noninteractive tzdata ;\
add-apt-repository ppa:nginx/stable -y ;\
curl https://packages.microsoft.com/keys/microsoft.asc > microsoft.asc
;\
apt-key add microsoft.asc ;\
curl https://packages.microsoft.com/config/ubuntu/18.04/prod.list >
/etc/apt/sources.list.d/mssql-release.list ;\
apt-get update ;\
ACCEPT_EULA=Y apt-get install -y nginx openjdk-8-jre php7.2 php7.2-
bcmath php7.2-fpm php7.2-mbstring php7.2-dev php-stomp php7.2-ldap php7.2-
gd php7.2-pgsql php7.2-curl php7.2-opcache php7.2-soap php7.2-zip php7.2-
xml php-pear msodbcsql17 mssql-tools unixodbc unixodbc-dev libengine-gost-
openssl1.1 ;\
ln -snf /opt/mssql-tools/bin/sqlcmd /usr/bin/sqlcmd ;\
ln -snf /opt/mssql-tools/bin/bcp /usr/bin/bcp ;\
pecl install mongodb sqlsrv pdo_sqlsrv dbase rar ;\
echo "extension=sqlsrv.so" > /etc/php/7.2/fpm/conf.d/sqlsrv.ini ;\
echo "extension=pdo_sqlsrv.so" >
/etc/php/7.2/fpm/conf.d/pdo_sqlsrv.ini ;\
echo "extension=mongodb.so" > /etc/php/7.2/fpm/conf.d/mongodb.ini ;\
echo "extension=dbase.so" > /etc/php/7.2/fpm/conf.d/dbase.ini ;\
echo "extension=rar.so" > /etc/php/7.2/fpm/conf.d/rar.ini

```

**Code Block 36** Пример установки ПО на ubuntu 18.04 с MSSQL драйвером и модулем php

```

user www-data;
worker_processes 1;
pid /run/nginx.pid;
include /etc/nginx/modules-enabled/*.conf;
worker_rlimit_nofile 16384;
events {
    worker_connections 2048;
}
http {
    sendfile on;
    tcp_nopush on;
    tcp_nodelay on;
    keepalive_timeout 65;
    types_hash_max_size 2048;
    server_tokens off;
    charset utf-8;
    client_max_body_size 400M;
    include /etc/nginx/mime.types;
    default_type application/octet-stream;
    access_log /var/log/nginx/access.log;
    error_log /var/log/nginx/error.log warn;
    include /etc/nginx/conf.d/*.conf;
    #include /etc/nginx/sites-enabled/*;
}

```

**Code Block 37** nginx.conf

```

# REG_NAME нужно заменить на название имя непосредственно того региона,
который вы настраиваете, например krasnoyarsk
server {
    listen 2080;
    root /srv/promed;
    access_log /var/log/nginx/promed.access.log;
    error_log /var/log/nginx/promed.error.log;
    location ~ /\.php$ {
        charset          utf-8;
        try_files $uri = 404;
        include fastcgi_params;
        fastcgi_pass      unix:/var/run/php/php7.2-fpm.sock;
        fastcgi_read_timeout 3600;
        fastcgi_send_timeout 3600;
        fastcgi_index index.php;
        fastcgi_param     REGION REG_NAME;
        fastcgi_param     SCRIPT_FILENAME $document_root$fastcgi_script_name;
    }
    location /
    {
        charset          utf-8;
        index index.php index.html index.htm;
        try_files $uri $uri/ /index.php$query_string;
    }
}

```

**Code Block 38 promed.conf**

```

[global]

pid = /run/php/php7.2-fpm.pid

error_log = /var/log/php7.0-fpm.log

; Эти 3 строки связаны с тем чтобы очищать незавершенные и зависшие
процессы php

emergency_restart_threshold = 12

emergency_restart_interval = 65s

process_control_timeout = 10s

; количество процессов

process.max = 1536

; подключение конфига php по аналогии с php.ini

include=/etc/php/7.2/fpm/pool.d/*.conf

```

**Code Block 39 php-fpm.conf**

```

[www]
user = www-data
group = www-data
listen = /run/php/php7.2-fpm.sock
listen.owner = www-data
listen.group = www-data
pm = dynamic
pm.max_children = 460
pm.start_servers = 8
pm.min_spare_servers = 6
pm.max_spare_servers = 24
pm.process_idle_timeout = 60s;
pm.max_requests = 1536
env[TMP] = /tmp
env[TMPDIR] = /tmp
env[TEMP] = /tmp
php_flag[display_errors] = off
php_admin_value[error_log] = /var/log/fpm-php.www.log
php_admin_flag[log_errors] = on
php_admin_value[memory_limit] = 2048M
php_admin_value[date.timezone] = Europe/Moscow
php_admin_value[post_max_size] = 400M
php_admin_value[upload_max_filesize] = 400M
php_admin_value[max_execution_time] = 3600
php_admin_value[session.gc_maxlifetime] = 7200

```

#### Code Block 40 www.conf

#### Примечание

Указанные ниже настройки зависят от мощности сервера. но надо быть с ними осторожными подробнее <https://www.php.net/manual/ru/install.fpm.configuration.php>

```

pm.max_children = 460
pm.start_servers = 8
pm.min_spare_servers = 6
pm.max_spare_servers = 24

```

```

#пишем в начале файла. например 3й не закомментируемой строкой
openssl_conf = openssl_def

#Пишем в конце файла
[openssl_def]
engines = engine_section
[engine_section]
gost = gost_section
[gost_section]
engine_id = gost
dynamic_path = /usr/lib/x86_64-linux-gnu/engines-1.1/gost.so
default_algorithms = ALL
CRYPTO_PARAMS = id-Gost28147-89-CryptoPro-A-ParamSet

```

#### Code Block 41 openssl.conf

## 17 Установка сервера хранения справочников, логов, кэша, сессий

Для функционирования Системы необходима установка и настройка СУБД MongoDB. Используется последняя стабильная версия mongodb 4.2

Функциональное назначение СУБД MongoDB - хранение локальных справочников и кэширование данных.

Установка СУБД производится на большинстве из платформ в соответствии с инструкциями официального сайта:

- инструкция по установке для Windows - <https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows/>
- инструкция по установке для Linux - <https://docs.mongodb.com/manual/administration/install-on-linux/>
- инструкция по установке для macOS - <https://docs.mongodb.com/manual/tutorial/install-mongodb-on-os-x/>

Примечание - Ссылки приведены в ознакомительных целях, месторасположение инструкций может быть изменено владельцем сайта.

Дополнительных настроек, кроме указанных в инструкциях по установке (приведенных по ссылкам выше), не требуется. По умолчанию используемый порт подключения к СУБД - 27017.

Инсталляция СУБД MongoDB производится либо на отдельно выделенном сервере для СУБД, либо на специализированном сервере, предназначенном для работы сервисов и приложений.

После установки СУБД MongoDB требуется загрузка локальных справочников и настройка конфигурационных файлов web-серверов:

- promed/condig/mongodb.php — файл конфигурации БД справочников.
- promed/condig/mongodblog.php — файл конфигурации БД логов пользователей.
- promed/condig/mongodbcache.php — файл конфигурации БД кэша приложения.
- promed/condig/mongodbsessions.php — файл конфигурации БД сессий пользователей.

В указанных конфигурациях указывается:

- \$config['mongo\_host'] — настройка IP машины с сервером mongodb;
- \$config['mongo\_port'] — настройка порта сервера mongodb;
- \$config['mongo\_db'] — имя БД на сервере mongodb;

Для мониторинга производительности и управления СУБД используется менеджер студия, например, Robotmongo, устанавливаемая на рабочем месте администратора.

Загрузка функциональных справочников производится согласно инструкции:

1. Авторизуйтесь в системе.
2. Откройте АРМ Администратора ЦОД, если заходите в первый раз, возможно у вас не откроется АРМ Администратор ЦОД, то в этом случае нужно открыть консоль браузера и ввести следующую команду **getWnd('swDBLocalVersionWindow').show();**
3. Нажмите кнопку «Система», выберите подпункт «Управление версиями локальных справочников». Отобразится форма состоящая из разделов:
  1. Версии - данный раздел содержит уже сгенерированные версии справочников.
  2. Справочники версии - можно посмотреть какие справочники и в какую версию вошли.



3. Все справочники - перечень справочников, доступных для регенерирования.
  - Добавьте локальный справочник. Для этого:
    1. В секции «Все справочники» нажмите кнопку "Добавить", откроется форма добавления справочника.
    2. Заполнить поля: Наименование (Название справочника), Префикс (совпадает с наименованием), Краткое наименование (совпадает с наименованием), Схема (Схема в которой находится справочник), Ключ-поле (Ключевое поле таблицы, обычно имя и \_id), Модуль (Promed).
    3. Нажмите кнопку «Сохранить» и сгенерируйте его.

**ВАЖНО. Перед самым первым формированием справочников нужно убедиться, что у базы указана правильная региональность. Если справочники сформировали раньше, то можно удалить базу со справочниками и повторить формирование.**

## 18 Фоновые процессы системы

### 18.1 Фоновые процессы базы данных.

#### 18.1.1 Создание реестра счетов

Для создания реестров счетов из очереди создается задание:

```
* /5 * * * * sudo -u postgres psql -d promedtest -U srv_rep50 -c "SELECT r50.registry_processing_region();"
* /5 * * * * sudo -u postgres psql -d promedtest -U srv_rep60 -c "SELECT r60.registry_processing_region();"
* /2 * * * * (/bin/date && sudo -u postgres psql -d promedadygea -U srv_rep1 -c "CALL r1.registry_processing_region();") #>>
/home/zartdinov/promedadygea.log 2>&1 #При выполнении скрипта создается файл с именем promedadygea.log и записывает информацию о выполнении скрипта.
* /2 * * * * (/bin/date && sudo -u postgres psql -d promedtest -U srv_rep1 -c "CALL r1.registry_processing_region();") #>>
/home/zartdinov/promedtest.log 2>&1 #При выполнении скрипта создается файл с именем promedtest.log и записывает информацию о выполнении скрипта.
```

Периодичность запуска от 2 до 5 мин.

Имя схемы зависит от настроек региона.rpt.registry\_lvn\_processing\_region();

#### 18.1.2 Чистка лога действия пользователя.

Задания запускаются на БД PHP\_LOG

Очистка лога с отправкой удаленного в архив

```
select dbo.clear_PHPLog2(mont:=1);
0 3 * * 1-5 (/bin/date && sudo -u postgres psql -d php_log -U srv_rep1 -c "select dbo.clear_PHPLog2(1);") >> /home/zartdinov/1-5.log 2>&1
```

mont кол-во месяцев, от текущей даты, данные за этот период остаются в логе, все что старше переносится в архив лог, по умолчанию =1.

Периодичность запуска с понедельника по пятницу раз в день.

Очистка архивного лога.

```
select dbo.clear_PHPLogArchive(mont:=3);
0 3 * * 6-0 (/bin/date && sudo -u postgres psql -d php_log -U srv_rep1 -c "select dbo.clear_PHPLogArchive(3);") >> /home/zartdinov/6-0.log 2>&1
```

mont кол-во месяцев, от текущей даты, данные за этот период остаются в архивном логе, все что старше удаляется, по умолчанию =3.

Периодичность запуска с суббота, воскресенье раз в день.

### 18.1.3 Формирование ЛВН

```
*/5 * * * * sudo -u postgres psql -d promedtest -U web_promed50 -c "select  
rpt.registry_lvn_processing_region();"
*/5 * * * * sudo -u postgres psql -d promedadygea -U web_promed1 -c  
"select rpt.registry_lvn_processing_adygea_region();"
```

### 18.1.4 Vacum

```
0 4 * * * su - postgres -c "vacuumdb --all --analyze --verbose"
```

### 18.1.5 Чистка логов PostgreSQL

```
0 0 * * * find /dbpostgres/postgresql/log/*.log -ctime +0 -delete
```

### 18.1.6 Обновление мат представлений

```
0 5 * * * sudo -u postgres psql -d promedlistest2 -c "select  
xp_materializedview_refresh()"
```

### 18.1.7 Backup

```
0 0 * * * /opt/scripts/backupfull.sh
```

### 18.1.8 Auto kill

```
*/10 * * * * sudo -u postgres psql -d postgres -c "SELECT  
pg_terminate_backend(pid) FROM pg_stat_activity WHERE pid <>  
pg_backend_pid() AND state = 'idle' AND state_change < current_timestamp -  
INTERVAL '10' MINUTE;"
```

### 18.1.9 Collect data iops for zabbix

```
* * * * * /etc/zabbix/scripts/iostat-collect.sh /tmp/iostat.out 60
```

## 18.2 Фоновые процессы WEB сервера

```
*/20 0-7 * * * /usr/bin/curl  
'http://URL_ADDRESS/?c=PhpLogService&m=transferDataFromMongoDB&swtoken=<по  
дставляем токен пользователя от которого запускать команду>'  
данное задание на прикладе веб-сервера ЕЦП.МИС пишет отдельный лог в  
котором можно смотреть ход работы - TransferDataFromMongoDB
```

**Code Block 42 Задание для запуска переноса логов из MongoDB в основную БД**

## **ЧАСТЬ 2 – Сервисы для взаимодействия с внешними федеральными системами**

# 1 swan-api

## 1.1 Разворачивание

### 1.1.1 Установка с применением docker

#### 1.1.1.1 Необходимое ПО

- docker v19.03
- docker-compose 1.25

#### 1.1.1.2 Структура папок

/srv/hosts/swan-api - основная папка сервиса  
├── api.yml - конфиг файл  
└── logs - папка с логами

#### 1.1.1.3 docker-compose.yml

```
version: "2"
services:
  swan-api:
    image: docker.promedweb.ru/swan-api:1.0.50
    container_name: swan-api_promed
    volumes:
      - /srv/hosts/swan-api/logs:/opt/tomee/logs
      - ./api.yml:/opt/tomee/conf/api.yml
    environment:
      - CATALINA_OPTS=-Xms512M -Xmx2048M -server -XX:+UseParallelGC -
Dfile.encoding=UTF-8
      - TIMEZONE=Asia/Yekaterinburg
    ports:
      - "8082:8080"
```

**Code Block 43 docker-compose.yml**

### 1.1.2 Установка напрямую на ОС

#### 1.1.2.1 Необходимое ПО

#### 1.1.2.2 Установка TomEE

#### 1.1.2.3 Настройка TomEE как сервиса

## 1.2 Описание файлов конфигурации.

```

# Настройки задаются в формате YAML
# http://www.yaml.org/
# https://ru.wikipedia.org/wiki/YAML

# ! Отступы - это не просто так, они задают структуру документа, см.
описание формата!

# Общие настройки API
api:
  # время жизни сессии
  # sessionTTL: 3600

  # загружать драйвер для профилирования запросов jdbc? В случае
отключения адреса вида jdbc:simon:* могут не работать
  # profilingUseJdbcDriver: true

  # загружать файл конфигурации для профилирования?
  profilingUseConfig: true

# специфические настройки для различных модулей
app:
  # интеграция с Архимед
  archimed:
    # идентификатор шаблона для /archimed/EvnFuncDiagResult
    EvnFuncDiagResult.xmlTemplateBaseId: 33

# Настройки подключения к LDAP. Необходимо использовать тот же сервер что
и для ПроМед
ldap:
  # URL LDAP-сервера
  url: <LDAP_IP>
  # порт
  # port: 389
  # домен авторизации
  domain: dc=еср,dc=ru
  # пользователь
  user: cn=admin
  # пароль
  pass: <LDAP_PASS>
  # путь для запросов групп
  # groupPath: ou=Groups
  # путь для запросов пользователей
  # userPath: ou=Users

# Настройки подключения к БД
db:
  # драйвер для подключения
  driver: org.postgresql.Driver

  # Настройки по регионам:

  # Если для региона в API не используются подключения с различными ролями
то указываем:
  # * ид региона - r2, r59.. или perm, ufa..
  # * url - URL для подключения к БД
  # * username - имя пользователя
  # * password - пароль
  # * isDefault - признак "региона по умолчанию", т.е. если в запросах к
API не указан регион - будет использоваться этот.
  # Если регион таковым не является - можно не указывать

  # r2:
  #   url:
jdbc:sqlserver://АДРЕС_СЕРВЕРА;databaseName=ИМЯ_БД;selectMethod=cursor
  #   username: ПОЛЬЗОВАТЕЛЬ
  #   password: ПАРОЛЬ

```

```

#   isDefault: true

# Если используются подключения с различными ролями, например API должен
# работать с основной и БД СМП, то подключения указываются в секции
connects:
# * ид региона - r2, r59.. или perm, ufa..
# * role - "роль" БД: рабочая - MAIN, реестровая - REGISTRY, отчетная -
REPORT, основная СМП - EMERGENCY_MAIN, реестровая СМП - EMERGENCY_REGISTRY
# * url - URL для подключения к БД
# * username - имя пользователя
# * password - пароль
# * isDefault - признак "коннекта по умолчанию", т.е. основная БД с
# которой работает API. Либо не указываем эту опцию если это не основная БД

# r2:
#   connects:
#     - role: MAIN
#       url:
jdbc:simon:sqlserver://АДРЕС_СЕРВЕРА;databaseName=ИМЯ_БД;selectMethod=curs
or
#       username: ПОЛЬЗОВАТЕЛЬ
#       password: ПАРОЛЬ
#     - role: EMERGENCY_MAIN
#       isDefault: true
#       url:
jdbc:simon:sqlserver://АДРЕС_СЕРВЕРА;databaseName=ИМЯ_БД;selectMethod=curs
or
#       username: ПОЛЬЗОВАТЕЛЬ
#       password: ПАРОЛЬ

# ВАЖНО!!!
# ! Для рабочего окружения указываем только один регион с которым
# работаем
# ! Если настроено несколько регионов, то для одного из них необходимо
# задать признак "региона по умолчанию" (isDefault: true).
# Если настроен только один регион - можно не указывать
# ! Если для региона используются соединения с различными ролями (секция
# connects) то необходимо задать для одного из соединений
# признак "по умолчанию" (isDefault: true)

regions:
  r50:
    role: MAIN
    url:
jdbc:postgresql://<DB_HOST>/<DB_NAME>?user=<DB_USER>&password=<DB_PASS>&st
ringtype=unspecified
    username: <DB_USER>
    password: <DB_PASS>
  r2:
    driver: com.microsoft.sqlserver.jdbc.SQLServerDriver
    url:
jdbc:sqlserver://<DB_HOST>;databaseName=<DB_NAME>;selectMethod=cursor
    username: <DB_USER>
    password: <DB_PASS>

```

#### Code Block 44 api.yaml (PostgreSQL)

Необходимо заменить <IP\_SERVER>

<LDAP\_HOST> - ip адрес сервера ldap

<LDAP\_PASS> - пароль сервера ldap (юзер прописал для примера в конфиге. они должны быть стандартные.)



<DB\_HOST> - IP Адрес Базы данных

<DB\_NAME> - Имя Базы данных

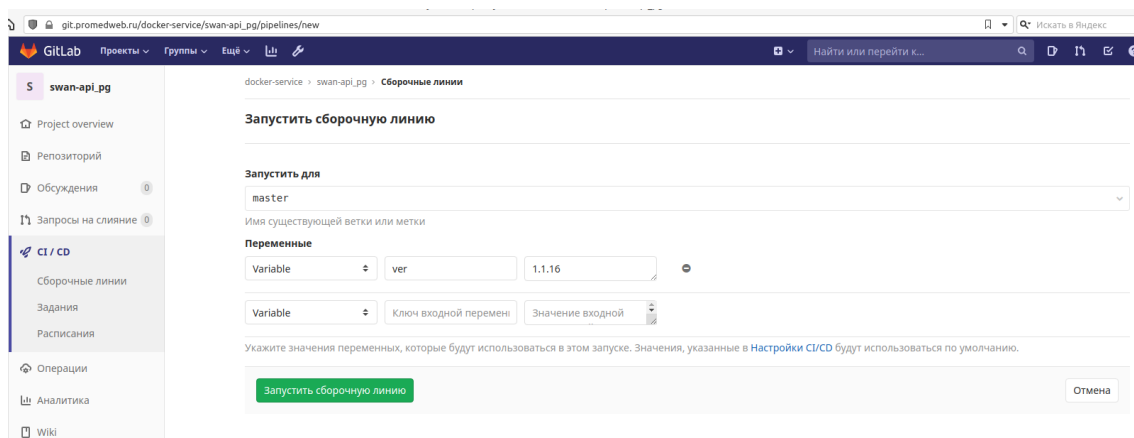
<DB\_USER> - Пользователь БД

<DB\_PASS> - Пароль от БД

## 1.3 Обновление

Для обновления MSSQL версии нужно перейти на следующий проект <https://git.promedweb.ru/docker-service/swan-api>

Запустить Pipeline с переменной ver и версией апи. Последняя на данный момент версия для 1.0.50



## 1.4 Действия после установки

Можно настроить nginx для работы с swan-api. Например для работы 1С или Android-приложений для СМП.

```
server {
    listen 8081;
    server_name api.ecp-mis.ru;
    access_log /var/log/nginx/api_acc.log main buffer=64k;
    error_log /var/log/nginx/api_err.log;
    open_file_cache_errors off;

    location /swan-api/ {
        index index.php;
        proxy_pass http://<IP_SERVER>:<PORT_SERVER>/swan-api/;
        proxy_redirect off;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    }
}
```

Code Block 45 api.conf

## 2 ЕСИА (ScanCodeService)

Операционная система:

OS	Version	Bit
CentOS	8.2.2004	x86_64 bit

Программный комплекс:

Application	Version
Java	1.8.0-openjdk-devel
CriptoPRO JCP	2.0.40035
Apache Tomcat (TomEE)	8.5.50(7.0.7)
net-tools	
unzip	

### Java

Обновляем пакеты операционной системы:

```
$ sudo dnf -y update
```

Проверяем версию java:

```
$ sudo java -version
```

Если java не установлена (отличается версия):

```
zartdinov@localhost:~
```

```
[zartdinov@localhost ~]$ sudo java -version
[sudo] password for zartdinov:
sudo: java: command not found
[zartdinov@localhost ~]$
```

Ищем нужную нам версию java:

```
$ sudo dnf search 1.8.0-openjdk
```

```
zartdinov@localhost:~$ sudo dnf search 1.8.0-openjdk
Last metadata expiration check: 1:32:27 ago on Fri 24 Jul 2020 01:42:49 PM +05.
===== Name Matched: 1.8.0-openjdk =====
java-1.8.0-openjdk.x86_64 : OpenJDK Runtime Environment 8
java-1.8.0-openjdk-src.x86_64 : OpenJDK Source Bundle 8
java-1.8.0-openjdk-demo.x86_64 : OpenJDK Demos 8
java-1.8.0-openjdk-devel.x86_64 : OpenJDK Development Environment 8
java-1.8.0-openjdk-javadoc.noarch : OpenJDK 8 API documentation
java-1.8.0-openjdk-headless.x86_64 : OpenJDK Headless Runtime Environment 8
java-1.8.0-openjdk-javadoc-zip.noarch : OpenJDK 8 API documentation compressed in single archive
java-1.8.0-openjdk-accessibility.x86_64 : OpenJDK 8 accessibility connector
[zartdinov@localhost ~]$
```

Устанавливаем java-1.8.0-openjdk-devel.x86\_64:

```
$ sudo dnf -y install java-1.8.0-openjdk-devel.x86_64
```

Проверяем версию java:

```
$ sudo java -version
```

```
zartdinov@localhost:~$ sudo java -version
openjdk version "1.8.0_262"
OpenJDK Runtime Environment (build 1.8.0_262-b10)
OpenJDK 64-Bit Server VM (build 25.262-b10, mixed mode)
[zartdinov@localhost ~]$
```

### CriptoPRO JCP

Для загрузки [CriptoPRO](#) понадобится зарегистрироваться на портале [CriptoPRO](#), так же актуальная версия CriptoPRO будет выложена на корпоративном [Cloud](#).

Скачиваем CriptoPRO с корпоративного [Cloud](#):

```
$ curl -o jtls.zip https://cloud.swan-it.ru/index.php/s/oPLFt84CXeYMgzb/download?path=%2FJCP%2FJavaTLS%2Fjcp-2.0.40035&files=jcp-2.0.40035.zip
```

Распакуем jtls.zip:

```
$ unzip jtls.zip
```

Распакуем jcp-2.0.40035.zip в каталог /opt:

```
$ sudo unzip jcp-2.0.40035/jcp-2.0.40035.zip -d /opt/
```

Назначим файлы с расширением `.sh` исполняемыми:

```
$ sudo chmod +x /opt/jcp-2.0.40035/*.sh
```

Перейдем в каталог `jcp-2.0.40035`:

```
$ cd /opt/jcp-2.0.40035
```

Проверяем используемую по умолчанию `java` и путь до каталога:

```
$ sudo alternatives --config java
```

Запускаем установку `jcp`:

```
$ sudo ./setup_console.sh /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.262.b10-0.e18_2.x86_64/ -force -en -install -jcp -jcryptop -cades -jre /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.262.b10-0.e18_2.x86_64/
```

Скопируем `jcp dependencies` библиотеки в каталог `ext`:

```
$ sudo cp /opt/jcp-2.0.40035/dependencies/* /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.262.b10-0.e18_2.x86_64/jre/lib/ext/
```

Скопируем `jcp javadoc` библиотеки в каталог `ext`:

```
$ sudo cp /opt/jcp-2.0.40035/javadoc/* /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.262.b10-0.e18_2.x86_64/jre/lib/ext/
```

Проверяем лицензию `CriptoPRO JCP`:

```
$ java ru.CriptoPro.JCP.tools.License
```

zartdinov@localhost:~

```
[zartdinov@localhost ~]$ java ru.CryptoPro.JCP.tools.License
License verify:
Type: Server, sign and encrypt
Allowed amount of cores: Unlimited
Serial number: CF20X-X0030-00BAA-1F
Validity: Until Oct 25, 2020
Valid license.
[zartdinov@localhost ~]$ █
```

## ESIA

Актуальная версия ЕСИА (ScanCodeServices) будет выкладываться на корпоративном [Cloud](#).

Создадим группу tomee:

```
$ sudo groupadd tomee
```

Создадим пользователя tomee:

```
$ sudo useradd tomee -s /bin/false -g tomee -d /opt/tomee
```

Скачиваем актуальную версию ЕСИА (ScanCodeServices) с корпоративного [Cloud](#):

```
$ curl -o ScanCodeServices.zip https://cloud.swan-
it.ru/index.php/s/8FaGYLEneRTmi8S/download?path=%2FTomEE%2FScanCodeService
s&files=ScanCodeServices_v202003.zip
```

Распакуем ScanCodeServices.zip:

```
$ unzip ScanCodeServices.zip
```

Распакуем ScanCodeServices\_v202003.zip в каталог /opt:

```
$ sudo unzip ScanCodeServices/ScanCodeServices_v202003.zip -d /opt/tomee
```

Назначим файлы с расширением .sh исполняемыми:

```
$ sudo chmod +x /opt/tome/bin/*.sh
```

Назначаем владельцем каталога /opt/tomee/ tomee:

```
$ sudo chown -R tomee: /opt/tomee/
```

Установим редактор vim:

```
$ sudo dnf -y install vim
```

## Настроим tomee

Сделаем резервную копию конфигурационного файла tom:

```
$ sudo cp /opt/tomee/conf/tomcat-users.xml /opt/tomee/conf/tomcat-  
users.xml.old
```

Отредактируем конфигурационный файл tomcat-users.xml:

```
$ sudo vim /opt/tomee/conf/tomcat-users.xml
```

Было:

```
zartdinov@localhost:~$  
[root@localhost ~]# cat /opt/tomee/conf/tomcat-users.xml  
 Licensed to the Apache Software Foundation (ASF) under one or more  
 contributor license agreements. See the NOTICE file distributed with  
 this work for additional information regarding copyright ownership.  
 The ASF licenses this file to You under the Apache License, Version 2.0  
 (the "License"); you may not use this file except in compliance with  
 the License. You may obtain a copy of the License at  
  
 http://www.apache.org/licenses/LICENSE-2.0  
  
 Unless required by applicable law or agreed to in writing, software  
 distributed under the License is distributed on an "AS IS" BASIS,  
 WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
 See the License for the specific language governing permissions and  
 limitations under the License.  
 <tomcat-users version="1.0" xmlns="http://tomcat.apache.org/xml" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd">  
 <!-- NOTE: By default, no user is included in the "manager-gui" role required  
 to operate the "/manager/html" web application. If you wish to use this app,  
 you must define such a user - the username and password are arbitrary. It is  
 strongly recommended that you do NOT use one of the users in the commented out  
 section below since they are intended for use with the examples web  
 application. -->  
 <!-- NOTE: The sample user and role entries below are intended for use with the  
 examples web application. They are wrapped in a comment and thus are ignored  
 when reading this file. If you wish to configure these users for use with the  
 examples web application, do not forget to remove the <!-- ... --> that surrounds  
 them. You will also need to set the passwords to something appropriate. -->  
 <!--  
 <role rolename="tomcat"/>  
 <role rolename="tomcat1"/>  
 <user username="tomcat" password="must-be-changed" roles="tomcat"/>  
 <user username="both" password="must-be-changed" roles="tomcat,role1"/>  
 <user username="role1" password="must-be-changed" roles="role1"/>  
 <!--  
 <!-- Activate these lines to get access to TOMEE GUI if added (tomee-webaccess) -->  
 <!--  
 <role rolename="tomee-admin" />  
 <user username="tomee" password="tomee" roles="tomee-admin,manager-gui" />  
 <!--  
 </tomcat-users>
```

Стало:

```
zartdinov@localhost:~$ cat tomcat-users.xml
<?xml version="1.0" encoding="UTF-8"?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<tomcat-users version="1.0" xmlns="http://tomcat.apache.org/xml" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd">
  <!-- NOTE: By default, no user is included in the "manager-gui" role required
to operate the "/manager/html" web application. If you wish to use this app,
you must define such a user - the username and password are arbitrary. It is
strongly recommended that you do NOT use one of the users in the commented out
section below since they are intended for use with the examples web
application. -->
  <!-- NOTE: The sample user and role entries below are intended for use with the
examples web application. They are wrapped in a comment and thus are ignored
when reading this file. If you wish to configure these users for use with the
examples web application, do not forget to remove the <!-- ... --> that surrounds
them. You will also need to set the passwords to something appropriate. -->
  <!--
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <user username="tomcat" password="tomcat-changed" roles="tomcat"/>
  <user username="tomcat" password="must-be-changed" roles="tomcat,role1"/>
  <user username="role1" password="must-be-changed" roles="role1"/>
  <!--
  <!-- Activate those lines to get access to TomEE GUI if added (tomcat-webapp) -->
  <!--
  <role rolename="tomcat-admin" />
  <user username="tomcat" password="tomcat" roles="tomcat-admin,manager-gui" />
  <!--
  <role rolename="admin-gui,manager-gui"/>
  <user username="login" password="password" roles="admin-gui,manager-gui"/>
  </tomcat-users>
-->
</tomcat-users>
```

В конфигурационный файл tomcat-users.xml добавили:

```
<role rolename="admin-gui,manager-gui"/>
<user username="login" password="password" roles="admin-gui,manager-gui"/>
```

Сделаем резервную копию конфигурационного файла context.xml:

```
$ sudo cp /opt/tomee/webapps/manager/META-INF/context.xml
/opt/tomcat/webapps/manager/META-INF/context.xml.old
```

Отредактируем конфигурационный файл context.xml:

```
$ sudo vim /opt/tomee/webapps/manager/META-INF/context.xml
```

Было:

```
zartdinov@localhost:~$ cat context.xml
<?xml version="1.0" encoding="UTF-8"?>
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the license at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" >
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127.0.0.1,:::1" />
  <Manager sessionAttributeValueClassNameFilter="java.lang.(?:Boolean|Integer|Long|Number|String|org.apache.catalina.filters.CsrfPreventionFilter|Set|Cache|List|Map|Util|LinkedHashMap|>
</Context>
```

Стало:

```
zartdinov@localhost:~$ cat /etc/tomcat/conf/context.xml
<!--
Licensed to the Apache Software Foundation (ASF) under one or more
contributor license agreements.  See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
The ASF licenses this file to you under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License.  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
-->
<Context antiResourceLocking="false" privileged="true" >
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" />
  <Manager className="org.apache.catalina.valves.RemoteAddrValve"
    allow="\d+\.\d+\.\d+\.\d+" />
</Context>
```

В конфигурационном файле context.xml изменили:

Было:

```
<Valve className="org.apache.catalina.valves.RemoteAddrValve"
  allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" />
```

Стало:

```
<Valve className="org.apache.catalina.valves.RemoteAddrValve"
  allow="\d+\.\d+\.\d+\.\d+" />
```

Установим сетевую утилиту net-tools:

```
$ sudo dnf -y install net-tools
```

Смотрим не занят ли порт 8080:

```
$ sudo netstat -tulnt | grep "8080"
```

Если порт 8080 не занят идем в раздел создания конфигурационного файла старта, остановки и автостарта tomee:

```
zartdinov@localhost:~$ sudo netstat -tulnp | grep "8080"
[zartdinov@localhost ~]$
```

Если порт 8080 занят, меняем его на другой:







```
$ sudo vim /opt/tomee/webapps/ScanCodeService/WEB-INF/classes/config.yaml
```

```
mc [root@localhost.localdomain]:/opt/tomee/webapps/ScanCodeService/WEB-INF/classes
/opt/tomee/webapps/ScanCodeService/WEB-INF/classes/config.yaml
# Configuration
configScanCode:
  # ComPort
  comPort: /dev/ttyACM0
  comPortExcludes: /dev/ttyS0;/dev/ttyS1;COM1
  checkComPorts: false
  checkComPortsInterval: 15

configPrinter:
  # printerName or default
  printerName: default
configLibrary:
  # path to Rutoken native library
  libRutokenFile: rtpksslcp

configScheduler:
  checkTerminals: false
  checkTerminalsInterval: 15

configCADES:
  # url to TSP service https://www.cryptopro.ru/tsp/tsp.srf http://pki.skbkontur.ru/tsp/tsp.srf http://tsp.ncarf.ru/tsp/tsp.srf
  urlTSPService: http://tsp.ncarf.ru/tsp/tsp.srf

configCryptoProJCP:
  keyMediatype: HDImageStore
  keyAlias: test2020
  keyPassword: 12345678

configPfxFile:
  fileName: c:\Work\ScanCodeService\test\pl2.pfx
  filePassword: 12345678
  aliasName: cp_exported
  keyPassword: 12345678
```

В конфигурационном файле config.yaml прописывается keyAlias и keyPassword сертификата:

```
configCryptoProJCP:
  keyMediatype: HDImageStore
  keyAlias: test2020
  keyPassword: 12345678
```

В конфигурационном файле config.yaml прописываем режим работы сканера штрих-кодов и кардридера. Устанавливаем режим опроса сканера checkComPorts: true или отключаем (false), аналогично для кардридера checkTerminals: true. Режим опроса необходим, если предполагается считывание полисов по событиям вставки электронного полиса и сканированию штрих-кода бумажного полиса. Указывается интервал опроса checkComPortsInterval: 15 в миллисекундах. Указывается порт сканера comPort: /dev/ttyACM0.

```
configScanCode:
  comPort: /dev/ttyACM0
  comPortExcludes: /dev/ttyS0;/dev/ttyS1;COM1
  checkComPorts: true
  checkComPortsInterval: 15
configScheduler:
  checkTerminals: true
  checkTerminalsInterval: 15
```

## systemd

Отредактируем конфигурационный файл tomee.service:

```
$ sudo vim /etc/systemd/system/tomee.service
```

```
[Unit]
Description=TomEE
After=network.target

[Service]
Type=forking

Environment=JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.262.b10-0.el8_2.x86_64
Environment=CATALINA_PID=/opt/tomee/temp/tomee.pid
Environment=CATALINA_HOME=/opt/tomee
Environment=CATALINA_BASE=/opt/tomee
Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"
Environment="JAVA_OPTS=-Djava.awt.headless=true -Djava.security.egd=file:/dev/./urandom"

ExecStart=/opt/tomee/bin/startup.sh
ExecStop=/opt/tomee/bin/shutdown.sh

User=tomee
Group=tomee

ExecReload=/bin/kill $MAINPID
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target
```

Перезагружаем конфигурацию сервиса:


```
$ sudo systemctl daemon-reload
```

Запускаем сервис tomee:

```
$ sudo systemctl start tomee.service
```

Проверяем, запущен ли сервис с детальным выводом состояния сервиса:

```
$ sudo systemctl status tomee.service
```



```
zaidinov@localhost:~$ sudo systemctl status tomee.service
● tomee.service - TomEE
   Loaded: loaded (/etc/systemd/system/tomee.service; disabled; vendor preset: disabled)
   Active: active (running) since Mon 2020-09-14 16:58:07 EDT; 3h 5min ago
     Process: 4168 ExecStart=/opt/tomee/bin/startup.sh (code=exited, status=0/SUCCESS)
    Main PID: 4168 (java)
      Tasks: 65 (limit: 11483)
     Memory: 471.3M
     CGroup: /system.slice/tomee.service
            └─171 /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.262.b10-0.el8_2.x86_64/bin/java -Djava.util.logging.config.file=/opt/tomee/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -javaagent:/opt
Jul 27 16:58:07 localhost.localdomain systemd[1]: Starting TomEE...
Jul 27 16:58:07 localhost.localdomain systemd[1]: Started TomEE.
https://www.ibm.com
```

Активируем сервис (позволяет стартовать во время запуска системы):

```
$ sudo systemctl enable tomee.service
```

Открываем порт 8080 на firewall:

```
$ sudo firewall-cmd --add-port=8080/tcp --permanent
```

Перезапускаем firewall:

```
sudo firewall-cmd --reload
```

Создадим каталог tomee:

```
$ sudo mkdir /var/opt/cprocsp/keys/tomee/
```

Распакуем архив esiatest.000.zip:

```
$ sudo unzip esiatest.000.zip -d /var/opt/cprocsp/keys/tomee/
```

Назначаем владельца каталога /var/opt/cprocsp/keys/tomee/ tomee:

```
$ sudo chown -R tomee: /var/opt/cprocsp/keys/tomee/
```

Останавливаем tomee:

```
$ sudo systemctl stop tomee.service
```

Запускаем tomee:

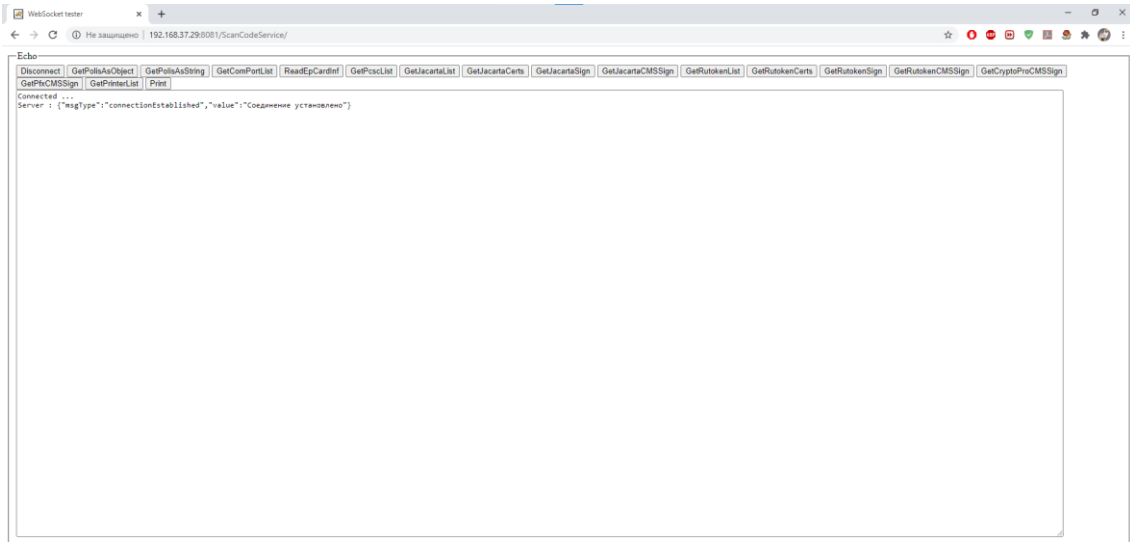
```
$ sudo systemctl start tomee.service
```

## Открываем tomee в браузере:

## Переходим в панель Manager App:

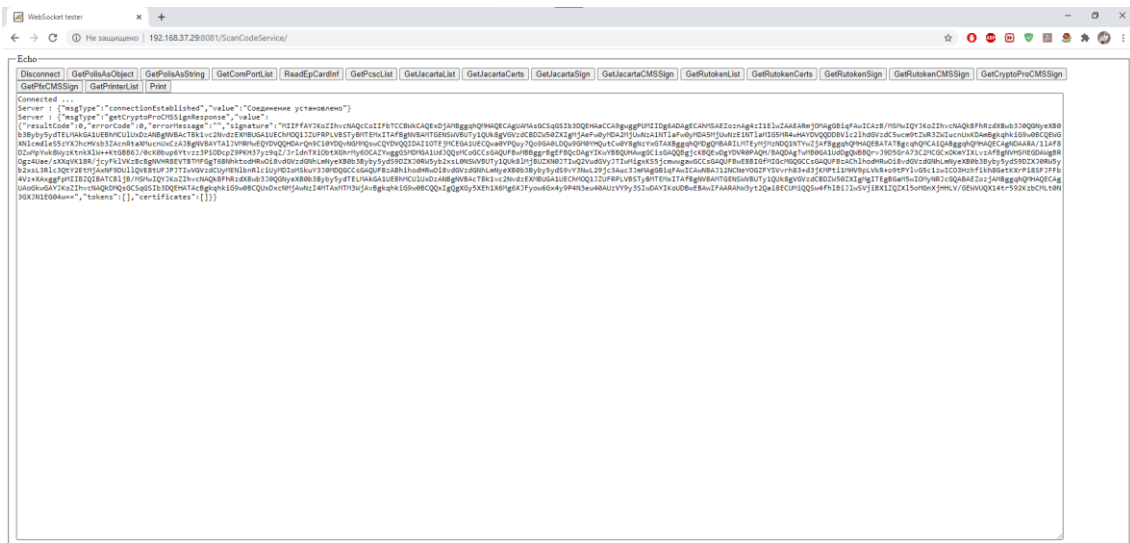
Путь	Версия	Название	Запущено	Сессии	Команды
/	Не указано	Welcome to Tomcat	true	0	Старт   Стоп   Перезагрузить   Удалить Завершить сессии   с неактивностью 2   30 минут
/ScanCodeService	Не указано	ScanCodeService	true	0	Старт   Стоп   Перезагрузить   Удалить Завершить сессии   с неактивностью 2   60 минут
/docs	Не указано	Tomcat Documentation	true	0	Старт   Стоп   Перезагрузить   Удалить Завершить сессии   с неактивностью 2   30 минут
/host-manager	Не указано	Tomcat Host Manager Application	true	0	Старт   Стоп   Перезагрузить   Удалить Завершить сессии   с неактивностью 2   30 минут
/manager	Не указано	Tomcat Manager Application	true	1	Старт   Стоп   Перезагрузить   Удалить Завершить сессии   с неактивностью 2   30 минут

## Переходим в servlet ScanCodeService:



Проверим работу servlet ScanCodeServices нажатием на кнопку GetCryptoProCMSSign:

Если СrptoPRO и TomEE настроены правильно, в ответ должны получить приблизительно такой ответ от сервиса:



## 3 ЕСИА (ScanCodeService) Docker

### 3.1 Описание сервиса:

Служба поддержки внешних устройств ScanCodeService предназначена для работы с внешними устройствами, подключаемыми на клиентском компьютере. Набор устройств включает в себя: токены авторизации (Рутокен ЭЦП, Jacarta, ETокен), электронные полисы, сканеры штрих-кодов. Также служба поддерживает формирование электронной подписи на ключах, записанных в хранилище КриптоПро JCP.

### 3.2 Репозиторий в Git:

### 3.3 Примечание:

В контейнере защиты тестовый сертификат esiatest.000, keystore.jks и [tomcat-users.xml](#) для тестового развертывания.

Для проверки работы веб-сокетов реализована html-страница <http://localhost:8080/ScanCodeService/>. Названия кнопок соответствуют названиям тестируемых методов.(проверка работы servlet ScanCodeServices нажатием на кнопку GetCryptoProCMSSign)

### 3.4 Развертывание с использованием docker-образа

#### 3.4.1 Необходимое ПО

10. docker v19.03
11. docker-compose 1.25

#### 3.4.2 Структура папок монтируемых на host-машину

/usr/local/tomee/

├─ logs - папка с логами

├─ webapps/ScanCodeService/WEB-INF/classes/config.yaml - конфигурационный файл сервиса

├─ webapps/host-manager/META-INF/context.xml - конфигурационный файл доступов

├─ webapps/manager/META-INF/context.xml - конфигурационный файл доступов

├─ conf/tomcat-users.xml - файл настройки пользователей tomcat

├─ conf/keystore.jks - java keystore

└─ constants.js - файл настроек сервера

/var/opt/cproscsp/keys/root/имя\_контейнер\_крипто-про.000

#### 3.4.3 Для развертывания необходимо

- Создать папку развертывания
- В корне папки развертывания создать файл [config.yaml](#)



- Создать папки **logs**
- Скопировать в текущую папку, папку-контейнер крипто-про
- Создать в папке развертывания файл [docker-compose.yaml](#)
- Создать конфигурационный файл доступов tomcat [context.xml](#)
- Создать файл настройки пользователей tomcat [tomcat-users.xml](#)
- Скопировать в текущую папку файл с парами ключей keystore.jks (контейнер сертификатов для закрытия сервиса по https. Ключ для контейнера должен быть "123456", временно, он зашит в конфиг контейнера)
- Проверить синтаксис конфигурации docker-compose командой: **docker-compose config**
- Запустить сервис с помощью команды: **docker-compose up -d**

```

# Раздел содержит настройки сканера.
configScanCode:
  # Наименование COM-порта в системе
  comPort: /dev/ttyACM0
  comPortExcludes: /dev/ttyS0;/dev/ttyS1;COM1
  checkComPorts: false
  checkComPortsInterval: 15

# Раздел содержит настройки принтера.
configPrinter:
  # Наименование принтера в системе (по умолчанию default).
  printerName: default

# path to Rutoken native library
configLibrary:
  libRutokenFile: librtpkcs11esp.so

# Раздел содержит настройки расписания опроса внешних устройств. Опрос
необходим при работе со смарткартами и сканером штрихкода для детекции
подключения устройства и немедленного чтения данных.
# В других случаях опрос излишне потребляет ресурсы.
configScheduler:
  # Включить/выключить опрос устройств
  checkTerminals: false
  # Интервал опроса (сек)
  checkTerminalsInterval: 15

# Раздел содержит параметры для квалифицированной подписи
configCADES:
  # URL службы штампов времени https://www.cryptopro.ru/tsp/tsp.srf
http://pki.skbkontur.ru/tsp/tsp.srf http://tsp.ncarf.ru/tsp/tsp.srf
  urlTSPService: http://tsp.ncarf.ru/tsp/tsp.srf

# Раздел содержит параметры подписи на ключе КриптоПро
configCryptoProJCP:
  keyMediatype: HDImageStore
  # Алиас ключа
  keyAlias: esiatest
  # Пароль ключа
  keyPassword: 12345678

configPfxFile:
  fileName: p12.pfx
  filePassword: 12345678
  aliasName: cp_exported
  keyPassword: 12345678

```

**Code Block 46 config.yaml**

```

version: '3.3'

services:
  scancode:
    image: d-repo.rtmis.ru/promed/scancode:1.0.0
    container_name: scancode
    volumes:
      - ./config.yaml:/usr/local/tomee/webapps/ScanCodeService/WEB-INF/classes/config.yaml
      - ./tomcat-users.xml:/usr/local/tomee/conf/tomcat-users.xml
      - ./logs:/usr/local/tomee/logs/
      # Контейнер JCP
      - ./esiatest.000:/var/opt/cproscsp/keys/root/esiatest.000
      # Контейнер java keystore
      - ./keystore.jks:/usr/local/tomee/conf/keystore.jks
      - ./context.xml:/usr/local/tomee/webapps/host-manager/META-INF/context.xml
      - ./context.xml:/usr/local/tomee/webapps/manager/META-INF/context.xml
    environment:
      TZ: Asia/Yekaterinburg
      # Раскомментировать при наличии лицензии
      #   # KEY LICENSE
      #   JCP_KEY:
      #   # COMPANY NAME
      #   JCP_NAME:
      CATALINA_OPTS: "-Xms512M -Xmx1024M -server -XX:+UseParallelGC"
      JAVA_OPTS: "-Djava.awt.headless=true -Djava.security.egd=file:/dev/./urandom"
    ports:
      - 8080:8080
      - 8089:8089
      - 8443:8443
      - 8005:8005
      - 8009:8009

```

**Code Block 47 docker-compose.yaml**

```

<?xml version="1.0" encoding="UTF-8"?>

<Context antiResourceLocking="false" privileged="true" >
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="\d+\.\d+\.\d+\.\d+" />
  <Manager
    sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Long|Number|string)|org\.apache\.catalina\.filters\.CsrfPreventionFilter$LruCache(?:\$1)?|java\.util\.(?:Linked)?HashMap"/>
</Context>

```

**Code Block 48 context.xml**

```
<?xml version="1.0" encoding="UTF-8"?>

<tomcat-users xmlns="http://tomcat.apache.org/xml"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-
users.xsd"
              version="1.0">

  <role rolename="manager-gui"/>
  <role rolename="admin-gui"/>
  <user username="tomcat" password="secret" roles="manager-gui,admin-gui"/>

</tomcat-users>
```

**Code Block 49 tomcat-users.xml**

## 4 ЕСИА (ScanCodeService) OLD

ОС	Version	Разрядность
CentOS minimal	8.2.2004	x86_64 bit

Application	Version
Java	1.8.0-openjdk-devel
CriptoPro JCP	2.0.40035
Apache Tomcat(TomEE)	8.5.50(7.0.7)
net-tools	
unzip	

Проверяем установлено ли на сервере java:

```
$ sudo java -version
```



```
ssrdz@ssrdz-test:~$ java -version
openjdk version "1.8.0_252"
OpenJDK Runtime Environment (build 1.8.0_252-b09-1-16.04-b09)
OpenJDK 64-bit Server VM (build 25.252-b09, mixed mode)
ssrdz@ssrdz-test:~$
```

Если на сервере не установлена java

Установка Java

Обновляем список packages из repository:

```
$ sudo dnf -y update
```

Устанавливаем Java:

```
$ sudo dnf -y install java-1.8.0-openjdk-devel
```

Установка net-tools

Устанавливаем net-tools:

```
$ sudo apt install net-tools
```

Установка Apache Tomcat (TomEE)/8.5.50 (7.0.7)

Создаем директорию tomee:

```
$ sudo mkdir /opt/tomee
```

Создаем группу tomee:

```
$ sudo groupadd tomee
```

Создаем пользователя tomee:

```
$ sudo useradd -s /bin/false -g tomee -d /opt/tomee tomee
```

Apache Tomcat(TomEE)

[Apache Tomcat\(TomEE\): ScanCodeServices](#)

Пример скачивания Apache Tomcat(TomEE): ScanCodeServices

```
curl -o ScanCodeServices_v202003.zip https://cloud.swan-it.ru/index.php/s/ZaNWYP3zF7DGR9A/download
```

Переходим в каталог tomee:

```
$ cd /opt/tomee
```

```
$ sudo chmod -R g+r conf
```

```
$ sudo chmod g+x conf
```

Назначаем пользователя tomee владельцем каталога tomee:

```
$ sudo chown -R tomee /opt/tomee
```

```
$ sudo update-java-alternatives -l
```

```
$ sudo vim /etc/systemd/system/tomee.service
```

Создаем даемон запуска, остановки и запуска при старте системы:

```

[Unit]
Description=TomEE
After=network.target

[Service]
Type=forking

Environment=JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-amd64
Environment=CATALINA_PID=/opt/tomee/temp/tomee.pid
Environment=CATALINA_HOME=/opt/tomee
Environment=CATALINA_BASE=/opt/tomee
Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"
Environment="JAVA_OPTS=-Djava.awt.headless=true -
Djava.security.egd=file:/dev/./urandom"

ExecStart=/opt/tomee/bin/startup.sh
ExecStop=/opt/tomee/bin/shutdown.sh

User=tomee
Group=tomee

ExecReload=/bin/kill $MAINPID
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target

```

Перезапускаем даемон:

```
$ sudo systemctl daemon-reload
```

Запускаем даемон tomee:

```
$ sudo systemctl start tomcat
```

Проверяем статус запуска даемон tomee:

```
$ systemctl status tomcat
```

```

artimon@emptytest ~$ sudo systemctl status tomcat
● tomcat.service - TomEE
   Loaded: loaded (/etc/systemd/system/tomee.service; disabled; vendor preset: enabled)
   Active: active (running) since Оч 2020-04-17 14:35:26 +03; 5 days ago
     Process: 7106 ExecStop=/opt/tomee/bin/shutdown.sh (code=exited, status=0/FALLING)
     Process: 7130 ExecStart=/opt/tomee/bin/startup.sh (code=exited, status=0/SUCCESS)
    Main PID: 7134 (java)
      Tasks: 54
     Memory: 532.2M
          CPU: 20min 19.48%
     CGroup: /system.slice/tomee.service
            └─7134 /usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java -Djava.util.logging.config.file=/opt/tomee/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Djavaagent=/opt/tomee/lib/openejb-java...

ноя 17 14:35:26 crm-test systemd[1]: Starting TomEE...
ноя 17 14:35:26 crm-test startup.sh[7120]: Existing PID file found during start.
ноя 17 14:35:26 crm-test startup.sh[7130]: Removing/creating stale PID file.
ноя 17 14:35:26 crm-test systemd[1]: Started TomEE.

```

Включаем запуск при старте системы:

```
$ systemctl enable tomcat
```

Настройка TomEE

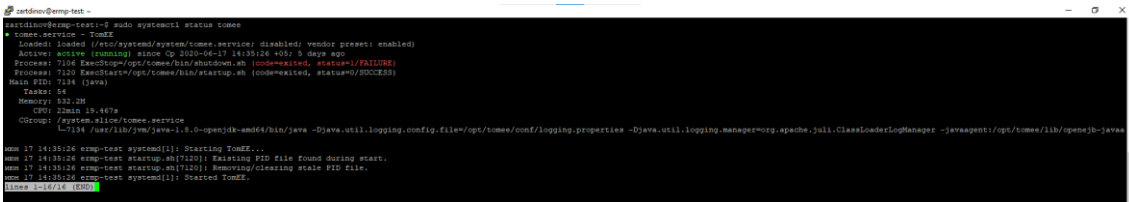
Делаем резервную копию конфигурационного файла tomcat-users.xml:





```
$ sudo vim /opt/tomee/webapps/manager/META-INF/context.xml
```

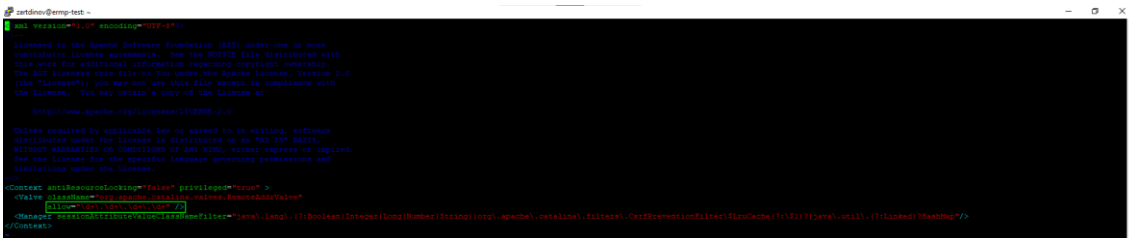
```
<Context antiResourceLocking="false" privileged="true" >  
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"  
    allow="127\.\d+\.\d+\.\d+:::1|0:0:0:0:0:0:1" />  
  <Manager  
    sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Long|  
    Number|String)|org\.apache\.catalina\.filters\.CsrfPreventionFilter\$LruCa  
    che(?:\$1)?|java\.util\.(?:Linked)?HashMap"/>  
</Context>
```



```
artdion@temp-test:~$ sudo systemctl status tomee  
● tomee.service - TomEE  
   Loaded: loaded (/etc/systemd/system/tomee.service; disabled; vendor preset: enabled)  
   Active: active (running) since Ов 2020-06-17 14:35:24 +03; 5 days ago  
     Process: 7106 ExecStop=/opt/tomee/bin/shutdown.sh (code=exited, status=1/kill/9)  
     Process: 7105 ExecStart=/opt/tomee/bin/startup.sh (code=exited, status=0/SUCCESS)  
    Main PID: 7104 (java)  
      Tasks: 54  
     Memory: 532.2M  
        CPU: 23min 19.467s  
   CGroup: /system.slice/tomee.service  
           └─┬─/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java -Djava.util.logging.config.file=/opt/tomee/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -javaagent:/opt/tomee/lib/openejb-java
```

Должно получиться:

```
<Context antiResourceLocking="false" privileged="true" >  
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"  
    allow="\d+\.\d+\.\d+\.\d+" />  
  <Manager  
    sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Long|  
    Number|String)|org\.apache\.catalina\.filters\.CsrfPreventionFilter\$LruCa  
    che(?:\$1)?|java\.util\.(?:Linked)?HashMap"/>  
</Context>
```



```
artdion@temp-test:~$ cat /opt/tomee/webapps/manager/META-INF/context.xml  
<Context antiResourceLocking="false" privileged="true" >  
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"  
    allow="\d+\.\d+\.\d+\.\d+" />  
  <Manager  
    sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Long|  
    Number|String)|org\.apache\.catalina\.filters\.CsrfPreventionFilter\$LruCa  
    che(?:\$1)?|java\.util\.(?:Linked)?HashMap"/>  
</Context>
```

Смотрим используемые port:

```
$ sudo netstat -tulnp
```





## 5 ИЕМК

### 5.1 Разворачивание

Небольшое предисловие по сервису

Как мне сказал разработчик при запуске сервиса он создает необходимые БД и т.д. потому что при первом старте нужны будут права учетной записи выше, чем при использовании потом в работе.

В целом работа идет примерно следующим образом. раз в определенное время (сделал раз в 15 минут) служба репликатора запускается смотрит что в БД поменялось выгружает id этих данных в очередь ActiveMQ. Сервис прокси имеет слушателей в ActiveMQ. забирает id необходимых данных обращается к основной БД через swan-api. Приводит к виду из шаблона(об это ниже), подписывает и отправляет в федеральные службы. Так же он получает callback.

Все логи по работе сервиса swan-iemc-проху с данными которые мы передали и получили он хранит в БД.

#### 5.1.1 Установка с применением docker

5.1.1.1 Необходимое ПО

- docker v19.03
- docker-compose 1.25

5.1.1.2 Структура папок и файлов

```
/srv/hosts/iemk
├── api.yaml
├── keys
├── logs
│   ├── проху
│   └── replicator
```

### 5.1.1.3 docker-compose.yml

```

version: "2"
services:
  activemq_iemk:
    image: docker.k-vrachu.ru/activemq:latest
    container_name: activemq_iemk
    ports:
      - "8161:8161"
      - "61616:61616"
    networks:
      net-fer:
        ipv4_address: 172.22.0.2
  api:
    image: docker.k-vrachu.ru/swan-api:1.0.52
    container_name: swanapi-iemk
    ports:
      - "8082:8080"
    volumes:
      - /srv/hosts/iemk/api.yaml:/opt/tomee/conf/api.yaml
    networks:
      net-fer:
        ipv4_address: 172.22.0.3
  replicator:
    image: docker.k-vrachu.ru/iemkreplicator:pglatest
    container_name: iemkreplicator
    volumes:
      - /srv/hosts/iemk/logs/replicator:/opt/tomee/logs
    environment:
      - "DB_MAIN_HOST="
      - "DB_MAIN_PORT="
      - "DB_MAIN_NAME="
      - "DB_MAIN_USER="
      - "DB_MAIN_PASS="
      - "DB_LOG_HOST="
      - "DB_LOG_PORT="
      - "DB_LOG_NAME=log_service"
      - "DB_LOG_USER="
      - "DB_LOG_PASS="
      - "ACTIVEMQHOST=172.22.0.2:61616"
      - "filterLpu=102"
      - "filterLpuUse=0"
      - "filterLpuBuilding=10010817"
      - "filterLpuBuildingUse=0"
      - "filterMedStaffFact=99560009698"
      - "filterMedStaffFactUse=0"
      - "filterMedSpecOms=99560009698"
      -
      "filterDiag=34,383,384,385,386,387,5,75,76,77,78,79,80,81,82,83,84,253,324,325,326,327,328,329,330,331,332,333,334,335,336,26"
      - "filterDiagUse=0"
    networks:
      net-fer:
        ipv4_address: 172.22.0.4
  proxy:
    image: docker.k-vrachu.ru/swaniemkproxy:pglatest
    container_name: swaniemkproxy
    ports:
      - "8080:8080"
    volumes:
      - /srv/hosts/iemk/keys:/var/opt/cproccsp/keys/root
      - /srv/hosts/iemk/logs/proxy:/opt/tomee/logs
      - /srv/configs/tomee.xml:/opt/tomee/conf/tomee.xml
      -
      /srv/configs/iemk_configs/crypto.properties:/opt/tomee/webapps/swan-iemc-proxy/WEB-INF/classes/crypto.properties

```

```

environment:
  - "DB_LOG_HOST="
  - "DB_LOG_PORT="
  - "DB_LOG_NAME=log_service"
  - "DB_LOG_USER="
  - "DB_LOG_PASS="
  - "ACTIVEMQHOST=172.22.0.2:61616"
  - "cryptoUser=test"
  - "cryptoPassword=12345678"
  - "cryptoSignatureAlgorithm=SIGNATURE_V2_ALGORITHM_2012_256"
  -
  "cryptoSignatureDigestAlgorithm=SIGNATURE_V2_DIGEST_ALGORITHM_2012_256"
  #- "JCP_KEY_LICENSE="
  #- "JCP_COMPANY_NAME="
extra_hosts:
  - "ips.rosminzdrav.ru:10.41.19.39"
networks:
  net-fer:
    ipv4_address: 172.22.0.5

networks:
  net-fer:
    driver: bridge
    ipam:
      driver: default
      config:
        - subnet: 172.22.0.0/16
          gateway: 172.22.0.254

```

**Code Block 50 docker-compose.yml**

## 5.1.2 Установка напрямую на ОС

5.1.2.1 Необходимое ПО

5.1.2.2 Установка TomEE

5.1.2.3 Настройка TomEE как сервиса

## 5.2 Описание конфигурации.

Настройка swan-api описана [по ссылке](#). swan-api должен сатроеть на основную БД промеда

Все остальные сервисы настраиваются через environvent к каждому сервису.  
Описание настроек сервисов

DB\_MAIN\_HOST - IP адрес основной БД ЕЦП.МИС

DB\_MAIN\_PORT - Порт основной БД ЕЦП.МИС

DB\_MAIN\_NAME - Имя основной БД ЕЦП.МИС

DB\_MAIN\_USER - Пользователь для авторизации в основной БД ЕЦП.МИС

DB\_MAIN\_PASS - Пароль для авторизации в основной БД ЕЦП.МИС

DB\_LOG\_HOST - IP адрес БД log\_service

DB\_LOG\_PORT - Порт БД log\_service

DB\_LOG\_NAME - Имя БД log\_service. По умолчанию это БД называется log\_service

DB\_LOG\_USER - Пользователь для авторизации в БД log\_service

DB\_LOG\_PASS - Пароль для авторизации в БД log\_service

ACTIVEMQHOST - Адрес для ActiveMQ. Вместе с портом

filterLpu=0 - Фильтр ЛПУ - по умолчанию 0.Параметр нужен, если выгрузка по

конкретному(-ым) лпу, тогда в параметр через запятую пишем Lpu\_id  
filterLpuUse=0 - Включить фильтр ЛПУ (0,1)- по умолчанию 0. Если нужна выгрузка по конкретным Лпу, то ставим= 1.  
filterLpuBuilding =0 -Фильтр по подразделениям. Аналогично filterLpu.  
filterLpuBuildingUse =0  
filterMedStaffFact = 0-Фильтр по MedStaffFact\_id. (Ставка врача). Аналогично filterLpu.  
filterMedStaffFactUse =0  
filterMedSpecOms =0 - Фильтр по медицинским специальностям . Аналогично filterLpu.  
filterMedSpecOmsUse =0 - Включить фильтр по медицинским специальностям (0,1)  
filterDiag =0 -Фильтр по диагнозам. Если указаны диагнозы , то они не передаются. Нужно , если регионы не хотят по конкретным диагнозам передавать данные. Аналогично filterLpu.  
filterDiagUse =0  
cryptoUser - Имя контейнера КриптоПро с открытым и закрытым ключом, для подписания отправляемых документов.  
cryptoPassword - Пароль от контейнера КриптоПро  
cryptoSignatureAlgorithm - Алгоритм используемого ключа. Можно использовать какой в примере указан, так как сейчас ИЕМК поддерживает в основном GOST 2012\_256  
cryptoSignatureDigestAlgorithm - Алгоритм используемого ключа. Можно использовать какой в примере указан, так как сейчас ИЕМК поддерживает в основном GOST 2012\_256



```

<?xml version="1.0" encoding="UTF-8"?>
<tomee>
  <Resource id="iemcDatabase" type="DataSource">
    JdbcDriver org.postgresql.Driver
    JdbcUrl
    jdbc:postgresql://<DB_LOG_HOST>:<DB_LOG_PORT>/<DB_LOG_NAME>?stringtype=un-
    pecified
    UserName <DB_LOG_USER>
    Password <DB_LOG_PASS>
    validationQuery = SELECT 1
    JtaManaged true
  </Resource>
  <Resource id="iemcDatabaseUnmanaged" type="DataSource">
    JdbcDriver org.postgresql.Driver
    JdbcUrl
    jdbc:postgresql://<DB_LOG_HOST>:<DB_LOG_PORT>/<DB_LOG_NAME>?stringtype=un-
    pecified
    UserName <DB_LOG_USER>
    Password <DB_LOG_PASS>
    validationQuery = SELECT 1
    JtaManaged false
  </Resource>
  <Resource id="IEMCJmsResourceAdapter"
type="ActiveMQResourceAdapter">
    BrokerXmlConfig =
    ServerUrl = tcp://<ACTIVEMQHOST>
    DataSource = iemcDatabase
  </Resource>
  <Resource id="IEMCJmsConnectionFactory"
type="javax.jms.ConnectionFactory"
class-
name="org.apache.activemq.ActiveMQConnectionFactory">
    poolMaxSize = 40
    ConnectionMaxIdleMinutes = 15
    ConnectionMaxWaitMilliseconds = 5000
    ResourceAdapter = IEMCJmsResourceAdapter
    brokerURL = tcp://<ACTIVEMQHOST>
    userName =
    password =
  </Resource>

  <Container id="IEMCJmsMdbContainer" ctype="MESSAGE">
    ResourceAdapter = IEMCJmsResourceAdapter
  </Container>
  <Resource id="PROMED.IN.QUEUE" type="javax.jms.Queue" />
  <Resource id="PROMED.OUT.QUEUE" type="javax.jms.Queue" />
</tomee>

```

**Code Block 51 tomee.xml**

```

# crypto provider settings
org.apache.ws.security.crypto.provider=ru.swan.soap.crypto.LocalMerlin
org.apache.ws.security.crypto.merlin.keystore.provider=JCP
org.apache.ws.security.crypto.merlin.keystore.type=HDImageStore

# crypto storage account settings
crypto.user=<cryptoUser>
crypto.signatureUser=<cryptoUser>
crypto.password=<cryptoPassword>

# trust settings
crypto.isTrustAllCerts=true

#signature digest algorithm
# CryptoPro
# v1 gostr3411 signature digest algorithm
#crypto.signature.digest.algorithm=SIGNATURE_DIGEST_ALGORITHM
# v2 gostr3411 signature digest algorithm
#crypto.signature.digest.algorithm=SIGNATURE_V2_DIGEST_ALGORITHM
# v2 gostr34112012-256 signature digest algorithm
crypto.signature.digest.algorithm=SIGNATURE_V2_DIGEST_ALGORITHM_2012_256
# v2 gostr34112012-512 signature digest algorithm
#crypto.signature.digest.algorithm=SIGNATURE_V2_DIGEST_ALGORITHM_2012_512

#signature algorithm
# CryptoPro
# v1 gostr3411 signature algorithm
#crypto.signature.algorithm=SIGNATURE_ALGORITHM
# v2 gostr34102001-gostr3411 signature algorithm
#crypto.signature.algorithm=SIGNATURE_V2_ALGORITHM
# v2 gostr34102012-gostr34112012-256 signature algorithm
crypto.signature.algorithm=SIGNATURE_V2_ALGORITHM_2012_256
# v2 gostr34102012-gostr34112012-512 signature algorithm
#crypto.signature.algorithm=SIGNATURE_V2_ALGORITHM_2012_512/

```

**Code Block 52 crypto.properties**

Так же часть настроек находятся в БД в таблице В БД log\_service. Будет в описании БД log\_service

## 5.3 Описание БД log\_service для использования сервиса ИЕМК

### 5.3.1 iemc.integrationservicesystemoption

Это БД с основными настройками сервиса ИЕМК

system	title	uniqueid	value
1	Адрес сервиса evn/EvnPS	promedServiceEvnPS	/evn/EvnPS
1	Адрес сервиса person/PersonState	promedServicePersonState	/person/PersonState

system	title	uniqueid	value
1	Адрес сервиса of person/PersonPrivilege	promedServicePersonPrivilege	/person/PersonPrivilege
1	Адрес сервиса login	promedServiceLogin	/login
1	Период релоадинга настроек (сек)	optionsReloadingPeriod	600
1	Регион	region	r59
1	Количество потоков для отправки сообщений	queue.listener.count	10
1	Уровень логирования в базе данных (INFO,DEBUG)	queue.listener.count	DEBUG
1	Временная зона	timezone	GMT+05
1	Идентификатор МИС (MIS GUID)	mis.guid	
1	Адрес коллбека для асинхронных сервисов	callback.uri	<a href="http://&lt;IEMK_IP&gt;/swan-iemc-test/webservices/CallbackSoapService?wsdl">http://&lt;IEMK_IP&gt;/swan-iemc-test/webservices/CallbackSoapService?wsdl</a>
1	Адрес АПИ ПроМед	restful.base.path	<a href="http://&lt;API_IP&gt;/swan-api/rest-api">http://&lt;API_IP&gt;/swan-api/rest-api</a>
1	Пароль пользователя промед для авторизации в RESTfull сервисе	promedPass	
1	Имя пользователя промед для авторизации в RESTfull сервисе	promedUser	

Примечания к настройкам в БД:

**promedServiceEvnPS, promedServicePersonState, promedServicePersonPrivilege, promedServiceLogin** - можно не менять

optionsReloadingPeriod - время обновления конфига из БД. указывается в секундах, но всё равно лучше перезапустить сервис.

promedUser, promedPass - логин и пароль от пользователя в Промеде. это обязательно. нужно для авторизации в АПИ

restful.base.path - Адрес и путь до swan-api, он должен быть виден сервису swan-iemc-проху

callback.uri - Адрес коллбека нашего сервиса. Для тестовой среды он должен быть виден из интернета. Для рабочей среды должен быть виден из ЗСПД. Регистрировать его не нужно. просто актуальный можно прописать сюда.

mis.guid - идентификатор. получить его можно при регистрации системы.

### 5.3.2 iemc.integrationserviceeventtype

В этой таблице находятся шаблоны для ответов в ИЕМК. приходят с обновлением БД

id	action	ас ун с	evenid	serviceuri	title	intem platei d	outte mplat eid
2001	urn:ihe:iti:2007:RetrieveDocumentSet	1	RETRIEVE.DOCUMENT	<a href="http://api-iemc-test.rosminzdrav.ru/xds/xdsClientRep?wsdl">http://api-iemc-test.rosminzdrav.ru/xds/xdsClientRep?wsdl</a>	Выборка документа СЭМД из ИЭМК	103	100
2002	urn:ihe:iti:2007:RegistryStoredQuery	1	RETRIEVE.DOCUMENT.METADATA	<a href="http://api-iemc-test.rosminzdrav.ru/xds/xdsClientReg?wsdl">http://api-iemc-test.rosminzdrav.ru/xds/xdsClientReg?wsdl</a>	Выборка метадаанных СЭМД из ИЭМК	104	105
2003	urn:hl7-org:v3:PRPA_IN201305	0	RETRIEVE.PATIENT	<a href="https://ips-test.rosminzdrav.ru/52d76ffc06419">https://ips-test.rosminzdrav.ru/52d76ffc06419</a>	Запрос данных о пациенте	107	106
2004	urn:hl7-org:v3:PRPA_IN201302	0	CHANGE.PATIENT	<a href="https://ips-test.rosminzdrav.ru/52d76ffc06419">https://ips-test.rosminzdrav.ru/52d76ffc06419</a>	Изменение пациента	108	109
2006	urn:ihe:iti:2007:ProvideAndRegisterDocumentSet-b	1	SEND.EPICRISIS	<a href="https://ips-test.rosminzdrav.ru/52f0d9db3c8c0">https://ips-test.rosminzdrav.ru/52f0d9db3c8c0</a>	Регистрация эпикриза стационара	113	112
2007	urn:ihe:iti:2007:ProvideAndRegisterDocumentSet-b	1	SEND.AMBULATORY	<a href="https://ips-test.rosminzdrav.ru/52f0d9db3c8c0">https://ips-test.rosminzdrav.ru/52f0d9db3c8c0</a>	Регистрация амбулаторного эпикриза	113	118
2008	urn:ihe:iti:2007:ProvideAndRegisterDocumentSet-b	1	CHANGE.AMBULATORY	<a href="https://ips-test.rosminzdrav.ru/52f0d9db3c8c0">https://ips-test.rosminzdrav.ru/52f0d9db3c8c0</a>	Изменение амбулаторного эпикриза	113	120
2009	urn:ihe:iti:2007:ProvideAndRegisterDocumentSet-b	1	SEND.DIRECTION	<a href="https://ips-test.rosminzdrav.ru/52f0d9db3c8c0">https://ips-test.rosminzdrav.ru/52f0d9db3c8c0</a>	Регистрация направления	113	122

id	action	ac yn c	evenid	serviceuri	title	intem platei d	outte mplat eid
					на госпита лизации ю		
2 1 0	urn:ihe:iti:2007:Provide AndRegisterDocumentS et-b	1	SEND.PROTOCO LLAB	<a href="https://ips-test.rosminzdrav.ru/52f0d9db3c8c0">https://ips- test.rosminzdrav.ru/5 2f0d9db3c8c0</a>	Регистр ация протоко ла лабора торного исслед ования	113	126
2 1 1	urn:ihe:iti:2007:Provide AndRegisterDocumentS et-b	1	SEND.PROTOCO LCONS	<a href="https://ips-test.rosminzdrav.ru/52f0d9db3c8c0">https://ips- test.rosminzdrav.ru/5 2f0d9db3c8c0</a>	Регистр ация протоко ла консуль тации	113	128
2 0 5	urn:hl7- org:v3:PRPA_IN201301	0	REGISTER.PATI ENT	<a href="https://ips-test.rosminzdrav.ru/52d76ffc06419">https://ips- test.rosminzdrav.ru/5 2d76ffc06419</a>	Регистр ация пациен та	110	111

в этой таблице меняются только адреса serviceuri

В данном примере актуальные тестовые адреса.

Для перевода в рабочий режим в таблице необходимо будет заменить

<https://ips-test.rosminzdrav.ru/52d76ffc06419> на <https://ips.rosminzdrav.ru/52dd1bfaca6c5>  
<https://ips-test.rosminzdrav.ru/52f0d9db3c8c0> на <https://ips.rosminzdrav.ru/52f3408e550d7>

При этом ips.rosminzdrav.ru нужно чтобы работал через ЗСПД

### 5.3.3 iemc.integrationserviceeventlog

В этой таблице находятся логи работы системы. По ней и определяется как работает система.

## 6 Разворачивание ФЭР

### 6.1 Установка с применением docker

#### 6.1.1 Необходимое ПО

- docker v19.03
- docker-compose 1.25

#### 6.1.2 Структура папок и файлов

/srv/hosts/fercon/

```
├── dbPG.properties
├── db.properties
└── fer.properties
```

### 6.1.3 docker-compose.yml

```
version: '2'
services:
  activemq_instance1:
    image: docker.k-vrachu.ru/activemq:latest
    container_name: activemq_fercon_1
    ports:
      - "8161:8161"
    networks:
      net-fer:
        ipv4_address: 172.23.0.2

  activemq_instance2:
    image: docker.k-vrachu.ru/activemq:latest
    container_name: activemq_fercon_2
    ports:
      - "8162:8161"
    networks:
      net-fer:
        ipv4_address: 172.23.0.3

  fercon_instance1:
    image: docker.k-vrachu.ru/fercon:pglatest
    container_name: fercon_1
    environment:
      - CATALINA_OPTS=-Xms2048M -Xmx4096M -server -XX:+UseParallelGC
      -Dfile.encoding=UTF-8
      - INSTANCE=1
      - ACTIVEMQ=172.23.0.2:61616
    ports:
      - "8080:8080"
    volumes:
      - "/srv/hosts/fercon:/srv/config:ro"
    networks:
      net-fer:
        ipv4_address: 172.23.0.4
    depends_on:
      - activemq_instance1

  fercon_instance2:
    image: docker.k-vrachu.ru/fercon:pglatest
    container_name: fercon_2
    environment:
      - CATALINA_OPTS=-Xms1024M -Xmx3072M -server -XX:+UseParallelGC
      -Dfile.encoding=UTF-8
      - INSTANCE=2
      - ACTIVEMQ=172.23.0.3:61616
    volumes:
      - "/srv/hosts/fercon:/srv/config:ro"
    networks:
      net-fer:
        ipv4_address: 172.23.0.5
    depends_on:
      - activemq_instance2

networks:
  net-fer:
    driver: bridge
    ipam:
      driver: default
      config:
        - subnet: 172.23.0.0/16
          gateway: 172.23.0.254
```

Code Block 53 docker-compose.yml

## 6.2 Описание файлов конфигурации

Настройки db.properties и dbPG.properties дублируются

необходимо заменить на соответствующие

**<SERVER\_IP>** - адрес сервера БД

**<SERVER\_PORT>** - порт сервера БД

**<DATABASE>** - имя БД

**<USERNAME>** - логин для авторизации БД

**<PASSWORD>** - пароль авторизации в БД

**baseProp** - отвечает за то, какую БД будет использовать сервис. 0 - MSSQL, 1 - PostgreSQL



```

# db connection settings
db.driver=com.microsoft.sqlserver.jdbc.SQLServerDriver
db.url=jdbc:sqlserver://<SERVER_IP>;databaseName=<DATABASE>;selectMethod=c
ursor;
db.username = <USERNAME>
db.password = <PASSWORD>

# identified patient 0|1 logic - 0 -MSSQL, 1 - Postgress
baseProp = 1

# identified patient 0|1 logic
patientIdentify = 0

#Lpu Available Attachment filter, comma-separated list or 0
lpuAvailFilter = -1

#9990000055
#Lpu Available Attachment filter, comma-separated list or 0
lpuAlwaysAvailFilter = 22,23,25,26,'9990000096',150185

#actual MO 0|1 logic
actualMainMoAttachment = 1
#person MO attachment 0|1 logic
mainMOAttachment = 0
#0|1 logic
gynecologMOAttachment = 1
#0|1 logic
stomatMOAttachment = 1
#0|1 logic
officeMOAttachment = 1
#0|1 logic
moMainTerritory = 0

#0|1 logic
assignMainAttachment = 0
#0|1 logic
noMainAttachmentAll = 0
#0|1 logic
noMainAttachmentAllTer = 0
#0|1 logic
assignNoMainAttachment = 0
#unspecified sign no main Attachment (region!) 0|1 logic

#Main Attachment nsi.MedSpec filter, comma-separated list or 0
filterNsiMedSpec = 8, 10, 11, 14, 16, 17, 19, 20, 22, 27, 30, 31, 118, 81,
260, 145, 42, 173, 174, 176, 175, 177, 178
#Other MO Main Attachment nsi.MedSpec filter, comma-separated list or 0
otherMOfilterNsiMedSpec = 173,174,176,175,177,178,8, 207, 10
#Other MO NO Main Attachment nsi.MedSpec filter, comma-separated list or 0
noMainfilterNsiMedSpec = 8,207,173,174,175,176,177,178

#age
ageGrown = 18
#NO child nsi.MedSpec filter, comma-separated list or 0
noChildfilterNsiMedSpec = 27
#NO grown nsi.MedSpec filter, comma-separated list or 0
noGrownfilterNsiMedSpec = 22, 173, 11,
41,42,43,44,45,46,47,48,49,50,51,73,77,79,80,81,82,83,84,85,86,89,90,91,92
,93,94,95,96,149,174
#NO male nsi.MedSpec filter, comma-separated list or 0
noMalefilterNsiMedSpec = 8,207
#NO female nsi.MedSpec filter, comma-separated list or 0
noFemalefilterNsiMedSpec = 0

#date
dateAvail = 14

```

```
#LpuUnitType_id filter, 0|1 logic
policlinicAvail=1
#LpuUnit_IsEnabled filter, 0|1 logic
operatorEntry = 1

#recType filter, comma-separated list or 0
recType = 1

#shaduleTimeOpen filter, comma-separated list or 0
shaduleTimeOpen = 20:00
#shaduleTimeCloseTomorrow filter, comma-separated list or 0
shaduleTimeCloseTomorrow = 17:00
#diffTime filter, comma-separated list or 0
diffTime = 30

#client
#misId test conc-50159-rmis      production conc-50114-rmis
misId = conc-50114-rmis
#EpguUserId
epguUserId = 999900
#sourceSlot, comma-separated list or 0
sourceSlot = 0
#timeFailApp, comma-separated list or 0
timeFailApp = 30
```

**Code Block 54 db.properties**

```

# db connection settings
db.driver=org.postgresql.Driver
db.url=jdbc:postgresql://<SERVER_IP>:<SERVER_PORT>/<DATABASE>?stringtype=unspecified
db.username = <USERNAME>
db.password = <PASSWORD>

# identified patient 0|1 logic - 0 -MSSQL, 1 - Postgress
baseProp = 1

# identified patient 0|1 logic
patientIdentify = 0

#Lpu Available Attachment filter, comma-separated list or 0
lpuAvailFilter = -1

#actual MO 0|1 logic
actualMainMoAttachment = 0
#person MO attachment 0|1 logic
mainMOAttachment = 1
#0|1 logic
gynecologMOAttachment = 0
#0|1 logic
stomatMOAttachment = 0
#0|1 logic
officeMOAttachment = 0
#0|1 logic
moMainTerritory = 0

#0|1 logic
assignMainAttachment = 0
#0|1 logic
noMainAttachmentAll = 1
#0|1 logic
noMainAttachmentAllTer = 0
#0|1 logic
assignNoMainAttachment = 0
#unspecified sign no main Attachment (region!) 0|1 logic

#Main Attachment nsi.MedSpec filter, comma-separated list or 0
filterNsiMedSpec = -1
#Other MO Main Attachment nsi.MedSpec filter, comma-separated list or 0
otherMOfilterNsiMedSpec = -1
#Other MO NO Main Attachment nsi.MedSpec filter, comma-separated list or 0
noMainfilterNsiMedSpec =
3,8,16,22,27,33,34,35,36,171,173,174,175,176,177,178,179

#age
ageGrown = 18
#NO child nsi.MedSpec filter, comma-separated list or 0
noChildfilterNsiMedSpec = 0
#NO grown nsi.MedSpec filter, comma-separated list or 0
noGrownfilterNsiMedSpec =
11,22,41,42,43,44,45,46,47,48,49,50,51,73,77,79,80,81,82,83,84,85,86,88,89
,90,91,92,93,94,95,96,174,149
#NO male nsi.MedSpec filter, comma-separated list or 0
noMalefilterNsiMedSpec = 8,207
#NO female nsi.MedSpec filter, comma-separated list or 0
noFemalefilterNsiMedSpec = 0

#date
dateAvail = 14

#LpuUnitType_id filter, 0|1 logic
policlinicAvail=1
#LpuUnit_IsEnabled filter, 0|1 logic

```

```

operatorEntry = 1

#recType filter, comma-separated list or 0
recType = 1

#shaduleTimeOpen filter, comma-separated list or 0
shaduleTimeOpen = 21:00
#shaduleTimeCloseTomorrow filter, comma-separated list or 0
shaduleTimeCloseTomorrow = 17:00
#diffTime filter, comma-separated list or 0
diffTime = 15

#client
#misId test conc-50143-rmis      production
misId = conc-50143-rmis
#misIdHC test conc-50143-rmis   production
misIdHC = conc-50239-rmis
#EpguUserId
epguUserId = 999900
#sourceSlot, comma-separated list or 0
sourceSlot = 1
#timeFailApp, comma-separated list or 0
timeFailApp = 1

```

#### Code Block 55 dbPG.properties

Пояснения по конфигу db.properties и dbPG.properties:

**misId** - ID сервиса записи к врачу. Для тестового и рабочего они разные. получить их можно через менеджера, после регистрации системы в соответствующей системе.

**misIdHC** - ID сервиса вызова на дом. Для тестового и рабочего они разные. получить их можно через менеджера, после регистрации системы в соответствующей системе.

**sourceSlot** = 0 должен стоять в первом экземпляре и 1 во втором. Настраивается автоматически от переменной в докере INSTANCE

```

# fer service
service.url=http://fer-
concentrator.egisz.rosminzdrav.ru/concentrator_ws/ws/mis/
serviceHouseCall.url=http://fer-
concentrator.egisz.rosminzdrav.ru/concentrator_ws/ws/housecallstatus/house
callstatus.wsd

# soap settings
soap.is12=true
soap.isLog=true

# listener settings
fer.UpdateAppointmentStatus.listener.count = 4
fer.HouseCallStatus.listener.count = 5

```

#### Code Block 56 fer.properties

Пояснения по конфигу fer.properties :

**service.url** - адрес для работы сервиса Запись к врачу. для тестового [http://test-api2-fer.rt-eu.ru/concentrator\\_ws/ws/mis/mis.wsd](http://test-api2-fer.rt-eu.ru/concentrator_ws/ws/mis/mis.wsd) Для рабочего [http://fer-concentrator.egisz.rosminzdrav.ru/concentrator\\_ws/ws/mis/](http://fer-concentrator.egisz.rosminzdrav.ru/concentrator_ws/ws/mis/)

**serviceHouseCall.url** - адрес для работы сервиса Вызов на дом. Для тестового [http://test-api2-fer.rt-eu.ru/concentrator\\_ws/ws/housecallstatus/housecallstatus.wsdl](http://test-api2-fer.rt-eu.ru/concentrator_ws/ws/housecallstatus/housecallstatus.wsdl) для рабочего [http://fer-concentrator.egisz.rosminzdrav.ru/concentrator\\_ws/ws/housecallstatus/housecallstatus.wsdl](http://fer-concentrator.egisz.rosminzdrav.ru/concentrator_ws/ws/housecallstatus/housecallstatus.wsdl)

Рабочие должны работать через ЗСПД

## 6.2.1 job.xml

Они находятся в отдельной папке внутри контейнера и берутся нужные в зависимости от номера экземпляра.

Их можно заменить на свой при необходимости.

1 Экземпляр сервис FerConcentrator

Путь в контейнере. /srv/job/ins1/FerConcentrator-2.0.0/WEB-INF/classes/job.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<job-scheduling-data
  xmlns="http://www.quartz-scheduler.org/xml/JobSchedulingData"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.quartz-
scheduler.org/xml/JobSchedulingData http://www.quartz-
scheduler.org/xml/job_scheduling_data_1_8.xsd"
  version="1.8">
  <schedule>
    <job>
      <name>UpdateAppointmentStatusReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotReplicatorJob</job-class>
    </job>

    <trigger>
      <cron>
        <name>SlotReplicatorTrigger</name>
        <job-name>UpdateAppointmentStatusReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0/30 * * * * ?</cron-expression>
      </cron>
    </trigger>
  </schedule>
</job-scheduling-data>
```

### Code Block 57 job.xml

1 Экземпляр сервис FerConcentrator-2.0.0

Путь в контейнере. /srv/job/ins1/FerConcentrator-2.0.0/WEB-INF/classes/job.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<job-scheduling-data
  xmlns="http://www.quartz-scheduler.org/xml/JobSchedulingData"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.quartz-
scheduler.org/xml/JobSchedulingData http://www.quartz-
scheduler.org/xml/job_scheduling_data_1_8.xsd"
  version="1.8">
  <schedule>
    <job>
      <name>UpdateAppointmentStatusSlotReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotReplicatorJob</job-class>
    </job>

    <job>
      <name>UpdateAppointmentStatusCancelReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotCancelReplicatorJob</job-class>
    </job>

    <job>
      <name>UpdateAppointmentStatusAppearReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotAppearReplicatorJob</job-class>
    </job>

    <job>
      <name>HCCancelReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по вызовам</description>
      <job-class>ru.swan.client.HCCancelReplicatorJob</job-class>
    </job>

    <job>
      <name>HCNotReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по вызовам</description>
      <job-class>ru.swan.client.HCNotReplicatorJob</job-class>
    </job>

    <job>
      <name>HCVizitReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по вызовам</description>
      <job-class>ru.swan.client.HCVizitReplicatorJob</job-class>
    </job>

    <job>
      <name>HCAccompReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по вызовам</description>
      <job-class>ru.swan.client.HCAccompReplicatorJob</job-class>
    </job>

    <trigger>
      <cron>
        <name>HCCancelReplicatorJob</name>
        <job-name>HCCancelReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0 0/10 * * * ?</cron-expression>
      </cron>
    </trigger>
    <trigger>
      <cron>
        <name>HCNotReplicatorJob</name>

```

```

        <job-name>HCNotReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0 0/10 * * * ?</cron-expression>
    </cron>
</trigger>
<trigger>
    <cron>
        <name>HCVizitReplicatorJob</name>
        <job-name>HCVizitReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0 0/10 * * * ?</cron-expression>
    </cron>
</trigger>
<trigger>
    <cron>
        <name>HCAccompReplicatorJob</name>
        <job-name>HCAccompReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0 0/10 * * * ?</cron-expression>
    </cron>
</trigger>
<trigger>
    <cron>
        <name>UpdateAppointmentStatusSlotReplicatorJob</name>
        <job-name>UpdateAppointmentStatusSlotReplicatorJob</job-
name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0 0/10 * * * ?</cron-expression>
    </cron>
</trigger>
    <trigger>
        <cron>
            <name>UpdateAppointmentStatusCancelReplicatorJob</name>
            <job-name>UpdateAppointmentStatusCancelReplicatorJob</job-
name>
            <job-group>ferConcentrator</job-group>
            <cron-expression>0 0/10 * * * ?</cron-expression>
        </cron>
    </trigger>
    <trigger>
        <cron>
            <name>UpdateAppointmentStatusAppearReplicatorJob</name>
            <job-name>UpdateAppointmentStatusAppearReplicatorJob</job-
name>
            <job-group>ferConcentrator</job-group>
            <cron-expression>0 0 13 * * ?</cron-expression>
        </cron>
    </trigger>
</schedule>
</job-scheduling-data>

```

### Code Block 58 job.xml

1 Экземпляр сервис FerConcentrator-3.0.0

Путь в контейнере. /srv/job/ins1/FerConcentrator-3.0.0/WEB-INF/classes/job.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<job-scheduling-data
  xmlns="http://www.quartz-scheduler.org/xml/JobSchedulingData"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.quartz-
scheduler.org/xml/JobSchedulingData http://www.quartz-
scheduler.org/xml/job_scheduling_data_1_8.xsd"
  version="1.8">
  <schedule>
    <job>
      <name>UpdateAppointmentStatusReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotReplicatorJob</job-class>
    </job>

    <trigger>
      <cron>
        <name>SlotReplicatorTrigger</name>
        <job-name>UpdateAppointmentStatusReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0/30 * * * * ?</cron-expression>
      </cron>
    </trigger>

  </schedule>
</job-scheduling-data>

```

**Code Block 59 job.xml**

## 1 Экземпляр сервис FerConcentratorHC

Путь в контейнере. /srv/job/ins1/FerConcentratorHC/WEB-INF/classes/job.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<job-scheduling-data
  xmlns="http://www.quartz-scheduler.org/xml/JobSchedulingData"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.quartz-
scheduler.org/xml/JobSchedulingData http://www.quartz-
scheduler.org/xml/job_scheduling_data_1_8.xsd"
  version="1.8">
  <schedule>
    <job>
      <name>UpdateAppointmentStatusReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotReplicatorJob</job-class>
    </job>

    <trigger>
      <cron>
        <name>SlotReplicatorTrigger</name>
        <job-name>UpdateAppointmentStatusReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0/30 * * * * ?</cron-expression>
      </cron>
    </trigger>

  </schedule>
</job-scheduling-data>

```

**Code Block 60 job.xml**



2 Экземпляр сервис FerConcentrator-2.0.0

Путь в контейнере. /srv/job/ins2/FerConcentrator-2.0.0/WEB-INF/classes/job.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<job-scheduling-data
  xmlns="http://www.quartz-scheduler.org/xml/JobSchedulingData"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.quartz-
scheduler.org/xml/JobSchedulingData http://www.quartz-
scheduler.org/xml/job_scheduling_data_1_8.xsd"
  version="1.8">
  <schedule>

    <job>
      <name>UpdateAppointmentStatusRecordReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotRecordReplicatorJob</job-class>
    </job>

    <job>
      <name>UpdateAppointmentStatusSlotReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotReplicatorJob</job-class>
    </job>

    <job>
      <name>UpdateAppointmentStatusCancelReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotCancelReplicatorJob</job-class>
    </job>

    <job>
      <name>UpdateAppointmentStatusAppearReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotAppearReplicatorJob</job-class>
    </job>

    <job>
      <name>HCCancelReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по вызовам</description>
      <job-class>ru.swan.client.HCCancelReplicatorJob</job-class>
    </job>

    <job>
      <name>HCNotReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по вызовам</description>
      <job-class>ru.swan.client.HCNotReplicatorJob</job-class>
    </job>

    <job>
      <name>HCVizitReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по вызовам</description>
      <job-class>ru.swan.client.HCVizitReplicatorJob</job-class>
    </job>

    <job>
      <name>HCAccompReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по вызовам</description>
      <job-class>ru.swan.client.HCAccompReplicatorJob</job-class>
    </job>

  <trigger>
    <cron>

```

```

        <name>HCCancelReplicatorJob</name>
        <job-name>HCCancelReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0 0/10 * * * ?</cron-expression>
    </cron>
</trigger>
<trigger>
    <cron>
        <name>HCNotReplicatorJob</name>
        <job-name>HCNotReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0 0/10 * * * ?</cron-expression>
    </cron>
</trigger>
<trigger>
    <cron>
        <name>HCVisitReplicatorJob</name>
        <job-name>HCVisitReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0 0/10 * * * ?</cron-expression>
    </cron>
</trigger>
<trigger>
    <cron>
        <name>HCAccompReplicatorJob</name>
        <job-name>HCAccompReplicatorJob</job-name>
        <job-group>ferConcentrator</job-group>
        <cron-expression>0 0/10 * * * ?</cron-expression>
    </cron>
</trigger>

name>
    <trigger>
        <cron>
            <name>UpdateAppointmentStatusRecordReplicatorJob</name>
            <job-name>UpdateAppointmentStatusRecordReplicatorJob</job-
                <job-group>ferConcentrator</job-group>
                <cron-expression>0 0/5 * * * ?</cron-expression>
            </cron>
        </trigger>

name>
    <trigger>
        <cron>
            <name>UpdateAppointmentStatusSlotReplicatorJob</name>
            <job-name>UpdateAppointmentStatusSlotReplicatorJob</job-
                <job-group>ferConcentrator</job-group>
                <cron-expression>0 0/5 * * * ?</cron-expression>
            </cron>
        </trigger>
        <trigger>
            <cron>
                <name>UpdateAppointmentStatusCancelReplicatorJob</name>
                <job-name>UpdateAppointmentStatusCancelReplicatorJob</job-
                    <job-group>ferConcentrator</job-group>
                    <cron-expression>0 0/5 * * * ?</cron-expression>
                </cron>
            </trigger>
            <trigger>
                <cron>

```

```
name>
    <name>UpdateAppointmentStatusAppearReplicatorJob</name>
    <job-name>UpdateAppointmentStatusAppearReplicatorJob</job-
    <job-group>ferConcentrator</job-group>
    <cron-expression>0 5 13 * * ?</cron-expression>
    </cron>
  </trigger>

</schedule>
</job-scheduling-data>
```

**Code Block 61 job.xml**

2 Экземпляр сервис FerConcentrator-3.0.0

Путь в контейнере. /srv/job/ins2/FerConcentrator-3.0.0/WEB-INF/classes/job.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<job-scheduling-data
  xmlns="http://www.quartz-scheduler.org/xml/JobSchedulingData"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.quartz-
scheduler.org/xml/JobSchedulingData http://www.quartz-
scheduler.org/xml/job_scheduling_data_1_8.xsd"
  version="1.8">
  <schedule>
    <job>
      <name>UpdateAppointmentStatusSlotReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotReplicatorJob</job-class>
    </job>

    <job>
      <name>UpdateAppointmentStatusCancelReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotCancelReplicatorJob</job-class>
    </job>

    <job>
      <name>UpdateAppointmentStatusAppearReplicatorJob</name>
      <group>ferConcentrator</group>
      <description>Запрос данных по биркам</description>
      <job-class>ru.swan.client.SlotAppearReplicatorJob</job-class>
    </job>

    <trigger>
      <cron><!--Добавление посещений по записям-->
      <name>UpdateAppointmentStatusSlotReplicatorJob</name>
      <job-name>UpdateAppointmentStatusSlotReplicatorJob</job-
name>
      <job-group>ferConcentrator</job-group>
      <cron-expression>0 0 3 ? * * *</cron-expression>
    </cron>
    </trigger>
    <trigger>
      <cron><!--Отмена записи-->
      <name>UpdateAppointmentStatusCancelReplicatorJob</name>
      <job-name>UpdateAppointmentStatusCancelReplicatorJob</job-
name>
      <job-group>ferConcentrator</job-group>
      <cron-expression>0 0 4 ? * * *</cron-expression>
    </cron>
    </trigger>
    <trigger>
      <cron><!--Записи без посещений в течении периода-->
      <name>UpdateAppointmentStatusAppearReplicatorJob</name>
      <job-name>UpdateAppointmentStatusAppearReplicatorJob</job-
name>
      <job-group>ferConcentrator</job-group>
      <cron-expression>0 0 5 ? * * *</cron-expression>
    </cron>
    </trigger>

  </schedule>
</job-scheduling-data>

```

**Code Block 62 job.xml**

## 6.3 Действия после установки

После установки необходимо написать заявку на подключение вашей системы к тестовой, а потом и к продуктивной среде.

После того как заявку примут выдадут id сервиса, который нужно будет прописать в настройках системы.

Адрес wsdl для предоставления заявки должен быть виден из интернета. необходимо настроить будет сервис через балансировщик

адрес wsdl в сервисе:

Запись к врачу: `http://<IP_ADDRESS>:<PORT>/FerConcentrator/services/FerConcentrator.svc?wsdl`

Вызов на дом: `http://<IP_ADDRESS>:<PORT>/FerConcentratorHC/HouseCallPortal?wsdl`

Пример настройки балансировщика привожу ниже:

```
server {
    listen                80;
    server_name           fer.promedweb.ru;
    open_file_cache_errors off;

    location /FerConcentrator/ {
        proxy_pass        http://<IP_SERVER>:<PORT>/FerConcentrator/;
        proxy_redirect    off;
        proxy_set_header  Host
$host:$server_port;
        proxy_set_header  X-Real-IP           $remote_addr;
        proxy_set_header  X-Forwarded-For
$proxy_add_x_forwarded_for;
        proxy_connect_timeout    5;
        proxy_send_timeout      60s;
        proxy_read_timeout      600s;
        proxy_http_version      1.1;
        proxy_set_header        Upgrade           $http_upgrade;
        proxy_set_header        Connection
$connection_upgrade;
    }

    location /FerConcentratorHC/ {
        proxy_pass        http://<IP_SERVER>:<PORT>/FerConcentratorHC/;
        proxy_redirect    off;
        proxy_set_header  Host
$host:$server_port;
        proxy_set_header  X-Real-IP           $remote_addr;
        proxy_set_header  X-Forwarded-For
$proxy_add_x_forwarded_for;
        proxy_connect_timeout    5;
        proxy_send_timeout      60s;
        proxy_read_timeout      600s;
        proxy_http_version      1.1;
        proxy_set_header        Upgrade           $http_upgrade;
        proxy_set_header        Connection
$connection_upgrade;
    }
}
```

Code Block 63 fer.conf (nginx)

## 7 КУ ФЭР Профосмотры (Dispensary)

### 7.1 Разворачивание

#### 7.1.1 Установка с применением docker

##### 7.1.1.1 Необходимое ПО

- docker v19.03
- docker-compose 1.25

##### 7.1.1.2 Структура папок

/srv/hosts/disp - основная папка сервиса  
├─ conf - папка с конфигами (сами файлы описываются ниже)  
└─ logs - папка с логами

##### 7.1.1.3 docker-compose.yml

```
version: '2'
services:
  dispensary:
    image: docker.k-vrachu.ru/dispensary:pglatest
    environment:
      - "CATALINA_OPTS=-Xms256M -Xmx1024M -server -XX:+UseParallelGC -
Dfile.encoding=UTF-8"
    ports:
      - 8080:8080
    extra_hosts:
      - "fer-concentrator.egisz.rosminzdrav.ru:10.41.19.36"
    volumes:
      - "/srv/hosts/disp/logs:/opt/tomee/logs"
      -
"/srv/hosts/disp/conf/application.properties:/opt/tomee/webapps/swan-
dispensary-service/WEB-INF/classes/application.properties:ro"
      # Если используется PostgreSQL База Данных
      - "/srv/hosts/disp/conf/db.postgres.xml:/opt/tomee/webapps/swan-
dispensary-service/WEB-INF/classes/db.postgres.xml:ro"
      -
"/srv/hosts/disp/conf/db.postgresql.properties:/opt/tomee/webapps/swan-
dispensary-service/WEB-INF/classes/db.postgresql.properties:ro"
      # Если используется MSSQL База Данных
      #- "/srv/hosts/disp/conf/db.mssql.xml:/opt/tomee/webapps/swan-
dispensary-service/WEB-INF/classes/db.mssql.xml:ro"
      #-
"/srv/hosts/disp/conf/db.mssql.properties:/opt/tomee/webapps/swan-
dispensary-service/WEB-INF/classes/db.mssql.properties:ro"
```

**Code Block 64 docker-compose.yml**

#### 7.1.2 Установка напрямую на ОС

##### 7.1.2.1 Необходимое ПО

Для работы сервиса необходимо использовать Java 8 и Apache TomEE - сервер приложений JavaEE версии 7.1

Скачать можно по ссылке: <https://tomee.apache.org/download-ng.html>

### 7.1.2.2 Установка TomEE

Производится согласно документации размещенной на сайте: <https://tomee.apache.org/docs.html>

### 7.1.2.3 Настройка TomEE как сервиса

После установки TomEE скопировать дополнительные библиотеки, в папку lib сервера приложений и перезапустить TomEE

Дополнительные библиотеки находятся в репозитории git:

<https://git.promedweb.ru/rtmis/disp/-/tree/master/tomee-lib>

## 7.2 Описание файлов конфигурации

Файл application.properties

```
# System User Id
swan.system.user.id = 1
swan.rmism.id = conc-<ID>-rmis
swan.dap.sys.nick = profosmotr
swan.dap.therapy.codes =
B01.047.001.001,B01.047.001.001,B04.047.002,02003509,B02.000.003,B04.026.0
02,B04.047.002,B04.047.002.999,B04.047.004,B04.047.004.999
swan.manual.update.pool = 10

db.data.source = db.postgres.xml
#db.data.source = db.mssql.xml

#scheduling the top of every hour of every day.
scheduling.job.cron=0 0 * * * *

#scheduling check service every 3 minute
scheduling.job.cron.check.service=0 0/3 * * * *

#scheduling from 20:00 to 06:00
#scheduling.job.cron=0 0 20-23,0-6 * * *

# Web Service
# Для подключения к рабочей среде
# spring.ws.dispensary.callback.uri = http://fer-
concentrator.egisz.rosminzdrav.ru/examination-
callback/DispensaryCallbackServicece
# spring.ws.booking.update.uri = https://fer-
concentrator.egisz.rosminzdrav.ru/update-booking-
details/UpdateBookingDetailsService
# Для подключения к тестовой среде
spring.ws.dispensary.callback.uri = http://test-api2-fer.rt-
eu.ru/examination-callback/DispensaryCallbackService
spring.ws.booking.update.uri = http://test-api2-fer.rt-eu.ru/update-
booking-details/UpdateBookingDetailsService
```

**Code Block 65 application.properties**

Пояснения к файлу application.properties

**swan.system.user.id** = 1 - id пользователя, от имени которого производятся записи в БД.

**swan.rmism.id** - ID сервиса. Для тестового и рабочего они разные. получить их можно через менеджера, после регистрации системы в соответствующей системе.

**swan.dap.sys.nick** - тип услуги, которая используется при записи на ДВН.



**swan.dap.therapy.codes** - коды услуг , которые могут осуществлять диспансеризацию взрослого населения

**swan.manual.update.pool** - количество потоков, которое будет использоваться при передаче измененных в Промеде направлений на диспансеризацию, когда этот метод вызывается REST запросом

**db.data.source** - в случае подключения к postgresql серверу подключаем db.postgres.xml и пробрасываем в контейнер файлы db.postgres.xml и db.postgresql.properties. Если у нас подключение к MSSQL серверу прописываем настройку db.mssql.xml и в контейнер пробрасываем файлы db.mssql.xml и db.mssql.properties

**scheduling.job.cron** - расписание для автоматического запуска выгрузки обновленных в Промеде направлений.

**scheduling.job.cron.check.service** - расписание для автоматического запуска проверки доступности федерального сервиса

**spring.ws.dispensary.callback.uri** и **spring.ws.booking.update.uri** - Для тестовой и рабочей среды уже прописаны выше. можно просто раскомментировать и закомментировать нужные строки. Но если сервис работает в продуктивной (рабочей) среде необходимо чтобы адрес [fer-concentrator.egisz.rosminzdrav.ru](http://fer-concentrator.egisz.rosminzdrav.ru) работал через ЗСПД (должен соответствовать адресу 10.41.19.36)

Файлы для конфигурации PostgreSQL

```

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE configuration PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>
  <properties resource="db.postgresql.properties" />
  <environments default="development">
    <environment id="development">
      <transactionManager type="JDBC" />
      <dataSource type="POOLED">
        <property name="driver" value="${db.driver}" />
        <property name="url" value="${db.url}" />
        <property name="username" value="${db.username}" />
        <property name="password" value="${db.password}" />
      </dataSource>
    </environment>
  </environments>
  <databaseIdProvider type="DB_VENDOR">
    <property name="PostgreSQL" value="postgres"/>
  </databaseIdProvider>
  <mapper>
    <mapper
resource="ru/swan/dispensary/db/postgres/ConcentratorSessionMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/postgres/IdentifyPatientMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/postgres/DispensaryInfoMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/postgres/QuestionTypeMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/postgres/ServicesSlotsMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/postgres/BookingDataMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/postgres/ReplicatorJobMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/postgres/UpdateBookingMapper.xml" />
  </mapper>
</configuration>

```

**Code Block 66 db.postgres.xml**

```

db.driver=org.postgresql.Driver
db.url=jdbc:postgresql://<IP_SERVER>:<PORT_SERVER>/<DB>
db.username = <USERNAME>
db.password = <PASSWORD>

```

**Code Block 67 db.postgresql.properties**

## Файлы конфигурации MSSQL

```

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE configuration PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>
  <properties resource="db.mssql.properties" />
  <environments default="development">
    <environment id="development">
      <transactionManager type="JDBC" />
      <dataSource type="POOLED">
        <property name="driver" value="${db.driver}" />
        <property name="url" value="${db.url}" />
        <property name="username" value="${db.username}" />
        <property name="password" value="${db.password}" />
      </dataSource>
    </environment>
  </environments>
  <databaseIdProvider type="DB_VENDOR">
    <property name="SQL Server" value="mssql"/>
    <property name="PostgreSQL" value="postgres"/>
  </databaseIdProvider>
  <mapppers>
    <mapper
resource="ru/swan/dispensary/db/mssql/ConcentratorSessionMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/mssql/IdentifyPatientMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/mssql/DispensaryInfoMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/mssql/QuestionTypeMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/mssql/ServicesSlotsMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/mssql/BookingDataMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/mssql/ReplicatorJobMapper.xml" />
    <mapper
resource="ru/swan/dispensary/db/mssql/UpdateBookingMapper.xml" />
  </mapppers>
</configuration>

```

**Code Block 68 db.mssql.xml**

```

# db connection settings
db.driver=com.microsoft.sqlserver.jdbc.SQLServerDriver
db.url=jdbc:sqlserver://<IP_SERVER>;databaseName=<DB>;selectMethod=cursor;
db.username = <USERNAME>
db.password = <PASSWORD>

```

**Code Block 69 db.mssql.properties**

## 7.3 Действия после установки

После установки необходимо написать заявку на подключение вашей системы к тестовой, а потом и к продуктивной среде.

После того как заявку примут выдадут id сервиса, который нужно будет прописать в настройках системы.

Адрес wsdl для предоставления заявки должен быть виден из интернета. необходимо настроить будет сервис через балансировщик

адрес wsdl в сервисе `http://<IP_ADDRESS>:<PORT>/swan-dispensary-service/ws/dispensaryCommons.wsdl?wsdl`

Пример настройки балансировщика привожу ниже:

```
server {
    listen                80;
    server_name           disp.promedweb.ru;
    open_file_cache_errors off;

    location /swan-dispensary-service/ {
        proxy_pass        http://<IP_SERVER>:<PORT>/swan-
dispensary-service/;
        proxy_redirect    off;
        proxy_set_header  Host
$host:$server_port;
        proxy_set_header  X-Real-IP           $remote_addr;
        proxy_set_header  X-Forwarded-For
$proxy_add_x_forwarded_for;
        proxy_connect_timeout 5;
        proxy_send_timeout   60s;
        proxy_read_timeout   600s;
        proxy_http_version   1.1;
        proxy_set_header     Upgrade         $http_upgrade;
        proxy_set_header     Connection
$connection_upgrade;
    }
}
```

**Code Block 70 fer.conf (nginx)**

## 8 РЭМД (Электронные медицинские документы)

### 8.1 Разворачивание

#### 8.1.1 Установка с применением docker

8.1.1.1 Необходимое ПО

8.1.1.2 Структура папок  
/srv/hosts/remd

```
├── config
├── keys
└── logs
```

8.1.1.3 docker-compose.yml

```
version: '2'

services:
  remd:
    image: docker.promedweb.ru/remd:pglatest
    environment:
      - "CATALINA_OPTS=-Xms512M -Xmx1536M -server -XX:+UseParallelGC -
Dfile.encoding=UTF-8"
      #- "JCP_KEY_LICENSE="
      #- "JCP_COMPANY_NAME="
    extra_hosts:
      - "ips.rosminzdrav.ru:10.41.19.39"
    volumes:
      - "./config/application.properties:/opt/tomee/webapps/Remd/WEB-
INF/classes/application.properties:ro"
      - "./config/crypto.properties:/opt/tomee/webapps/Remd/WEB-
INF/classes/crypto.properties:ro"
      - "./logs:/opt/tomee/logs"
      - "./keys:/var/opt/cprocsp/keys/root/"
    ports:
      - "8080:8080"
```

Code Block 71 docker-compose.yml

#### 8.1.2 Установка напрямую на ОС

8.1.2.1 Необходимое ПО

8.1.2.2 Установка TomEE

8.1.2.3 Настройка TomEE как сервиса

### 8.2 Описание конфигурации.

#### 8.2.1 Описание настроек сервиса

```
# crypto provider settings
org.apache.ws.security.crypto.provider=ru.swan.soap.crypto.LocalMerlin
org.apache.ws.security.crypto.merlin.keystore.provider=JCP
org.apache.ws.security.crypto.merlin.keystore.type=HDImageStore

# crypto storage account settings
crypto.user=${crypto.user}
crypto.signatureUser=${crypto.user}
crypto.password=${crypto.password}
crypto.signature.algorithm=${crypto.signature.algorithm}
crypto.signature.digest.algorithm=${crypto.signature.digest.algorithm}

# trust settings
crypto.isTrustAllCerts=true
```

**Code Block 72 crypto.properties**

```

#CONNECTION DEVELOP
spring.datasource.url = jdbc:postgresql://${DB_HOST}:${DB_PORT}/${DB_NAME}
spring.datasource.username = ${DB_USER}
spring.datasource.password = ${DB_PASS}

# KEEP THE CONNECTION ALIVE IF IDLE FOR A LONG TIME (NEEDED IN PRODUCTION)
spring.datasource.testWhileIdle = true
spring.datasource.validationQuery = SELECT 1

# USE SPRING.JPA.PROPERTIES.* FOR HIBERNATE NATIVE PROPERTIES (THE PREFIX
IS
# STRIPPED BEFORE ADDING THEM TO THE ENTITY MANAGER).

# SHOW OR NOT LOG FOR EACH SQL QUERY
spring.jpa.show-sql = false
spring.jpa.hibernate.naming.physical-
strategy=org.hibernate.boot.model.naming.PhysicalNamingStrategyStandardImp
l

# HIBERNATE DDL AUTO (CREATE, CREATE-DROP, UPDATE): WITH "UPDATE" THE
DATABASE
# SCHEMA WILL BE AUTOMATICALLY UPDATED ACCORDINGLY TO JAVA ENTITIES FOUND
IN
# THE PROJECT
spring.jpa.hibernate.ddl-auto = none
spring.jpa.properties.hibernate.default_schema=EMD

# NAMING STRATEGY

#spring.jpa.hibernate.naming-strategy =
org.hibernate.cfg.ImprovedNamingStrategy

# =====
# = JPA / HIBERNATE
# =====
spring.datasource.driverClassName=org.postgresql.Driver
spring.jpa.properties.hibernate.show_sql = false
spring.jpa.properties.hibernate.format_sql = false
spring.jpa.properties.hibernate.globally_quoted_identifiers = true
spring.jpa.properties.hibernate.dialect =
org.hibernate.dialect.PostgreSQL94Dialect

#SYSTEM USER ID
swan.system.user.id = 1

#TRIAL MODE
swan.trial.mode = ${trial_mode}

#SCHEDULING FROM 20:00 TO 06:00
scheduling.job.cron=${schedule}

#IDENTIFICATION KEY RMIS IN IPS
rmis.identification.key = ${ident_key}

rmis.system.sid = ${system_sid}

rmis.download.url = ${promed_url}

#WEB SERVICE
spring.ws.remd.egisz.uri=${egisz_uri}
spring.ws.remd.callback.uri=http://www.w3.org/2005/08/addressing/anonymous
spring.ws.remd.callback.ips.uri={ips_url}

# Crypto settings

```

```
crypto.passwordCallbackClass=ru.swan.soap.crypto.PasswordFromCryptoPropertiesFileCallback
ips.handle.headers=true
ips.message.id=09fa0dfc-a975-42ce-9739-d8afac7df2d0
ips.action=registerDocument
ips.response.action=http://emu.callback.mis.service.nr.eu.rt.ru/Callback/s
endResponseResponse
```

#### Code Block 73 application.properties

**egisz\_uri** - для рабочего сервера <http://ips.rosminzdrav.ru/59a6aa9dcd045>, для тестового <http://ips-test.rosminzdrav.ru/598c424055f7d> . [ips.rosminzdrav.ru](http://ips.rosminzdrav.ru) - должен быть виден через ЗСПД

**trial\_mode** - true или false. На рабочей системе trial должен быть отключен.

**schedule** - крон для запуска задания

**ident\_key** - ключ формата UUID. получается после регистрации.

**system\_sid** - ID системы. получается после регистрации.

**promed\_url** - адрес ЕЦП.МИС для получения файлов ЕМД из папки export. если используется https то сертификат должен быть валидный. он должен быть виден сервису и виден через ЗСПД (доступен для запроса федеральными информационными ресурсами)

**ips\_url** - адрес похож на egisz\_uri. получается после регистрации.

**crypto.user** - имя контейнера КриптоПро для подписания документов

**crypto.password** - пароль от контейнера КриптоПро

**crypto.signature.algorithm** - алгоритм шифрования контейнера КриптоПро

**crypto.signature.digest.algorithm** - алгоритм шифрования контейнера КриптоПро

## 8.2.2 Написание настроек ЕЦП.МИС

```
/**
 * Электронные медицинские документы
 */
$db['emd'] = $db['default'];
$db['emd']['hostname'] = '192.168.0.0';
$db['emd']['database'] = 'promed';
$db['emd']['username'] = '';
$db['emd']['password'] = '';
$db['emd']['dbdriver'] = "postgre";
$db['emd']['pconnect'] = false;
```

#### Code Block 74 database.php

```
// Включить использование базы emd для хранения электронных медицинских документов
$config['EMD_ENABLE'] = true;
// Отключение проверки сертификата, в основном для теста
$config['EMD_DISABLE_CERT_CENTER_CHECK'] = true;
```

#### Code Block 75 promed.php



## 8.2.3 Настройка балансировщика nginx

```
map $http_upgrade $connection_upgrade {
    default upgrade;
    ""      close;
}

server {
    listen      80;
    server_name remd.ecp-mis.ru;

    location /Remd {
        proxy_pass http://localhost:8089/Remd;
        proxy_redirect      off;
        proxy_set_header    Host      $host;
        proxy_set_header    X-Real-IP $remote_addr;
        proxy_set_header    X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_connect_timeout      5;
        proxy_send_timeout        60s;
        proxy_read_timeout        600s;
        proxy_http_version        1.1;
        proxy_set_header          Upgrade $http_upgrade;
        proxy_set_header          Connection $connection_upgrade;
    }
}
```

**Code Block 76 remd.conf**

## 8.3 Действия после установки

После того как сервис развернут нужно зарегистрировать сервис WSDL сервиса для регистрации [http://<IP\\_SERVER>:<PORT\\_SERVER>/Remd/ws/EmdrClientCallback.wsdl](http://<IP_SERVER>:<PORT_SERVER>/Remd/ws/EmdrClientCallback.wsdl)

Проверка версии и некоторых настроек сервиса  
[http://<IP\\_SERVER>:<PORT\\_SERVER>/Remd/api/getRemdVersion](http://<IP_SERVER>:<PORT_SERVER>/Remd/api/getRemdVersion)

## 9 ТФОМС

- [Настройка RabbitMQ](#)
  - d. [Устанавливаем epel и erlang](#)
  - e. [Далее устанавливаем сам RabbitMQ](#)
  - f. [Меняем ограничение на количество открытых файлов](#)
- [Настройка ЭЦП](#)
- [Настройка заданий ЭЦП](#)

### 9.1 Настройка RabbitMQ

Важно! RabbitMQ может быть как на нашей стороне, так и может располагаться на стороне ТФОМС. в последнем случае RabbitMQ разворачивать не нужно.

#### 9.1.1 Устанавливаем epel и erlang

##### Установка epel и erlang

```
yum update
wget https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
sudo rpm -Uvh epel-release-latest-7*.rpm
wget http://rpms.famillecollet.com/enterprise/remi-release-7.rpm
sudo rpm -Uvh remi-release-7*.rpm
```

#### 9.1.2 Далее устанавливаем сам RabbitMQ

##### Установка RabbitMQ

```
cd /tmp
wget https://packagecloud.io/rabbitmq/rabbitmq-server/packages/el/7/rabbitmq-server-3.8.5-1.el7.noarch.rpm
rpm --import https://www.rabbitmq.com/rabbitmq-release-signing-key.as
yum -y install rabbitmq-server-3.6.11-1.el7.noarch.rpm
```

#### 9.1.3 Меняем ограничение на количество открытых файлов

Открываем файл `/etc/systemd/system/rabbitmq-server.service.d/limits.conf` и выставим содержимое, что бы получилось вот так

```
[Service]
LimitNOFILE=infinity
```

Если указанного файла нет, то создаем его.

выполняем команду

```
systemctl --system daemon-reload
```

Помечаем rabbitmq для автоматического запуска при перезагрузках

```
chkconfig rabbitmq-server on
```

Запускаем сервер

```
service rabbitmq-server start
```

Ставим стандартные плагины

```
rabbitmq-plugins enable rabbitmq_management
chown -R rabbitmq:rabbitmq /var/lib/rabbitmq/
```

По умолчанию создается сразу юзер **guest** с одноименным паролем. У него админский доступ по localhost. Рекомендую создать собственного пользователя, а **guest** удалить.

Нужно создать учётки для администрирования, для ЕЦП.МИС и ТФОМС.

## 9.2 Настройка ЭЦП

```
$config['TFOMSAutoInteract'] = array(
    'host' => '<IP_SERVER>',
    'port' => '5672',
    'user' => '<USER_SERVER>',
    'password' => '<PASS_SERVER>',
    'publisher_queues' => array(
        'common' => 'test_input',
        'answer' => 'test_answer',
    ),
    'consumer_queues' => array(
        'common' => 'test_output',
        'answer' => 'tfomsToPromed.answer',
    ),
    'init_date' => '2020-01-01',
    'read_only' => false,
);

$config['ExchInspectPlan'] = array(
    'host' => '<IP_SERVER>',
    'port' => '5672',
    'user' => '<USER_SERVER>',
    'password' => '<PASS_SERVER>',
    'allowed_lpus' => array(100,101,102),
    'publisher_queues' => array(
        'common' => 'test_input',
        'answer' => 'test_answer',
    ),
    'consumer_queues' => array(
        'common' => 'test_output',
        'answer' => 'tfomsToPromed.answer',
    ),
    // 'allowed_lpus' => array(100,101,102), //ограничение доступа к
    // функции экспорта планов диспансеризации в ТФОМС
    'init_date' => '2020-01-01',
    'read_only' => true,
);
```

### Code Block 77 promed.php

<IP\_SERVER> - IP сервера RabbitMQ  
<USER\_SERVER>- Пользователь отRabbitMQ  
<PASS\_SERVER> - Пароль отRabbitMQ

### 9.3 Настройка заданий ЭЦП

```

currentDate=`date +%Y-%m-%d`
currentTime=`date +%Y-%m-%d %T`

# runPublisher - Диспансерное наблюдения
echo >> /var/logs/TFOMS/DN_${currentDate}.txt
echo $currentTime >> /var/logs/TFOMS/DN_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runPublisher&packageType=DISP&packageLimit=100&getDebug=1&swtoken=<TOKEN>"
>>/var/logs/TFOMS/DN_${currentDate}.txt

# runConsumer - Ответы от фонда
echo >> /var/logs/TFOMS/ANSWER_${currentDate}.txt
echo $currentTime >> /var/logs/TFOMS/ANSWER_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runConsumer&packageLimit=100&getDebug=1&swtoken=<TOKEN>"
>>/var/logs/TFOMS/ANSWER_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runConsumer&queueNick=answer&packageLimit=100&getDebug=1&swtoken=<TOKEN>" >>
/var/logs/TFOMS/ANSWER_${currentDate}.txt

# runPublisher - Участки и прикрепления
echo >> /var/logs/TFOMS/DISTRICT_PERS_${currentDate}.txt
echo $currentTime >> /var/logs/TFOMS/DISTRICT_PERS_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runPublisher&packageType=PERS&packageLimit=100&getDebug=1&swtoken=<TOKEN>" >>
/var/logs/TFOMS/DISTRICT_PERS_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runPublisher&packageType=DISTRICT&packageLimit=100&getDebug=1&swtoken=<TOKEN>" >>
/var/logs/TFOMS/DISTRICT_PERS_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runPublisher&packageType=PERSONATTACHDISTRICT&packageLimit=100&getDebug=1&swtoken=<TOKEN>" >>
/var/logs/TFOMS/DISTRICT_PERS_${currentDate}.txt

# runPublisher - ОНК, ВМП и прочее
echo >> <ECP_IP>:<ECP_PORT>ONKO_${currentDate}.txt
echo $currentTime >> <ECP_IP>:<ECP_PORT>ONKO_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runPublisher&packageType=ONKOTRAMADOL&packageLimit=100&getDebug=1&swtoken=<TOKEN>" >>
/var/logs/TFOMS/ONKO_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runService&packageType=ONKODRUG&packageLimit=50&getDebug=1&swtoken=<TOKEN>" >>
/var/logs/TFOMS/ONKO_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runPublisher&packageType=HTMC_REFERRAL&packageLimit=100&getDebug=1&swtoken=<TOKEN>" >>
/var/logs/TFOMS/ONKO_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runPublisher&packageType=ECO_REFERRAL&packageLimit=10&getDebug=1&swtoken=<TOKEN>" >>
/var/logs/TFOMS/ONKO_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runPublisher&packageType=PERSCANCER&packageLimit=10&getDebug=1&swtoken=<TOKEN>" >>
/var/logs/TFOMS/ONKO_${currentDate}.txt
0 20-23,00-08 * * * /usr/bin/curl
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runPublisher&packageType=

```

```
e=PERSHEPATITIS&packageLimit=10&getDebug=1&swtoken=<TOKEN>" >>  
/var/logs/TFOMS/ONKO_$(date +%Y%m%d).txt  
0 20-23,00-08 * * * /usr/bin/curl  
"http://<ECP_IP>:<ECP_PORT>/?c=TFOMSAutoInteract&m=runPublisher&packageTyp  
e=GEBT&packageLimit=50&getDebug=1&swtoken=<TOKEN>" >>  
/var/logs/TFOMS/ONKO_$(date +%Y%m%d).txt
```

#### Code Block 78 cron

<TOKEN> - токен авторизации описано как его получить [ТУТ](#)

<ECP\_IP> - IP адрес веб сервера ЕЦП.МИС

<ECP\_PORT> - ПОРТ веб сервера ЕЦП.МИС

## 10 ФРБТ / ФРВИЧ / ФР14Н

### 10.1 Принцип работы сервиса

По расписанию выполняется поиск пациентов по регистру соответствующего заболевания, данные по которым изменились

(описание что конкретно проверяется будет приложено позднее)

По каждому пациенту начинается сеанс обмена с patient.read;

Асинхронные ответы приходят на общий listener (реализованный в рамках FRMO), который по типу сессии распределяет ответы по очередям activeMQ

Сервис получает ответ через activeMQ и в зависимости от ответа выполняет дальнейшие действия.

Для отладки/тестирования есть страничка default-ext.html с которой можно задать список передачи и инициировать сессию вручную по конкретным пациентам

### 10.2 Развертывание

#### 10.2.1 Необходимое ПО

URL проекта: <https://git.promedweb.ru/rtmis/javaapps/FRHIV>

Для сборки проекта необходимо:

4. java 1.8 (возможно использование openJDK) с установленным CryptoPro (использовалась версия jcr-2.0.40035 )
5. gradle 6.3 или старше
6. доступ к <https://repo.rtmis.ru>

сборка осуществляется командой

```
gradle clean assemble
```

В каталоге ./build/libs будет создан FRHIV-X.X.X.war

Для запуска сервиса необходимо:

- java 1.8 (возможно использование openJDK) с установленным CryptoPro (использовалась версия jcr-2.0.40035 )
- tomcat 9.X (под tomee сервис не работает)

Инструкции по сборке docker контейнера будут приложены позднее

#### 10.2.2 Установка

Для установки необходимо

- развернуть содержимое FRHIV-X.X.X.war в tomcat/webapps/<желаемое имя сервиса>



- скопировать в `/var/opt/cproscsp/keys/<tomcat user>/` ключ для подписывания документов

### 10.2.3 Настройка

- Настройка подписи документов
  - а. в `tomcat/webapps/<желаемое имя сервиса>/WEB-INF/classes/crypto.properties` установить
    - `crypto.user =`
    - `crypto.signatureUser =`
    - `crypto.password=`
- Настройка расписания
  - в `tomcat/webapps/<желаемое имя сервиса>/WEB-INF/classes/job.xml` установить
    - `<cron-expression>...</cron-expression>`
- Настройка логирования - `tomcat/webapps/<желаемое имя сервиса>/WEB-INF/classes/logback.xml`
- Основная настройка сервиса - `tomcat/webapps/<желаемое имя сервиса>/WEB-INF/classes/configuration.yaml`

```
db:
  url: <jdbc url>
  username: <jdbc login>
  password: <jdbc password>
  driver: <jdbc driver>
  packagePrefixes:
    - ru.rtmis.srv
    - ru.swan.db
  databaseIds:
    "PostgreSQL": "postgres"
    "Microsoft SQL Server": "mssql"
send:
  systemId: <system ID>
  dest: <destination url - для тестового сервиса https://ips-
test.rosminzdrav.ru/57234d87b0838>
  moId: <default moID>
JMS:
  listenerThreadCount: <кол-во потоков для чтения асинхронных ответов.
рекомендованное значение 4>
  destinations: #описание очередей для получения асинхронных ответов.
Если работают не все сервисы - закомментировать ненужное
    - frhiv.inqueue.destination
    - frbt.inqueue.destination
```

## 10.3 Описание конфигурации

## 10.4 Описание БД

# 11 ФРМО

## 11.1 Разворачивание

Список необходимого ПО:

1. Java JDK 1.8
2. TomEE Plus 1.7.4, в составе которого используется apache-cxf-2.6.16
3. Крипто Про JCP версии 2.0.40035
4. ActiveMQ 5.15

### 11.1.1 Установка напрямую на ОС

#### 11.1.1.1 Установка Крипто Про JCP

Установить пакет Crypto Pro JCP 2.0. Для работы с ГОСТ 2012 требуется версия 2.0.40035 и выше. Согласно инструкции из пакета ЖТЯИ.00091-01 90 01. Инструкция по использованию.pdf нужно выполнить следующее:

Установка КриптоПро JCP должна проводиться администратором из командной строки, находясь в папке с инсталлятором:

`setup_console.bat <JRE> <JRE>` указывается как путь до папки JRE, не включая саму папку. КриптоПро JCP должен быть установлен в ту же папку JRE, с которой работает TomEE.

Скопируйте `jar` файлы которые находятся в папках `dependencies` и `javadoc` которые находятся в папке с КриптоПроJCP в папку `<JRE>/lib/ext/`

Если система не имеет графического интерфейса, то лицензию можно ввести путём или вызовом класса `ru.CryptoPro.JCP.tools.License` с параметрами:

```
<JRE>/bin/java ru.CryptoPro.JCP.tools.License -serial "serial_number" -company "company_name" -store
```

или

```
<JRE>/bin/java ru.CryptoPro.JCP.tools.License -serial "serial_number" -combase "company_name_in_base64" -store
```

Также можно проверить заданную лицензию без ее установки.

```
<JRE>/bin/java ru.CryptoPro.JCP.tools.License -serial "serial_number" -company "company_name"
```

или

```
<JRE>/bin/java ru.CryptoPro.JCP.tools.License -serial "serial_number" -combase "company_name_in_base64"
```

При использовании параметра `"-combase"` имя компании вводится в base64 кодировке.

Вызов класса `ru.CryptoPro.JCP.tools.License` без параметров проверит установленную лицензию.

Дату первой установки можно узнать с помощью команды:

```
<JRE>/bin/java ru.CryptoPro.JCP.tools.License -first
```

Для вывода справки:

```
<JRE>/bin/java ru.CryptoPro.JCP.tools.License ?
```

Подробнее о вводе лицензии можно ознакомиться в документации к КриптоПро JCP "Руководство администратора безопасности".

КриптоПро JCP должен быть установлен в ту же папку JRE, с которой работает TomEE (TomEE по умолчанию читает из системной переменной `JAVA_HOME`).

Далее региональный менеджер получает у региона ключ и регистрирует его в тестовой или рабочей среде. Это ключ нужно будет положить в директорию `/var/opt/cproscsp/keys/<user>/`

`<user>` - пользователь от которого работает TomEE

#### 11.1.1.2 Установка TomEE 1.7.4

Скачать можно по ссылке

<https://repo.maven.apache.org/maven2/org/apache/openejb/apache-tomee/1.7.4/apache-tomee-1.7.4-plus.tar.gz>

```
#!/bin/bash
cd /tmp
curl https://repo.maven.apache.org/maven2/org/apache/openejb/apache-tomee/1.7.4/apache-tomee-1.7.4-plus.tar.gz -O apache-tomee-1.7.4-plus.tar.gz
tar -xvf apache-tomee-1.7.4-plus.tar.gz
mv ./apache-tomee-plus-1.7.4/ /opt/tomee-frmo/
useradd -r -s /sbin/nologin tomcat
chown -R tomcat: /opt/tomee-frmo
chmod +x /opt/tomee-frmo/bin/*.sh
apt update
apt install openjdk-8-jdk -y

cat <<EOF > /etc/systemd/system/tomee-frmo.service
[Unit]
Description=Apache TomEE
After=network.target

[Service]
Type=forking

Environment=JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-amd64/jre
Environment=JRE_HOME=/usr/lib/jvm/java-1.8.0-openjdk-amd64/jre
Environment=CATALINA_PID=/opt/tomee/temp/tomee.pid
Environment=CATALINA_HOME=/opt/tomee
Environment=CATALINA_BASE=/opt/tomee
Environment='CATALINA_OPTS=-Xms512M -Xmx1536M -server -XX:+UseParallelGC -Dfile.encoding=UTF-8'
ExecStart=/opt/tomee-frmo/bin/startup.sh
ExecStop=/opt/tomee-frmo/bin/shutdown.sh

User=tomcat
Group=tomcat
UMask=0007
RestartSec=10
Restart=always

[Install]
WantedBy=multi-user.target

EOF

systemctl daemon-reload
systemctl enable tomee-frmo.service
systemctl start tomee-frmo.service
```

#### Code Block 79 Tomee install Ubuntu

### 11.1.1.3 Настройка ActiveMQ

Скачать можно по

ссылке <http://www.apache.org/dyn/closer.cgi?filename=/activemq/5.15.12/apache-activemq-5.15.12-bin.tar.gz&action=download>

```
#!/bin/bash
cd /tmp

apt update
apt install -y \
    wget \
    openjdk-8-jdk

wget http://archive.apache.org/dist/activemq/5.15.12/apache-activemq-5.15.12-bin.tar.gz
tar -xvf apache-activemq-5.15.12-bin.tar.gz
mv /tmp/apache-activemq-5.15.12 /opt/activemq

cat << EOF > /etc/systemd/system/activemq.service
[Unit]
Description=activemq message queue
After=network.target
[Service]
PIDFile=/opt/activemq/data/activemq.pid
Environment='ACTIVEMQ_OPTS=-Xms128M -Xmx1024M -server -XX:+UseParallelGC -Dfile.encoding=UTF-8'
ExecStart=/opt/activemq/bin/activemq start
ExecStop=/opt/activemq/bin/activemq stop
User=root
Group=root
[Install]
WantedBy=multi-user.target
EOF

systemctl daemon-reload
systemctl enable activemq.service
systemctl start activemq.service
```

**Code Block 80** Установка ActiveMQ на Ubuntu

## 11.1.2 Установка с помощью docker.

### 11.1.2.1 docker-compose.yml

```
version: '2'
services:
  activemq:
    image: docker.k-vrachu.ru/activemq:latest
    container_name: activemq
    ports:
      - "8161:8161"
      - "61616:61616"
  frmo:
    image: docker.k-vrachu.ru/frmo:pglatest
    container_name: frmo
    volumes:
      - /srv/hosts/FRMO/crypto.properties:/opt/tomee/webapps/FRMO/WEB-INF/classes/crypto.properties
      - /srv/hosts/FRMO/frmo.properties:/opt/tomee/webapps/FRMO/WEB-INF/classes/frmo.properties
      - /srv/hosts/FRMO/db.xml:/opt/tomee/webapps/FRMO/WEB-INF/classes/db.xml
      - /srv/hosts/FRMO/db.properties:/opt/tomee/webapps/FRMO/WEB-INF/classes/db.properties
      - /srv/hosts/FRMO/db.postgres.xml:/opt/tomee/webapps/FRMO/WEB-INF/classes/db.postgres.xml
      - /srv/hosts/FRMO/db.postgresql.properties:/opt/tomee/webapps/FRMO/WEB-INF/classes/db.postgresql.properties
      - /srv/hosts/FRMO/context.xml:/opt/tomee/webapps/FRMO/META-INF/context.xml
      - /srv/hosts/FRMO/keys:/var/opt/cprosp/keys/root
      - /srv/hosts/FRMO/logs:/opt/tomee/logs
    ports:
      - "8080:8080"
    environment:
      #Такие параметры уже включены в конетйнер можно изменить их при необходимости
      - TZ=Asia/Yekaterinburg
      - CATALINA_OPTS=-Xms512M -Xmx1536M -server -XX:+UseParallelGC -Dfile.encoding=UTF-8
      - JCP_KEY_LICENSE=''
      - JCP_COMPANY_NAME=''
    depends_on:
      - activemq
```

#### Code Block 81 docker-compose.yml

## 11.1.3 Установка и настройка сервиса ФРМО

Файлы сервиса можно получить по ссылке <https://cloud.swan-it.ru/index.php/s/Hd4gRDg32ktSmKe> в папке adugeya будет архив FRMO.zip

Установить под TomEE сервис FRMO, для чего скопировать папку FRMO с файлами сервиса в папку webapps TomEE.

1. В файле frmo.properties настраиваются следующие параметры:
2. Идентификатор сервиса, полученный в ИПС (Федеральной подсистеме интеграции прикладных систем): ips.clientEntityId=4005d388-3b63-4846-971b-7a0a13399bf8
3. Регион, например, Пермь: region=r59
4. Учетка для сервиса в БД Промед (user\_id): user=1

5. Адрес федерального сервиса, к которому подключается сервис интеграции с ФРМО (ЕДИНЫЙ для всех регионов): `frmo.serviceUrl=http://ips-test.rosminzdrav.ru/57234d87b0838`
6. Адрес регионального сервиса обратного вызова в системе ИПС: `frmo.callbackIpsUrl=http://ips-test.rosminzdrav.ru/xxxxxxxxxx`
7. В файле `crypto.properties` настраиваются следующие параметры:
8. `crypto.user`=имя контейнера ключей в КриптоПро JCP
9. `crypto.signatureUser`=имя контейнера ключей в КриптоПро JCP
10. `crypto.password`=пароль к контейнеру ключей
11. В файле `db.postgresql.properties` настраиваются следующие параметры:
12. `r59.url=jdbc:postgressql://192.168.36.24:5432/promedtest?stringtype=unspecified` -- подключение к БД
13. `r59.username` = учетка к БД
14. `r59.password` = пароль учетки
15. В файле `db.xml` настраиваются следующие параметры:
16. `<environments default="r59">` -- регион, где развернут сервис
17. для каждого региона вводится секция, имена переменных соответствуют указанным в `db.properties`

```

<!-- Buryatia -->
<environment id="r3">
<transactionManager type="JDBC" />
<dataSource type="POOLED">
<property name="driver" value="{db.driver}" />
<property name="url" value="{r3.url}" />
<property name="username" value="{r3.username}" />
<property name="password" value="{r3.password}" />
</dataSource>
</environment>

```
18. В файле `job.xml` настраиваются следующие параметры:

```

<cron-expression>0 0/2 * * * ?</cron-expression>

```

-- расписание запуска службы в крон-формате (по умолчанию каждые 2 минуты)
19. В файле `context.xml` настраиваются следующие параметры:
20. `brokerURL=tcp://172.22.0.2:61616` -- url брокера очередей ActiveMQ
21. `<Resource name="jms/frmo.inqueue.destination" auth="Container"` -- очередь сообщений для клиента ФРМО (строчку не менять)

```

type="org.apache.activemq.command.ActiveMQQueue"
factory="org.apache.activemq.jndi.JNDIReferenceFactory"
physicalName="ru.swan.frmo.answers" />

```

-- имя очереди сообщений для клиента ФРМО в ActiveMQ
22. `<Resource name="jms/frmr.sendResponse.destination" auth="Container"` -- очередь сообщений для клиента ФРМО (строчку не менять)

```

type="org.apache.activemq.command.ActiveMQQueue"
factory="org.apache.activemq.jndi.JNDIReferenceFactory"
physicalName="ru.swan.frmr.sendResponse" />

```

-- имя очереди сообщений для клиента ФРМО в ActiveMQ
23. При разворачивании в докере приложений под Linux необходимо указать кодировку по умолчанию при запуске JVM: `-Dfile.encoding=UTF-8`. Пример: переменная среды `CATALINA_OPTS`, значение `-Xms128M -Xmx1024M -server -XX:+UseParallelGC -Dfile.encoding=UTF-8`

24. Для тестового контура необходимо предоставить адрес до этого сервиса через интернет. Сделать можно через балансировщик. И передать адрес вида `http://<ADDRESS>/FRMO/callback.svc` менеджеру для регистрации в тестовой среде.

Для запуска сервиса нужно стартовать службу TomEE.

#### 11.1.4 Описание файлов конфигурации

```
# crypto provider settings
org.apache.ws.security.crypto.provider=ru.swan.soap.crypto.LocalMerlin
org.apache.ws.security.crypto.merlin.keystore.provider=JCP
org.apache.ws.security.crypto.merlin.keystore.type=HDImageStore

# crypto storage account settings
crypto.user=<KEY_NAME>
crypto.signatureUser=<KEY_NAME>
crypto.password=<KEY_PASS>

# trust settings
crypto.isTrustAllCerts=true
crypto.signature.algorithm=SIGNATURE_V2_ALGORITHM_2012_256
crypto.signature.digest.algorithm=SIGNATURE_V2_DIGEST_ALGORITHM_2012_256
```

##### Code Block 82 crypto.properties

**crypto.user** - обычно совпадает с `crypto.signatureUser`. Это имя контейнера КриптоПро

**crypto.password** - Пароль контейнера КриптоПро

```

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE configuration
  PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
  "http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>
  <properties resource="db.postgresql.properties" />
  <settings>
    <setting name="defaultStatementTimeout" value="3600"/>
  </settings>
  <environments default="r59">
    <!-- Perm -->
    <environment id="r59">
      <transactionManager type="JDBC" />
      <dataSource type="POOLED">
        <property name="driver" value="${db.driver}" />
        <property name="url" value="${r59.url}" />
        <property name="username" value="${r59.username}" />
        <property name="password" value="${r59.password}" />
      </dataSource>
    </environment>
  </environments>
  <databaseIdProvider type="DB_VENDOR">
    <property name="PostgreSQL" value="postgres"/>
  </databaseIdProvider>
  <mappers>
    <mapper resource="ru/swan/frmo/data/postgres/SessionMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/MOMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/BuildingMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/DepartMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/StaffMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/LogServiceMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/MediaDataMapper.xml" />
  </mappers>
</configuration>

```

**Code Block 83 db.xml**

В этом конфиге нужно заменить схему БД с r59 на необходимую региональную схему.

```

db.driver=org.postgresql.Driver
r59.url=jdbc:postgresql://<DB_HOST>:<DB_PORT>/<DB_NAME>?stringtype=unspecified
r59.username = <DB_USER>
r59.password = <DB_PASS>

```

**Code Block 84 db.properties**

В данном конфиге описывается подключение к БД . используется основная БД ЕЦП.МИС

**<DB\_HOST>** - IP адрес сервера БД

**<DB\_PORT>** - порт сервера БД

**<DB\_NAME>** - имя базы данных

**<DB\_USER>** - имя пользователя для авторизации в БД

**<DB\_PASS>** - пароль для авторизации в БД



```

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE configuration
  PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
  "http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>
  <properties resource="db.postgresql.properties" />
  <settings>
    <setting name="defaultStatementTimeout" value="3600"/>
  </settings>
  <environments default="r59">
    <!-- Perm -->
    <environment id="r59">
      <transactionManager type="JDBC" />
      <dataSource type="POOLED">
        <property name="driver" value="${db.driver}" />
        <property name="url" value="${r59.url}" />
        <property name="username" value="${r59.username}" />
        <property name="password" value="${r59.password}" />
      </dataSource>
    </environment>
  </environments>
  <databaseIdProvider type="DB_VENDOR">
    <property name="PostgreSQL" value="postgres"/>
  </databaseIdProvider>
  <mappers>
    <mapper resource="ru/swan/frmo/data/postgres/SessionMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/MOMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/BuildingMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/DepartMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/StaffMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/LogServiceMapper.xml" />
    <mapper resource="ru/swan/frmo/data/postgres/MediaDataMapper.xml" />
  </mappers>
</configuration>

```

**Code Block 85 db.poetgres.xml**

В этом конфиге нужно заменить схему БД с r59 на необходимую региональную схему.

```

db.driver=org.postgresql.Driver
r59.url=jdbc:postgresql://<DB_HOST>:<DB_PORT>/<DB_NAME>?stringtype=unspeci
fied
r59.username = <DB_USER>
r59.password = <DB_PASS>

```

**Code Block 86 db.postgresql.properties**

В данном конфиге описывается подключение к БД . используется основная БД ЕЦП.МИС

- <DB\_HOST> - IP адрес сервера БД
- <DB\_PORT> - порт сервера БД
- <DB\_NAME> - имя базы данных
- <DB\_USER> - имя пользователя для авторизации в БД
- <DB\_PASS> - пароль для авторизации в БД

```

frmo.serviceUrl=http://ips-test.rosminzdrav.ru/57234d87b0838
frmo.callbackUrl=http://www.w3.org/2005/08/addressing/anonymous
frmo.callbackIpsUrl=https://ips-test.rosminzdrav.ru/XXXXXXXXXXXXXX

# ips settings
ips.isHandleIpsHeaders=true
ips.clientEntityId=4005d388-3b63-4846-971b-7a0a13399bf8
ips.action=sendDocument
ips.responseAction=http://emu.callback.mis.service.nr.eu.rt.ru/Callback/sendResponseResponse

region=r59
# user id
user=1
# mode id: 1 = export to FRMO, 2=import from FRMO, 3=read FRMO only
mode=1
saveFileBasePath=/srv/files
saveFileRelativePath=frmo
# limitation of session execution time (hours)
sessionTimeLimit=24
# waiting delay time (milliseconds)
waitDelayTime=5000
# perform verification phase
isVerify=true
# perform FRMR call
isFRMRRun=true
# FRMR session ActionType code
# 51 -- list
sessionFRMRActionCode=10075
# test methods running
isTesting=true

```

#### Code Block 87 frmo.properties

**frmo.serviceUrl** - для рабочего сервера <https://ips.rosminzdrav.ru/57406573a4083>, для тестового <http://ips-test.rosminzdrav.ru/57234d87b0838> . [ips.rosminzdrav.ru](https://ips.rosminzdrav.ru) - должен быть виден через ЗСПД

**frmo.callbackIpsUrl** - адрес похож на frmo.serviceUrl . Получается после регистрации.

**ips.clientEntityId** - ID сервиса в формате UUID. Получается после регистрации.

**region** - Написать схему региона

**saveFileBasePath**=/srv/files

**saveFileRelativePath**=frmo - Тут напишу сразу по обоим настройкам. Скорее всего на данный момент артефакт. но на всякий случай можно сделать папку /srv/files/frmo (то есть в пусти собрать обе настройки и сделать из них путь)

```

<?xml version="1.0" encoding="UTF-8"?>
  <job-scheduling-data
    xmlns="http://www.quartz-scheduler.org/xml/JobSchedulingData"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.quartz-
scheduler.org/xml/JobSchedulingData http://www.quartz-
scheduler.org/xml/job_scheduling_data_1_8.xsd"
    version="1.8">
    <schedule>
      <job>
        <name>FRMOJob</name>
        <group>frmo</group>
        <description>Проверка таблицы dbo.FRMOSession</description>
        <job-class>ru.swan.frmo.job.FRMOJob</job-class>
      </job>
      <trigger>
        <cron>
          <name>FRMOTrigger</name>
          <job-name>FRMOJob</job-name>
          <job-group>frmo</job-group>
          <cron-expression>0/45 * * * * ?</cron-expression>
        </cron>
      </trigger>
    </schedule>
  </job-scheduling-data>

```

**Code Block 88 job.xml**

периодический запуск сервиса.

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xml>
<Context>
  <Resource
    name="jms/ConnectionFactory"
    auth="Container"
    type="org.apache.activemq.ActiveMQConnectionFactory"
    description="JMS Connection Factory"
    factory="org.apache.activemq.jndi.JNDIReferenceFactory"

    brokerURL="tcp://<ACTIVEMQ_HOST>:<ACTIVEMQ_PORT>?jms.redeliveryPolicy.maximumRedeliveries=5&jms.redeliveryPolicy.backOffMultiplier=2&jms.redeliveryPolicy.useExponentialBackOff=true"/>
    <Resource name="jms/frmo.inqueue.destination" auth="Container"
    type="org.apache.activemq.command.ActiveMQQueue"
    factory="org.apache.activemq.jndi.JNDIReferenceFactory"
    physicalName="ru.swan.frmo.answers" />
    <Resource name="jms/frmr.sendResponse.destination"
    auth="Container"
    type="org.apache.activemq.command.ActiveMQQueue"
    factory="org.apache.activemq.jndi.JNDIReferenceFactory"
    physicalName="ru.swan.frmr.sendResponse" />
    <Resource name="jms/frbt.inqueue.destination" auth="Container"
    type="org.apache.activemq.command.ActiveMQQueue"
    factory="org.apache.activemq.jndi.JNDIReferenceFactory"
    physicalName="ru.swan.frbt.answers" />
  </Context>

```

**Code Block 89 context.xml**

<ACTIVEMQ\_HOST> - IP адрес сервера activemq



P.S. такую проверку лучше проводить сразу после запуска сервиса. так как первый раз после запуска обычно обрабатывает долго. после данного "пинга" сервис начинает следующие запросы обрабатывать быстрее.

Если этот xml сохранить в файл request.xml, то из командной строки можно выполнить проверку следующей командой:

```
curl -X POST -H "SOAPAction=sendResponse" -H "Content-Type=text/xml;charset=UTF-8" -d @request.xml http://<IP\_SERVER>:8080/FRMO/callback.svc
```

Если в ответе есть <wsse:BinarySecurityToken - то сервис правильно подхватил ключ Кристо Про который мы настроили

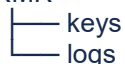
## 12 ФРМР

### 12.1 Разворачивание

#### 12.1.1 Установка с применением docker

##### 12.1.1.1 Необходимое ПО

##### 12.1.1.2 Структура папок и файлов /srv/hosts/FRMR



##### 12.1.1.3 docker-compose.yml

**ВАЖНО! ActiveMQ должен быть тот же что и у сервиса FRMO**

```
version: '2'
services:
  activemq:
    image: docker.k-vrach.ru/activemq:latest
    container_name: activemq
    ports:
      - "8161:8161"
      - "61616:61616"
  frmr:
    image: docker.k-vrach.ru/frmr:pglatest
    container_name: frmr
    volumes:
      - /srv/hosts/FRMR/tomee.xml:/opt/tomee/conf/tomee.xml
      - /srv/hosts/FRMR/context.xml:/opt/tomee/webapps/FRMR/META-INF/context.xml
      - /srv/hosts/FRMR/crypto.properties:/opt/tomee/webapps/FRMR/WEB-INF/classes/crypto.properties
      - /srv/hosts/FRMR/db.properties:/opt/tomee/webapps/FRMR/WEB-INF/classes/db.properties
      - /srv/hosts/FRMR/db.postgresql.properties:/opt/tomee/webapps/FRMR/WEB-INF/classes/db.postgresql.properties
      - /srv/hosts/FRMR/frmr.properties:/opt/tomee/webapps/FRMR/WEB-INF/classes/frmr.properties
      - /srv/hosts/FRMR/fer.properties:/opt/tomee/webapps/FRMR/WEB-INF/classes/fer.properties
      - /srv/hosts/FRMR/keys:/var/opt/cprosp/keys/root
      - /srv/hosts/FRMR/logs:/opt/tomee/logs
    environment:
      #Такие параметры уже включены в контейнер можно изменить их при необходимости
      - TZ=Asia/Yekaterinburg
      - CATALINA_OPTS=-Xms512M -Xmx1536M -server -XX:+UseParallelGC -Dfile.encoding=UTF-8
      - JCP_KEY_LICENSE=''
      - JCP_COMPANY_NAME=''
    depends_on:
      - activemq
```

**Code Block 90 docker-compose.yml**

## 12.1.2 Установка напрямую на ОС

### 12.1.2.1 Необходимое ПО

### 12.1.2.2 Установка TomEE

### 12.1.2.3 Настройка TomEE как сервиса

## 12.2 Описание файлов конфигурации.

```
<?xml version="1.0" encoding="UTF-8"?>
<tomee>

<!-- внешний ActiveMQ -->

<Resource id="ConnectionFactoryResourceAdapter"
type="ActiveMQResourceAdapter">
# Do not start the embedded ActiveMQ broker
BrokerXmlConfig =
    ServerUrl = tcp://<ACTIVEMQ_HOST>:<ACTIVEMQ_PORT>
</Resource>

<Resource id="ConnectionFactory" type="javax.jms.ConnectionFactory">
ResourceAdapter = ConnectionFactoryResourceAdapter
</Resource>

<Resource id="frmr.sendResponse.destination" type="javax.jms.Queue">
destination = ru.swan.frmr.sendResponse
</Resource>

<Resource id="frmr.sendResponseSlot.destination" type="javax.jms.Queue">
destination = ru.swan.frmr.sendResponse
</Resource>

<Resource id="frmr.integrationSession.destination" type="javax.jms.Queue">
destination = ru.swan.frmr.integrationSession
</Resource>

<Resource id="frmr.FRMRSessionReplicatorJob.destination"
type="javax.jms.Queue">
destination = ru.swan.frmr.integrationSession
</Resource>

</tomee>
```

### Code Block 91 tomee.xml

**<ACTIVEMQ\_HOST>** - IP адрес ActiveMQ (Должен совпадать с адресом ActiveMQ ФРМО)  
**<ACTIVEMQ\_PORT>** - ПОРТ AvtiveMQ

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE xml>
<Context>
  <Resource
    name="jms/ConnectionFactory"
    auth="Container"
    type="org.apache.activemq.ActiveMQConnectionFactory"
    description="JMS Connection Factory"
    factory="org.apache.activemq.jndi.JNDIReferenceFactory"

    brokerURL="tcp://<ACTIVEMQ_HOST>:<ACTIVEMQ_PORT>?jms.redeliveryPolicy.maximumRedeliveries=5&jms.redeliveryPolicy.backOffMultiplier=2&jms.redeliveryPolicy.useExponentialBackOff=true"
  />

  <Resource name="jms/frmr.sendResponse.destination"
    auth="Container"
    type="org.apache.activemq.command.ActiveMQQueue"
    factory="org.apache.activemq.jndi.JNDIReferenceFactory"
    physicalName="ru.swan.frmr.sendResponse" />

  <Resource name="jms/frmr.integrationSession.destination"
    auth="Container"
    type="org.apache.activemq.command.ActiveMQQueue"
    factory="org.apache.activemq.jndi.JNDIReferenceFactory"
    physicalName="ru.swan.frmr.integrationSession" />

  <Resource name="jms/frmr.sendResponseSlot.destination"
    auth="Container"
    type="org.apache.activemq.command.ActiveMQQueue"
    factory="org.apache.activemq.jndi.JNDIReferenceFactory"
    physicalName="ru.swan.frmr.sendResponse" />

</Context>

```

**Code Block 92 context.xml**

**<ACTIVEMQ\_HOST>** - IP адрес ActiveMQ (Должен совпадать с адресом ActiveMQ ФРМО)

**<ACTIVEMQ\_PORT>** - ПОРТ AvtiveMQ

```

org.apache.ws.security.crypto.provider=ru.swan.soap.crypto.LocalMerlin
org.apache.ws.security.crypto.merlin.keystore.provider=JCP
org.apache.ws.security.crypto.merlin.keystore.type=HDImageStore
crypto.user=<CRYPTO_USER>
crypto.signatureUser=<CRYPTO_USER>
crypto.password=<CRYPTO_PASS>
crypto.isTrustAllCerts=true
crypto.signature.algorithm=<CRYPTO_ALGORITHM>
crypto.signature.digest.algorithm=<CRYPTO_DIGEST_ALGORITHM>

```

**Code Block 93 crypto.properties**

**<CRYPTO\_USER>** - имя контейнера КриптоПро

**<CRYPTO\_PASS>** - пароль от контейнера КриптоПро

**<CRYPTO\_ALGORITHM>** - алгоритм шифрования контейнера КриптоПро. пример для ГОСТ 2012\_256 (SIGNATURE\_V2\_ALGORITHM\_2012\_256)

**<CRYPTO\_DIGEST\_ALGORITHM>** - алгоритм шифрования контейнера КриптоПро. пример для ГОСТ 2012\_256 (SIGNATURE\_V2\_DIGEST\_ALGORITHM\_2012\_256)



```
db.driver=org.postgresql.Driver
db.url=jdbc:postgresql://<DB_HOST>:<DB_PORT>/<DB_NAME>?stringtype=unspecif
ied
db.username = <DB_USER>
db.password = <DB_PASS>

frmrSnilsTest='13577513780'
frmrLpuTest='10010833'
frmrBulkLoad=0
```

#### Code Block 94 db.properties

<DB\_HOST> - IP Адрес сервера БД  
<DB\_PORT> - Порт сервера БД  
<DB\_NAME> - Имя БД  
<DB\_USER> - Пользователь для авторизации в БД  
<DB\_PASS> - Пароль для авторизации в БД

**misId** - Идентификатор системы. получается после регистрации системы. Имеет вид conc-00000-rmis

**frmrSnilsTest** - Данная настройка используется для теста системы. СНИЛС тестовой организации

**frmrLpuTest** - Данная настройка используется для теста системы. Идентификатор тестовой организации

**frmrBulkLoad** - Данная настройка используется для теста системы.

ДАЛЕЕ НЕОБХОДИМО ОПИСАТЬ ВСЕ ОСТАВШИЕСЯ НАСТРОЙКИ

```
db.driver=org.postgresql.Driver
db.url=jdbc:postgresql://<DB_HOST>:<DB_PORT>/<DB_NAME>?stringtype=unspecif
ied
db.username = <DB_USER>
db.password = <DB_PASS>

frmrSnilsTest='13577513780'
frmrLpuTest='10010833'
frmrBulkLoad=0
```

#### Code Block 95 db.postgresql.properties

Совпадает с предыдущим конфигом. Описание можно посмотреть там.

```
frmo.serviceUrl=http://ips-test.rosminzdrav.ru/57234d87b0838
frmo.callbackUrl=http://www.w3.org/2005/08/addressing/anonymous
frmo.callbackIpsUrl=https://ips-test.rosminzdrav.ru/wro0662w70vd9

ips.isHandleIpsHeaders=true
ips.clientEntityId=
ips.action=sendDocument
ips.responseAction=http://emu.callback.mis.service.nr.eu.rt.ru/Callback/sendResponseResponse

region=r59
user=1

db=postgresql
```

#### Code Block 96 frmr.properties

**frmo.serviceUrl** - для рабочего сервера <https://ips.rosminzdrav.ru/57406573a4083>, для тестового <http://ips-test.rosminzdrav.ru/57234d87b0838> . [ips.rosminzdrav.ru](https://ips.rosminzdrav.ru) - должен быть виден через ЗСПД

**frmo.callbackIpsUrl** - адрес похож на frmo.serviceUrl . Получается после регистрации.

**ips.clientEntityId** - ID сервиса в формате UUID. Получается после регистрации.

**region** - Написать схему региона

**db** - postgresql или mssql в зависимости какую БД вы используете

```
service.url=http://test-api2-fer.rt-eu.ru:80/concentrator_ws/ws/mis/

soap.is12=true
soap.isLog=true

fer.integrationSession.listener.count = 10
fer.SendResponse.listener.count = 1
```

#### Code Block 97 fer.properties

# 13 Инструкция по внедрению сервиса ФРМР / ФРМО в МИС (РМИС)

## 1. Подключение МИС (РМИС) к тестовой версии сервиса ФРМР / ФРМО

При подключении одновременно ФРМО и ФРМР некоторые пункты необходимо выполнять только один раз. ФРМО и ФРМР — это практически один сервис, callback разворачивается один и так же, он используется при регистрации.

1) В первую очередь необходимо зарегистрировать систему в тестовой версии ИПС. Для этого необходимо прислать на электронный адрес СТП ЕГИСЗ [egisz@rt-eu.ru](mailto:egisz@rt-eu.ru) заявку по форме, указанной в П.1.1. методических материалов по подключению, расположенных по адресу <https://portal.egisz.rosminzdrav.ru/materials/11> (заявка №1 архива с инструкцией) и приложить к ней сертификат открытого ключа в формате .cer. Данный сертификат будет использоваться в сервисах ИЭМК, ФРМО/ФРМР, РЭМД как сертификат для авторизации в фед. сервисах. Подается одна заявка для ФРМО и ФРМР.

2) Далее необходимо зарегистрировать тестовую версию web-службы в тестовой версии ИПС.

1. Поставить задачу эксплуатации на развертку тестовой web-службы. (Инструкция для Эксплуатации есть в архиве с инструкциями и заявками)

2. После выполнения по разворачивании тестовой web-службы, заполнить заявку согласно формам, приведенным в п.1.3 документа «Методические материалы по подключению к ИПС», размещенным на портале ЕГИСЗ по адресу: <http://portal.egisz.rosminzdrav.ru/materials/11>. (Заявка №2 архива с инструкциями) и приложить к ней сертификат открытого ключа в формате .cer.

3) Заявки оформляются в строгом соответствии с форматом. На заявках должна быть поставлена дата, подпись и расшифровка подписи ответственного лица органа государственной власти субъекта РФ в сфере охраны здоровья (для организаций, подведомственных ОУЗ) или МО (для иных ведомственных принадлежностей), печать органа исполнительной власти субъекта Российской Федерации в сфере здравоохранения. Также необходимо направить на электронный адрес службы технической поддержки ЕГИСЗ [egisz@rt-eu.ru](mailto:egisz@rt-eu.ru) скан-копию подписанной заявки в формате PDF (JPG) и заполненную заявку в формате DOC (DOCX).

4) В течении 2 (двух) рабочих дней после получения заявки СТП ЕГИСЗ регистрирует ИС в тестовой версии ИПС и предоставляет идентификатор ИС в ИПС.

5) Следующая заявка подается на регистрацию в тестовой версии Системы. (Заявка №3 архива с инструкциями). Поля таблицы 2. Заявки №3, заполняются в соответствии с ранее полученными данными, а также важно заполнить правильно Методы, используемые в системе. Список возможных доступных методов, находится в документе ФРМО\_Описание\_интеграционных\_профилей.doc. Актуальная версия документа размещена по адресу: <https://portal.egisz.rosminzdrav.ru/materials/481>. Подаются 2 отдельных заявки для ФРМО и ФРМР

6) В течение 2 (двух) рабочих дней после получения заявки СТП ЕГИСЗ регистрирует ИС в тестовой версии Системы и предоставляет подтверждение о регистрации ИС в ответ на заявку.

## 2. Тестирование и массовые испытания ФРМР / ФРМО

1) Для создания отчета о тестировании по ФРМО/ФРМР просто копируем из логов сообщения и добавляем их в новый xml файл. Обычно требуются запрос, синхронный ответ, асинхронный ответ и асинхронный запрос. Проводится так же как массовая выгрузка только в меньшем объеме. Инструкция по проведению Массовой выгрузки находится в Conf: [Сервис интеграции с ФРМР](#)

2) В течение 30 (тридцати) рабочих дней со дня выделения идентификатора ИС необходимо провести тестирование при взаимодействии с тестовой версией Системы и направить отчет о результатах тестирования, а также протокол приемочных испытаний на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)), в соответствии с формой, приведенной в данном архиве (№4 Отчет о проведении тестирования). В рамках тестирования должны быть пройдены все проверки в соответствии с программой и методикой испытаний интеграции ИС с Системой. Результатом тестирования является успешное выполнение всех проверок.

3) В случае успешного прохождения тестирования интеграционного взаимодействия СТП ЕГИСЗ в ответном письме уведомляет нас о необходимости проведения массовой загрузки/выгрузки данных в тестовую версию Системы.

СТП ЕГИСЗ в течение 4 (четырёх) рабочих дней с момента получения отчета о проведении тестирования ожидает заявки. Заявки мы должны направить в соответствии с формой, приведенной в архиве (№5 Заявка на массовую выгрузку). Дата и время проведения и время проведения ФРМО и ФРМР может различаться. Рекомендуем ставить дату проведения с разницей не более 1 недели.

4) Согласно плановой дате проведения массовой загрузки/выгрузки ИС осуществляет массовую загрузку/выгрузку данных в тестовую версию Системы, но плановая дата должна быть не раньше ответа от СТП о том, что можно проводить массовую выгрузку согласно заявке №5.

Минимальное количество запросов на каждую доступную подсистему и каждый из ее методов при массовой загрузке должно составлять 20. Данное значение указано в документе ФРМР\_Описание\_интеграционных\_профилей.doc. Актуальная версия документа размещена по адресу: <http://portal.egisz.rosminzdrav.ru/materials/483>. ФРМО\_Описание\_интеграционных\_профилей.doc. Актуальная версия документа размещена по адресу: <https://portal.egisz.rosminzdrav.ru/materials/481>.

5) После выгрузки необходимо уведомить СТП ЕГИСЗ, что была проведена массовая выгрузка С ... ПО ... Вставить время по часовому поясу MSK и указать дату проведения. Также рекомендуется приложить идентификатор ИПС тестовой среды.

Пример: "Была проведена массовая выгрузка ФРМР проведена 7:57-11:32 MSK. Дата 21.07.2020, идентификатор системы в тестовой версии ИПС Ярославской Области - 61073f90-ae31-4486-3f61-b35a04aacc58"

В течении 4 рабочих дней СТП ЕГИСЗ проверяет нашу выгрузку и направляет заключение об успешности проведения массовой загрузки/выгрузки данных и о возможности подключения к промышленной версии.

### **3. Подключение МИС(РМИС) к продуктивной версии сервиса ФРМР**

1) Получить квалифицированную электронную подпись в аккредитованном УЦ, зарегистрировать ИС в рабочей версии ИПС. Подпись предоставляет регион. Обязательна подпись должна соответствовать ГОСТ Р 34.10-2012 (256 бит). Другой ГОСТ или разрядность сертификата СТП ЕГИСЗ не примет, например ГОСТ Р 34.10-2012 (512 бит) не принимается по причинам несоответствия разрядности. Заявка №6 Данного Архива. Ключ приобретается регионом в 1 экземпляре, для работы ФРМО и ФРМР.

2) Делаем задачу на разворачивание рабочей версии веб службы, на эксплуатацию (группу ОЭИР. РМИС). Им необходимо предоставить. Лицензию JSP для крипто про. По одной для ФРМО или 2 если ставим сразу ФРМР. Также прикладываем ключ для обмена, его предоставляет нам регион. Так же необходим Закрытый ключ (контейнер) от сертификата, запрашиваем одновременно с сертификатом, регион может потребовать официальное письмо для предоставления контейнера, закрытый ключ (контейнер) нужен от сертификата, который мы регистрировали в СТП ЕГИСЗ на предыдущем шаге.

3) После настройки рабочей версии web-службы, потребуется передать региону IP-адреса для настройки сети ЗСПД. Доступ на указанные IP-адреса настраивает МИАЦ региона. Так-же потребовать от региона чтобы они сделали доступ к 2IP Адресам

федерального сервиса. Возможны корректировки нужных IP. Актуальные IP можно взять с <http://portal.egisz.rosminzdrav.ru/materials/11> Пункт 2.3.2. Порядок публикации веб-службы документации "Методические материалы по подключению к сервису ИПС". На данный момент актуальны IP:

1. Открытая сеть - 185.194.33.39;
2. Закрытая сеть - 10.41.19.39.

4) После настройки рабочей версии web-службы. Подаем заявку №7 (В архиве). на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)) согласно формам, приведенным в п.1.4 документа «Методические материалы по подключению к ИПС», размещенным на портале ЕГИСЗ по адресу: <http://portal.egisz.rosminzdrav.ru/materials/11>. Одновременно сканированную копию заявки направить в Минздрав России, по адресу: [di@rosminzdrav.ru](mailto:di@rosminzdrav.ru).

5) Регистрируем ИС в промышленной версии Системы. Заполняем Заявку №3 Данного архива и подчеркиваем в заявке Промышленная версия ИС. Отправляем заявку в СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)). Одновременно сканированную копию заявки направить в Минздрав России, по адресу: [di@rosminzdrav.ru](mailto:di@rosminzdrav.ru).

В течение 2 (двух) рабочих дней после получения заявки СТП ЕГИСЗ логин/пароль доступа, а также ссылку к web-интерфейсу рабочей версии системы.

6) Передаем информацию эксплуатации для внесения изменений в настройки рабочей версии ФРМР.

#### 4. Архив со всеми необходимыми заявками

Файл предоставляется разработчиком Сервиса.

## 1 Общая информация

ФРМО - Федеральный Реестр Медицинских организаций.

ФРМР - Федеральный Реестр Медицинских Работников.

Оба модуля проверяются на рабочих серверах.

Существуют следующие режимы работы:

5. регулярный режим запуска
6. Режим первого запуска
7. Массовая выгрузка (только ФРМР)

При работе модулей могут выполняться следующие действия:

2. Передача новых данных(create) - осуществляется передача данных из РМИС в систему ФРМО/ФРМР.
3. Обновление уже существующих данных(update) - сервис передает измененные данные в систему ФРМО/ФРМР.
4. Удаление существующих данных(delete) - удаление данных из систем ФРМО/ФРМР
5. Чтение существующих данных (read) - считываем данные из систем ФРМО/ФРМР.

## 2 Способы тестирования

**!!!ВАЖНО!!!:**

**Должно быть полное соответствие данных на портале ФРМО/ФРМР и в РМИС для успешного выполнения сервиса.**

Для проверки обновления данных в РМИС должны быть одинаковые ключи на портале и в РМИС.

Для ФРМО ключи - OID, для ФРМР - СНИЛС, ФИО.

!!!!

## 2.1 Регулярный запуск

Регулярный, или обычный режим запуска. Для проверки необходим доступ АРМ Цод, формы Журнал работы сервисов.

Запуск сервиса ФРМО/ФРМР производится с формы. При этом

2. Передача данных в сервис ФРМО/ФРМР - выполняется list/create/update/delete данных
3. Сервис обновления ФРМО - выполняется read и обновление/создание/удаление данных в РМИС

Логи и ошибки смотрятся в детальном логе работы сервисов.

## 2.2 Режим первого запуска

Осуществляется в том случае, если сервис ранее не запускался.

Для тестирования переводится в режим первого запуска вручную через Админов/Разработчиков.

Во время режима первого запуска запускаются и проверяются оба сервиса. Проверка начинается с ФРМО.

Для успешного прохождения необходима полное соответствие структуры на портале и в РМИС. Если есть какое либо несоответствие(пример: не совпали название отделение, тип отделения, или отделения вообще нет) - будет ошибка.

После выполнения успешного выполнения ФРМО автоматически будет запущен ФРМР, который произведет выгрузки медперсонала.

## 2.3 Режим массовой выгрузки

Режим используется в ФРМР.

Перед массовой выгрузкой производится выгрузка 1 медработника и составление отчета (собираются синхронные и асинхронные запросы/ответы из логов. на каждый регион свои логи, ссылку уточнять отдельно через hd).

Для массовой выгрузки создается не менее 100 людей, 20 - добавляются в систему как врачи, 80 - остаются массовойкой.

Для успешного проведения массовой выгрузки у врачей должно быть полностью одинаковые данные об образовании.

При создании данных об образовании и выборе учебных заведений - лучше сверяться со справочником psi, но это не всегда может помочь.

Массовая выгрузка проводится на новых регионах, для проведения необходим доступ к БД.

Данные для выгрузки собираются в таблицу, запуск сервиса производится через скрипты из БД.

Все запросы/ответы смотрятся через ошибки. Во время проведения массовой выгрузки **НЕ ДОЛЖНО БЫТЬ ОШИБОК.**

Если в процессе проведения массовой выгрузки были получены ошибки, то изменяются данные врачей/массовки:

- СНИЛС - если был выполнен запрос person.create - **МЕНЯЕТСЯ ВСЕГДА. !!!ВАЖНО!!! СНИЛС ПЕРЕД ОТПРАВКОЙ/ДОБАВЛЕНИЕМ В ТАБЛИЦУ - ВСЕГДА ПРОВЕРЯТЬ НА ВАЛИДНОСТЬ !!!!**
- Паспортные данные - если был выполнен метод document.create, то меняется номер и серия паспорта. **!!! ВАЖНО !!! СМОТРИМ И ВЫБИРАЕМ ТАКИЕ НОМЕР И СЕРИЮ, ЧТО БЫ НЕ ПЕРЕСЕКАЛИСЬ С РЕАЛЬНЫМИ ЛЮДЬМИ !!!!**

## 2.4 Дополнительная информация и ссылки

Для создания отчета по ФРМО/ФРМП просто копируем из логов сообщения и добавляем их в новый xml файл. Обычно требуются запрос, синхронный ответ, асинхронный ответ и асинхронный запрос.

На новых регионах ФРМО/ФРМП может отсутствовать в журнале работы сервисов. Тогда запускается через БД, или через разработку добавляются сервисы.

При проверке ФРМО/ФРМП всегда **ВАЖНО СООТВЕТСТВИЕ ДАННЫХ**.

Если для ФРМП меняются данные врача, особенно для массовой выгрузки - менять **ТОЛЬКО ЧЕРЕЗ ПЕРЕОДИКИ**

[SQL Скрипты и порядок действий для массовой выгрузки](#)

Для первого запуска есть [инструкция как привести структуру в порядок](#) и есть [Отчет о расхождении структуры МО в РМИС и ФРМО](#)

## 3 Postgre ФРМО. Отчет о тестировании и массовая выгрузка.

### 3.1 Общие сведения:

3.1.1 Для запуска единичной сессии ФРМО можно использовать скрипт FRMOsession\_run.sql:

```
do
$$
DECLARE
RC int;
Lpu_id bigint;
FRMOsession_Name varchar(100);
FRMOsession_begDT timestamp;
FRMOsessionMode_id bigint = 1 ;-- 1 - экспорт в ФРМО 2 -- импорт из ФРМО 3 -- только
чтение данных из ФРМО
pmUser_id bigint;
Error_Code VARCHAR;
Error_Message VARCHAR(4000);
begin

p_Lpu_id := 10010833; -- 13004038 -- 10011030 -- 13101710
p_FRMOsession_Name := 'FRMO export test';
p_FRMOsession_begDT := '2020-08-18';
p_pmUser_id := 1;

/*SELECT s.FRMOsession_id, st.FRMOsessionActionType_Code,
st.FRMOsessionHist_doneDT, st.FRMOsessionHist_pid
FROM dbo.FRMOsession s
left join lateral (select a.FRMOsessionActionType_Code, h.FRMOsessionHist_doneDT,
h.FRMOsessionHist_pid
from dbo.FRMOsessionHist h
inner join dbo.FRMOsessionActionType a on a.FRMOsessionActionType_id =
h.FRMOsessionActionType_id
```

```

where s.FRMOSession_id = h.FRMOSession_id
order by h.FRMOSessionHist_insDT desc
limit 1
) st on true
WHERE and s.FRMOSession_endDT is NULL and s.FRMOSession_Service = 'FRMO' and
s.Lpu_id = Lpu_id;*/

```

```

select fnc.Error_Code,fnc.Error_Message
into Error_Code,Error_Message
from
dbo.p_FRMOSession_run(Lpu_id,FRMOSession_Name,FRMOSession_begDT,FRMOSession
Mode_id,pmUser_id) fnc;
if (Error_Code is not null)
then
RAISE USING ERRCODE = error_code,MESSAGE =Error_Message;
end if;
end;
$$;

```

В параметрах указываются:

- 1) Режим сессии FRMOSessionMode\_id bigint := 1 -- 1 - экспорт в ФРМО 2 -- импорт из ФРМО 3 -- только чтение данных из ФРМО
- 2) МО: Lpu\_id := 10010833 -- 13004038 -- 10011030 -- 13101710
- 3) имя сессии: FRMOSession\_Name := 'FRMO export test'
- 4) дата запуска сессии: FRMOSession\_begDT := '2020-08-18'
- 5) пользователь: pmUser\_id := 1

3.1.2 Ошибки по сессии можно посмотреть по скрипту FRMOSessionError.sql:  
--В ФРМО отсутствует подразделение с ОИД подразделения  
1.2.643.5.1.13.13.12.2.59.5834.0.17454  
--В ФРМО отсутствует отделение с ОИД подразделения  
1.2.643.5.1.13.13.12.2.59.5834.0.17454 ОИД отделения  
1.2.643.5.1.13.13.12.2.59.5834.0.17454.159761  
--Для отделения Кабинет функциональной диагностики стационара Параклиника ЖК с  
ОИД 1.2.643.5.1.13.13.12.2.59.5834.0.34090 в НСИ указан другой ОИД  
1.2.643.5.1.13.13.12.2.59.5834.0.17454  
--ОИД подразделения 1.2.643.5.1.13.13.12.2.59.5834.0.35803 ставка 149 отсутствует в  
РМИС

```

select
FE.FRMOSessionError_id, fe.FRMOSessionHist_id, st.FRMOSessionHist_id
, coalesce(fet.FRMOSessionErrorType_Name, fe.FRMOSessionError_Message) as error_text
, a.FRMOSessionActionType_descr
, L.Lpu_Nick
, lb.LpuBuildingPass_Name, lu.LpuUnit_Name, LS.LpuSection_Name, fe.LpuStaff_Num,
lu.LpuUnit_ID, FE.LpuSection_id, fe.LpuBuildingPass_id
, fe.FRMOSessionError_Message
, FE.FRMOSessionError_insdt
FROM dbo.FRMOSession ses
inner join v_Lpu L on L.Lpu_id = ses.Lpu_id
left join lateral (select h.FRMOSessionHist_id, h.FRMOSessionHist_insDT
from dbo.FRMOSessionHist h
inner join dbo.FRMOSessionActionType a on a.FRMOSessionActionType_id =
h.FRMOSessionActionType_id
where ses.FRMOSession_id = h.FRMOSession_id and a.FRMOSessionActionType_code in
(60,64)

```



```

-- and h.FRMOSessionHist_id = 1334
order by h.FRMOSessionHist_id desc
limit 1
) st on true
inner join dbo.FRMOSessionError fe on fe.FRMOSession_id = ses.FRMOSession_id and
fe.FRMOSessionHist_id >= st.FRMOSessionHist_id
inner join dbo.FRMOSessionHist h on fe.FRMOSessionHist_id = h.FRMOSessionHist_id
inner join dbo.FRMOSessionActionType a on a.FRMOSessionActionType_id =
h.FRMOSessionActionType_id
left join dbo.FRMOSessionErrorType fet on fe.FRMOSessionErrorType_id =
fet.FRMOSessionErrorType_id
left JOIN dbo.v_LpuBuildingPass AS lb ON lb.LpuBuildingPass_id = fe.LpuBuildingPass_id
left JOIN dbo.v_LpuUnit AS lu ON lu.LpuUnit_id = fe.LpuUnit_id
left join v_LpuSection LS on LS.LpuSection_id = FE.LpuSection_id
where (1 = 1) and ses.FRMOSession_id = :FRMOSession_id
and h.FRMOSessionHist_insDT >= coalesce(st.FRMOSessionHist_insDT,
ses.FRMOSession_insDT)
-- and fe.LpuBuildingPass_id in (383, 2400)
order by error_text, fe.FRMOSessionHist_id

Параметр Ид сессии: :FRMOSession_id bigint = 1.

```

### 3.2 Запуск массовой выгрузки:

### 3.3 Запуск массовой выгрузки выполняется скриптом MassiveTestCycle0.sql:

```

do
$$
DECLARE
p_lpu_id bigint := 13101710; -- 150007 -- Карелия -- 12600015 -- Вологда 21 -- Уфа 150519 --
13003272 -- Астра 150001 -- Пенза 150519 -- Крым 150447 -- Хакасия -- 13003345 --Псков --
10010833 Пермь
p_pmUser_iD bigint := 1;
p_today timestamp := getDate();

-- для теста Обновления mo.update
p_LpuPmuType_id bigint := 3;
p_LpuPmuType_nid bigint := 35;

-- здание для создания нового /удаления, которое не используется в подразделениях и
медоборудовании
p_LpuBuildingPass_id bigint := 1874; -- Ярославль 383 -- Уфа Административное здание
305 -- Хакасия 10185 -- Карелия Тестовое здание 369 -- Псков Здание поликлиники 193 --
Астра Поликлиническое отделение 752 --Пенза Гараж 1 -- 147 -- Вологда 17 -- ПЕРмь раб
Поликлиника 72 -- Вологда Тестовая площадка 2400 -- Уфа Николаевская Амбул 283 --
193 -- Астра Поликлиническое отделение 752 -- Пенза Гараж 1 1620 -- Крым ООВП (СМ)
с.Межводное 282 -- Хакасия 283 -- Псков
p_MOArea_id bigint := 10004 ;-- Яросл 1526 -- Уфа 25 -- Хакасия 23 -- Карелия 46 -- Псков
780 -- Пермь рабоч 32 -- Вологда 993 -- Уфа Николаевская амб 336 -- Астра
Поликлиническое отделение №1 616 -- Неверкино Гараж 1 25 -- Тест Ширинская РБ --1 --
Псков
-- площадка для нового здания
p_MOArea_nid bigint := 10003; -- Яросл 1528 -- Уфа 75 -- Хакасия 24 -- Карелия 48 -- Псков
645 -- Астра Поликлиническое отделение №1 тест --1022 -- Крым Хмелевский ФАП 77 --
Вологда Первомайская 993 -- Уфа Николаевская амб 336 -- Астра Поликлиническое
отделение №1 605 -- Неверкино Поликлиннака 25 -- Тест Ширинская РБ --1 --Псков
-- для теста создания/удаления здания
p_LpuBuildingPass_Name_new varchar(50) := 'Здание в парке'; -- Ufa 'Тестовое здание' --
Karelia, Nakasia 'Здание поликлиники' -- Pskov Крым 'Здание тестовое нов 22'--Vologda

```

```

p_LpuBuildingPass_Name_old varchar(50) := 'Здание в саду'; -- Ufa 'Тестовое здание' --
Karelia, Hakasia 'Здание поликлиники' -- Pskov -- Крым 'Здание тестовое' --Vologda
'Николаевская врачебная амбулатория' --Astra 'ФАП Благодатка' --Пенза 'ООВП (СМ) с
Межводное' -- Крым --Шира, Хак '5эт.здание больницы с поликлиникой' -- Псков здесь
сохранить оригинальное название здания

-- для теста Обновления mo_building.update
-- здание для редактирования, которое можно использовать в подразделениях
p_LpuBuildingPass2_id bigint := 1873; -- Яросл 2400 -- Уфа Николаевская Амбул 282 --
Хакасия Большое здание больницы с поликлиникой 76 -- Карелия Поликлиника 283 --
Псков 5эт.здание больницы с поликлиникой 192 -- Astra Женская консультация -- 366 --
Крым Поликлинико-административное здание 100 -- Вологда Базовый корпус
p_LpuBuildingPass_YearBuilt_new timestamp := '20180101':timestamp;
p_LpuBuildingPass_YearBuilt_old timestamp := '2003-01-01 00:00:00.000':timestamp;-- Псков
Astra Крым '1987-01-01 00:00:00.000'-- Пенза здесь сохранить оригинальное год постройки

-- подразделение для создания/удаления, которое не используется в штатке
p_LpuUnit_id bigint := 99560015008; -- Яросл 252 -- Ufa Травмпункт 344 -- Хакасия Стомат
30920 -- Карелия здравпункты 495 --Pskov Клинико-диагностическая лаборатория 440 --
Астра Дневной стационар -- 45 -- Пенза Яснополянская ВА 3306 -- Крым Межводное ОВП
1819 -- Вологда Рентгенологическое отделение 5758 -- Уфа Пол-ка Николаевская -- 2008 -
- Astra Поликлиническое отделение №1 (ул. С. Перовской, д. 71) 48 -- Пенза Благодатский
ФАП 3306 -- Крым Межводное ООВП 344 -- Хакасия Стомат 1001 -- Хакасия --761 -- Псков
-- для создания/удаления подразделения
p_LpuUnit_Name_new varchar(50) := 'Полка вторая'; -- Ufa 'Стоматологическая
поликлиника' -- Hakasia 'Здравпункты' -- Karelia 'Клинико-диагностическая лаборатория' --
Pskov 'Женская консультация №1 стационар' --Astra -- Крым --'Яснополянская врачебная
амбулатория' -- Пенза 'Рентгенологическое отделение новое 22' -- Вологда
p_LpuUnit_Name_old varchar(50) := 'Полка 2'; -- Яросл -- Ufa 'Стоматологическая
поликлиника' -- Hakasia 'Здравпункты' -- Karelia 'Клинико-диагностическая лаборатория' --
Pskov 'Женская консультация №1 стационар' --Astra -- Крым -- 'Яснополянская врачебная
амбулатория' -- Пенза 'Рентгенологическое отделение' -- Вологда -- 'Поликлиника
Николаевская ВА' --'Благодатский ФАП' -- здесь сохранить оригинальное название
Подразделения

-- для теста Обновления mo_depart.update
-- подразделение для редактирования
p_LpuUnit_id2 bigint := 99560015007; -- Yarosl 5758 -- Ufa Поликлиника Николаевская ВА
1001 -- Hakasia Коммунарковская поликлиника 30919 -- Карелия Центр здоровья 761 --
Pskov поликлиника 276 -- Astra Женская консультация 2153 -- Крым ООВП см
Новосельское
p_LpuUnitProfile_fid_new bigint := 1043 ;-- Yarosl --Ufa Ambulatory
-- а также для подготовки LpuUnit к create
p_LpuUnitProfile_fid_old bigint := 1011; --Yarosl Амбулатории -- Ufa 1124 -- Hakasia 1189 --
Карелия Центры здоровья для взрослых --1005 --Pskov Консультативно-диагностические
центры 1000 -- Astra Акушерско-гинекологические 1005 -- Крым, Пенза ООВП см
Новосельское 1005 -- 1021 --Vologda 1005 -- Астра, Уфа, Крым 1175 -- Функц.диагностики
1005 -- Амбулатории

-- для создания/удаления расписания
p_LpuStaff_id bigint := 2; --Яросл 14 -- Уфа 94 -- Hakasia 9 -- Карелия 70 -- Pskov 81 -- Astra
55 -- 102 -- Пенза 55 -- Crimea 184 -- Vologda расписание №55 85 -- Vologda 14 -- Ufa 55 --
Crimea 69 -- Псков 20 -- Astra 94 -- Пенза 40 -- Хакасия
p_LpuStaff_NumOld varchar(10) := '2'; -- Яросл -- Ufa '2' -- Pskov '1' -- Astra '34' -- Пенза-- '55'
--Vologda '1' --Астра, Пенза, Хакасия, Крым
p_LpuStaff_NumNew varchar(10) := '3'; -- Вологда, Pskov нов '2' --Астра, Пенза, Хакасия
p_LpuStaff_BeginDate_new timestamp := '20170101';
p_LpuStaff_EndDate_new timestamp := '20171231';
p_LpuStaff_BeginDate_old timestamp := '20180101';
p_LpuStaff_EndDate_old timestamp := '20181231';

-- для обновления расписания
-- LpuStaff_id = 336 -- Ufa 98 -- Hakasia 73 -- Karelia 69 -- Pskov

```

```

p_Staff_id varchar(10) := '22'; -- Ярослав Врач-гастроэнтеролог 42193 -- Ufa Врач-терапевт
8711 -- Hakasia 60041 -- Karelia 16153 -- Pskov
p_Staff_RateNew varchar(10) := '10'; -- Ярослав Karelia, Ufa, Pskov
p_Staff_RateOld varchar(10) := '100'; -- Ярослав Karelia, Ufa 4 -- Pskov

-- для Обновления mo_equipment.update
p_MedProductCard_id2 bigint := 135854 ;
p_MedProductCard_id bigint := 135853;
p_MedProductCard_SerialNumber_old varchar(100) := '2';
p_MedProductCard_SerialNumber_new varchar(100) := '3';
-- для создания/удаления mo_equipment.delete
p_MedProductCard_InventoryNumber_old varchar(100) := '11';
p_MedProductCard_InventoryNumber_new varchar(100) := '12';
p_RegCertificate_id bigint := 80158;
p_MedProductOrg_id_old bigint := 6880000889;
p_MedProductOrg_id_new bigint := 68320115679;

p_RC int;
p_FRMOSession_Name varchar(100);
p_FRMOSession_begDT timestamp;
Error_Code varchar;
Error_Message varchar(4000);

p_i int := 16;
p_count int := 22;

p_timestamp timestamp := dbo.tzGetDate();
p_FRMOSession_id bigint := null;
p_FRMOSessionActionType_Code int := null;
p_FRMOSessionHist_doneDT timestamp := null;
p_FRMOSessionHist_pid bigint := null;
p_FRMOSessionHist_id bigint := null;
p_err varchar;

begin
while (p_i < p_count)
loop
-- ТАКТ 1 обработки данных при массовой выгрузке
if (p_i % 2 = 0) then
RAISE notice 'такт 1 % %', p_i, GetDate() ;
-- mo.update
-- присваиваем новое значение
update dbo.Lpu set LpuPmuType_id = p_LpuPmuType_nid where Lpu_id=p_Lpu_id;

-- mo_building.update
-- присваиваем новое значение
update dbo.LpuBuildingPass set LpuBuildingPass_YearBuilt =
p_LpuBuildingPass_YearBuilt_new where LpuBuildingPass_id=p_LpuBuildingPass2_id;

-- mo_building.create+delete
-- присваиваем новое название и адрес
update dbo.LpuBuildingPass set
LpuBuildingPass_Name = p_LpuBuildingPass_Name_new || ' ' || p_i::varchar,
LpuBuildingPass_FRMOName = p_LpuBuildingPass_Name_new || ' ' || p_i::varchar,
MOArea_id = p_MOArea_nid
where LpuBuildingPass_id=p_LpuBuildingPass_id;

-- mo_depart.update
-- присваиваем новое значение
update dbo.LpuUnit set LpuUnitProfile_fid = p_LpuUnitProfile_fid_new where
LpuUnit_id=p_LpuUnit_id2;

```

```

-- тест mo_depart.delete+create
-- присваиваем новое название
update dbo.LpuUnit set LpuUnit_Name = p_LpuUnit_Name_new + '+LTRIM(str(p_i)) where
LpuUnit_id=p_LpuUnit_id;
--1.2.643.5.1.13.13.12.2.35.3259.0.33920 Vologda old
--1.2.643.5.1.13.13.12.2.35.3259.0.34484 Vologda new

-- тест mo_staff.delete
-- присваиваем новый номер
update dbo.LpuStaff set
LpuStaff_Num = p_LpuStaff_NumNew || p_i::varchar,
LpuStaff_begDate = p_LpuStaff_BeginDate_new,
LpuStaff_ApprovalDT = p_LpuStaff_BeginDate_new,
LpuStaff_endDate = p_LpuStaff_EndDate_new
where LpuStaff_id = p_LpuStaff_id;

-- mo_staff.update
-- присваиваем новое значение
update persis.Staff set Rate = p_Staff_RateNew where Staff.id = p_Staff_id;

-- mo_equipment.update
-- присваиваем новое значение
update passport.MedProductCard set MedProductCard_SerialNumber =
p_MedProductCard_SerialNumber_new where MedProductCard_id=p_MedProductCard_id2;

-- тест mo_equipment.delete+create
-- присваиваем новый инвентарный номер
update passport.MedProductCard set MedProductCard_InventoryNumber =
p_MedProductCard_InventoryNumber_new where MedProductCard_id=p_MedProductCard_id;
update passport.RegCertificate set Org_prid = p_MedProductOrg_id_new where
RegCertificate_id = p_RegCertificate_id;

else
-- ТАКТ 2 обработки данных при массовой выгрузке
-- mo_building.update
RAISE notice 'такт 2 % %', p_i, GetDate() ;
-- mo.update
-- возвращаем настоящее значение
update dbo.Lpu set LpuPmuType_id = p_LpuPmuType_id where Lpu_id=p_Lpu_id;

-- возвращаем настоящее значение
update dbo.LpuBuildingPass set LpuBuildingPass_YearBuilt =
p_LpuBuildingPass_YearBuilt_old where LpuBuildingPass_id=p_LpuBuildingPass2_id;

-- mo_building.create+delete
-- возвращаем настоящее название
update dbo.LpuBuildingPass set
LpuBuildingPass_Name = p_LpuBuildingPass_Name_old,
LpuBuildingPass_FRMOName = p_LpuBuildingPass_Name_old,
MOArea_id = p_MOArea_id
where LpuBuildingPass_id=p_LpuBuildingPass_id;

-- mo_depart.update
-- возвращаем прежнее значение
update dbo.LpuUnit set LpuUnitProfile_fid = p_LpuUnitProfile_fid_old where
LpuUnit_id=p_LpuUnit_id2;

-- тест mo_depart.delete+create
-- возвращаем настоящее название

```

```

update dbo.LpuUnit set LpuUnit_Name = p_LpuUnit_Name_old where
LpuUnit_id=p_LpuUnit_id;
--1.2.643.5.1.13.13.12.2.35.3259.0.34485 Vologda new 2

-- тест mo_staff.delete
-- возвращаем прежний номер
update dbo.LpuStaff set
LpuStaff_Num = p_LpuStaff_NumOld,
LpuStaff_begDate = p_LpuStaff_BeginDate_old,
LpuStaff_ApprovalDT = p_LpuStaff_BeginDate_old,
LpuStaff_endDate = p_LpuStaff_EndDate_old
where LpuStaff_id = p_LpuStaff_id;

-- mo_staff.update
-- возвращаем старое значение
update persis.Staff set Rate = p_Staff_RateOld where Staff.id = p_Staff_id;

-- mo_equipment.update
-- возвращаем настоящее значение
update passport.MedProductCard set MedProductCard_SerialNumber =
p_MedProductCard_SerialNumber_old where MedProductCard_id=p_MedProductCard_id2;

-- тест mo_equipment.delete+create
-- возвращаем настоящий инвентарный номер
update passport.MedProductCard set MedProductCard_InventoryNumber =
p_MedProductCard_InventoryNumber_old where MedProductCard_id=p_MedProductCard_id;
update passport.RegCertificate set Org_prid = p_MedProductOrg_id_old where
RegCertificate_id = p_RegCertificate_id;
end if;

/*select LpuBuildingPass_Name, LpuBuildingPass_YearBuilt, LpuBuildingPass_BuildingIdent
from dbo.LpuBuildingPass where LpuBuildingPass_id in (p_LpuBuildingPass_id,
p_LpuBuildingPass2_id);

select LpuUnit_Name, LpuUnitProfile_fid, LpuUnit_IsHomeVisit, UnitDepartType_fid,
LpuUnit_Phone
from dbo.LpuUnit where LpuUnit_id in (p_LpuUnit_id, p_LpuUnit_id2);

select * from dbo.v_LpuStaff where Lpu_id = p_Lpu_id;

select mc.MedProductCard_InventoryNumber, mc.MedProductCard_SerialNumber, rc.Org_prid
from passport.MedProductCard mc
left join passport.RegCertificate rc on rc.MedProductCard_id = mc.MedProductCard_id
where mc.MedProductCard_id in (p_MedProductCard_id, p_MedProductCard_id2);*/

p_FRMOSession_Name := 'frmo test '||p_j::varchar(2);
p_FRMOSession_begDT := dbo.tzGetDate();

SELECT s.FRMOSession_id, st.FRMOSessionActionType_Code,
st.FRMOSessionHist_doneDT, st.FRMOSessionHist_pid
into
p_FRMOSession_id,p_FRMOSessionActionType_Code,p_FRMOSessionHist_doneDT,p_FRM
OSessionHist_pid
FROM dbo.FRMOSession s
left join lateral (select a.FRMOSessionActionType_Code, h.FRMOSessionHist_doneDT,
h.FRMOSessionHist_pid
from dbo.FRMOSessionHist h
inner join dbo.FRMOSessionActionType a on a.FRMOSessionActionType_id =
h.FRMOSessionActionType_id
where s.FRMOSession_id = h.FRMOSession_id

```

```

order by h.FRMOsessionHist_insDT desc
limit 1 ) st on true
WHERE (1 = 1) and s.FRMOsession_endDT is NULL and s.FRMOsession_Service = 'FRMO'
and s.Lpu_id = p_Lpu_id
limit 1;

```

```

if p_FRMOsession_id is null then
select fnc.Error_Code,fnc.Error_Message,FRMOsession_id
into Error_Code,Error_Message,p_FRMOsession_id
from dbo.p_FRMOsession_ins (
FRMOsession_id := p_FRMOsession_id ,
FRMOsession_Name := p_FRMOsession_Name,
FRMOsession_begDT := p_FRMOsession_begDT,
Lpu_id := p_Lpu_id,
FRMOsession_Service := 'FRMO',
pmUser_id := p_pmUser_id) fnc;

```

```

if (Error_Code is not null)
then
RAISE USING ERRCODE = error_code,MESSAGE =Error_Message;
end if;
else
RAISE '*** ERROR CREATING SESSION';
end if;

```

```

RAISE notice 'FRMOsession_id=% %', p_FRMOsession_id,
p_FRMOsessionActionType_Code;

```

```

RAISE notice 'WAITING VERIFICATION PHASE %', GetDate() ;
p_FRMOsessionHist_id := null;

```

```

while (p_FRMOsessionHist_id is null) loop
-- select 'WAITING VERIFICATION PHASE'
SELECT pg_sleep(1);
select h.FRMOsessionHist_id
into p_FRMOsessionHist_id
from dbo.FRMOsessionHist h
where p_FRMOsession_id = h.FRMOsession_id and h.FRMOsessionActionType_id = 10057;
end loop;
RAISE notice 'WAITING ERRORS %', GetDate();
SELECT pg_sleep(1);
RAISE notice 'FRMOsessionHist_id=%', p_FRMOsessionHist_id;

```

```

if (exists (select * FROM dbo.FRMOsessionError fe WHERE fe.FRMOsessionHist_id =
p_FRMOsessionHist_id)) then
RAISE notice 'DELETING ERRORS';
delete FROM dbo.FRMOsessionError WHERE FRMOsessionHist_id =
p_FRMOsessionHist_id;
end if;

```

```

while ((SELECT s.FRMOsession_success FROM dbo.FRMOsession s
where p_FRMOsession_id = s.FRMOsession_id limit 1) is NULL)
loop
-- select 'WAITING SESSION COMPLETED'
SELECT pg_sleep(3);
end loop;

```

```

RAISE notice 'SESSION № % IS DONE %',p_i, GetDate();

```

```

SELECT coalesce(s.FRMOsession_comment,'') into p_err FROM dbo.FRMOsession s

```

```
where p_FRMOSession_id = s.FRMOsession_id
limit 1;
```

```
if (p_err <> '') then
p_err:= '*** ERROR EXECUTING SESSION '|| p_err;
raise '%',p_err;
end if;
```

```
p_i := p_i + 1;
end loop;
end;
$$;
```

```
--1.2.643.5.1.13.13.12.2.2.145.0.35074
```

В течение циклов происходит редактирование, создание и удаление объектов структуры MO

Указываются параметры:

12. Начальный счетчик циклов p\_i int = 0

13. Количество циклов p\_count int = 20

14. MO: p\_Lpu\_id = 10010833 -- 13004038 -- 10011030 -- 13101710

15. пользователь: p\_pmUser\_id = 1

-- для теста Обновления mo.update

- p\_LpuPmuType\_id bigint = 3
- p\_LpuPmuType\_nid bigint = 35

-- здание для создания нового /удаления, которое не используется в подразделениях и медоборудовании

- p\_LpuBuildingPass\_id bigint = 1874
- p\_MOArea\_id bigint = 10004 -- площадка здания

-- площадка для нового здания

- p\_MOArea\_nid bigint = 10003

-- для теста создания/удаления здания

- p\_LpuBuildingPass\_Name\_new varchar(50) = 'Здание в парке'
- p\_LpuBuildingPass\_Name\_old varchar(50) = 'Здание в саду'

-- для теста Обновления mo\_building.update

-- здание для редактирования, которое можно использовать в подразделениях

- p\_LpuBuildingPass2\_id bigint = 1873
- p\_LpuBuildingPass\_YearBuilt\_new timestamp= '20180101'
- p\_LpuBuildingPass\_YearBuilt\_old timestamp= '2003-01-01 00:00:00.000'

-- подразделение для создания/удаления, которое не используется в штатке

- p\_LpuUnit\_id bigint = 99560015008 -- для создания/удаления подразделения
- p\_LpuUnit\_Name\_new varchar(50) = 'Полка вторая'
- p\_LpuUnit\_Name\_old varchar(50) = 'Полка 2'

- для теста Обновления mo\_depart.update
- подразделение для редактирования
  - p\_LpuUnit\_id2 bigint = 99560015007
  - p\_LpuUnitProfile\_fid\_new bigint = 1043
  - p\_LpuUnitProfile\_fid\_old bigint = 1011
  
- для создания/удаления расписания
  - p\_LpuStaff\_id bigint = 2
  - p\_LpuStaff\_NumOld varchar(10) = '2'
  - p\_LpuStaff\_NumNew varchar(10) = '3'
  - p\_LpuStaff\_BeginDate\_new timestamp= '20170101'
  - p\_LpuStaff\_EndDate\_new timestamp= '20171231'
  - p\_LpuStaff\_BeginDate\_old Datetime = '20180101'
  - p\_LpuStaff\_EndDate\_old Datetime = '20181231'
  
- для обновления расписания
  - p\_Staff\_id varchar(10) = 22
  - p\_Staff\_RateNew varchar(10) = 10
  - p\_Staff\_RateOld varchar(10) = 100

#### **4 Postgre ФРМР. Отчет о тестировании и массовая выгрузка. Подробная инструкция.**

- [Общие сведения](#)
- [Подготовка.](#)
- [Проверка ФРМР.](#)
  - [Создание врача.](#)
    - i. [Заполнение персональных данных человека.](#)
    - ii. [Заполнение данных об образовании](#)
    - iii. [Заполнение данных о месте работы.](#)
- [Составление отчета о тестировании.](#)
- [Массовая выгрузка.](#)
  - a. [Проведение Массовой выгрузки.](#)
  - b. [Составление отчета](#)
- [SQL Scripts](#)
  - [Проверка и подготовка Сессии.](#)
  - [Запуск методов ФРМР:](#)



- [Работа с данными](#)
- [Проверка данных Врача](#)

## 4.1 Общие сведения

Массовая выгрузка ФРМР подразумевает под собой передачу данных сразу по нескольким врачам. Как правило, для тестирования создаются 100 тестовых людей, из которых 20 людей являются врачами, и 80 - массовой. Проводится массовая выгрузка только после успешно проведенной выгрузки ФРМО, т.к при выгрузке так же передается значение OID структурного подразделения. Так же следует учитывать, что при выгрузке ФРМО так же должны быть выгружены штатное расписание и должности, при этом количество ставок на должности должно быть не менее 100.

Сама массовая выгрузка делится на 3 этапа:

- Подготовка
- Проверка ФРМР.
- Массовая выгрузка

## 4.2 Подготовка.

Перед проведение проведением массовой выгрузки и началом работы с новым регионом необходимо будет выполнить следующие шаги:

- Установить [DBeaver](#) или любой аналогичный клиент для работы с БД. [Подробнее, как установить DBeaver](#)
- Запросить доступ к БД региона, на котором проводится массовая выгрузка. При этом:
  - Если выгрузка проводится с тестового сервера региона, то доступ необходим к тестовой БД
  - Если выгрузка проводится с рабочего сервера региона, то доступ необходим к рабочей БД.
  - В обоих случаях, доступ к БД необходим с правами на чтение, редактирование, добавление, удаление
- Подключиться через DBeaver или аналогичный клиент для работы с БД к той БД, к которой был предоставлен доступ. [Подробнее о том, как настроить подключение к DBeaver](#)

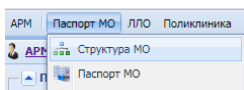
После выполнения этих шагов можно переходить к дальнейшим этапам.

## 4.3 Проверка ФРМР.

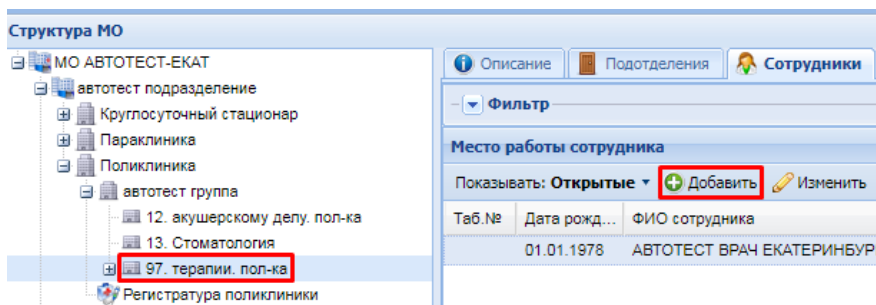
Данный необходим для проверки работы ФРМР и подразумевает под собой создание 1 тестового врача, выгрузка его на портал и составление отчета о результатах.

### 4.3.1 Создание врача.

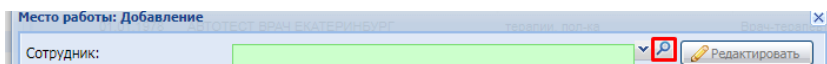
При проверке ФРМР создается только 1 врач. Для этого, зайти и авторизоваться в РМИС Промед. В верхнем меню найти кнопку "Паспорт МО", в выпадающем меню выбрать "Структура МО":



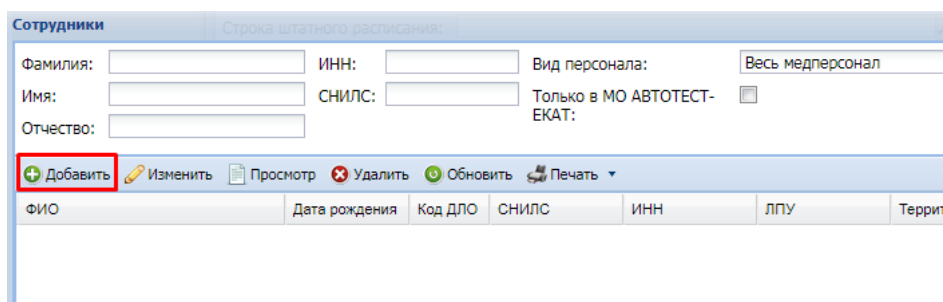
Раскрываем дерево МО и выбираем подразделение МО. Нажимаем кнопку "Добавить"



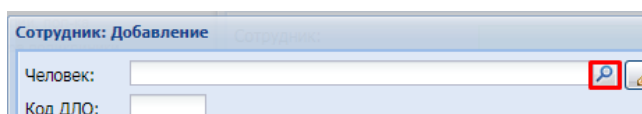
В открывшейся форме "Место работы: Добавление" нажимаем на изображение лупы напротив поля сотрудник:



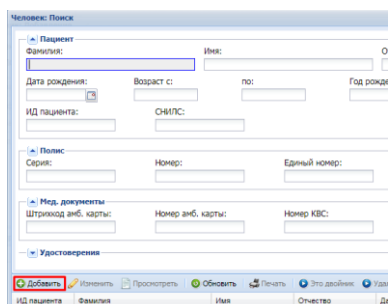
На форме "Сотрудники" снова нажимаем добавить:



На форме "Сотрудники: Добавление" ищем изображение лупы напротив поля "Человек" и нажимаем на него:



И последним шагом в этой цепочке - нажать на кнопку "Добавить" на форме "Человек: Поиск"



#### 4.3.1.1 Заполнение персональных данных человека.

На форме "Человек: Поиск" заполняем следующие поля:

2. ФИО человека: ТЕСТ ФРМР ОДИН
3. Дата рождения: любая дата, например 01.01.1980
4. Телефон человека: любой телефон, например: +71111111111
5. Пол человека.
6. СНИЛС человека: СНИЛС обязательно должен быть валидным, и не должен использоваться ранее для выгрузки на этом регионе. Валидность СНИЛС проверяется по [формуле](#), или можно воспользоваться [генератором](#) (СНИЛС в генераторе получаются почти всегда валидными. Но из-за почти всегда - лучше проверять)

7. Социальный статус.
8. Адрес проживания и Адрес регистрации. Лучший вариант при заполнении - указывать ул.Ленина, номер дома 999, квартира 999,
9. Заполняются данные полиса. Важными при заполнении параметрами будут - номер, который тоже должен быть валидным(используем [генератор](#)).
10. Заполняются паспортные данные. При заполнении серии и номера необходимо учитывать, что бы серия и номер не пересекались с серией и номером реально существующих людей. Например, если на регионе есть люди, серия и номер которых начинаются на 5\*\*\* и 1\*\*\*\*\*, то серия и номер для ПД создаваемого челока не должны начинаться с указанных цифр. Определить диапазон можно простым SQL запросом в таблицу v\_personstate.
11. Заполняется место работы. Местом работы указывается МО, на которую добавляется врач. Должность должна соответствовать той должности, на которую добавляется врач.
12. ИНН человека. Он тоже должен быть валидным, и высчитывается по [формуле](#). Или используется все тот же генератор.
13. Проверяем еще раз введенные данные и сохраняем человека.

Пример итогового результат заполнения формы "Человек: Добавление":

#### 4.3.1.2 Заполнение данных об образовании

После сохранения человека необходимо добавить информацию об образовании врача. Для этого на форме "Сотрудники: Добавление" заполняются:

- На вкладке Квалификационная категория, добавляется 1 квалификационная категория. Специальность выбирается с id = 13. Категория - высшая. Дату прохождения указываем 2017 год

- На вкладке Специальности по диплому. Добавляются специальность по диплому для Высшего и Среднего образования.

- Выбирается учебное заведение: примера нет, на каждом регионе будет какое то свое учебное заведение, которое зависит от справочников на регионе, как давно их обновили, от справочников на портале ФРМО и как давно их обновили.
- Год поступления/год окончания - должен попадать в период работы учебного заведения. Например, если УЗ - работало с 1995 по 2005, то врач должен учиться в период с 2001 по 2004
- Серия диплома - рандомные 4 цифры. **\*ВАЖНО\* ЦИФР ДОЛЖНО БЫТЬ - 4. НЕ 3, НЕ 5, А ИМЕННО 4**
- Номер диплома - рандомные 6(для среднего образования) и рандомные 7(для высшего образования) цифр.
- Дата выдачи диплома. Не должна превышать дату работы УЗ.
- Тип образования - Среднее образования для Среднего образования. Высшее образование - для Высшего образования.
- Специальность - Врач с кодом 15
- Квалификация - Врач общей практики.

Среднее образование	Высшее образование

- На вкладке Послевузовское образование. Создается 1 послевузовское образование и заполняются поля:
  - Выбирается учебное заведение: примера нет
  - Даты начала, окончания, получения диплома. Главное - что бы даты попадали в диапазон работы УЗ
  - Серия и номер диплома - рандомные цифры. Важно, что бы Серия была - не больше и не меньше 4 цифр. Номер диплома - должен быть 5 цифр.
  - Тип образования - аспирантура
  - Специальность - для интернатуры выбираем специальность с кодом 75. Для аспирантуры - Микробиология
  - Ученая степень - Доктор медицинских наук

Пример заполнения Послевузовского образования:

- На вкладке Курсы Переподготовки создаем запись об 1 курсе. Заполняются следующие поля:
  - a. Выбирается учебное заведение: примера нет
  - b. Год прохождения. Так же главное, что бы попадало в период действия УЗ
  - c. Количество часов - 360
  - d. Серия и номер документа. Рандомные цифры, но важно, что бы Серия была не более и не менее 3х цифр, Номер - 4 цифры.
  - e. Дата получения документа.
  - f. Специальность - выбираем специальность с кодом 3200

#### Пример заполнения Курса Переподготовки

- На вкладке Курсы Повышения Квалификации создаем 1 запись о повышении квалификации. Заполняем следующие поля:
  - g. Выбирается учебное заведение: примера нет
  - h. Цикл - указываем 1 цикл
  - i. Год прохождения. И он снова должен попадать в период действия УЗ
  - j. Количество часов. Указываем 240
  - k. Серия и Номер документа. Рандомные цифры, но ВАЖНО - Серия должна быть 3 цифры, не больше и не меньше. Номер документа должен быть не больше и не меньше 5 цифр.
  - l. Дата получения документа. Как и ранее, должен попадать в действие УЗ
  - m. Специальность - выбираем специальность с кодом 17

#### Пример заполнения Курса Повышения Квалификации

- На вкладке Сертификаты создается 1 запись о сертификате. Заполняются следующие поля:
  - Выбирается учебное заведение: примера нет
  - Серия и Номер сертификата. Рандомные цифры, НО! Сертификат должен быть - 1 цифра, Номер Сертификата - 3 цифры.
  - Специальность - выбираем специальность с кодом 36.

- Дата сдачи экзамена и Дата получения сертификата. Как и все заполняемые ранее даты должна попадать в период работы УЗ

Пример заполнения данных о Сертификате:

7. На вкладке Награды. Добавляем 1 награду, заполняем все 4 поля и не важно, какими данными.

Пример заполнения данных о Награде

Дополнительные сведения не заполняются, как и Аккредитация (Федералы убрали необходимость в аккредитации и с 2019 года она ими при выгрузке не принимается).

После добавления всех необходимых данных об образовании сохраняем Врача и переходим к заполнению следующих форм.

#### 4.3.1.3 Заполнение данных о месте работы.

При заполнении данных о месте работы Врача учитываем, что у Врача не должно быть других ставок. Заполняются следующие поля:

- Запись на начало - указываем Принят
- Прибыл - Из мед. организации данного субъекта Российской Федерации
- Номер приказа - 1к
- Дата приема - не должна совпадать ни с одной датой, указанной в данных об обучении.

Пример:

- Тип занятия должности - всегда должна быть основная должность!
- Ставка - 1

- Режим работы - работающий на постоянной работе.
- Отношение к военной службе - Военнообязанный

Пример:





## 4.4 Составление отчета о тестировании.

После создания Врача и добавления его в структуру можно приступить непосредственно к выгрузке.

Для этого через DBeaver запускаем последовательно SQL скрипты

- проверка таблицы для хранения Врача
- создание таблицы хранения Врача (опционально, если таблицы нет на регионе)
- добавление данных Врача
- проверка данных Врача. Если какие то данные отличаются/отсутствуют - исправляем и проверяем еще раз.
- **Скрипты ниже**
- комментируем данные Врача (все, кроме person)
- Запускаем скрипт create
- Переходим в логи. У каждого сервера свой адрес для доступа к логам. В самих логах необходимо найти папку fmg/дата\_отправки.log. Ищем отправленный запрос. Находим полученный синхронный ответ. Копируем message\_id, и ищем по нему асинхронный ответ. Если асинхронный ответ пришел без ошибок, то сохраняем все запросы и ответы по данному методу. **Важно:** Отправленный запрос, синхронный ответ, асинхронный ответ, асинхронный запрос, каждый должен быть в отдельном xml файле, т.е должно получиться 4 файла. **В итоге:** Должен получиться архив с файлами XML и для каждого запрос/ответа свой XML.

**Пример названия файлов для одного метода.**

 person.create	Документ XML
 person.create.sync_async	Документ XML
 person.create_async	Документ XML
 person.create_sync	Документ XML

- Если после отправки в ответе были получены ошибки, то внимательно читаем, в чем заключается ошибка и исправляем. При этом у Врача перед следующим запуском необходимо будет изменить СНИЛС и ПД(если ошибка случалась после метода person\_document.create)
- Переходим в DBeaver, комментируем person и раскомментируем common (Среднее образование)(комментируем и раскомментируем строго по порядку, т.е сначала 1 метод в 1 блоке, потом 1 метод во втором блоке и т.д)
- Повторяем пункты 6,7,8(в 8 пункте меняется каждый раз метод).

- Повторяем, пока по всем данным не будет выполнен create
- ????
- Profit! Все методы create успешно выполнены, ошибок нет и можно перейти к следующему методу - update
- Повторяем пункты 6(только будет запуска скрипта update), 7, 8
- ???
- Если ошибок не было, то переходим на метод read и снова повторяем пункты 6, 7, 8
- После выполнения всех read раскоментируем все строки и запускаем delete. При этом по person и qualification

№	Метод
1	create
2	update
3	read
4	delete

## 4.5 Массовая выгрузка.

Проведение Массовой выгрузки по шагам ничем не отличается от этапа проверки ФРМР. За исключением того, что Врачей необходимо будет создать 20+ раз.

Остальные 80+ представляют собой массовку, которая не добавляется в штатную структуру.

При создании врачей все данные, за исключением СНИЛС, ПД и Серий и Номеров документов должны быть одинаковы. Т.е все должны жить в одной квартире по одному адресу, закончить одно и тоже УЗ по одной и той же специальности, работать в одной и той же МО на одной и тоже ставке.

Если на каком либо враче данные отличаются от остальных при проверке - срочно меняем на правильные.

### 4.5.1 Проведение Массовой выгрузки.

Массовая выгрузка проводится аналогично шагам, указанным в этапе Проверка ФРМР.

Единственное отличие заключается в том, что сначала все действия проводятся над Врачами, массовка предварительно комментится в 'zzz'.

После успешного завершения всех действий над Врачами массовка раскоментируется и выполняется(Врачей при этом можно закоментировать в \*название\_метода1\* или вовсе удалить). После выполнения person.create проверяем, что у нас при отправке массовки вернулось не менее 80 успешных ответов(т.е не должно быть ошибок).

### 4.5.2 Составление отчета

Т.к при массовой выгрузке отправляется очень много данных, то ответы в xml не собираются.

Вместо этого фиксируем общее время по МСК старта(т.е когда был запущен первый метод) и общее время завершения(т.е когда все методы выполнили, получили последний ответ и больше ничего запускать не надо).

При запуске метода - фиксируем время по МСК, когда был запущен метод.



## 4.6 SQL Scripts

Список всех используемых в работе SQL Скриптов.

### 4.6.1 Проверка и подготовка Сессии.

--проверяем, есть ли ФРМР/ФРМО сессия на сервере

```
select * from FRMRIntegrationSession
```

```
select * from FRMOSession
```

----создание сессии при проверке, если ее не нашли.

```
do
$$
DECLARE FRMOSessionActionType_id BIGINT;
Error_Code VARCHAR;
Error_Message VARCHAR(4000);
begin
select fnc.Error_Code,fnc.Error_Message
into Error_Code,Error_Message
from dbo.p_FRMRIntegrationSession_ins(
FRMOSession_id := ::bigint,--подставить id найденной/созданной сессии, -- bigint
FRMRIntegrationSession_isSuccess := 2::bigint, -- bigint
-- FRMRSession_Guid := NULL, -- uniqueidentifier
--FRMRIntegrationSession_ErrorText: = ", -- varchar(200)
lpu_id := ::bigint,--подставить lpu_id, -- bigint
lpu_oid := ::varchar, --подставить OID , -- varchar(256)
FRMRIntegrationSession_isUsed := 2::bigint, -- bigint
FRMOSessionActionType_id:= 10053::bigint, -- bigint
pmUser_id := 1::bigint) fnc;
if (Error_Code is not null)
then
RAISE USING ERRCODE = error_code,MESSAGE =Error_Message;
end if;
end;
$$;
```

### 4.6.2 Запуск методов ФРМР:

--выполнение create

```
do
$$
DECLARE
Error_Code VARCHAR;
Error_Message VARCHAR(4000);
begin
select fnc.Error_Code,fnc.Error_Message
into Error_Code,Error_Message
from dbo.p_FRMRIntegrationSession_upd (FRMRIntegrationSession_id := ::bigint, -- bigint
FRMOSession_id := ::bigint, -- bigint
lpu_id := ::bigint,
lpu_oid:=::varchar,-- varchar
FRMRIntegrationSession_isUsed:=NULL,
FRMOSessionActionType_id:=10054::bigint,
pmUser_id := 1::bigint) fnc;
if (Error_Code is not null)
then
RAISE USING ERRCODE = error_code,MESSAGE =Error_Message;
end if;
```

```

end;
$$;

--update
do
$$
DECLARE
Error_Code VARCHAR;
Error_Message VARCHAR(4000);
begin
select fnc.Error_Code,fnc.Error_Message
into Error_Code,Error_Message
from dbo.p_FRMRIntegrationSession_upd (FRMRIntegrationSession_id := ::bigint, -- bigint
FRMOSession_id := ::bigint, -- bigint
lpu_id := ::bigint,
Lpu_oid:=::varchar,-- varchar
FRMRIntegrationSession_isUsed:=NULL,
FRMOSessionActionType_id:=10050::bigint,
pmUser_id := 1::bigint) fnc;
if (Error_Code is not null)
then
RAISE USING ERRCODE = error_code,MESSAGE =Error_Message;
end if;
end;
$$;

--read
do
$$
DECLARE
Error_Code VARCHAR;
Error_Message VARCHAR(4000);
begin
select fnc.Error_Code,fnc.Error_Message
into Error_Code,Error_Message
from dbo.p_FRMRIntegrationSession_upd (FRMRIntegrationSession_id := ::bigint, -- bigint
FRMOSession_id := ::bigint, -- bigint
lpu_id := ::bigint,
Lpu_oid:=::varchar,-- varchar
FRMRIntegrationSession_isUsed:=NULL,
FRMOSessionActionType_id:=10053::bigint,
pmUser_id := 1::bigint) fnc;
if (Error_Code is not null)
then
RAISE USING ERRCODE = error_code,MESSAGE =Error_Message;
end if;
end;
$$;

--delete
do
$$
DECLARE
Error_Code VARCHAR;
Error_Message VARCHAR(4000);
begin
select fnc.Error_Code,fnc.Error_Message
into Error_Code,Error_Message
from dbo.p_FRMRIntegrationSession_upd (FRMRIntegrationSession_id := ::bigint, -- bigint
FRMOSession_id := ::bigint, -- bigint
lpu_id := ::bigint,
Lpu_oid:=::varchar,-- varchar

```

```

FRMRIntegrationSession_isUsed:=NULL,
FRMOSessionActionType_id:=10052::bigint,
pmUser_id := 1::bigint) fnc;
if (Error_Code is not null)
then
RAISE USING ERRCODE = error_code,MESSAGE =Error_Message;
end if;
end;
$$;

```

#### 4.6.3 Работа с данными

```

--Создание таблицы FRMRUpload. в ней хранится Врач
CREATE table if not exists tmp.FRMRUpload (
FRMRUpload_id bigint NOT NULL GENERATED BY DEFAULT AS IDENTITY,
FRMRUpload_Method varchar (60) ,
Snils_id varchar (60) NULL,
FRMRUpload_insDT timestamp NULL,
FRMRUpload_updDT timestamp null,
CONSTRAINT pk_FRMRUpload_id PRIMARY KEY (FRMRUpload_id)
);

```

--поиск врачей и вывод их снилсов

```

SELECT ""||Person_Snils||""||','||Person_SecNameR,* FROM v_personstate
WHERE Person_SurNameR='TECT' AND Person_FirNameR='ФРМР' /*AND
Person_SecName NOT LIKE '%тест'*/ ORDER BY Person_id
limit 1;

```

--добавление методов по врачам в таблицу. добавить метод в таблицу,  
закомментировать строку и раскомментировать следующую

```

INSERT INTO tmp.FRMRUpload (FRMRUpload_Method, Snils_id)
select --'zzz'
'person',
--'document',
--'common',
--'address',
--'prof',
--'postgraduate',
--'ext',
--'cert',
--'accreditation',
--'qualification',
--'card',
--'nomination',
v_PersonState.Person_Snils
/*DISTINCT,
v_PersonState.Person_FirName,v_PersonState.Person_SecName,
v_PersonState.Person_SurName,v_PersonState.Sex_id,
v_PersonState.Person_BirthDay,v_PersonState.Person_Inn,
'1' as citizenShipId,'171' as oksmId, --Страна
MilitaryRelation_id,v_PersonState.Person_Phone*/
from dbo.v_PersonState
WHERE Person_Snils IN
(–подставить снилс Врача)
ORDER BY Person_id
limit 105;

```

```

--вывод добавленных методов
select * from tmp.FRMRUpload
SELECT * FROM tmp.FRMRUpload limit 10000;

--удаление всех методов, которые не называются 'zzz'
delete from tmp.FRMRUpload
where FRMRUpload_Method != 'zzz';

-----
--update для игнорирования методом.
--3
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='card1' WHERE
FRMRUpload_Method='card';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='qualification1' WHERE
FRMRUpload_Method='qualification';
--2.5
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='ext1' WHERE
FRMRUpload_Method='ext';
--
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='postgraduate1' WHERE
FRMRUpload_Method='postgraduate';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='accreditation1' WHERE
FRMRUpload_Method='accreditation';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='nomination1' WHERE
FRMRUpload_Method='nomination';
--2
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='prof1' WHERE
FRMRUpload_Method='prof';
--1
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='address1' WHERE
FRMRUpload_Method='address';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='document1' WHERE
FRMRUpload_Method='document';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='common1' WHERE
FRMRUpload_Method='common';
--0
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='person1' WHERE
FRMRUpload_Method='person';

update tmp.FRMRUpload set FRMRUpload_Method='zzz' where
FRMRUpload_Method='person';

--DELETE FROM tmp.FRMRUpload WHERE FRMRUpload_Method!='zzz';
-----

--update что бы метод увидел
--3
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='card' WHERE
FRMRUpload_Method='card1';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='qualification' WHERE
FRMRUpload_Method='qualification1';
--2.5
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='cert' WHERE
FRMRUpload_Method='cert1';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='ext' WHERE
FRMRUpload_Method='ext1';
--

```

```

UPDATE tmp.FRMRUpload SET FRMRUpload_Method='postgraduate' WHERE
FRMRUpload_Method='postgraduate1';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='accreditation' WHERE
FRMRUpload_Method='accreditation1';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='nomination' WHERE
FRMRUpload_Method='nomination1';
--2
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='prof' WHERE
FRMRUpload_Method='prof1';
--1
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='address' WHERE
FRMRUpload_Method='address1';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='document' WHERE
FRMRUpload_Method='document1';
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='common' WHERE
FRMRUpload_Method='common1';
--0
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='person' WHERE
FRMRUpload_Method='person1';

--массовка

update tmp.FRMRUpload set FRMRUpload_Method='person' where
FRMRUpload_Method='zzz';

```

**\*ВАЖНО. Запускать от меньшего к большему. Т.е начинаем с нуля и поднимаемся к 3\***

#### 4.6.4 Проверка данных Врача

-----ищем и выводим данные для метода person

```

select distinct
Person_Snils,
Person_FirName,
Person_SecName,
Person_SurName,
Sex_id,
Person_BirthDay,
Person_Inn,
'1' as citizenShipId,
'171' as oksmId, --Страна
'2' AS MilitaryRelation_id,
Person_Phone
from dbo.v_personstate
where Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='person')
ORDER BY Person_SecName;

```

-----ищем и выводим данные для метода document

```

SELECT
v_PersonState.Person_Snils,
v_PersonDocument.Document_Ser,
v_PersonDocument.Document_Num,
v_PersonDocument.PersonDocument_updDT,
v_Document.Document_begDate,
v_Document.Document_updDT,
v_Org.Org_Nick,
v_DocumentType.Frmr_id
FROM persis.MedWorker v_MedStaffFactCache
LEFT JOIN LATERAL(

```

```

SELECT Ipu_id,MilitaryRelation_id FROM
persis.WorkPlace
INNER JOIN persis.v_Staff staff ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
limit 1) AS staff on true
INNER JOIN lateral (select * from dbo.v_PersonState
where v_PersonState.Person_id = v_MedStaffFactCache.Person_id limit 1) v_PersonState on
true
INNER JOIN dbo.v_PersonDocument ON v_PersonDocument.Person_id =
v_PersonState.Person_id
INNER JOIN dbo.v_Document ON v_Document.Document_id =
v_PersonDocument.Document_id
INNER JOIN dbo.v_DocumentType ON v_DocumentType.DocumentType_id =
v_Document.DocumentType_id
left JOIN dbo.OrgDep ON OrgDep.OrgDep_id = v_Document.OrgDep_id
left JOIN dbo.v_Org ON v_Org.Org_id = OrgDep.Org_id
where v_PersonState.Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='document')
AND v_Document.Document_begDate IS NOT NULL
AND Org_Nick IS NOT NULL
AND Person_Snils is not NULL
ORDER BY Person_Snils;

```

-----выводим данные для метода common

```

SELECT
Person_Snils,
coalesce (EducationInstitution.name,OtherEducationalInstitution) AS institution,
DiplomaSeries AS docSerial,--Серия аттестата docSerial
DiplomaNumber AS docNumber,--Номер аттестата docNumber
SpecialityDiploma.insDT AS docDate--Дата выдачи аттестата docDate
FROM persis.MedWorker v_MedStaffFactCache
LEFT JOIN LATERAL(
SELECT MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace
INNER JOIN persis.v_Staff staff ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
limit 1) AS staffon on true
INNER JOIN lateral (select * from dbo.v_PersonState
where v_PersonState.Person_id = v_MedStaffFactCache.Person_id limit 1) v_PersonState on
true
--LEFT JOIN dbo.v_MedPersonal ON v_MedPersonal.Person_id = v_PersonState.Person_id
--LEFT JOIN dbo.v_MedPersonal ON
LEFT JOIN persis.SpecialityDiploma ON SpecialityDiploma.MedWorker_id =
v\_MedStaffFactCache.id AND EducationType_id=2
LEFT JOIN persis.EducationInstitution ON EducationInstitution.id =
SpecialityDiploma.EducationInstitution_id
WHERE
v_PersonState.Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='common')
AND v\_MedStaffFactCache.id IS NOT NULL
AND DiplomaSeries IS NOT NULL
AND DiplomaNumber IS NOT NULL
AND SpecialityDiploma.insDT IS NOT NULL
AND Person_Snils is not null
ORDER BY Person_Snils;

```

-----address

```

SELECT
v_PersonState.Person_Snils,

```

```

ua.Address_updDT as regDate,
CASE WHEN ua.Address_id=v_PersonState.UAddress_id THEN 1 ELSE 2 end as
addressTypeId,
ua.Address_id,
coalesce (KLTown_AOID,KLCity_AOID) AS aoidArea,--Идентификатор населенного пункта
по ФИАС aoidArea
KLStreet_AOID as aoidStreet,--Идентификатор улицы по ФИАС aoidStreet
ua.KLRgn_id as region,--Код региона region(id)
coalesce(KLTown_Name,KLCity_Name) AS areaName,--Наименование населенного пункта
areaName
coalesce(v_KLTown.KLSocr_Nick,v_KLCity.KLSocr_Nick) AS prefixArea,--Префикс
населенного пункта prefixArea
v_KLStreet.KLStreet_Name AS streetName,--Наименование улицы streetName
v_KLSocr.KLSocr_Nick,--Префикс улицы prefixStreet
dbo.ConcatPunkt('д',CAST(Address_House AS VARCHAR(30)),'',3,0),
dbo.ConcatPunkt('к',CAST(Address_Corpus AS VARCHAR(30)),'',3,0),'',0,null) AS house,--
Номер дома, корпуса, строения в формате дХ кХ сХ house
Address_Flat AS flat--Номер квартиры flat
FROM persis.MedWorker v_MedStaffFactCache
LEFT JOIN LATERAL(
SELECT Ipu_id,MilitaryRelation_id
FROM persis.WorkPlace
INNER JOIN persis.v_Staff staff ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
limit 1) AS staff on true
INNER JOIN lateral (select * from dbo.v_PersonState
where v_PersonState.Person_id = v_MedStaffFactCache.Person_id limit 1) v_PersonState on
true
left JOIN dbo.v_Address ua ON ua.Address_id=v_PersonState.UAddress_id OR
ua.Address_id=v_PersonState.PAddress_id
LEFT JOIN dbo.v_KLTown on v_KLTown.KLTown_id = ua.KLTown_id
LEFT JOIN dbo.v_KLCity on v_KLCity.KLCity_id = ua.KLCity_id
LEFT JOIN dbo.v_KLStreet ON v_KLStreet.KLStreet_id = ua.KLStreet_id
LEFT join dbo.v_KLSocr ON v_KLSocr.KLSocr_id = v_KLStreet.KLSocr_id
WHERE Person_Snils IS NOT NULL
AND ua.Address_updDT IS NOT NULL
AND KLStreet_AOID IS NOT null
AND Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='address')
limit 50;

```

-----prof

```

SELECT
Person_Snils,
CASE WHEN EducationInstitution_id IS null THEN 2 ELSE 0 end as educPlace,
EducationType.Frmr_id AS educationTypeId , --уровень образования
CASE WHEN IsSpecSet=1 THEN 'true' ELSE 'false' end as isTargeted,
CASE WHEN IsSpecSet=1 THEN FRMPTerritories_id ELSE null end as targetedRegionId ,
YearOfAdmission AS beginYear, --год поступления
YearOfGraduation,
DocumentRecieveDate AS docDate, --дата выдачи
DiplomaSpeciality.frmr_id AS specId, --специальность
CASE WHEN EducationInstitution_id IS null THEN OtherEducationInstitution ELSE null end
AS foreignInstitution,--Образовательная организация institution
DiplomaSeries AS docSerial,--Серия аттестата docSerial
DiplomaNumber AS docNumber,--Номер аттестата docNumber
SpecialityDiploma.insDT AS docDate,--Дата выдачи аттестата docDate
--EducationInstitution.code AS institutionId ,
Qualification_id as qualificationId

```

```

FROM persis.MedWorker v_MedStaffFactCache
LEFT JOIN LATERAL(
SELECT MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace
INNER JOIN persis.v_Staff staff ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
limit 1
) AS staff on true
INNER JOIN lateral (select * from dbo.v_PersonState
where v_PersonState.Person_id = v_MedStaffFactCache.Person_id limit 1) v_PersonState on
true
LEFT JOIN persis.SpecialityDiploma ON SpecialityDiploma.MedWorker_id =
v\_MedStaffFactCache.id AND EducationType_id IN (3,1,6,7)
INNER JOIN persis.DiplomaSpeciality ON DiplomaSpeciality.id =
SpecialityDiploma.DiplomaSpeciality_id
INNER JOIN persis.EducationType ON EducationType.id =
SpecialityDiploma.EducationType_id
left JOIN persis.EducationInstitution ON EducationInstitution.id =
SpecialityDiploma.EducationInstitution_id
WHERE v\_MedStaffFactCache.id IS NOT NULL
AND DiplomaSeries IS NOT NULL
AND DiplomaNumber IS NOT NULL
AND SpecialityDiploma.insDT IS NOT NULL
AND
Person_Snils IS NOT NULL
AND Person_Snils IN (-указываем СНИЛС выгружаемого врача)
limit 50;

```

-----postgraduate

```

SELECT Person_Snils, CASE WHEN PostgraduateEducation.EducationInstitution_id IS null
THEN 2 ELSE 0 end as educPlace,
EducationInstitution.code AS institutionId , PostgraduateEducationType_id AS
educationStageld,
CAST(YEAR(startDate) AS int) AS beginYear, --год поступления
CASE WHEN endDate IS null THEN 'true' ELSE 'false' end AS isStudying,
PostgraduateEducation.DiplomaSeries AS docSerial,--Серия аттестата docSerial
PostgraduateEducation.DiplomaNumber AS docNumber,--Номер аттестата docNumber
graduationDate AS docDate, --дата выдачи
CASE WHEN PostgraduateEducationType_id IN (3,4,5) AND endDate IS Not null THEN
AcademicMedicalDegree.Frmr_id
ELSE null end as academicDegreeld,
CASE WHEN PostgraduateEducationType_id IN (1,2) THEN Speciality.code ELSE
PostgraduateEducation.SpecialityAspirant_id end as allSpecId,
coalesce(PostgraduateEducation.ScienceBranch_id,
SpecialityAspirantScienceLink.ScienceBranch_id) AS scienceBranchId,
CASE WHEN PostgraduateEducationType_id IN (1,2) THEN Speciality.code ELSE null end as
specId,
CASE WHEN PostgraduateEducationType_id IN (3,4) THEN
PostgraduateEducation.SpecialityAspirant_id ELSE null end as doctSpecId,
--'408' as doctSpecId,
CASE WHEN PostgraduateEducationType_id not IN (3,4,5) AND
coalesce(PostgraduateEducation.IsSpecSet,44)=1
THEN 'true' WHEN PostgraduateEducationType_id not IN (3,4,5) AND
coalesce(PostgraduateEducation.IsSpecSet,44)=0
THEN 'false' ELSE null end as isTargeted, CASE WHEN
coalesce(PostgraduateEducation.IsSpecSet,44)=1
THEN FRMPTerritories_id ELSE null end as targetedRegionId
FROM persis.MedWorker v_MedStaffFactCache

```



```

LEFT JOIN LATERAL ( SELECT MedWorker_id,staff.Lpu_id FROM persis.WorkPlace
INNER JOIN persis.v_Staff staff ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id limit 1) AS staff on true
INNER JOIN dbo.v_PersonState ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
inner join persis.PostgraduateEducation ON PostgraduateEducation.MedWorker_id =
v\_MedStaffFactCache.id
INNER JOIN persis.AcademicMedicalDegree ON AcademicMedicalDegree.id =
PostgraduateEducation.AcademicMedicalDegree_id
INNER JOIN persis.Speciality ON Speciality.id = PostgraduateEducation.Speciality_id
INNER JOIN persis.EducationInstitution ON EducationInstitution.id =
PostgraduateEducation.EducationInstitution_id
left join nsi.SpecialityAspirantScienceLink ON
SpecialityAspirantScienceLink.SpecialityAspirant_id =
PostgraduateEducation.SpecialityAspirant_id
WHERE Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='postgraduate')
and v\_MedStaffFactCache.id IS NOT NULL
AND PostgraduateEducation.DiplomaSeries IS NOT NULL
AND PostgraduateEducation.DiplomaNumber IS NOT NULL
AND coalesce(Speciality.code,PostgraduateEducation.SpecialityAspirant_id) IS NOT null
AND Person_Snils IS NOT NULL
limit 50;

```

-----ext

```

SELECT
Person_Snils AS snils,
EducationInstitution.code AS institutionId,
'1' as profEducationKindId,
QualificationImprovementCourse.hourscount AS hoursCount,
DocumentSeries AS docSerial,
DocumentNumber AS docNumber,
QualificationImprovementCourse.DocumentRecieveDate AS docDate,
--Speciality.code,
DiplomaSpeciality.frmr_id AS code,
CASE WHEN position('/') in Speciality.name -1 > 0 THEN
SUBSTRING(Speciality.name,0,position('/')-1) ELSE Speciality.name END
as name
FROM persis.MedWorker v_MedStaffFactCache
LEFT JOIN LATERAL(
SELECT MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace
INNER JOIN persis.v_Staff staff ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id limit 1
) AS staff on true
INNER JOIN dbo.v_PersonState ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
LEFT JOIN persis.SpecialityDiploma ON SpecialityDiploma.MedWorker_id =
v\_MedStaffFactCache.id AND EducationType_id IN (3,1,6,7)
INNER JOIN persis.DiplomaSpeciality ON DiplomaSpeciality.id =
SpecialityDiploma.DiplomaSpeciality_id
INNER JOIN persis.EducationType ON EducationType.id =
SpecialityDiploma.EducationType_id
INNER JOIN persis.QualificationImprovementCourse AS QualificationImprovementCourse ON
QualificationImprovementCourse.MedWorker_id = v\_MedStaffFactCache.id
LEFT join persis.Speciality AS Speciality ON Speciality.id =
QualificationImprovementCourse.Speciality_id
INNER JOIN persis.EducationInstitution ON EducationInstitution.id =
QualificationImprovementCourse.EducationInstitution_id

```

```

WHERE
v_MedStaffFactCache.id IS NOT NULL
and DocumentSeries IS NOT NULL
and DocumentNumber IS NOT NULL
AND Person_Snils IS NOT NULL
AND QualificationImprovementCourse.DocumentRecieveDate IS NOT NULL
and Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='ext')
UNION
SELECT
Person_Snils AS snils,
EducationInstitution.code AS institutionId,
'3' as profEducationKindId,
RetrainingCourse.hourscount AS hoursCount,
DocumentSeries AS docSerial,
DocumentNumber AS docNumber,
RetrainingCourse.DocumentRecieveDate AS docDate,
--Speciality.code,
DiplomaSpeciality.frmr_id AS code,
CASE WHEN position('/') in Speciality.name)-1>0 THEN
SUBSTRING(Speciality.name,0,position('/')-1) ELSE Speciality.name END
as name
FROM persis.MedWorker v_MedStaffFactCache
LEFT JOIN LATERAL(
SELECT MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace
INNER JOIN persis.v_Staff staff ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
limit 1
) AS staff on true
INNER JOIN dbo.v_PersonState ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
LEFT JOIN persis.SpecialityDiploma ON SpecialityDiploma.MedWorker_id =
v\_MedStaffFactCache.id AND EducationType_id IN (3,1,6,7)
INNER JOIN persis.DiplomaSpeciality ON DiplomaSpeciality.id =
SpecialityDiploma.DiplomaSpeciality_id
INNER JOIN persis.EducationType ON EducationType.id =
SpecialityDiploma.EducationType_id
INNER JOIN persis.RetraingCourse ON RetraingCourse.MedWorker_id =
v\_MedStaffFactCache.id
LEFT join persis.Speciality ON Speciality.id = RetraingCourse.Speciality_id
INNER JOIN persis.EducationInstitution ON EducationInstitution.id =
RetraingCourse.EducationInstitution_id
WHERE v\_MedStaffFactCache.id IS NOT NULL
AND DocumentSeries IS NOT NULL
AND DocumentNumber IS NOT null
AND Person_Snils IS NOT NULL
AND RetraingCourse.DocumentRecieveDate IS NOT NULL
and Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='ext')
limit 50;

```

-----cert

```

SELECT
Certificate.updDT,
Person_Snils,
EducationInstitution.code AS institutionId,
Speciality.code AS specId,
Certificate_ExamDate as examDate,

```

```

CertificateReceipDate AS passDate,
CertificateSeries as certSerial,
CertificateNumber AS certNumber
FROM persis.MedWorker v_MedStaffFactCache
LEFT JOIN LATERAL
(SELECT MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace
INNER JOIN persis.v_Staff staff ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
limit 1
) AS staff on true
INNER JOIN dbo.v_PersonState ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
INNER JOIN persis.Certificate ON Certificate.MedWorker_id = v\_MedStaffFactCache.id
LEFT join persis.Speciality AS Speciality ON Speciality.id = Certificate.Speciality_id
INNER JOIN persis.EducationInstitution ON EducationInstitution.id =
Certificate.EducationInstitution_id
WHERE
--v_PersonState.Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='cert')
Person_Snils IN (--указать СИЛС выгружаемых врачей)
-- and CertificateNumber IS NOT NULL
--AND Certificate_ExamDate IS NOT NULL
--AND Person_Snils IS NOT NULL
--AND EducationInstitution_id IS NOT NULL
--AND Speciality.code IS NOT NULL
limit 50;

-----qualification

SELECT
Person_Snils AS snils,
Category.code AS qualifyCategoryId,
AssigmentDate AS beginDate,
Speciality.code AS specId
FROM persis.MedWorker v_MedStaffFactCache
LEFT JOIN LATERAL
(SELECT MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace
INNER JOIN persis.v_Staff staff ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
limit 1
) AS staff on true
INNER JOIN dbo.v_PersonState ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
INNER JOIN persis.QualificationCategory ON QualificationCategory.MedWorker_id =
v\_MedStaffFactCache.id
INNER JOIN persis.Category ON Category.id = QualificationCategory.Category_id
INNER JOIN persis.Speciality ON Speciality.id = QualificationCategory.Speciality_id
WHERE
Person_Snils IS NOT NULL AND Category.code IS NOT NULL AND Speciality.code IS NOT
NULL
AND AssigmentDate IS NOT NULL
and Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='qualification')
limit 50;

-----card

SELECT
MedStaffFact_id,
Person_Snils AS snils,

```

```

PassportToken_tid AS organizationId,
'false' AS targeted,
LpuUnit_FRMOUnitID AS nrPmuDepartId,
LpuSection_FRMOBuildingOid AS nrPmuDepartHospitalSubdivisionId,
Frmr_id AS postId,
CASE WHEN PostOccupationType_id=3 THEN null ELSE CAST(MedStaffFact_Stavka AS
decimal(16,4)) END AS rate,
v_MedStaffFactCache.WorkData_begDate beginDate,
v_MedStaffFactCache.WorkData_endDate endDate,
CASE WHEN DismissalReasonType_id IN (2,3) THEN 2 WHEN DismissalReasonType_id NOT
IN (2,3) THEN 1 ELSE null END AS endTypeld,
DismissalReasonType_id AS fireReasonId,
PostOccupationType.code AS positionTypeld
FROM persis.v_MedWorker
INNER JOIN dbo.v_MedStaffFactCache ON v_MedStaffFactCache.Person_id =
v_MedWorker.Person_id
INNER JOIN fed.PassportToken ON PassportToken.Lpu_id = v_MedStaffFactCache.Lpu_id
--отделение
INNER JOIN dbo.v_LpuSection ON v_LpuSection.LpuSection_id =
v_MedStaffFactCache.LpuSection_id
--подразделение
INNER JOIN v_lpuunit ON v_LpuUnit.LpuUnit_id = v_MedStaffFactCache.LpuUnit_id
inner join persis.v_Post Post on v_MedStaffFactCache.Post_id = Post.id
INNER JOIN persis.FRMPPost ON Post.frmEntry_id= FRMPPost.id
INNER JOIN persis.PostOccupationType ON PostOccupationType.id =
v_MedStaffFactCache.PostOccupationType_id
where LpuUnit_FRMOUnitID IS NOT NULL
and Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='card')
AND CAST(MedStaffFact_Stavka AS decimal(16,4)) != 0.0000
AND CAST(MedStaffFact_Stavka AS decimal(16,4)) != 0.0000
limit 50;

-----nomination

SELECT
name AS nomination,
Person_Snils AS Snils,
FRMPNomination_id AS nominationId,
number AS nomNumber,
date AS nomDate
FROM persis.MedWorker v_MedStaffFactCache
LEFT JOIN LATERAL
(SELECT MedWorker_id,staff.Lpu_id FROM persis.WorkPlace
INNER JOIN persis.v_Staff staff ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
limit 1
) AS staff on true
INNER JOIN dbo.v_PersonState ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
INNER JOIN persis.Reward ON Reward.MedWorker_id = staff.MedWorker_id
WHERE
Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='nomination')
and name IS NOT NULL
AND Person_Snils IS NOT NULL
AND FRMPNomination_id IS NOT NULL
AND number IS NOT NULL
AND date IS NOT null
limit 50;

```

## 5 ФРМО/ФРМР. Инструкция по внедрению.

### 5.1 Порядок подключения ИС к Системе.

Для обеспечения обмена данными между Системой и ИС посредством интеграционных профилей необходимо обеспечить подключение ИС к версиям Системы, расположенным на тестовой площадке ЕГИСЗ и основной площадке ЕГИСЗ, и пройти необходимые проверки (контрольные примеры) в соответствии с порядком, определенным программой и методикой испытаний ИС с Системой.

Версия Системы, расположенная на тестовой площадке ЕГИСЗ, находится по адресу: <https://nr-test.egisz.rosminzdrav.ru>.

Версия Системы, расположенная на основной площадке ЕГИСЗ, находится по адресу <https://nr.egisz.rosminzdrav.ru>.

Для получения возможности работы с тестовой средой Системы через web-интерфейс с целью проверки результатов работы по средствам интеграционного взаимодействия, сотруднику, ответственному за отладку интеграционного взаимодействия, предоставляется возможность получить доступ к Системе.

Необходимость получения доступа к тестовой среде Системы, а также присваиваемую пользователю роль, для разработчиков ИС определяет лицо от ответственной организации, эксплуатирующей информационную систему.

Для получения доступа в тестовую версию Системы учетная запись на едином портале государственных услуг не нужна. Авторизация происходит с помощью логина и пароля, направленного от СТП ЕГИСЗ в ответ на заявку.

Разработчики ИС, ответственные за отладку интеграционного взаимодействия, не допускаются на промышленную среду Системы. Проверка результатов работы интеграционного взаимодействия на промышленной среде Системы осуществляется ответственными сотрудниками МО и/или ответственной организацией.

Для получения возможности работы с тестовой версией Системы через web-интерфейс лицо от ответственной организации, эксплуатирующей информационную систему, направляет соответствующую заявку в формате Word, а также сканированную копию с подписью ответственного лица и печатью организации на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)) согласно форме, приведенной в Приложении 1 документа, расположенного по адресу <https://portal.egisz.rosminzdrav.ru/materials/481> для доступа в ФРМО, и Приложении 1 документа, расположенного по адресу <https://portal.egisz.rosminzdrav.ru/materials/483> для ФРМР.

Если в субъекте РФ для работы в Системе необходимо подключить более одной ИС, процедура, описанная ниже, выполняется для каждой информационной системы.

### 5.2 Подключение к тестовой версии сервисов ФРМР/ФРМО.

- **Сгенерировать самоподписанную электронную подпись(эксплуатация), зарегистрировать ИС в тестовой версии ИПС.**

Для этого ответственная организация направляет заявку на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)) согласно формам, приведенным в п.1.1 документа «Методические материалы по подключению к ИПС», размещенным на портале ЕГИСЗ по адресу: <http://portal.egisz.rosminzdrav.ru/materials/11>. При заполнении заявки необходимо указать тестовую среду. Заявки в обязательном порядке подаются в 2 форматах – DOCX и PDF, при этом заявка в формате PDF должна содержать подпись и расшифровку подписи ответственного лица, подтвержденную печатью организации. К заявке нужно приложить сертификат в формате сег. В течение 2 (двух) рабочих дней после получения заявки СТП ЕГИСЗ регистрирует ИС в тестовой версии ИПС и предоставляет идентификатор информационной системы в ИПС.

2. **Опубликовать тестовую версию web-службы в тестовой версии ИПС.**

Для этого ответственная организация направляет заявку на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)) согласно формам, приведенным в п.1.3 документа «Методические материалы по подключению к ИПС», размещенным на портале ЕГИСЗ по адресу: <http://portal.egisz.rosminzdrav.ru/materials/11>. В столбце "Прямой адрес веб-службы" нужно указать адрес колбэка, который предоставляет эксплуатация. Версия веб службы - SOAP 1.1. Тип веб-службы - асинхронный. При заполнении заявки необходимо указать тестовую среду. Для ФРМО и ФРМР нужно формировать отдельные заявки, но с одинаковым колбэком. Заявки в обязательном порядке подаются в 2 форматах – DOCX и PDF, при этом заявка в формате PDF должна содержать подпись и расшифровку подписи ответственного лица, подтвержденную печатью организации.

**Необходимо помнить, что колбэк для сервисов ФРМО и ФРМР используется один. Как в тестовой версии, так и в продуктивной. Инструкции по разворачиванию сервисов для эксплуатации находятся здесь: [ФРМО](#), [ФРМР](#).**

### **3. Зарегистрировать ИС в тестовой версии Системы.**

Для этого ответственная организация направляет заявку на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)) в соответствии с формой, приведенной в Приложении 2 документа, расположенного по адресу <https://portal.egisz.rosminzdrav.ru/materials/481> для доступа в ФРМО, и Приложении 2 документа, расположенного по адресу <https://portal.egisz.rosminzdrav.ru/materials/483> для ФРМР. Заявки в обязательном порядке подаются в 2 форматах – DOCX и PDF, при этом заявка в формате PDF должна содержать подпись и расшифровку подписи ответственного лица, подтвержденную печатью организации.

**4. В течение 2 (двух) рабочих дней после получения заявки СТП ЕГИСЗ регистрирует ИС в тестовой версии Системы и предоставляет подтверждение о регистрации ИС в ответ на заявку.**

### **5.3 Отчет о тестировании:**

**1. В течение 30 (тридцати) рабочих дней со дня выделения идентификатора ИС необходимо провести тестирование при взаимодействии с тестовой версией Системы и направить отчет о результатах тестирования, а также протокол приемочных испытаний на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)), в соответствии с формой, приведенной в Приложении 3 документа, расположенного по адресу <https://portal.egisz.rosminzdrav.ru/materials/481> для доступа в ФРМО, и Приложении 3 документа, расположенного по адресу <https://portal.egisz.rosminzdrav.ru/materials/483> для ФРМР.**

Пример приложения к отчёту (запросы отправленные и полученные при взаимодействии): [Пример результатов тестирования ФРМО и ФРМР.zip](#)

Отчет и протокол в обязательном порядке подаются в 2 форматах – DOCX и PDF, при этом документы в формате PDF должны содержать подпись и расшифровку подписи ответственного лица, подтвержденную печатью организации. В рамках тестирования должны быть пройдены все проверки в соответствии с программой и методикой испытаний интеграции ИС с Системой. Результатом тестирования является успешное выполнение всех проверок. В случае необходимости консультирования по техническим вопросам, сотрудник ответственной организации направляет описание проблемы в виде описания последовательности произведенных действий с указанием ссылок обращений и с приложением файлов запросов и ответов сервиса ИС на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)). В течение 3 (трех) рабочих дней после получения запроса, СТП ЕГИСЗ предоставляет ответ или направляет запрос на предоставление дополнительных сведений необходимых для выявления причин проблемы.

**2. СТП ЕГИСЗ в ответном письме направляет заключение об успешном прохождении тестирования или о необходимости доработки ИС и проведения повторного тестирования. Заключение предоставляется в течение 4 (четырёх) рабочих дней с момента получения отчета о проведении тестирования.**

**3. В случае успешного прохождения тестирования интеграционного взаимодействия СТП ЕГИСЗ в ответном письме уведомляет ИС о необходимости проведения массовой загрузки/выгрузки данных в тестовую версию Системы.**

#### **5.4 Массовая выгрузка:**

Для прохождения этапа массовой выгрузки необходимо, после успешного проведения отчета о тестировании по ФРМО и ФРМР, отправить заявку на проведение массовой выгрузки в соответствии с формой, приведенной в Приложении 4 документа, расположенного по адресу <https://portal.egisz.rosminzdrav.ru/materials/481> для доступа в ФРМО, и Приложении 4 документа, расположенного по адресу <https://portal.egisz.rosminzdrav.ru/materials/483> для ФРМР.

**1. Согласно плановой дате проведения массовой загрузки/выгрузки ИС осуществляет массовую загрузку/выгрузку данных в тестовую версию Системы, предварительно направив уведомление на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)).**

Минимальное количество запросов на каждую доступную подсистему и каждый из ее методов при массовой загрузке должно составлять 20.

**2. По результату анализа журнала аудита интеграционных профилей СТП ЕГИСЗ в ответном письме направляет заключение об успешности проведения массовой загрузки/выгрузки данных и о возможности подключения к промышленной версии Системы или необходимости доработки ИС и проведения повторного тестирования и массовой загрузки/выгрузки данных. Массовая загрузка/выгрузка считается успешно выполненной, если число корректных запросов достигает 100 %. Если при первой попытке массовой загрузки/выгрузки не был получен необходимый результат, достижение 100 %-ой корректности массовой загрузки/выгрузки осуществляется в несколько подходов:**

- если количество корректных запросов составило 95 % и более, то необходимо повторить массовую загрузку/выгрузку только для тех подсистем, в методах которых были выявлены ошибки. Повторная загрузка/выгрузка выполняется для всех методов подсистемы, а не только для ошибочных.

- если количество корректных запросов составило менее 95 %, то необходимо повторить массовую загрузку/выгрузку в полном объеме.

**Подробная инструкция по составлению отчета о тестировании и проведению массовой выгрузки по ФРМО: [ФРМО. Отчет о тестировании и массовая выгрузка.](#)**

**Подробная инструкция по составлению отчета о тестировании и проведению массовой выгрузки по ФРМР: [ФРМР. Отчет о тестировании и массовая выгрузка.](#)**  
**[Подробная инструкция.](#)**

Заключение предоставляется в течении 4 (четырёх) рабочих дней с момента получения уведомления от ИС об осуществлении массовой загрузки/выгрузки данных в тестовую версию Системы.

#### **5.5 Подключение к промышленной версии:**

- Получить квалифицированную электронную подпись в аккредитованном УЦ, зарегистрировать ИС в рабочей версии ИПС.

Для этого сотрудник ответственной организации направляет заявку на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)) согласно формам, приведенным в п.1.2 документа «Методические материалы по подключению к ИПС», размещенным на портале ЕГИСЗ по адресу: <http://portal.egisz.rosminzdrav.ru/materials/11>. При заполнении заявки необходимо указать промышленную среду. Заявки в обязательном порядке подаются в 2 форматах – DOCX и PDF, при этом заявка в формате PDF должна содержать подпись и расшифровку подписи ответственного лица, подтвержденную печатью организации. Одновременно сканированную копию заявки направить в Минздрав России, по адресу: [di@rosminzdrav.ru](mailto:di@rosminzdrav.ru).

- Опубликовать рабочую версию web-службы в рабочей версии ИПС. Для этого сотрудник ответственной организации направляет заявку на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)) согласно формам, приведенным в п.1.4 документа «Методические материалы по подключению к ИПС», размещенным на портале ЕГИСЗ по адресу: <http://portal.egisz.rosminzdrav.ru/materials/11>. В столбце "Прямой адрес веб-службы" нужно указать адрес колбэка, который предоставляет эксплуатация. Версия веб-службы - SOAP 1.1. Тип веб-службы - асинхронный. При заполнении заявки необходимо указать промышленную среду. Для ФРМО и ФРМП нужно формировать отдельные заявки, но с одинаковым колбэком. Заявки в обязательном порядке подаются в 2 форматах – DOCX и PDF, при этом заявка в формате PDF должна содержать подпись и расшифровку подписи ответственного лица, подтвержденную печатью организации.
 

**Необходимо помнить, что колбэк для сервисов ФРМО и ФРМП используется один. Как в тестовой версии, так и в продуктивной. Инструкции по разворачиванию сервисов для эксплуатации находятся здесь: ФРМО, ФРМП.** Заявки в обязательном порядке подаются в 2 форматах – DOCX и PDF, при этом заявка в формате PDF должна содержать подпись и расшифровку подписи ответственного лица, подтвержденную печатью организации. Одновременно сканированную копию заявки направить в Минздрав России, по адресу: [di@rosminzdrav.ru](mailto:di@rosminzdrav.ru).
- Зарегистрировать ИС в промышленной версии Системы. Для этого сотрудник ответственной организации направляет заявку на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)) в соответствии с формой, приведенной в Приложении 2 документа, расположенного по адресу <https://portal.egisz.rosminzdrav.ru/materials/481> для доступа в ФРМО, и Приложении 2 документа, расположенного по адресу <https://portal.egisz.rosminzdrav.ru/materials/483> для ФРМП. Заявки в обязательном порядке подаются в 2 форматах – docx и pdf, при этом заявка в формате PDF должна содержать подпись и расшифровку подписи ответственного лица, подтвержденную печатью организации. Одновременно сканированную копию заявки направить в Минздрав России, по адресу: [di@rosminzdrav.ru](mailto:di@rosminzdrav.ru).
- В течение 2 (двух) рабочих дней после получения заявки СТП ЕГИСЗ предоставляет идентификатор ИС (идентификатор генерируется на основании алгоритма генерации, настроенного в БД рабочей версии Системы), логин/пароль доступа, а также ссылку к веб-интерфейсу рабочей версии системы.
- В случае необходимости осуществить настройку ИС и начать взаимодействие с рабочей версией Системы.
- В случае возникновения ошибок при интеграционном взаимодействии с рабочей версией Системы на стороне ИС, которые будут отрицательно влиять на функционирование и производительность Системы, ИС будет отключен от работы с интеграционными профилями промышленной площадки с уведомлением лица, от ответственной организации за эксплуатацию ИС, до восстановления корректного интеграционного взаимодействия на тестовой среде. После устранения ошибок интеграционного взаимодействия на тестовой среде необходимо направить отчет о результатах тестирования и протокол приемочных испытаний на адрес СТП ЕГИСЗ ([egisz@rt-eu.ru](mailto:egisz@rt-eu.ru)), в соответствии с формой, приведенной в приложении (Приложение 3). СТП ЕГИСЗ в ответном письме направляет заключение о восстановлении возможности работы с промышленной версией Системы или о необходимости доработки ИС и проведения повторного тестирования. Заключение предоставляется в течение 3 (трех) рабочих дней с момента получения отчета о проведении тестирования.



## 6 ФРМО. Отчет о тестировании и массовая выгрузка.

### 6.1 Общие сведения:

Отчет о тестировании включает для каждого метода данные четырех сообщений: 1) запрос клиента в ФРМО, 2) GUID асинхронного ответа из синхронного ответа ФРМО, 3) асинхронный ответ ФРМО. 4) ответ клиента на асинхронный ответ ФРМО

Все эти данные могут быть получены из логов сервиса. Логируются осуществляется параллельно в файлы <папка tomee>\logs\catalina.<дата логов>.log и <папка tomee>\logs\frmo\frmo<дата логов>.log.

Данные запроса <?xml ... </SOAP-ENV:Envelope> в логе следуют за фразой **INFO: \*\*\* SOAP message to send:**

GUID асинхронного ответа и данные асинхронного ответа <ips\_response\_id>f0fb959d-d1a0-440f-85f0-c066cff9259b</ips\_response\_id><?xml ... </departs> в логе следуют за фразой: **INFO: \*\*\* Working answer from the queue:**

Данные ответа клиента на асинхронный ответ <?xml ...</SOAP-ENV:Envelope>следуют за фразой **INFO: Ответ на асинхронный ответ:**

В процессе подготовки отчетов и для проведения массовой выгрузки используются sql-скрипты. Ниже приведены варианты запросов для MSSQL. Имеются варианты для PostgreSQL (во вложении на примере для Якутской ГБЗ [YGB3-Scripts.ZIP](#)).

6.1.1 Для запуска единичной сессии ФРМО можно использовать скрипт FRMOSession\_run.sql (альтернатива способу запуска через Промед):

```
DECLARE @RC int
DECLARE @Lpu_id bigint
DECLARE @FRMOSession_Name varchar(100)
DECLARE @FRMOSession_begDT datetime
declare @FRMOSessionMode_id bigint = 1 -- 1 - экспорт в ФРМО 2 -- импорт из ФРМО 3 --
только чтение данных из ФРМО
DECLARE @pmUser_id bigint
DECLARE @Error_Code int
DECLARE @Error_Message varchar(4000)

set @Lpu_id = 10010833
set @FRMOSession_Name = 'FRMO export test'
set @FRMOSession_begDT = '2020-08-18'
set @pmUser_id = 1

SELECT top 1 s.FRMOsession_id, st.FRMOsessionActionType_Code,
st.FRMOsessionHist_doneDT, st.FRMOsessionHist_pid
FROM dbo.FRMOsession s (NOLOCK)
outer apply (select top 1 a.FRMOsessionActionType_Code, h.FRMOsessionHist_doneDT,
h.FRMOsessionHist_pid
from dbo.FRMOsessionHist h (NOLOCK)
inner join dbo.FRMOsessionActionType a (NOLOCK) on a.FRMOsessionActionType_id =
h.FRMOsessionActionType_id
where s.FRMOsession_id = h.FRMOsession_id
order by h.FRMOsessionHist_insDT desc
) st
WHERE (1 = 1) and s.FRMOsession_endDT is NULL and s.FRMOsession_Service = 'FRMO'
and s.Lpu_id = @Lpu_id

EXECUTE @RC = [dbo].[p_FRMOsession_run]
@Lpu_id
,@FRMOsession_Name
```

```
,@FRMOsession_begDT
,@FRMOsessionMode_id
,@pmUser_id
,@Error_Code OUTPUT
,@Error_Message OUTPUT
select @Error_Code ,@Error_Message
```

#### 6.1.2 Описание скрипта:

В параметрах указываются:

- 1) Режим сессии @FRMOsessionMode\_id bigint = 1 -- 1 - экспорт в ФРМО 2 -- импорт из ФРМО 3 -- только чтение данных из ФРМО
- 2) МО: @Lpu\_id = 10010833 -- 13004038 -- 10011030 -- 13101710
- 3) имя сессии: @FRMOsession\_Name = 'FRMO export test'
- 4) дата запуска сессии: @FRMOsession\_begDT = '2020-08-18'
- 5) пользователь: @pmUser\_id = 1

#### 6.1.3 Ошибки по сессии можно посмотреть по скрипту FRMOsessionError.sql (также доступны в Промеде):

```
declare @FRMOsession_id bigint = 1
select
FE.FRMOsessionError_id, fe.FRMOsessionHist_id, st.FRMOsessionHist_id
, ISNULL(fet.FRMOsessionErrorType_Name, fe.FRMOsessionError_Message) as error_text
, a.FRMOsessionActionType_descr
, L.Lpu_Nick
, lb.LpuBuildingPass_Name, lu.LpuUnit_Name, LS.LpuSection_Name, fe.LpuStaff_Num,
lu.LpuUnit_ID, FE.LpuSection_id, fe.LpuBuildingPass_id
, fe.FRMOsessionError_Message
, FE.FRMOsessionError_insdt
FROM dbo.FRMOsession ses (NOLOCK)
inner join v_Lpu L with (nolock) on L.Lpu_id = ses.Lpu_id
outer apply (select top 1 h.FRMOsessionHist_id, h.FRMOsessionHist_insDT
from dbo.FRMOsessionHist h (NOLOCK)
inner join dbo.FRMOsessionActionType a (NOLOCK) on a.FRMOsessionActionType_id =
h.FRMOsessionActionType_id
where ses.FRMOsession_id = h.FRMOsession_id and a.FRMOsessionActionType_code in
(60,64)
-- and h.FRMOsessionHist_id = 1334
order by h.FRMOsessionHist_id desc
) st
inner join [dbo].[FRMOsessionError] fe (NOLOCK) on fe.FRMOsession_id =
ses.FRMOsession_id and fe.FRMOsessionHist_id >= st.FRMOsessionHist_id
inner join dbo.FRMOsessionHist h (NOLOCK) on fe.FRMOsessionHist_id =
h.FRMOsessionHist_id
inner join dbo.FRMOsessionActionType a (NOLOCK) on a.FRMOsessionActionType_id =
h.FRMOsessionActionType_id
left join [dbo].[FRMOsessionErrorType] fet (NOLOCK) on fe.FRMOsessionErrorType_id =
fet.FRMOsessionErrorType_id
left JOIN dbo.v_LpuBuildingPass AS lb with (NOLOCK) ON lb.LpuBuildingPass_id =
fe.LpuBuildingPass_id
left JOIN dbo.v_LpuUnit AS lu with (NOLOCK) ON lu.LpuUnit_id = fe.LpuUnit_id
left join v_LpuSection LS with (nolock) on LS.LpuSection_id = FE.LpuSection_id
WHERE (1 = 1) and ses.FRMOsession_id = @FRMOsession_id
and h.FRMOsessionHist_insDT >= isnull(st.FRMOsessionHist_insDT,
ses.FRMOsession_insDT)
-- and fe.LpuBuildingPass_id in (383, 2400)
order by error_text, fe.FRMOsessionHist_id
```

6.1.4 Описание скрипта:  
Параметр Ид сессии: @FRMOSession\_id bigint = 1.

## 6.2 Запуск массовой выгрузки:

6.2.1 Для запуска массовой выгрузки необходимо создание контрольного примера структуры МО: одна МО и по две единицы структурных элементов каждого типа. Описание данных с указанием форм Системы изложены в инструкции:

6.2.2 Описание данных для массовой выгрузки ФРМО

6.2.3 Запуск массовой выгрузки выполняется скриптом MassiveTestCycle.sql. Этот скрипт также может использоваться для получения тестовых отчетов. Перед запуском скрипта должна быть выполнена правка ошибок форматно-логического контроля, полученных на стадии верификации сессии экспорта в ФРМО. Во время выполнения скрипта должно быть гарантировано, что данные используемых объектов структуры МО не будут изменяться каким-либо внешним процессом.

6.2.4 Версия для MSSQL:

```
declare @lpu_id bigint = 13101710
declare @pmUser_id bigint = 1
declare @today Datetime = GetDate()

-- для теста Обновления mo.update
declare @LpuPmuType_id bigint = 3
declare @LpuPmuType_nid bigint = 35

-- здание для создания нового /удаления, которое не используется в подразделениях и
медоборудовании
declare @LpuBuildingPass_id bigint = 1874
declare @MOArea_id bigint = 10004
-- площадка для нового здания
declare @MOArea_nid bigint = 10003
-- для теста создания/удаления здания
declare @LpuBuildingPass_Name_new varchar(50) = 'Здание в парке'
declare @LpuBuildingPass_Name_old varchar(50) = 'Здание в саду'

-- для теста Обновления mo_building.update
-- здание для редактирования, которое можно использовать в подразделениях
declare @LpuBuildingPass2_id bigint = 1873
declare @LpuBuildingPass_YearBuilt_new Datetime = '20180101'
declare @LpuBuildingPass_YearBuilt_old Datetime = '2003-01-01 00:00:00.000'

-- подразделение для создания/удаления, которое не используется в штатке
declare @LpuUnit_id bigint = 99560015008
-- для создания/удаления подразделения
declare @LpuUnit_Name_new varchar(50) = 'Полка вторая'
declare @LpuUnit_Name_old varchar(50) = 'Полка 2' -- Яросл

-- для теста Обновления mo_depart.update
-- подразделение для редактирования
declare @LpuUnit_id2 bigint = 99560015007
declare @LpuUnitProfile_fid_new bigint = 1043
-- а также для подготовки LpuUnit к create
declare @LpuUnitProfile_fid_old bigint = 1011

-- для создания/удаления расписания
declare @LpuStaff_id bigint = 2
declare @LpuStaff_NumOld varchar(10) = '2'
```

```

declare @LpuStaff_NumNew varchar(10) = '3'
declare @LpuStaff_BeginDate_new Datetime = '20170101'
declare @LpuStaff_EndDate_new Datetime = '20171231'
declare @LpuStaff_BeginDate_old Datetime = '20180101'
declare @LpuStaff_EndDate_old Datetime = '20181231'

-- для обновления расписания
-- LpuStaff_id = 336
declare @Staff_id varchar(10) = 22
declare @Staff_RateNew varchar(10) = 10
declare @Staff_RateOld varchar(10) = 100

-- для Обновления mo_equipment.update
declare @MedProductCard_id2 bigint = 135854
declare @MedProductCard_id bigint = 135853
declare @MedProductCard_SerialNumber_old varchar(100) = '2'
declare @MedProductCard_SerialNumber_new varchar(100) = '3'
-- для создания/удаления mo_equipment.delete
declare @MedProductCard_InventoryNumber_old varchar(100) = '11'
declare @MedProductCard_InventoryNumber_new varchar(100) = '12'
declare @RegCertificate_id bigint = 80158
declare @MedProductOrg_id_old bigint = 6880000889
declare @MedProductOrg_id_new bigint = 68320115679

DECLARE @RC int
DECLARE @FRMOsession_Name varchar(100)
DECLARE @FRMOsession_begDT datetime
DECLARE @Error_Code int
DECLARE @Error_Message varchar(4000)

declare @i int = 0
declare @count int = 20

begin try

while (@i < @count) begin

-- ТАКТ 1 обработки данных при массовой выгрузке
if (@i % 2 = 0) begin

select 'такт 1', @i, GetDate()

-- mo.update
-- присваиваем новое значение
update dbo.Lpu set LpuPmuType_id = @LpuPmuType_nid where Lpu_id=@Lpu_id

-- mo_building.update
-- присваиваем новое значение
update dbo.LpuBuildingPass set LpuBuildingPass_YearBuilt =
@LpuBuildingPass_YearBuilt_new where LpuBuildingPass_id=@LpuBuildingPass2_id

-- mo_building.create+delete
-- присваиваем новое название и адрес
update dbo.LpuBuildingPass set LpuBuildingPass_Name = @LpuBuildingPass_Name_new +
'+LTRIM(str(@i)), LpuBuildingPass_FRMOname = @LpuBuildingPass_Name_new +
'+LTRIM(str(@i)), MOArea_id = @MOArea_nid
where LpuBuildingPass_id=@LpuBuildingPass_id

-- mo_depart.update
-- присваиваем новое значение
update dbo.LpuUnit set LpuUnitProfile_fid = @LpuUnitProfile_fid_new where
LpuUnit_id=@LpuUnit_id2

-- тест mo_depart.delete+create
-- присваиваем новое название
update dbo.LpuUnit set LpuUnit_Name = @LpuUnit_Name_new + ' '+LTRIM(str(@i)) where

```

```

LpuUnit_id=@LpuUnit_id
--1.2.643.5.1.13.13.12.2.35.3259.0.33920 Vologda old
--1.2.643.5.1.13.13.12.2.35.3259.0.34484 Vologda new

-- тест mo_staff.delete
-- присваиваем новый номер
update dbo.LpuStaff set LpuStaff_Num = @LpuStaff_NumNew + LTRIM(str(@i)),
LpuStaff_begDate = @LpuStaff_BeginDate_new, LpuStaff_ApprovalDT =
@LpuStaff_BeginDate_new, LpuStaff_endDate = @LpuStaff_EndDate_new where LpuStaff_id
= @LpuStaff_id

-- mo_staff.update
-- присваиваем новое значение
update persis.Staff set Rate = @Staff_RateNew where Staff.id = @Staff_id

-- mo_equipment.update
-- присваиваем новое значение
update passport.MedProductCard set MedProductCard_SerialNumber =
@MedProductCard_SerialNumber_new where MedProductCard_id=@MedProductCard_id2

-- тест mo_equipment.delete+create
-- присваиваем новый инвентарный номер
update passport.MedProductCard set MedProductCard_InventoryNumber =
@MedProductCard_InventoryNumber_new where MedProductCard_id=@MedProductCard_id
update passport.RegCertificate set Org_prid = @MedProductOrg_id_new where
RegCertificate_id = @RegCertificate_id

end else begin

-- ТАКТ 2 обработки данных при массовой выгрузке
-- mo_building.update
select 'такт 2', @i, GetDate()

-- mo.update
-- возвращаем настоящее значение
update dbo.Lpu set LpuPmuType_id = @LpuPmuType_id where Lpu_id=@Lpu_id

-- возвращаем настоящее значение
update dbo.LpuBuildingPass set LpuBuildingPass_YearBuilt =
@LpuBuildingPass_YearBuilt_old where LpuBuildingPass_id=@LpuBuildingPass2_id

-- mo_building.create+delete
-- возвращаем настоящее название
update dbo.LpuBuildingPass set LpuBuildingPass_Name = @LpuBuildingPass_Name_old,
LpuBuildingPass_FRMOname = @LpuBuildingPass_Name_old, MOArea_id = @MOArea_id
where LpuBuildingPass_id=@LpuBuildingPass_id

-- mo_depart.update
-- возвращаем прежнее значение
update dbo.LpuUnit set LpuUnitProfile_fid = @LpuUnitProfile_fid_old where
LpuUnit_id=@LpuUnit_id2

-- тест mo_depart.delete+create
-- возвращаем настоящее название
update dbo.LpuUnit set LpuUnit_Name = @LpuUnit_Name_old where LpuUnit_id=@LpuUnit_id
--1.2.643.5.1.13.13.12.2.35.3259.0.34485 Vologda new 2

-- тест mo_staff.delete
-- возвращаем прежний номер
update dbo.LpuStaff set LpuStaff_Num = @LpuStaff_NumOld, LpuStaff_begDate =
@LpuStaff_BeginDate_old, LpuStaff_ApprovalDT = @LpuStaff_BeginDate_old,
LpuStaff_endDate = @LpuStaff_EndDate_old where LpuStaff_id = @LpuStaff_id

-- mo_staff.update
-- возвращаем старое значение
update persis.Staff set Rate = @Staff_RateOld where Staff.id = @Staff_id

```

```

-- mo_equipment.update
-- возвращаем настоящее значение
update passport.MedProductCard set MedProductCard_SerialNumber =
@MedProductCard_SerialNumber_old where MedProductCard_id=@MedProductCard_id2

-- тест mo_equipment.delete+create
-- возвращаем настоящий инвентарный номер
update passport.MedProductCard set MedProductCard_InventoryNumber =
@MedProductCard_InventoryNumber_old where MedProductCard_id=@MedProductCard_id
update passport.RegCertificate set Org_prid = @MedProductOrg_id_old where
RegCertificate_id = @RegCertificate_id
end

select LpuBuildingPass_Name, LpuBuildingPass_YearBuilt, LpuBuildingPass_BuildingIdent
from dbo.LpuBuildingPass where LpuBuildingPass_id in (@LpuBuildingPass_id,
@LpuBuildingPass2_id)

select LpuUnit_Name, LpuUnitProfile_fid, LpuUnit_IsHomeVisit, UnitDepartType_fid,
LpuUnit_Phone from dbo.LpuUnit where LpuUnit_id in (@LpuUnit_id, @LpuUnit_id2)

select * from dbo.v_LpuStaff where Lpu_id = @Lpu_id

select mc.MedProductCard_InventoryNumber, mc.MedProductCard_SerialNumber, rc.Org_prid
from passport.MedProductCard mc with (nolock) left join passport.RegCertificate rc with
(nolock) on rc.MedProductCard_id = mc.MedProductCard_id
where mc.MedProductCard_id in (@MedProductCard_id, @MedProductCard_id2)

set @FRMOsession_Name = 'frmo test '+str(@i,2)
set @FRMOsession_begDT = dbo.tzGetDate()

declare @datetime datetime = dbo.tzGetDate()
declare @FRMOsession_id bigint = null
declare @FRMOsessionActionType_Code int = null
declare @FRMOsessionHist_doneDT datetime = null
declare @FRMOsessionHist_pid bigint = null
declare @FRMOsessionHist_id bigint = null

SELECT top 1 @FRMOsession_id = s.FRMOsession_id, @FRMOsessionActionType_Code =
st.FRMOsessionActionType_Code, @FRMOsessionHist_doneDT =
st.FRMOsessionHist_doneDT, @FRMOsessionHist_pid = st.FRMOsessionHist_pid
FROM dbo.FRMOsession s (NOLOCK)
outer apply (select top 1 a.FRMOsessionActionType_Code, h.FRMOsessionHist_doneDT,
h.FRMOsessionHist_pid
from dbo.FRMOsessionHist h (NOLOCK)
inner join dbo.FRMOsessionActionType a (NOLOCK) on a.FRMOsessionActionType_id =
h.FRMOsessionActionType_id
where s.FRMOsession_id = h.FRMOsession_id
order by h.FRMOsessionHist_insDT desc
) st
WHERE (1 = 1) and s.FRMOsession_endDT is NULL and s.FRMOsession_Service = 'FRMO'
and s.Lpu_id = @Lpu_id

if @FRMOsession_id is null begin
EXECUTE [dbo].[p_FRMOsession_ins]
@FRMOsession_id = @FRMOsession_id output,
@FRMOsession_Name = @FRMOsession_Name,
@FRMOsession_begDT = @FRMOsession_begDT,
@Lpu_id = @Lpu_id,
@FRMOsession_Service = 'FRMO',
@pmUser_id = @pmUser_id,
@Error_Code = @Error_Code OUTPUT,
@Error_Message = @Error_Message OUTPUT

if (ISNULL(@Error_Message, '') <> '')
RAISERROR (@Error_Message, 16, 1)

```

```

end else begin
RAISERROR ('*** ERROR CREATING SESSION', 16, 1)
end
select 'FRMOSession_id=', @FRMOSession_id, @FRMOSessionActionType_Code

select 'WAITING VERIFICATION PHASE', GetDate()
set @FRMOSessionHist_id = null
while (@FRMOSessionHist_id is null) begin
-- select 'WAITING VERIFICATION PHASE'
WAITFOR DELAY '00:00:01';
set @FRMOSessionHist_id = (select top 1 h.FRMOSessionHist_id from dbo.FRMOSessionHist
h (NOLOCK)
where @FRMOSession_id = h.FRMOSession_id and h.FRMOSessionActionType_id = 10057)
end
select 'WAITING ERRORS', GetDate()
WAITFOR DELAY '00:00:10';
select 'FRMOSessionHist_id=', @FRMOSessionHist_id

if (exists (select * FROM [dbo].[FRMOSessionError] fe (NOLOCK) WHERE
fe.FRMOSessionHist_id = @FRMOSessionHist_id)) begin
select 'DELETING ERRORS'
delete FROM [dbo].[FRMOSessionError] WHERE FRMOSessionHist_id =
@FRMOSessionHist_id
end

while ((SELECT top 1 s.FRMOSession_success FROM dbo.FRMOSession s (NOLOCK)
where @FRMOSession_id = s.FRMOSession_id) is NULL)
begin
-- select 'WAITING SESSION COMPLETED'
WAITFOR DELAY '00:00:03';
end

select 'SESSION №'+STR(@i)+' IS DONE', GetDate()

declare @err varchar(200) = (SELECT top 1 isnull(s.FRMOSession_comment,") FROM
dbo.FRMOSession s (NOLOCK)
where @FRMOSession_id = s.FRMOSession_id)

if (@err <> ") begin
set @err = '*** ERROR EXECUTING SESSION '+ @err
RAISERROR (@err, 16, 1)
end

set @i = @i + 1
end

end try
begin catch
set @Error_Code = error_number()
set @Error_Message = error_message()
end catch

select 'Error', @Error_Code, @Error_Message

```

6.2.5 Версия для PostgreSQL:  
/\*

```

-- Создание таблицы параметров
drop table if exists tmp$parameters;
create temp table tmp$parameters as
select
null::bigint as error_code,
null::varchar(4000) as error_message,

```

```

null::boolean as success,
null::int as RC,
null::varchar(100) as FRMOsession_Name,
null::timestamp as FRMOsession_begDT,
null::int as i,
null::int as count,
null::bigint as Lpu_id,
null::bigint as pmUser_id,
null::timestamp as today,
null::bigint as LpuPmuType_id,
null::bigint as LpuPmuType_nid,
null::bigint as LpuBuildingPass_id,
null::bigint as MOArea_id,
null::bigint as MOArea_nid,
null::varchar(50) as LpuBuildingPass_Name_new,
null::varchar(50) as LpuBuildingPass_Name_old,
null::bigint as LpuBuildingPass2_id,
null::timestamp as LpuBuildingPass_YearBuilt_new,
null::timestamp as LpuBuildingPass_YearBuilt_old,
null::bigint as LpuUnit_id,
null::varchar(50) as LpuUnit_Name_new,
null::varchar(50) as LpuUnit_Name_old,
null::bigint as LpuUnit_id2,
null::bigint as LpuUnitProfile_fid_new,
null::bigint as LpuUnitProfile_fid_old,
null::bigint as LpuStaff_id,
null::varchar(10) as LpuStaff_NumNew,
null::varchar(10) as LpuStaff_NumOld,
null::timestamp as LpuStaff_BeginDate_new,
null::timestamp as LpuStaff_EndDate_new,
null::timestamp as LpuStaff_BeginDate_old,
null::timestamp as LpuStaff_EndDate_old,
null::bigint as Staff_id,
null::float8 as Staff_RateNew,
null::float8 as Staff_RateOld,
null::bigint as MedProductCard_id2,
null::bigint as MedProductCard_id,
null::varchar(100) as MedProductCard_SerialNumber_old,
null::varchar(100) as MedProductCard_SerialNumber_new,
null::varchar(100) as MedProductCard_InventoryNumber_old,
null::varchar(100) as MedProductCard_InventoryNumber_new,
null::bigint as RegCertificate_id,
null::bigint as MedProductOrg_id_old,
null::bigint as MedProductOrg_id_new,

null::bigint as FRMOsession_id,
null::boolean as FRMOsession_success,
null::bigint as FRMOsessionHist_pid,
null::bigint as FRMOsessionHist_id,
null::int as FRMOsessionActionType_Code,
null::timestamp as FRMOsessionHist_doneDT;

update tmp$parameters set error_code = 0, error_message = null, success = true
, i = 0, count = 2

-- Данные для Якутской ГБ №3
, Lpu_id = 7, pmUser_id = 1, today = dbo.GetDate()

-- для теста Обновления mo.update
, LpuPmuType_id = 3
, LpuPmuType_nid = 35

```



```

-- здание для создания нового /удаления, которое не используется в подразделениях и
медоборудовании
, LpuBuildingPass_id = 140101000000008
, MOArea_id = 140101000000002
-- площадка для нового здания
, MOArea_nid = 140101000000002

-- для теста создания/удаления здания
, LpuBuildingPass_Name_new = 'Здание тест новое'
, LpuBuildingPass_Name_old = 'Здание тест'

-- для теста Обновления mo_building.update
-- здание для редактирования, которое можно использовать в подразделениях
, LpuBuildingPass2_id = 140101000000007
, LpuBuildingPass_YearBuilt_new = '20070101'
, LpuBuildingPass_YearBuilt_old = '2005-01-01 00:00:00.000'

-- подразделение для создания/удаления, которое не используется в штатке
, LpuUnit_id = 140101000000069
-- для создания/удаления подразделения
, LpuUnit_Name_new = 'Дневной стационар при стационаре новый'
, LpuUnit_Name_old = 'Дневной стационар при стационаре'

-- для теста Обновления mo_depart.update
-- подразделение для редактирования
, LpuUnit_id2 = 140101000000086
, LpuUnitProfile_fid_new = 1043
-- а также для подготовки LpuUnit к create
, LpuUnitProfile_fid_old = 1005

-- для создания/удаления расписания
, LpuStaff_id = 140101000000015
, LpuStaff_NumOld = '3'
, LpuStaff_NumNew = '4'
, LpuStaff_BeginDate_new = '2018-01-01 00:00:00'
, LpuStaff_EndDate_new = '2018-12-31 00:00:00'
, LpuStaff_BeginDate_old = '2018-01-01 00:00:00'
, LpuStaff_EndDate_old = '2018-12-01 00:00:00'

-- для обновления расписания
, Staff_id = 140101000011108
, Staff_RateNew = 20
, Staff_RateOld = 10

-- для Обновления mo_equipment.update
, MedProductCard_id2 = 135854
, MedProductCard_id = 135853
, MedProductCard_SerialNumber_old = '2'
, MedProductCard_SerialNumber_new = '3'
-- для создания/удаления mo_equipment.delete
, MedProductCard_InventoryNumber_old = '11'
, MedProductCard_InventoryNumber_new = '12'
, RegCertificate_id = 80158
, MedProductOrg_id_old = 6880000889
, MedProductOrg_id_new = 68320115679;
*/
--select * from tmp$parameters;

-- Выполнение одного шага цикла

do $$
declare
vars tmp$parameters%ROWTYPE;

```

```

begin
select * from tmp$parameters into vars limit 1;

SELECT s.FRMOsession_id, st.FRMOsessionActionType_Code,
st.FRMOsessionHist_doneDT, st.FRMOsessionHist_pid
INTO vars.FRMOsession_id, vars.FRMOsessionActionType_Code,
vars.FRMOsessionHist_doneDT, vars.FRMOsessionHist_pid
FROM dbo.FRMOsession s
left join lateral (select a.FRMOsessionActionType_Code, h.FRMOsessionHist_doneDT,
h.FRMOsessionHist_pid
from dbo.FRMOsessionHist h
inner join dbo.FRMOsessionActionType a on a.FRMOsessionActionType_id =
h.FRMOsessionActionType_id
where s.FRMOsession_id = h.FRMOsession_id
order by h.FRMOsessionHist_insDT desc
limit 1 ) st on true
WHERE (1 = 1) and s.FRMOsession_endDT is NULL and s.FRMOsession_Service = 'FRMO'
and s.Lpu_id = vars.Lpu_id
limit 1;

if vars.FRMOsession_id is not null then
raise sqlstate 'U0002';
end if;

if (vars.error_code = 0) and (vars.i < vars.count) then
if (vars.i % 2 = 0) then

raise notice 'такт 1: шаг % время %', vars.i, dbo.GetDate();

-- mo.update
-- присваиваем новое значение
update dbo.Lpu set LpuPmuType_id = vars.LpuPmuType_nid where Lpu_id=vars.Lpu_id;

-- mo_building.update
-- присваиваем новое значение
update dbo.LpuBuildingPass set LpuBuildingPass_YearBuilt =
vars.LpuBuildingPass_YearBuilt_new where LpuBuildingPass_id=vars.LpuBuildingPass2_id;

-- mo_building.create+delete
-- присваиваем новое название и адрес
update dbo.LpuBuildingPass set LpuBuildingPass_Name = vars.LpuBuildingPass_Name_new ||
'' || cast(vars.i as varchar), LpuBuildingPass_FRMOname = vars.LpuBuildingPass_Name_new
|| '' || cast(vars.i as varchar), MOArea_id = vars.MOArea_nid
where LpuBuildingPass_id=vars.LpuBuildingPass_id;

-- mo_depart.update
-- присваиваем новое значение
update dbo.LpuUnit set LpuUnitProfile_fid = vars.LpuUnitProfile_fid_new where
LpuUnit_id=vars.LpuUnit_id2;

-- тест mo_depart.delete+create
-- присваиваем новое название
update dbo.LpuUnit set LpuUnit_Name = vars.LpuUnit_Name_new || '' || cast(vars.i as varchar)
where LpuUnit_id=vars.LpuUnit_id;

-- тест mo_staff.delete
-- присваиваем новый номер
update dbo.LpuStaff set LpuStaff_Num = vars.LpuStaff_NumNew || CAST(vars.i as varchar),
LpuStaff_begDate = vars.LpuStaff_BeginDate_new, LpuStaff_ApprovalDT =
vars.LpuStaff_BeginDate_new, LpuStaff_endDate = vars.LpuStaff_EndDate_new where
LpuStaff_id = vars.LpuStaff_id;

```

```

-- mo_staff.update
-- присваиваем новое значение
update persis.Staff set Rate = vars.Staff_RateNew where Staff.id = vars.Staff_id;

-- mo_equipment.update
-- присваиваем новое значение
-- update passport.MedProductCard set MedProductCard_SerialNumber =
vars.MedProductCard_SerialNumber_new where
MedProductCard_id=vars.MedProductCard_id2

-- тест mo_equipment.delete+create
-- присваиваем новый инвентарный номер
-- update passport.MedProductCard set MedProductCard_InventoryNumber =
vars.MedProductCard_InventoryNumber_new where
MedProductCard_id=vars.MedProductCard_id
-- update passport.RegCertificate set Org_prid = vars.MedProductOrg_id_new where
RegCertificate_id = vars.RegCertificate_id

else

-- ТАКТ 2 обработки данных при массовой выгрузке
-- mo_building.update

raise notice 'такт 2: шаг % время %', vars.i, dbo.GetDate();

-- mo.update
-- возвращаем настоящее значение
update dbo.Lpu set LpuPmuType_id = vars.LpuPmuType_id where Lpu_id=vars.Lpu_id;

-- возвращаем настоящее значение
update dbo.LpuBuildingPass set LpuBuildingPass_YearBuilt =
vars.LpuBuildingPass_YearBuilt_old where LpuBuildingPass_id=vars.LpuBuildingPass2_id;

-- mo_building.create+delete
-- возвращаем настоящее название
update dbo.LpuBuildingPass set LpuBuildingPass_Name = vars.LpuBuildingPass_Name_old,
LpuBuildingPass_FRMOname = vars.LpuBuildingPass_Name_old, MOArea_id =
vars.MOArea_id where LpuBuildingPass_id=vars.LpuBuildingPass_id;

-- mo_depart.update
-- возвращаем прежнее значение
update dbo.LpuUnit set LpuUnitProfile_fid = vars.LpuUnitProfile_fid_old where
LpuUnit_id=vars.LpuUnit_id2;

-- тест mo_depart.delete+create
-- возвращаем настоящее название
update dbo.LpuUnit set LpuUnit_Name = vars.LpuUnit_Name_old where
LpuUnit_id=vars.LpuUnit_id;

-- тест mo_staff.delete
-- возвращаем прежний номер
update dbo.LpuStaff set LpuStaff_Num = vars.LpuStaff_NumOld, LpuStaff_begDate =
vars.LpuStaff_BeginDate_old, LpuStaff_ApprovalDT = vars.LpuStaff_BeginDate_old,
LpuStaff_endDate = vars.LpuStaff_EndDate_old where LpuStaff_id = vars.LpuStaff_id;

-- mo_staff.update
-- возвращаем старое значение
update persis.Staff set Rate = vars.Staff_RateOld where Staff.id = vars.Staff_id;

```

```

-- mo_equipment.update
-- возвращаем настоящее значение
-- update passport.MedProductCard set MedProductCard_SerialNumber =
vars.MedProductCard_SerialNumber_old where
MedProductCard_id=vars.MedProductCard_id2

-- тест mo_equipment.delete+create
-- возвращаем настоящий инвентарный номер
-- update passport.MedProductCard set MedProductCard_InventoryNumber =
vars.MedProductCard_InventoryNumber_old where
MedProductCard_id=vars.MedProductCard_id
-- update passport.RegCertificate set Org_prid = vars.MedProductOrg_id_old where
RegCertificate_id = vars.RegCertificate_id
end if;

select dbo.GetDate() into vars.FRMOSession_begDT;
select 'frmo test ' || cast(vars.i as varchar) into vars.FRMOSession_Name;

select fnc.FRMOSession_id, cast(fnc.Error_Code as bigint), fnc.Error_Message
into vars.FRMOSession_id, vars.error_code, vars.error_message
from dbo.p_FRMOSession_ins (
FRMOSession_id := vars.FRMOSession_id,
FRMOSession_Name := vars.FRMOSession_Name,
FRMOSession_begDT := vars.FRMOSession_begDT,
FRMOSessionMode_id := 1,
Lpu_id := vars.Lpu_id,
FRMOSession_Service := 'FRMO',
pmUser_id := vars.pmUser_id) fnc;

if coalesce(vars.error_code, 0) != 0 then
raise sqlstate 'U0001';
end if;

raise notice 'FRMOSession_id= % FRMOSessionName= % created', vars.FRMOSession_id,
vars.FRMOSession_Name;

select vars.i + 1 into vars.i;
update tmp$parameters set i = vars.i, FRMOSession_id = vars.FRMOSession_id,
FRMOSession_name = vars.FRMOSession_name;

else
raise notice 'CYCLES ARE OVER';
end if;

exception
when others then
if SQLSTATE = 'U0002' then
vars.error_code := 2;
vars.error_message := 'Session already exists!';
else
if SQLSTATE != 'U0001' then
vars.error_code := 1;
vars.error_message := SQLSTATE || ':' || SQLERRM;
end if;
end if;
vars.success := case when vars.error_code = 0 then 1 else 0 end;
update tmp$parameters set error_code = vars.error_code, error_message =
vars.error_message, success = vars.success, FRMOSession_id = null;
end $$;

```

```

do $$
declare
vars tmp$parameters%ROWTYPE;
begin
perform pg_sleep(1);
select * from tmp$parameters into vars limit 1;

if (vars.i <= vars.count) and (vars.error_code = 0) then

raise notice 'Processing FRMOsession_id= % FRMOsession_Name= % ',
vars.FRMOsession_id, vars.FRMOsession_Name;

raise notice 'WAITING VERIFICATION PHASE';
select null into vars.FRMOsessionHist_id;
while (vars.FRMOsessionHist_id is null) loop
perform pg_sleep(1);
select h.FRMOsessionHist_id into vars.FRMOsessionHist_id
from dbo.FRMOsessionHist h
where vars.FRMOsession_id = h.FRMOsession_id and h.FRMOsessionActionType_id =
10057 limit 1;
end loop;
raise notice 'WAITING ERRORS';
perform pg_sleep(10);
raise notice 'FRMOsessionHist_id= % ', vars.FRMOsessionHist_id;

raise notice 'DELETING ERRORS';
delete FROM dbo.FRMOsessionError WHERE FRMOsessionHist_id =
vars.FRMOsessionHist_id;

end if;

exception
when others then
vars.error_code := 1;
vars.error_message := SQLSTATE || ': ' || SQLERRM;
vars.success := case when vars.error_code = 0 then 1 else 0 end;
update tmp$parameters set error_code = vars.error_code, error_message =
vars.error_message, success = vars.success, FRMOsession_id = vars.FRMOsession_id;
end $$;

do $$
declare
vars tmp$parameters%ROWTYPE;
begin
perform pg_sleep(1);
select * from tmp$parameters into vars limit 1;

if (vars.i <= vars.count) and (vars.error_code = 0) then

select null into vars.FRMOsession_success;
while (vars.FRMOsession_success is NULL) loop
SELECT s.FRMOsession_success into vars.FRMOsession_success
FROM dbo.FRMOsession s where vars.FRMOsession_id = s.FRMOsession_id limit 1;
raise notice 'WAITING SESSION COMPLETED';
PERFORM pg_sleep(10);
end loop;

raise notice 'SESSION № % name % IS DONE', vars.i, vars.FRMOsession_Name;

SELECT coalesce(s.FRMOsession_comment, '') into vars.error_message
FROM dbo.FRMOsession s where vars.FRMOsession_id = s.FRMOsession_id limit 1;
if (vars.error_message <> '') then
raise sqlstate 'U0003';
end if;

```

```

end if;
exception
when others then
if SQLSTATE = 'U0003' then
vars.error_code := 3;
vars.error_message := '*** ERROR EXECUTING SESSION ' || vars.error_message;
else
vars.error_code := 1;
vars.error_message := SQLSTATE || ': ' || SQLERRM;
end if;
vars.success := case when vars.error_code = 0 then 1 else 0 end;
update tmp$parameters set error_code = vars.error_code, error_message =
vars.error_message, success = vars.success, FRMOsession_id = vars.FRMOsession_id;
end $$;

select i, error_code, error_message, success, FRMOsession_id from tmp$parameters;

```

### Описание скрипта:

Скрипт содержит цикл, на каждом шаге которого запускается сессия экспорта в ФРМО и отработывают все необходимые методы. Для этого на каждом шаге выполняется изменение данных структуры МО в БД Промед, что влечет за собой посылку в ФРМО сообщений create, update и delete. В процессе участвует два здания МО, два подразделения МО, два штатных расписания. Один из этих объектов служит для метода update, для чего изменяется его неключевое поле, другой для методов create + delete, для чего изменяются его ключевые поля в ФРМО. Нужно учитывать, что здание для методов create + delete не должно использоваться в посылаемых подразделениях, а подразделение для методов create + delete не должно использоваться в штатных расписаниях.

Параметры скрипта включают в себя ключи таблиц, которые могут быть получены выполнением скриптов для МО get\_MO, для зданий get\_Buildings, для подразделений get\_Departments, для штатного расписания get\_Staff.

Версия для MSSQL запускается однократно. В консоли сообщений отображается ход процесса. По завершении цикла на вкладке результатов запроса выводится результат: ошибка, если есть, успешность/неуспешность.

Версия для PostgreSQL запускается в несколько шагов. На первом шаге инициализируется временная таблица параметров. Код представлен в закомментированном блоке "Создание таблицы параметров" и запускается как выделенный фрагмент. Затем запускается скрипт столько раз, сколько шагов должно быть выполнено. Каждый шаг завершается выводом в консоль сообщений текста "SESSION № ### name frm test ## IS DONE". Также на вкладке результатов запроса выводится результат на шаге: ошибка, если есть, успешность/неуспешность и идентификатор сессии. Для прохождения массовой выгрузки должно быть выполнено 20 успешных шагов.

Параметры скрипта:

- Начальный счетчик циклов @i int = 0
- Количество циклов @count int = 20 (для тестовых отчетов можно указать 2)
- Id МО: @Lpu\_id = 10010833
- Id пользователя: @pmUser\_id = 1

-- для теста Обновления mo.update меняем вид медицинской деятельности

4. Имеющееся значение @LpuPmuType\_id bigint = 3 (можно получить скриптом get\_MO)
5. Значение для подстановки @LpuPmuType\_nid bigint = 35

```

-- здание для создания нового /удаления, которое не используется в подразделениях и
медоборудовании
4. Id здания @LpuBuildingPass_id bigint = 1874 (можно получить скриптом get_Buildings)
5. Имеющееся наименование здания @LpuBuildingPass_Name_old varchar(50) = 'Здание
в саду'
6. Id площадки здания @MOArea_id bigint = 10004 (можно получить скриптом get_
Buildings)
7. Наименование здания для подстановки @LpuBuildingPass_Name_new varchar(50) =
'Здание в парке'
8. Id площадки здания для подстановки @MOArea_nid bigint = 10003 (можно получить
скриптом get_Buildings)
-- для теста Обновления mo_building.update
-- здание для метода update, которое можно использовать в подразделениях
• Id здания @LpuBuildingPass2_id bigint = 1873 (можно получить скриптом get_Buildings)
• Год постройки здания имеющийся @LpuBuildingPass_YearBuilt_old Datetime = '2003-
01-01 00:00:00.000'
• Год постройки здания для подстановки @LpuBuildingPass_YearBuilt_new Datetime =
'20180101'

-- для теста создания+удаления mo_depart.create + mo_depart.delete
-- подразделение для создания/удаления, которое не используется в штатке
1. Id подразделения @LpuUnit_id bigint = 99560015008 (можно получить скриптом get_
Departments)
2. Наименование подразделения имеющееся @LpuUnit_Name_old varchar(50) = 'Полка 2'
3. Наименование подразделения для подстановки @LpuUnit_Name_new varchar(50) =
'Полка вторая'

-- для теста Обновления mo_depart.update
-- подразделение для редактирования
1. Id подразделения @LpuUnit_id2 bigint = 99560015007 (можно получить скриптом get_
Departments)
2. Профиль подразделения имеющийся @LpuUnitProfile_fid_old bigint = 1011 (можно
получить скриптом get_Departments)
3. Профиль подразделения для подстановки @LpuUnitProfile_fid_new bigint = 1043

-- для создания/удаления расписания
1. Id расписания @LpuStaff_id bigint = 2 (можно получить скриптом get_Staff)
2. Номер расписания имеющийся @LpuStaff_NumOld varchar(10) = '2'
3. Номер расписания для подстановки @LpuStaff_NumNew varchar(10) = '3'
4. Дата начала действия расписания @LpuStaff_BeginDate_old Datetime = '20180101'
(можно получить скриптом get_Staff)
5. Дата окончания действия расписания @LpuStaff_EndDate_old Datetime =
'20181231'(можно получить скриптом get_Staff)

```

6. Дата начала действия расписания для подстановки @LpuStaff\_BeginDate\_new  
Datetime = '20170101'
7. Дата окончания действия расписания для подстановки @LpuStaff\_EndDate\_new  
Datetime = '20171231'

-- для обновления расписания

- Id должности @Staff\_id varchar(10) = 22 (можно получить скриптом get\_Staff)
- Кол-во ставок по должности имеющееся @Staff\_RateOld varchar(10) = 100
- Кол-во ставок по должности для подстановки @Staff\_RateNew varchar(10) = 10

6.2.6 Скрипт get\_MO.sql для Медицинской организации, экспортируемой в ФРМО:

-- ОИД МО

```
declare @oid varchar(100)
set @oid = '1.2.643.5.1.13.13.12.2.59.5834'
```

```
select
Org.Org_Id,
PassportToken.PassportToken_id,
PassportToken.PassportToken_tid as lpu_oid,
Lpu.Lpu_id as Lpuld,
Lpu.Lpu_Name as FullName,
ISNULL(Lpu.Lpu_Nick,"") as ShortName,
Lpu.LpuPmuType_id,
LpuPmuType.LpuPmuType_Code,
ISNULL(Lpu.Lpu_Email,"") as SEmail,
ISNULL(Org.Org_INN,"") as Inn,
ISNULL(Org.Org_KPP,"") as Kpp,
ISNULL(Lpu.Lpu_OGRN,"") as Ogrn,
ISNULL(Lpu.LpuOwnership_id,0) as organizationType,
mainbuild.MOArea_id,
```

```
case when Okfs.Okfs_OID in (5,4,10,9) then 1 else 0 end as stateOrganization,
case when (Lpu.LpuOwnership_id = 3) or
(isnull(PassportTokenLpuHead.PassportToken_tid,'0') = '0') then null else
PassportTokenLpuHead.PassportToken_tid end as par_lpu_oid,
```

-- Ведомственная принадлежность

```
case when Lpu.LpuOwnership_id = 3 then null else DepartAffilType.DepartAffilType_Code end
as DepartAffilType_Code, --ID Идентификатор
DepartAffilType.DepartAffilType_Name, --Name Имя
DepartAffilTypeParent.DepartAffilType_Code as DepartAffilType_Parent, --Parent Родитель
```

-- Форма собственности

```
case when Okfs.Okfs_OID in (5,4,10,9) and PassportTokenLpuHead.PassportToken_tid is
NULL then Okfs.Okfs_OID else null end as Okfs_Code,
Okfs.Okfs_Name, --Name Имя
OkfsParent.Okfs_OID as Okfs_Parent, --Parent Родитель
```

-- Организационно-правовая форма

```
Okopf.Okopf_OID as Okopf_Code, --ID Идентификатор
Okopf.Okopf_Name, --Name Имя
OkopfParent.Okopf_OID as Okopf_Parent, --Parent Родитель
```

-- Адрес

```
Address.Address_id,
```



Address.AddressType\_id, --ID long Идентификатор  
 Address.Address\_Zip, --PostIndex int Почтовый индекс  
 LOWER(Address.KLArea\_AOID) as KLArea\_AOID, -- код территории по ФИАС  
 LOWER(Address.KLStreet\_AOID) as KLStreet\_AOID, -- код улицы по ФИАС  
 convert(bigint, null) as houseid, -- Идентификатор дома по ФИАС

Address.KLRgn\_id,  
 Address.KLArea\_id,  
 Address.AddressType\_Name, --Name string Имя  
 Address.Area\_Prefix,  
 Address.Street\_Prefix,  
 Address.Address\_Flat, --Квартира  
 Address.Address\_Corpus, --Корпус  
 Address.Address\_House, --Дом  
 Address.Address\_KLAdr, --КЛАДР Идентификатор  
 Address.Address\_Level, --Уровень  
 Address.Street\_Name,  
 Address.KLArea\_Name,  
 STR(mainbuild.LpuBuildingPass\_CoordLat,9,6) as latitude,  
 STR(mainbuild.LpuBuildingPass\_CoordLong,9,6) as longitude,

Address.Address\_Parent, --Родитель  
 convert(bigint, null) as Address\_ParentType, --Тип родителя

0 as deleted, --Флаг удаления  
 convert(datetime, null) as deleteDate,  
 convert(varchar(100), null) as deleteReason,

convert(int, dbo.getRegion()) as region,  
 Lpu.Lpu\_insDT,  
 Lpu.Lpu\_updDT,  
 Lpu.Lpu\_begDate,  
 Lpu.Lpu\_endDate,  
 Lpu.Lpu\_Founder as founder,

convert(bigint, 2) as medicalSubjectId, -- 2 = Медицинская организация  
 Okved.Okved\_Name as ProfileLpu, -- Профиль учреждения

case when Lpu.LpuOwnership\_id = 3 then null else MOAreaFeature.MOAreaFeature\_code end  
 as TerritoryId,  
 --case when Lpu.LpuOwnership\_id = 3 then null else case when LpuLevel.LpuLevel\_code  
 between 1 and 3 then LpuLevel.LpuLevel\_code else 4 end end as LevelId,  
 case when Lpu.LpuOwnership\_id = 3 then null else LpuLevel.LpuLevel\_code end as LevelId,

org.Org\_OKATO as OKATO, --okato  
 org.Org\_OKDP as OKDP, --okdp  
 Okfs.Okfs\_code as Okfs, --okfs,  
 Okopf.Okopf\_code as Okopf, --okopf  
 org.Org\_OKPO as OKPO, --okpo  
 Okved.Okved\_code as Okved, --okved

--Тип учреждения  
 case when Lpu.LpuOwnership\_id = 3 then null else  
 isnull(LpuPmuTypeParent.LpuPmuType\_Code, LpuPmuType.LpuPmuType\_Code) end as  
 InstTypeParent\_Code, -- Идентификатор вида деятельности корневые fed.LpuPmuType  
 LpuPmuTypeParent.LpuPmuType\_Name as InstType\_Name\_Parent, --Parent string Родитель  
 case when Lpu.LpuOwnership\_id = 3 or LpuPmuTypeParent.LpuPmuType\_Code is NULL then  
 null else LpuPmuType.LpuPmuType\_Code end as InstType\_Code, -- Идентификатор  
 профиля деятельности дочерние fed.LpuPmuType  
 LpuPmuType.LpuPmuType\_Name as InstType\_Name, --Name string Имя

Lpu.Lpu\_RegDate as RegDate, --RegDate DateTime Дата регистрации  
 Lpu.Lpu\_DocReg as RegDoc, --RegDoc string Документ о регистрации  
 Lpu.Lpu\_RegNum as RegNumber, --RegNumber string Регистрационный номер

```

OrgRegistration.Org_Name as RegOrgName --RegOrgName string Наименование
регистрирующего органа

from dbo.v_Lpu Lpu with (nolock)
inner join dbo.v_Org Org with (nolock) on Lpu.Org_id = Org.Org_id
inner join fed.v_PassportToken PassportToken with (nolock) on Lpu.Lpu_id =
PassportToken.Lpu_id
left join fed.v_PasportMO PassportMO with (nolock) on Lpu.Lpu_id = PassportMO.Lpu_id
left join fed.v_PassportToken PassportTokenLpuHead with (nolock) on PassportMO.Lpu_gid =
PassportTokenLpuHead.Lpu_id
left join fed.v_DepartAffilType DepartAffilType with (nolock) on PassportMO.DepartAffilType_id
= DepartAffilType.DepartAffilType_id
left join fed.v_DepartAffilType DepartAffilTypeParent with (nolock) on
DepartAffilType.DepartAffilType_pid = DepartAffilTypeParent.DepartAffilType_id
left join dbo.v_Okfs Okfs with (nolock) on Org.Okfs_id = Okfs.Okfs_id
left join dbo.v_Okfs OkfsParent WITH (NOLOCK) ON Okfs.Okfs_pid = OkfsParent.Okfs_id
left join dbo.v_Okopf Okopf with (nolock) on Org.Okopf_id = Okopf.Okopf_id
left join dbo.v_Okopf OkopfParent WITH (NOLOCK) ON Okopf.Okopf_pid =
Okopfparent.Okopf_id
left join fed.v_LpuPmuType LpuPmuType with (nolock) on LpuPmuType.LpuPmuType_id =
Lpu.LpuPmuType_id
left join fed.v_LpuPmuType LpuPmuTypeParent with (nolock) on
LpuPmuType.LpuPmuType_pid = LpuPmuTypeParent.LpuPmuType_id
left join fed.MOAreaFeature MOAreaFeature with (nolock) on Lpu.MOAreaFeature_id =
MOAreaFeature.MOAreaFeature_id
left join fed.LpuLevel LpuLevel with (nolock) on PassportMO.LpuLevel_id =
LpuLevel.LpuLevel_id

left join dbo.v_LpuBuildingPass mainbuild with (NOLOCK) on mainbuild.LpuBuildingPass_id =
Lpu.LpuBuildingPass_mid

-- Адреса
outer apply (select top 1
Address_id,
KLRgn_id,
PArea.KLArea_id,
case when Address_id = Org.UAddress_id then 1 else 2 end as AddressType_id,
case when Address_id = Org.UAddress_id then 'Юридический' else 'Фактический' end as
AddressType_Name,
rtrim(ltrim(PAddress.Address_House)) as Address_House,
rtrim(ltrim(PAddress.Address_Zip)) as Address_Zip,
rtrim(ltrim(PAddress.Address_Flat)) as Address_Flat,
rtrim(ltrim(PAddress.Address_Corpus)) as Address_Corpus,
PAddress.KLADR_Code as Address_Kladr,
5 as Address_Level,
PStreet.KLStreet_Name as Street_Name,
PArea.KLArea_Name as KLArea_Name,
PArea.KLAdr_Code as Address_Parent,
PArea.KLArea_AOID, PStreet.KLStreet_AOID,
SStreet.KLSocr_Nick as Street_Prefix, SArea.KLSocr_Nick as Area_Prefix
from dbo.v_Address_KLAdr PAddress with (nolock)
left join dbo.v_KLStreet PStreet with (nolock) on PAddress.KLStreet_id = PStreet.KLStreet_id
left join dbo.v_KLSocr SStreet with (nolock) on SStreet.KLSocr_id = PStreet.KLSocr_id
left join dbo.v_KLArea PArea with (nolock) on PStreet.KLArea_id = PArea.KLArea_id
left join dbo.v_KLSocr SArea with (nolock) on SArea.KLSocr_id = PArea.KLSocr_id
where Address_id in (Org.UAddress_id)
) Address

-- Регистрационные данные
left join dbo.v_Org OrgRegistration with (nolock) on Lpu.Org_id = OrgRegistration.Org_id
-- Классификационные данные
left join dbo.v_Okved Okved with (nolock) on org.Okved_id = Okved.Okved_id

```

```

where (1 = 1)
and PassportToken.PassportToken_tid is not null

-- Даты закрытия
and isnull(Lpu_begDate, dbo.tzGetDate()) <= dbo.tzGetDate() and (Lpu_endDate >
dbo.tzGetDate() or Lpu_endDate is null)
and PassportToken.PassportToken_tid = @oid

```

6.2.7 Скрипт get\_Buildings.sql для списка зданий, экспортируемых в ФРМО:

-- Ид МО

```

declare @lpu_id bigint = 13101706

select
LpuBuildingPass.LpuBuildingPass_id,
LpuBuildingPass.MOArea_id,
Address.Address_id,
Lpu.Lpu_id as Lpu_Id,
PassportToken.PassportToken_tid as oid,
LpuBuildingPass.LpuBuildingPass_id as id,
LpuBuildingPass.LpuBuildingPass_Name AS buildName,
YEAR(LpuBuildingPass.LpuBuildingPass_YearBuilt) AS buildYear,--Дата постройки
LpuBuildingPass.LpuBuildingPass_Floors AS floorCount,--Этажность
CASE when ISNULL(LpuBuildingPass.LpuBuildingPass_IsBuildEmerg,0) = 2 then 'True' else
'False' end as hasTrouble,--Находится в аварийном состоянии
LpuBuildingPass.LpuBuildingPass_insDT,
LpuBuildingPass.LpuBuildingPass_updDT,
LpuBuildingPass.LpuBuildingPass_BuildingIdent,

-- Адрес
Address.Address_Zip, --PostIndex int Почтовый индекс
LOWER(Address.KLArea_AOID) as KLArea_AOID, -- код территории по ФИАС
LOWER(Address.KLStreet_AOID) as KLStreet_AOID, -- код улицы по ФИАС
convert(varchar(36), null) as houseid, -- Идентификатор дома по ФИАС
Address.KLRgn_id,
Address.Area_Prefix,
Address.Street_Prefix,
Address.Address_Flat, --Квартира
Address.Address_Corpus, --Корпус
Address.Address_House, --Дом
Address.Address_KLAdr, --КЛАДР Идентификатор
Address.Address_Level, --Уровень
Address.Street_Name,
Address.KLArea_Name,
Address.KLArea_id, Address.KLStreet_id,
STR(LpuBuildingPass.LpuBuildingPass_CoordLat,9,6) as latitude,
STR(LpuBuildingPass.LpuBuildingPass_CoordLong,9,6) as longitude

from dbo.v_Lpu Lpu with (nolock)
inner join dbo.v_Org Org with (nolock) on Lpu.Org_id = Org.Org_id
inner join fed.v_PassportToken PassportToken with (nolock) on Lpu.Lpu_id =
PassportToken.Lpu_id

inner JOIN dbo.v_LpuBuildingPass AS LpuBuildingPass with (NOLOCK) ON
LpuBuildingPass.Lpu_id = Lpu.Lpu_id
LEFT JOIN fed.v_MOArea AS MOArea with (NOLOCK) ON LpuBuildingPass.MOArea_id =
MOArea.MOArea_id
left join nsi.v_OKATO nsiOKATO with (nolock) on nsiOKATO.OKATO_id = MOArea.OKATO_id
left join dbo.v_Address_KLAdr MOAreaAddress with (NOLOCK) ON MOArea.Address_id =
MOAreaAddress.Address_id

```

```

outer apply (select top 1
--coord.KLHouseCoords_id,
Address_id,
KLRgn_id,
rtrim(ltrim(PAddress.Address_House)) as Address_House,
rtrim(ltrim(PAddress.Address_Zip)) as Address_Zip,
rtrim(ltrim(PAddress.Address_Flat)) as Address_Flat,
rtrim(ltrim(PAddress.Address_Corpus)) as Address_Corpus,
PAddress.KLADR_Code as Address_Kladr,
5 as Address_Level,
PStreet.KLStreet_Name as Street_Name,
PArea.KLArea_Name as KLArea_Name,
PArea.KLAdr_Code as Address_Parent,
PArea.KLArea_AOID, PStreet.KLStreet_AOID,
SStreet.KLSocr_Nick as Street_Prefix, SArea.KLSocr_Nick as Area_Prefix
, PStreet.KLArea_id, PAddress.KLStreet_id
from dbo.v_Address_KLAdr PAddress with (nolock)
left join dbo.v_KLStreet PStreet with (nolock) on PAddress.KLStreet_id = PStreet.KLStreet_id
left join dbo.v_KLSocr SStreet with (nolock) on SStreet.KLSocr_id = PStreet.KLSocr_id
left join dbo.v_KLArea PArea with (nolock) on PStreet.KLArea_id = PArea.KLArea_id
left join dbo.v_KLSocr SArea with (nolock) on SArea.KLSocr_id = PArea.KLSocr_id
where PAddress.Address_id = MOArea.Address_id
) Address
where (1 = 1)
and LpuBuildingPass.LpuBuildingPass_CoordLat is not NULL
and LpuBuildingPass.LpuBuildingPass_CoordLong is not NULL
and PassportToken.PassportToken_tid is not null
-- Даты закрытия
and isnull(Lpu_begDate, dbo.tzGetDate()) <= dbo.tzGetDate() and (Lpu_endDate >
dbo.tzGetDate() or Lpu_endDate is null)
and Lpu.Lpu_id = @lpu_id
order by Address.KLArea_AOID, Address.KLStreet_AOID

```

6.2.8 Скрипт get\_Departments.sql для списка подразделений и отделений, экспортируемых в ФРМО:

**-- Ид МО**

```

declare @lpu_id bigint = 13101703
declare @ndate datetime
set @ndate = GetDate()
declare @isIdRegional bit
set @isIdRegional = convert(bit, 1)
select
regs.rid,
LpuUnit.LpuUnit_id,
LpuBuilding.LpuBuilding_id,
LpuBuilding.LpuBuilding_IsExport,
LpuUnit.LpuUnit_IsNotFRMO,
LpuUnit.LpuUnit_updDT,
LpuUnit.LpuUnit_id,
LpuUnit.LpuUnit_FRMOUnitID as oid,
LpuUnit_FRMOid as fedid,
Lpu.Lpu_id as Lpulid,
LpuUnit.UnitDepartType_fid,

```

LpuUnit.LpuUnit\_Name,  
 REPLACE(REPLACE(REPLACE(REPLACE(LpuUnit.LpuUnit\_Phone,' ',''),'-',''),(','),',')) as  
 LpuUnit\_Phone,  
 UnitType.LpuUnitType\_Code,  
 UnitType.LpuUnitType\_Nick,  
  
 PassportToken.PassportToken\_tid as LpuIdentificator,  
 LpuBuilding.LpuBuilding\_id as LpuBuilding\_id,  
 LpuBuilding.LpuBuilding\_Code as LpuBuilding\_Code,  
 LpuBuilding.LpuBuilding\_Name as LpuBuilding\_Name,  
 LpuBuilding.LpuBuildingType\_id,  
 BuildingType.LpuBuildingType\_Name,  
  
 LpuUnit.LpuUnitProfile\_fid as departKindId,  
 LpuUnit.UnitDepartType\_fid as departTypeId,  
 CASE when ISNULL(LpuUnit.LpuUnit\_IsStandalone,0) = 2 then 'True' else 'False' end as  
 separateDepart,--Обособленное подразделение  
 CASE when ISNULL(LpuUnit.LpuUnit\_IsStandalone,0) = 2 then  
 UnitBuilding.LpuBuildingPass\_Name else NULL end as UnitBuildingName,  
 LpuUnit.LpuUnit\_endDate,  
  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 1 then CASE when  
 ISNULL(LpuUnit.LpuUnit\_IsHomeVisit,0) = 2 then 'True' else 'False' end else NULL end as  
 visitHome,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 1 then att.quant else NULL end as  
 attached,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 1 then att.child\_quant else NULL end as  
 child\_attached,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 1 then CASE when  
 ISNULL(reg.LpuSection\_PlanVisitShift,0) = 0 then 10 else reg.LpuSection\_PlanVisitShift end  
 else NULL end as visitPerShift,--Плановое число посещений в смену  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 1 then LpuSection.FRMPSubdivision\_id  
 else NULL end as RegRoomSubdivisionId,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 1 and LpuSection.FRMPSubdivision\_id  
 is not NULL then regs.roomCount else NULL end as RegRoomCount,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 1 and LpuSection.FRMPSubdivision\_id  
 is not NULL then regs.building else NULL end as RegRoomBuilding,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 1 and LpuSection.FRMPSubdivision\_id  
 is not NULL then regs.sub\_oid else NULL end as RegRoomOID,  
  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) in (3,4) then  
 LpuSection.FRMPSubdivision\_id else NULL end as LabRoomSubdivisionId,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) in (3,4) and  
 LpuSection.FRMPSubdivision\_id is not NULL then labs.resCount else NULL end as  
 LabRoomCount,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) in (3,4) and  
 LpuSection.FRMPSubdivision\_id is not NULL then labs.resCount else NULL end as  
 LabRoomRes,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) in (3,4) and  
 LpuSection.FRMPSubdivision\_id is not NULL then labs.building else NULL end as  
 LabRoomBuilding,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) in (3,4) and  
 LpuSection.FRMPSubdivision\_id is not NULL then labs.sub\_oid else NULL end as  
 LabRoomOID,  
  
 LpuSection.LpuSection\_id,  
 LpuSection.LpuSection\_Code,  
 LpuSection.LpuSection\_IsNotFRMO,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 2 then LpuSection.LpuSection\_Name  
 else NULL end as LpuSection\_Name,  
 CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 2 and UnitType.LpuUnitType\_Code = 2  
 then CASE when ISNULL(LpuUnit.LpuUnit\_IsCMP,0) = 2 then 'True' else 'False' end else NULL  
 end as ambulance,

```

CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 2 then CASE when
UnitType.LpuUnitType_Code = 2 then 1 when UnitType.LpuUnitType_Code in (3,5) then 2 else
3 end else NULL end as hospitalModeld,
CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 2 and UnitType.LpuUnitType_Code in
(2,3,4,5) then case when isnull(@isIdRegional,0) = 1 then LpuSection.LpuSection_id else
LpuSection.LpuSection_FRMOSectionId end else NULL end as sub_id,
CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 2 and UnitType.LpuUnitType_Code in
(2,3,4,5) then LpuSection.LpuSection_FRMOBuildingOid else NULL end as sub_oid,
CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 2 and UnitType.LpuUnitType_Code in
(2,3,4,5) then LpuSection.FRMPSsubdivision_id else NULL end as subdivisionId,
CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 2 and UnitType.LpuUnitType_Code in
(2,3,4,5) then LpuUnit.LpuUnit_endDate else NULL end as liquidationDate,
CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 2 and UnitType.LpuUnitType_Code in
(2,3,4,5) then SectionBuilding.LpuBuildingPass_Name else NULL end as SectionBuildingName,
CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 2 and UnitType.LpuUnitType_Code in
(2,3,4,5) then LpuSection.LpuSection_disDate else NULL end as LpuSection_disDate,

CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 2 and UnitType.LpuUnitType_Code in (4)
then bedstatal.LpuSectionBedState_Fact else NULL end as homeBedCount,

CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 5 then LpuSection.LpuSection_KolAmbul
else NULL end as brigadeCount,
CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 5 then LpuSection.LpuSection_PlanTrip
else NULL end as departurePerShift,
CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 5 then
UnitBuilding.LpuBuildingPass_Name else NULL end as SMPBuildingName,

CASE when ISNULL(LpuUnit.UnitDepartType_fid,0) = 6 then
SectionBuilding.LpuBuildingPass_Name else NULL end as AdminBuildingName, -- здание
админ-хоз подразделения

LpuSection.LpuCostType_id,
LpuSection.LpuSectionProfile_Code,
LpuSection.LpuSectionProfile_Name

from
dbo.v_Lpu Lpu with (nolock)
inner join dbo.v_Org Org with (nolock) on Lpu.Org_id = Org.Org_id
inner join fed.v_PassportToken PassportToken with (nolock) on Lpu.Lpu_id =
PassportToken.Lpu_id
inner join dbo.v_LpuSection LpuSection with (nolock) ON LpuSection.Lpu_id = Lpu.Lpu_id
inner join dbo.v_LpuUnit LpuUnit with (nolock) ON ISNULL(LpuSection.LpuUnit_sid,
LpuSection.LpuUnit_id) = LpuUnit.LpuUnit_id
inner join dbo.v_LpuUnitType UnitType with (nolock) ON LpuUnit.LpuUnitType_id =
UnitType.LpuUnitType_id
inner join dbo.v_LpuBuilding LpuBuilding with (nolock) ON LpuUnit.LpuBuilding_id =
LpuBuilding.LpuBuilding_id
and (
-- ПРИЗНАК группы подразделений "Экспорт в ПМУ" включен
LpuBuilding.LpuBuilding_IsExport = 2 and
isnull(LpuBuilding_begDate, dbo.tzGetDate()) <= dbo.tzGetDate() and (LpuBuilding_endDate >
dbo.tzGetDate() or LpuBuilding_endDate is null))
inner join dbo.v_LpuBuildingType BuildingType with (nolock) ON
BuildingType.LpuBuildingType_id = LpuBuilding.LpuBuildingType_id

left join dbo.v_LpuBuildingPass UnitBuilding with (nolock) ON LpuUnit.LpuBuildingPass_id =
UnitBuilding.LpuBuildingPass_id
left join dbo.v_LpuBuildingPass SectionBuilding with (nolock) ON
LpuSection.LpuBuildingPass_id = SectionBuilding.LpuBuildingPass_id

outer apply (
select sum(BedState.LpuSectionBedState_Fact) as LpuSectionBedState_Fact
from dbo.v_LpuSectionBedState (NOLOCK) AS BedState
where BedState.LpuSection_id = LpuSection.LpuSection_id

```

```

and BedState.LpuSectionBedState_begDate < @ndate and
BedState.LpuSectionBedState_endDate is null
) bedstatal

outer apply (
select sum(LpuSec.LpuSection_PlanVisitShift) as LpuSection_PlanVisitShift
from dbo.v_LpuSection LpuSec with (nolock)
where LpuSec.LpuUnit_id = LpuUnit.LpuUnit_id
) reg

outer apply (
select max(LpuSec.LpuSection_id) as rid, count(distinct LpuSec.LpuSection_id) as roomCount,
max(SecBuilding.LpuBuildingPass_Name) as building,
max(LpuSec.LpuSection_FRMOBuildingOid) as sub_oid
from dbo.v_LpuSection LpuSec with (nolock)
left join dbo.v_LpuBuildingPass SecBuilding with (nolock) ON LpuSec.LpuBuildingPass_id =
SecBuilding.LpuBuildingPass_id
where LpuSec.LpuUnit_id = LpuUnit.LpuUnit_id and ISNULL(LpuUnit.UnitDepartType_fid,0) = 1
and LpuSection.FRMPSubdivision_id = LpuSec.FRMPSubdivision_id
and ISNULL(LpuSec.LpuSection_IsNotFRMO, 1) = 1
) regs

outer apply (
select sum(LpuSec.LpuSection_PlanResShift) as resCount,
max(SecBuilding.LpuBuildingPass_Name) as building,
max(LpuSec.LpuSection_FRMOBuildingOid) as sub_oid
from dbo.v_LpuSection LpuSec with (nolock)
left join dbo.v_LpuBuildingPass SecBuilding with (nolock) ON LpuSec.LpuBuildingPass_id =
SecBuilding.LpuBuildingPass_id
where LpuSec.LpuUnit_id = LpuUnit.LpuUnit_id and ISNULL(LpuUnit.UnitDepartType_fid,0) in
(3,4) and LpuSection.FRMPSubdivision_id = LpuSec.FRMPSubdivision_id
and ISNULL(LpuSec.LpuSection_IsNotFRMO, 1) = 1
) labs

outer apply (
select count(distinct pc.Person_id) as quant
, count (distinct case when dbo.Age(ISNULL(pall.PersonBirthDay_BirthDay, @ndate), @ndate)
< 17 then pc.Person_id else NULL end) as child_quant
from PersonCard pc with (nolock)
inner join PersonState pall with (nolock) on pall.Person_id = pc.Person_id
inner join Person pr with (nolock) on pall.Person_id = pr.Person_id and pr.Server_id=0
where pc.Lpu_id=Lpu.Lpu_id and pc.LpuAttachType_id=1
and pc.PersonCard_begDate < @ndate and pc.PersonCard_endDate is null
) att

where (1 = 1)

and LpuSection.LpuSection_pid is NULL -- подразделения не включаем
and PassportToken.PassportToken_tid is not null

and LpuUnit.LpuUnit_FRMOUnitID is not null
and LpuUnit.LpuUnitProfile_fid is not NULL
and ISNULL(LpuUnit.LpuUnit_IsNotFRMO, 1) = 2 -- признак "не передавать на ФРМО" для
подразделения не включен
and ISNULL(LpuSection.LpuSection_IsNotFRMO, 1) = 1 -- признак "не передавать на ФРМО"
для отделения не включен

and (LpuSection.FRMPSubdivision_id is not NULL or LpuSection.LpuSection_id is NULL or not
(ISNULL(LpuUnit.UnitDepartType_fid,0) = 2 and UnitType.LpuUnitType_Code in (2,3,5)))

and (not (ISNULL(SectionBuilding.LpuBuildingPass_Name,"") = "") or (LpuSection.LpuSection_id
is null)
or not (ISNULL(LpuUnit.UnitDepartType_fid,0) = 2 and UnitType.LpuUnitType_Code in (2,3,5)))

```

```
and (not (ISNULL(UnitBuilding.LpuBuildingPass_Name, '') = '') or not
(ISNULL(LpuUnit.LpuUnit_IsStandalone, 0) = 2))
```

```
-- Даты закрытия
```

```
and isnull(Lpu_begDate, dbo.tzGetDate()) <= dbo.tzGetDate() and (Lpu_endDate >
dbo.tzGetDate() or Lpu_endDate is null)
and Lpu.Lpu_id = @lpu_id
```

6.2.9 Скрипт get\_Staff.sql для списка штатных расписаний, экспортируемых в ФРМО:  
select

```
LpuStaff.LpuStaff_Num as staffNum
, LpuStaff.LpuStaff_id
, max(isnull(LpuStaff.LpuStaff_ApprovalDT, Lpu.Lpu_begDate)) as staffCreateDate
, max(isnull(LpuStaff.LpuStaff_begDate, Lpu.Lpu_begDate)) as beginDate
, max(ISNULL(LpuStaff.LpuStaff_endDate, '20991231')) as endDate
, Lpu.Lpu_id as LpuId
, LpuUnit.LpuUnit_FRMOUnitID as nrPmuDepartId
, max(LpuUnit.UnitDepartType_fid) as UnitDepartType_fid

, CASE when ISNULL(LpuUnit.UnitDepartType_fid, 0) = 2 and UnitType.LpuUnitType_Code in
(2,3,5) then LpuSection.LpuSection_FRMOBuildingOid else NULL end as
nrPmuDepartHospitalSubdivisionId
, FRMPPost.frmr_id as post_id
, max(post.id) as postid
, max(st.Rate) as totalRate
, max(FRMPPost.id) as post_code, max(FRMPPost.name) as post_name
, max(PassportToken.PassportToken_tid) as LpuIdenticator
, LpuBuilding.LpuBuilding_id as LpuBuilding_id
, max(LpuUnit.LpuUnit_Name) as LpuUnit_Name
, max(LpuSection.LpuSection_Name) as LpuSection_Name
, max(st.id) as staff_id
, max(LpuUnit.LpuUnit_id) as LpuUnit_id
, max(LpuBuilding.LpuBuilding_id) as LpuBuilding_id
, max(LpuBuilding.LpuBuilding_name) as LpuBuilding_name

from dbo.v_Lpu Lpu with (nolock)
inner join dbo.v_Org Org with (nolock) on Lpu.Org_id = Org.Org_id
left join dbo.v_LpuStaff LpuStaff with (nolock) on Lpu.Lpu_id = LpuStaff.Lpu_id
inner join fed.v_PassportToken PassportToken with (nolock) on Lpu.Lpu_id =
PassportToken.Lpu_id
inner join dbo.v_LpuBuilding LpuBuilding with (nolock) ON LpuBuilding.Lpu_id = Lpu.Lpu_id
inner join dbo.v_LpuUnit LpuUnit with (nolock) ON LpuUnit.LpuBuilding_id =
LpuBuilding.LpuBuilding_id
inner join dbo.v_LpuUnitType UnitType with (nolock) ON LpuUnit.LpuUnitType_id =
UnitType.LpuUnitType_id
inner join dbo.v_LpuSection LpuSection with (nolock) ON LpuSection.LpuUnit_id =
LpuUnit.LpuUnit_id

inner join persis.v_Staff st with (nolock) ON st.LpuSection_id = LpuSection.LpuSection_id
and st.LpuBuilding_id = LpuBuilding.LpuBuilding_id and st.LpuUnit_id = LpuUnit.LpuUnit_id
inner join persis.Staff st_ with (nolock) ON st.id = st_.id
inner join persis.post post with (nolock) on post.id = st_.Post_id
inner join persis.FRMPPost FRMPPost with (nolock) on FRMPPost.id = post.frmr_id

where (1 = 1)

and PassportToken.PassportToken_tid is not null
and LpuUnit.LpuUnit_FRMOUnitID is not NULL

and ISNULL(LpuSection.LpuSection_IsNotFRMO, 1) = 1 -- признак "не передавать на ФРМО"
не включен
```



and (LpuSection.LpuSection\_FRMOBuildingOid is not NULL or not  
 (ISNULL(LpuUnit.UnitDepartType\_fid,0) = 2 and UnitType.LpuUnitType\_Code in (2,3,5)))  
 and (LpuBuilding.LpuBuilding\_IsExport = 2 and isnull(LpuBuilding\_begDate, dbo.tzGetDate())  
 <= dbo.tzGetDate() and (LpuBuilding\_endDate > dbo.tzGetDate() or LpuBuilding\_endDate is  
 null))  
 and st.Rate > 0.2 and dbo.tzGetDate() between st.BeginDate and isnull(st.EndDate,  
 '21000101')  
 -- Даты актуальности  
 and isnull(Lpu.Lpu\_begDate, dbo.tzGetDate()) <= dbo.tzGetDate() and (Lpu.Lpu\_endDate >  
 dbo.tzGetDate() or Lpu.Lpu\_endDate is null)

– **Ид МО**

and Lpu.Lpu\_id = 10010833

group by Lpu.Lpu\_id, LpuStaff.LpuStaff\_Num, LpuStaff.LpuStaff\_id,  
 LpuBuilding.LpuBuilding\_id, LpuUnit.LpuUnit\_FRMOUnitID, CASE when  
 ISNULL(LpuUnit.UnitDepartType\_fid,0) = 2 and UnitType.LpuUnitType\_Code in (2,3,5) then  
 LpuSection.LpuSection\_FRMOBuildingOid else NULL end, FRMPPost.fmr\_id

order by Lpu.Lpu\_id, LpuStaff.LpuStaff\_Num, LpuStaff.LpuStaff\_id, LpuBuilding.LpuBuilding\_id,  
 LpuUnit.LpuUnit\_FRMOUnitID, CASE when ISNULL(LpuUnit.UnitDepartType\_fid,0) = 2 and  
 UnitType.LpuUnitType\_Code in (2,3,5) then LpuSection.LpuSection\_FRMOBuildingOid else  
 NULL end, FRMPPost.fmr\_id

## 7 ФРМР. Отчет о тестировании и массовая выгрузка. Подробная инструкция.

- [Общие сведения](#)
- [Подготовка.](#)
- [Проверка ФРМР.](#)
  - а. [Создание врача.](#)
    - [Заполнение персональных данных человека.](#)
    - [Заполнение данных об образовании](#)
    - [Заполнение данных о месте работы.](#)
- [Составление отчета о тестировании.](#)
  - [Порядок выполнения методов](#)
- [Массовая выгрузка.](#)
  - [Проведение Массовой выгрузки.](#)
  - [Составление отчета](#)
- [SQL Scripts](#)
  - [Проверка и подготовка Сессии.](#)
  - [Запуск методов ФРМР:](#)
  - [Работа с данными](#)
  - [Проверка данных Врача](#)

## 7.1 Общие сведения

Массовая выгрузка ФРМР подразумевает под собой передачу данных сразу по нескольким врачам. Как правило, для тестирования создаются 100 тестовых людей, из которых 20 людей являются врачами, и 80 - массовой. Проводится массовая выгрузка только после успешно проведенной выгрузки ФРМО, т.к при выгрузке так же передается значение OID структурного подразделения. Так же следует учитывать, что при выгрузке ФРМО так же должны быть выгружены штатное расписание и должности, при этом количество ставок на должности должно быть не менее 100.

Сама массовая выгрузка делится на 3 этапа:

1. Подготовка
2. Проверка ФРМР.
3. Массовая выгрузка

## 7.2 Подготовка.

Перед проведение проведением массовой выгрузки и началом работы с новым регионом необходимо будет выполнить следующие шаги:

1. Установить [DBeaver](#) или любой аналогичный клиент для работы с БД. [Подробнее, как установить DBeaver](#)
2. Запросить доступ к БД региона, на котором проводится массовая выгрузка. При этом:
  - a. Если выгрузка проводится с тестового сервера региона, то доступ необходим к тестовой БД
  - b. Если выгрузка проводится с рабочего сервера региона, то доступ необходим к рабочей БД.
  - c. В обоих случаях, доступ к БД необходим с правами на чтение, редактирование, добавление, удаление
3. Подключиться через DBeaver или аналогичный клиент для работы с БД к той БД, к которой был предоставлен доступ. [Подробнее о том, как настроить подключение к DBeaver](#)

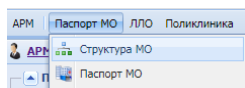
После выполнения этих шагов можно переходить к дальнейшим этапам.

## 7.3 Проверка ФРМР.

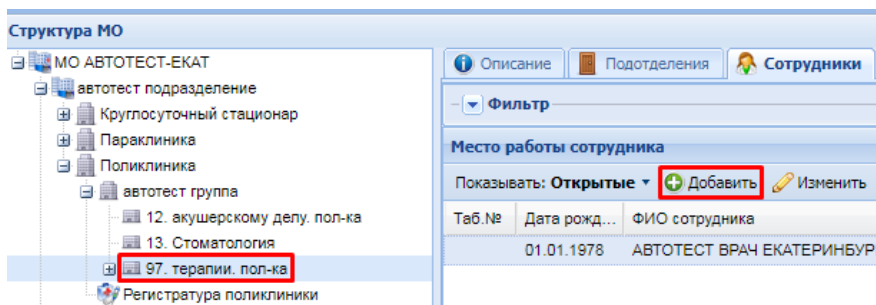
Данный необходим для проверки работы ФРМР и подразумевает под собой создание 1 тестового врача, выгрузка его на портал и составление отчета о результатах.

### 7.3.1 Создание врача.

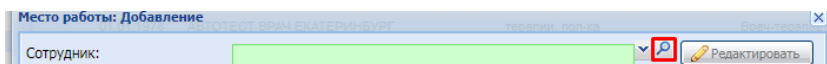
При проверке ФРМР создается только 1 врач. Для этого, зайти и авторизоваться в РМИС Промед. В верхнем меню найти кнопку "Паспорт МО", в выпадающем меню выбрать "Структура МО":



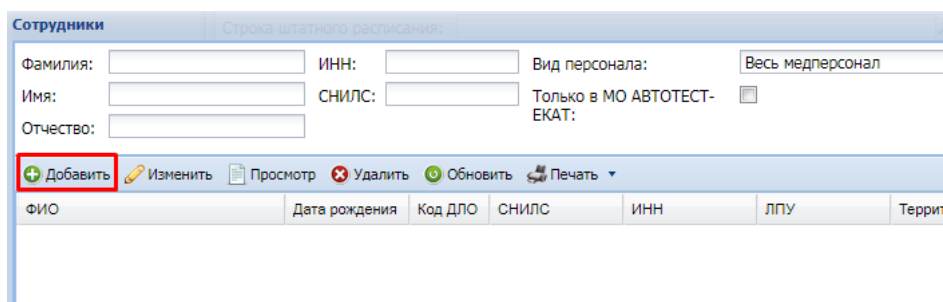
Раскрываем дерево МО и выбираем подразделение МО. Нажимаем кнопку "Добавить"



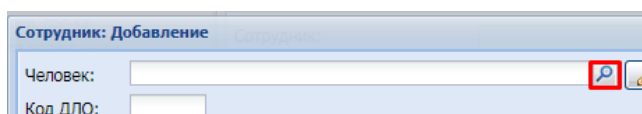
В открывшейся форме "Место работы: Добавление" нажимаем на изображение лупы напротив поля сотрудник:



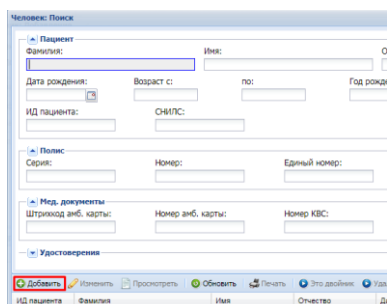
На форме "Сотрудники" снова нажимаем добавить:



На форме "Сотрудники: Добавление" ищем изображение лупы напротив поля "Человек" и нажимаем на него:



И последним шагом в этой цепочке - нажать на кнопку "Добавить" на форме "Человек: Поиск"



### 7.3.1.1 Заполнение персональных данных человека.

На форме "Человек: Поиск" заполняем следующие поля:

1. ФИО человека: ТЕСТ ФРМР ОДИН
2. Дата рождения: любая дата, например 01.01.1980
3. Телефон человека: любой телефон, например: +71111111111
4. Пол человека.
5. СНИЛС человека: СНИЛС обязательно должен быть валидным, и не должен использоваться ранее для выгрузки на этом регионе. Валидность СНИЛС проверяется по [формуле](#), или можно воспользоваться [генератором](#) (СНИЛС в генераторе получаются почти всегда валидными. Но из-за почти всегда - лучше проверять)

6. Социальный статус.
7. Адрес проживания и Адрес регистрации. Лучший вариант при заполнении - указывать ул.Ленина, номер дома 999, квартира 999,
8. Заполняются данные полиса. Важными при заполнении параметрами будут - номер, который тоже должен быть валидным(используем [генератор](#)).
9. Заполняются паспортные данные. При заполнении серии и номера необходимо учитывать, что бы серия и номер не пересекались с серией и номером реально существующих людей. Например, если на регионе есть люди, серия и номер которых начинаются на 5\*\*\* и 1\*\*\*\*\*, то серия и номер для ПД создаваемого челока не должны начинаться с указанных цифр. Определить диапазон можно простым SQL запросом в таблицу v\_personstate.
10. Заполняется место работы. Местом работы указывается МО, на которую добавляется врач. Должность должна соответствовать той должности, на которую добавляется врач.
11. ИНН человека. Он тоже должен быть валидным, и высчитывается по [формуле](#). Или используется все тот же генератор.
12. Проверяем еще раз введенные данные и сохраняем человека.

Пример итогового результат заполнения формы "Человек: Добавление":

### 7.3.1.2 Заполнение данных об образовании

После сохранения человека необходимо добавить информацию об образовании врача. Для этого на форме "Сотрудники: Добавление" заполняются:

- На вкладке Квалификационная категория, добавляется 1 квалификационная категория. Специальность выбирается с id = 13. Категория - высшая. Дату прохождения указываем 2017 год

- На вкладке Специальности по диплому. Добавляются специальность по диплому для Высшего и Среднего образования.

- Выбирается учебное заведение: примера нет, на каждом регионе будет какое то свое учебное заведение, которое зависит от справочников на регионе, как давно их обновили, от справочников на портале ФРМО и как давно их обновили.
- Год поступления/год окончания - должен попадать в период работы учебного заведения. Например, если УЗ - работало с 1995 по 2005, то врач должен учиться в период с 2001 по 2004
- Серия диплома - рандомные 4 цифры. **\*ВАЖНО\* ЦИФР ДОЛЖНО БЫТЬ - 4. НЕ 3, НЕ 5, А ИМЕННО 4**
- Номер диплома - рандомные 6(для среднего образования) и рандомные 7(для высшего образования) цифр.
- Дата выдачи диплома. Не должна превышать дату работы УЗ.
- Тип образования - Среднее образования для Среднего образования. Высшее образование - для Высшего образования.
- Специальность - Врач с кодом 15
- Квалификация - Врач общей практики.

Среднее образование	Высшее образование

- На вкладке Послевузовское образование. Создается 1 послевузовское образование и заполняются поля:
  - Выбирается учебное заведение: примера нет
  - Даты начала, окончания, получения диплома. Главное - что бы даты попадали в диапазон работы УЗ
  - Серия и номер диплома - рандомные цифры. Важно, что бы Серия была - не больше и не меньше 4 цифр. Номер диплома - должен быть 5 цифр.
  - Тип образования - аспирантура
  - Специальность - для интернатуры выбираем специальность с кодом 75. Для аспирантуры - Микробиология
  - Ученая степень - Доктор медицинских наук

Пример заполнения Послевузовского образования:

- На вкладке Курсы Переподготовки создаем запись об 1 курсе. Заполняются следующие поля:
  - a. Выбирается учебное заведение: примера нет
  - b. Год прохождения. Так же главное, что бы попадало в период действия УЗ
  - c. Количество часов - 360
  - d. Серия и номер документа. Рандомные цифры, но важно, что бы Серия была не более и не менее 3х цифр, Номер - 4 цифры.
  - e. Дата получения документа.
  - f. Специальность - выбираем специальность с кодом 3200

#### Пример заполнения Курса Переподготовки

- На вкладке Курсы Повышения Квалификации создаем 1 запись о повышении квалификации. Заполняем следующие поля:
  - a. Выбирается учебное заведение: примера нет
  - b. Цикл - указываем 1 цикл
  - c. Год прохождения. И он снова должен попадать в период действия УЗ
  - d. Количество часов. Указываем 240
  - e. Серия и Номер документа. Рандомные цифры, но ВАЖНО - Серия должна быть 3 цифры, не больше и не меньше. Номер документа должен быть больше и не меньше 5 цифр.
  - f. Дата получения документа. Как и ранее, должен попадать в действие УЗ
  - g. Специальность - выбираем специальность с кодом 17

#### Пример заполнения Курса Повышения Квалификации

- На вкладке Сертификаты создается 1 запись о сертификате. Заполняются следующие поля:
  - a. Выбирается учебное заведение: примера нет
  - b. Серия и Номер сертификата. Рандомные цифры, НО! Сертификат должен быть - 1 цифра, Номер Сертификата - 3 цифры.
  - c. Специальность - выбираем специальность с кодом 36.

- d. Дата сдачи экзамена и Дата получения сертификата. Как и все заполняемые ранее даты должна попадать в период работы УЗ

Пример заполнения данных о Сертификате:

- На вкладке Награды. Добавляем 1 награду, заполняем все 4 поля и не важно, какими данными.

Пример заполнения данных о Награде

Дополнительные сведения не заполняются, как и Аккредитация (Федералы убрали необходимость в аккредитации и с 2019 года она ими при выгрузке не принимается).

После добавления всех необходимых данных об образовании сохраняем Врача и переходим к заполнению следующих форм.

### 7.3.1.3 Заполнение данных о месте работы.

При заполнении данных о месте работы Врача учитываем, что у Врача не должно быть других ставок. Заполняются следующие поля:

- Запись на начало - указываем Принят
- Прибыл - Из мед. организации данного субъекта Российской Федерации
- Номер приказа - 1к
- Дата приема - не должна совпадать ни с одной датой, указанной в данных об обучении.

Пример:

- Тип занятия должности - всегда должна быть основная должность!
- Ставка - 1

- Режим работы - работающий на постоянной работе.
- Отношение к военной службе - Военнообязанный

Пример:





## 7.4 Составление отчета о тестировании.

После создания Врача и добавления его в структуру можно приступить непосредственно к выгрузке.

Для этого через DBeaver запускаем последовательно SQL скрипты

1. проверка таблицы для хранения Врача
2. создание таблицы хранения Врача (опционально, если таблицы нет на регионе)
3. добавление данных Врача
4. проверка данных Врача. Если какие то данные отличаются/отсутствуют - исправляем и проверяем еще раз.
5. **Скрипты ниже**
6. комментируем данные Врача (все, кроме person)
7. Запускаем скрипт create
8. Переходим в логи. У каждого сервера свой адрес для доступа к логам. В самих логах необходимо найти папку fmg/дата\_отправки.log. Ищем отправленный запрос. Находим полученный синхронный ответ. Копируем message\_id, и ищем по нему асинхронный ответ. Если асинхронный ответ пришел без ошибок, то сохраняем все запросы и ответы по данному методу. **Важно:** Отправленный запрос, синхронный ответ, асинхронный ответ, асинхронный запрос, каждый должен быть в отдельном xml файле, т.е должно получиться 4 файла. **В итоге:** Должен получиться архив с файлами XML и для каждого запрос/ответа свой XML.

**Пример названия файлов для одного метода.**

 person.create	Документ XML
 person.create.sync_async	Документ XML
 person.create_async	Документ XML
 person.create_sync	Документ XML

9. Если после отправки в ответе были получены ошибки, то внимательно читаем, в чем заключается ошибка и исправляем. При этом у Врача перед следующим запуском необходимо будет изменить СНИЛС и ПД(если ошибка случалась после метода person\_document.create)
10. Переходим в DBeaver, комментируем person и раскомментируем common (Среднее образование)(комментируем и раскомментируем строго по порядку, т.е сначала 1 метод в 1 блоке, потом 1 метод во втором блоке и т.д)
11. Повторяем пункты 6,7,8(в 8 пункте меняется каждый раз метод).



12. Повторяем, пока по всем данным не будет выполнен create
13. ????
14. Profit! Все методы create успешно выполнены, ошибок нет и можно перейти к следующему методу - update
15. Повторяем пункты 6(только будет запуску скрипта update), 7, 8
16. ???
17. Если ошибок не было, то переходим на метод read и снова повторяем пункты 6, 7, 8
18. После выполнения всех read раскоментируем все строки и запускаем delete. При этом по person и qualification

#### 7.4.1 Порядок выполнения методов

№	Название метода
1	create
2	update
3	read
4	delete

## 7.5 Массовая выгрузка.

Для проведения массовой выгрузки необходимо создание данных в соответствии с указаниями в разделе [Создание врача](#) данной инструкции, а также статьи [Описание данных для массовой выгрузки ФРМР](#). Должна быть создана 1 запись врача с указанием данных об образовании и месте работы. Для нового врача должен быть указан СНИЛС и номер документа удостоверяющего личность отличные от ранее введённых в Систему. Массовая выгрузка выполняется запуском сервиса в специальном режиме соответствующим скриптом.

На первом этапе выполняется циклически (20 раз): отправка данных о врачах, затем корректировкой данных врача. При этом создание нового человека выполняется один раз.

Далее выполняется отправка данных о человеке 99 раз по аналогичному принципу (без информации о месте работы и образовании)

При работе сервиса в режиме массовой выгрузки все данные врачей, за исключением СНИЛС, ПД и Серий и Номеров документов одинаковы. Т.е все живут в одной квартире по одному адресу, закончить одно и тоже УЗ по одной и той же специальности, работать в одной и той же МО на одной и тоже ставке.

#### 7.5.1 Проведение Массовой выгрузки.

 Скрипт запуска массовой выгрузки для Postgre.

После успешного завершения всех действий над Врачами массовка раскоментируется и выполняется(Врачей при этом можно закоментировать в \*название\_метода1\* или вовсе удалить). После выполнения person.create проверяем, что у нас при отправке массовой вернулось не менее 80 успешных ответов(т.е не должно быть ошибок).

#### 7.5.2 Составление отчета

Т.к при массовой выгрузке отправляется очень много данных, то ответы в xml не собираются.

Вместо этого фиксируем общее время по МСК старта(т.е когда был запущен первый метод) и общее время завершения(т.е когда все методы выполнили, получили последний ответ и больше ничего запускать не надо).

При запуске метода - фиксируем время по МСК, когда был запущен метод.

## 7.6 SQL Scripts

Список всех используемых в работе SQL Скриптов.

7.6.1 Проверка и подготовка Сессии.

--проверяем, есть ли ФРМР/ФРМО сессия на сервере

```
select * from FRMRIntegrationSession
```

```
select * from FRMOSession
```

----создание сессии при проверке, если ее не нашли.

```
DECLARE @FRMOSessionActionType_id BIGINT,  
        @Error_Code INT,  
        @Error_Message VARCHAR(4000);  
EXEC dbo.p_FRMRIntegrationSession_ins  
@FRMOSession_id = --подставить id найденной/созданной сессии, -- bigint  
@FRMRIntegrationSession_isSuccess = 2, -- bigint  
-- @FRMRSession_Guid = NULL, -- uniqueidentifier  
--@FRMRIntegrationSession_ErrorText = ", -- varchar(200)  
@lpu_id = --подставить lpu_id, -- bigint  
@Lpu_oid = --подставить OID , -- varchar(256)  
@FRMRIntegrationSession_isUsed = 2, -- bigint  
@FRMOSessionActionType_id = 10053, -- bigint  
@pmUser_id = 1,  
@Error_Code = @Error_Code OUTPUT, -- int  
@Error_Message = @Error_Message OUTPUT -- varchar(4000)  
SELECT @Error_Code,@Error_Message
```

7.6.2 Запуск методов ФРМР:

--выполнение create

```
DECLARE  
@Error_Code INT,  
@Error_Message VARCHAR(4000);  
EXEC dbo.p_FRMRIntegrationSession_upd @FRMRIntegrationSession_id = , -- bigint  
@FRMOSession_id = , -- bigint  
@lpu_id = ,  
@Lpu_oid=, -- bigint  
@FRMRIntegrationSession_isUsed=NULL,  
@FRMOSessionActionType_id=10054,  
@pmUser_id = 1  
@Error_Code = @Error_Code OUTPUT, --int  
@Error_Message = @Error_Message OUTPUT --varchar(4000)  
SELECT @Error_Code,@Error_Message
```

--update

```
EXEC dbo.p_FRMRIntegrationSession_upd @FRMRIntegrationSession_id = , -- bigint  
@FRMOSession_id = , -- bigint  
@lpu_id = ,  
@Lpu_oid=, -- bigint  
@FRMRIntegrationSession_isUsed=NULL,  
@FRMOSessionActionType_id=10050,  
@pmUser_id = 1
```

--read

```
EXEC dbo.p_FRMRIntegrationSession_upd @FRMRIntegrationSession_id = , -- bigint  
@FRMOSession_id = , -- bigint
```

```

@lpu_id = ,
@Lpu_oid=, -- bigint
@FRMRIntegrationSession_isUsed=NULL,
@FRMOSessionActionType_id=10053,
@pmUser_id = 1

--delete
EXEC dbo.p_FRMRIntegrationSession_upd @FRMRIntegrationSession_id = , -- bigint
@FRMOSession_id = , -- bigint
@lpu_id = ,
@Lpu_oid=, -- bigint
@FRMRIntegrationSession_isUsed=NULL,
@FRMOSessionActionType_id=10052,
@pmUser_id = 1

```

### 7.6.3 Работа с данными

--Создание таблицы FRMRUpload. в ней хранится Врач

```

CREATE TABLE tmp.FRMRUpload
(
[FRMRUpload_id] [bigint] NOT NULL IDENTITY,
[FRMRUpload_Method] [varchar] (60) COLLATE Cyrillic_General_CI_AS NULL,
[Snils_id] [varchar] (60) NULL,
[FRMRUpload_insDT] [datetime] NULL,
[FRMRUpload_updDT] [datetime] NULL
) ON [PRIMARY]
GO
ALTER TABLE [tmp].[FRMRUpload] ADD CONSTRAINT [pk_FRMRUpload_id] PRIMARY KEY
CLUSTERED ([FRMRUpload_id]) ON [PRIMARY]
GO

```

--поиск врачей и вывод их снилсов

```

SELECT TOP 1000 ""'+Person_Snils+""+',',Person_SecNameR,* FROM v_personstate
WHERE Person_SurNameR='ТЕСТ' AND Person_FirNameR='ФРМР' /*AND
Person_SecName NOT LIKE '%тест*/ ORDER BY Person_id

```

--добавление методов по врачам в таблицу. добавить метод в таблицу,  
закомментировать строку и раскомментировать следующую

```

INSERT INTO tmp.FRMRUpload (FRMRUpload_Method, Snils_id)
SELECT TOP 105
--'zzz'
'person',
--'document',
--'common',
--'address',
--'prof',
--'postgraduate',
--'ext',
--'cert',
--'accreditation',
--'qualification',
--'card',
--'nomination',
v_PersonState.Person_Snils
/*DISTINCT,
v_PersonState.Person_FirName,v_PersonState.Person_SecName,
v_PersonState.Person_SurName,v_PersonState.Sex_id,

```

```
v_PersonState.Person_BirthDay,v_PersonState.Person_Inn,
'1' as citizenShipId,'171' as oksmId, --Страна
MilitaryRelation_id,v_PersonState.Person_Phone*/
FROM dbo.v_PersonState (NOLOCK)
WHERE Person_Snils IN
(–подставить снилс Врача)
ORDER BY Person_id
```

```
--вывод добавленных методов
```

```
select * from tmp.FRMRUpload
SELECT TOP 1000 * FROM tmp.FRMRUpload
```

```
--удаление всех методов, которые не называются 'zzz'
```

```
delete from tmp.FRMRUpload
where FRMRUpload_Method != 'zzz'
```

```
-----
--update для игнорирования методом.
```

```
--3
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='card1' WHERE
FRMRUpload_Method='card'
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='qualification1' WHERE
FRMRUpload_Method='qualification'
```

```
--2.5
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='ext1' WHERE
FRMRUpload_Method='ext'
```

```
--
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='postgraduate1' WHERE
FRMRUpload_Method='postgraduate'
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='accreditation1' WHERE
FRMRUpload_Method='accreditation'
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='nomination1' WHERE
FRMRUpload_Method='nomination'
```

```
--2
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='prof1' WHERE
FRMRUpload_Method='prof'
```

```
--1
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='address1' WHERE
FRMRUpload_Method='address'
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='document1' WHERE
FRMRUpload_Method='document'
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='common1' WHERE
FRMRUpload_Method='common'
```

```
--0
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='person1' WHERE
FRMRUpload_Method='person'
```

```
update tmp.FRMRUpload set FRMRUpload_Method='zzz' where
FRMRUpload_Method='person'
```

```
--DELETE FROM tmp.FRMRUpload WHERE FRMRUpload_Method!='zzz'
```

```
-----
--update что бы метод увидел
```

```
--3
```

```
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='card' WHERE
```

```

FRMRUpload_Method='card1'
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='qualification' WHERE
FRMRUpload_Method='qualification1'
--2.5
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='cert' WHERE
FRMRUpload_Method='cert1'
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='ext' WHERE
FRMRUpload_Method='ext1'
--
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='postgraduate' WHERE
FRMRUpload_Method='postgraduate1'
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='accreditation' WHERE
FRMRUpload_Method='accreditation1'
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='nomination' WHERE
FRMRUpload_Method='nomination1'
--2
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='prof' WHERE
FRMRUpload_Method='prof1'
--1
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='address' WHERE
FRMRUpload_Method='address1'
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='document' WHERE
FRMRUpload_Method='document1'
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='common' WHERE
FRMRUpload_Method='common1'
--0
UPDATE tmp.FRMRUpload SET FRMRUpload_Method='person' WHERE
FRMRUpload_Method='person1'

--массовка

update tmp.FRMRUpload set FRMRUpload_Method='person' where
FRMRUpload_Method='zzz'

*ВАЖНО. Запускать от меньшего к большему. Т.е начинаем с нуля и поднимаемся к 3*

```

#### 7.6.4 Проверка данных Врача

----ищем и выводим данные для метода person

```

SELECT
DISTINCT(v_PersonState.Person_Snils),
v_PersonState.Person_FirName,
v_PersonState.Person_SecName,
v_PersonState.Person_SurName,
v_PersonState.Sex_id,
v_PersonState.Person_BirthDay,
v_PersonState.Person_Inn,
'1' as citizenShipId,
'171' as oksmId, --Страна
'2' AS MilitaryRelation_id,
v_PersonState.Person_Phone
FROM dbo.v_PersonState (NOLOCK)

WHERE Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='person')
ORDER BY Person_SecName
--GROUP BY Person_Snils
--WHERE zzz.Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='person')

```

-----ищем и выводим данные для метода document

```
SELECT
v_PersonState.Person_Snils,
v_PersonDocument.Document_Ser,
v_PersonDocument.Document_Num,
v_PersonDocument.PersonDocument_updDT,
v_Document.Document_begDate,
v_Document.Document_updDT,
v_Org.Org_Nick,
v_DocumentType.Frmr_id
FROM persis.MedWorker v_MedStaffFactCache (nolock)
OUTER APPLY
(
SELECT TOP 1 lpu_id,MilitaryRelation_id FROM
persis.WorkPlace (nolock)
INNER JOIN persis.v_Staff staff (nolock)ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
) AS staff
INNER JOIN dbo.v_PersonState (nolock)ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
INNER JOIN dbo.v_PersonDocument (nolock) ON v_PersonDocument.Person_id =
v_PersonState.Person_id
INNER JOIN dbo.v_Document (nolock) ON v_Document.Document_id =
v_PersonDocument.Document_id
INNER JOIN dbo.v_DocumentType (nolock) ON v_DocumentType.DocumentType_id =
v_Document.DocumentType_id
left JOIN dbo.OrgDep (nolock) ON OrgDep.OrgDep_id = v_Document.OrgDep_id
left JOIN dbo.v_Org (nolock) ON v_Org.Org_id = OrgDep.Org_id
where
v_PersonState.Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='document')
AND v_Document.Document_begDate IS NOT NULL
AND Org_Nick IS NOT NULL
AND Person_Snils is not NULL

ORDER BY Person_Snils
--GROUP BY Person_Snils
```

-----выводим данные для метода common

```
SELECT
Person_Snils,
ISNULL(EducationInstitution.name,OtherEducationalInstitution) AS institution,
DiplomaSeries AS docSerial,--Серия аттестата docSerial
DiplomaNumber AS docNumber,--Номер аттестата docNumber
SpecialityDiploma.insDT AS docDate--Дата выдачи аттестата docDate
FROM persis.MedWorker v_MedStaffFactCache (nolock)
OUTER APPLY
(
SELECT TOP 1 MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace (nolock)
INNER JOIN persis.v_Staff staff (nolock)ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
) AS staff
INNER JOIN dbo.v_PersonState (nolock) ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
--LEFT JOIN dbo.v_MedPersonal ON v_MedPersonal.Person_id = v_PersonState.Person_id
--LEFT JOIN dbo.v_MedPersonal ON
LEFT JOIN persis.SpecialityDiploma ON SpecialityDiploma.MedWorker_id =
v\_MedStaffFactCache.id AND EducationType_id=2
```

```

LEFT JOIN persis.EducationInstitution ON EducationInstitution.id =
SpecialityDiploma.EducationInstitution_id
WHERE
v_PersonState.Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='common')
AND v\_MedStaffFactCache.id IS NOT NULL
AND DiplomaSeries IS NOT NULL
AND DiplomaNumber IS NOT NULL
AND SpecialityDiploma.insDT IS NOT NULL
AND Person_Snils is not null
ORDER BY Person_Snils

```

-----address

```

SELECT TOP 50
v_PersonState.Person_Snils,
ua.Address_updDT as regDate,
CASE WHEN ua.Address_id=v_PersonState.UAddress_id THEN 1 ELSE 2 end as
addressTypeid,
ua.Address_id,
LOWER(ISNULL(KLTown_AOID,KLCity_AOID)) AS aoidArea,--Идентификатор населенного
пункта по ФИАС aoidArea
LOWER(KLStreet_AOID) as aoidStreet,--Идентификатор улицы по ФИАС aoidStreet
ua.KLRgn_id as region,--Код региона region(id)
ISNULL(KLTown_Name,KLCity_Name) AS areaName,--Наименование населенного пункта
areaName
ISNULL(v_KLTown.KLSocr_Nick,v_KLCity.KLSocr_Nick) AS prefixArea,--Префикс
населенного пункта prefixArea
v_KLStreet.KLStreet_Name AS streetName,--Наименование улицы streetName
v_KLSocr.KLSocr_Nick,--Префикс улицы prefixStreet
dbo.ConcatPunkt(dbo.ConcatPunkt('д',CAST(Address_House AS VARCHAR(30))),'',3,0),
dbo.ConcatPunkt('к',CAST(Address_Corpus AS VARCHAR(30))),'',3,0),'',0,null) AS house,--
Номер дома, корпуса, строения в формате дX кX сX house
Address_Flat AS flat--Номер квартиры flat
FROM persis.MedWorker v_MedStaffFactCache (nolock)
OUTER APPLY
(
SELECT TOP 1 lpu_id,MilitaryRelation_id FROM
persis.WorkPlace (nolock)
INNER JOIN persis.v_Staff staff (nolock)ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
) AS staff
INNER JOIN dbo.v_PersonState (nolock)ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
left JOIN dbo.v_Address ua ON ua.Address_id=v_PersonState.UAddress_id OR
ua.Address_id=v_PersonState.PAddress_id
LEFT JOIN dbo.v_KLTown on v_KLTown.KLTown_id = ua.KLTown_id
LEFT JOIN dbo.v_KLCity on v_KLCity.KLCity_id = ua.KLCity_id
LEFT JOIN dbo.v_KLStreet ON v_KLStreet.KLStreet_id = ua.KLStreet_id
LEFT join dbo.v_KLSocr ON v_KLSocr.KLSocr_id = v_KLStreet.KLSocr_id
WHERE Person_Snils IS NOT NULL
AND ua.Address_updDT IS NOT NULL
AND KLStreet_AOID IS NOT null
AND Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='address')

```

-----prof

```

SELECT TOP 50
Person_Snils,
CASE WHEN EducationInstitution_id IS null THEN 2 ELSE 0 end as educPlace,
EducationType.Frmr_id AS educationTypeId , --уровень образования
CASE WHEN IsSpecSet=1 THEN 'true' ELSE 'false' end as isTargeted,
CASE WHEN IsSpecSet=1 THEN FRMPTerritories_id ELSE null end as targetedRegionId ,
YearOfAdmission AS beginYear, --год поступления
YearOfGraduation,
DocumentRecieveDate AS docDate, --дата выдачи
DiplomaSpeciality.frmr_id AS specId, --специальность
CASE WHEN EducationInstitution_id IS null THEN OtherEducationalInstitution ELSE null end
AS foreignInstitution,--Образовательная организация institution
DiplomaSeries AS docSerial,--Серия аттестата docSerial
DiplomaNumber AS docNumber,--Номер аттестата docNumber
SpecialityDiploma.insDT AS docDate,--Дата выдачи аттестата docDate
--EducationInstitution.code AS institutionId ,
Qualification_id as qualificationId

FROM persis.MedWorker v_MedStaffFactCache (nolock)
OUTER APPLY
(
SELECT TOP 1 MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace (nolock)
INNER JOIN persis.v_Staff staff (nolock)ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
) AS staff
INNER JOIN dbo.v_PersonState (nolock)ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
LEFT JOIN persis.SpecialityDiploma ON SpecialityDiploma.MedWorker_id =
v\_MedStaffFactCache.id AND EducationType_id IN (3,1,6,7)
INNER JOIN persis.DiplomaSpeciality ON DiplomaSpeciality.id =
SpecialityDiploma.DiplomaSpeciality_id
INNER JOIN persis.EducationType ON EducationType.id =
SpecialityDiploma.EducationType_id
left JOIN persis.EducationInstitution ON EducationInstitution.id =
SpecialityDiploma.EducationInstitution_id
WHERE
v\_MedStaffFactCache.id IS NOT NULL
AND DiplomaSeries IS NOT NULL
AND DiplomaNumber IS NOT NULL
AND SpecialityDiploma.insDT IS NOT NULL
AND
Person_Snils IS NOT NULL
AND Person_Snils IN (-указываем СНИЛС выгружаемого врача)

```

-----postgraduate

```

SELECT TOP 50 Person_Snils, CASE WHEN PostgraduateEducation.EducationInstitution_id
IS null THEN 2 ELSE 0 end as educPlace,
EducationInstitution.code AS institutionId , PostgraduateEducationType_id AS
educationStagId,
CAST(YEAR(startDate) AS int) AS beginYear, --год поступления
CASE WHEN endDate IS null THEN 'true' ELSE 'false' end AS isStudying,
PostgraduateEducation.DiplomaSeries AS docSerial,--Серия аттестата docSerial
PostgraduateEducation.DiplomaNumber AS docNumber,--Номер аттестата docNumber
graduationDate AS docDate, --дата выдачи
CASE WHEN PostgraduateEducationType_id IN (3,4,5) AND endDate IS Not null THEN
AcademicMedicalDegree.Frmr_id
ELSE null end as academicDegreeId,
CASE WHEN PostgraduateEducationType_id IN (1,2) THEN Speciality.code ELSE

```



```

PostgraduateEducation.SpecialityAspirant_id end as allSpecId,
ISNULL(PostgraduateEducation.ScienceBranch_id,
SpecialityAspirantScienceLink.ScienceBranch_id) AS scienceBranchId,
CASE WHEN PostgraduateEducationType_id IN (1,2) THEN Speciality.code ELSE null end as
specId,
CASE WHEN PostgraduateEducationType_id IN (3,4) THEN
PostgraduateEducation.SpecialityAspirant_id ELSE null end as doctSpecId,
--'408' as doctSpecId,
CASE WHEN PostgraduateEducationType_id not IN (3,4,5) AND
isnull(PostgraduateEducation.IsSpecSet,44)=1
THEN 'true' WHEN PostgraduateEducationType_id not IN (3,4,5) AND
isnull(PostgraduateEducation.IsSpecSet,44)=0
THEN 'false' ELSE null end as isTargeted, CASE WHEN
isnull(PostgraduateEducation.IsSpecSet,44)=1
THEN FRMPTerritories_id ELSE null end as targetedRegionId
FROM persis.MedWorker v_MedStaffFactCache (nolock)
OUTER APPLY ( SELECT TOP 1 MedWorker_id,staff.Lpu_id FROM persis.WorkPlace (nolock)
INNER JOIN persis.v_Staff staff (nolock)ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id ) AS staff
INNER JOIN dbo.v_PersonState (nolock)ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
inner join persis.PostgraduateEducation ON PostgraduateEducation.MedWorker_id =
v\_MedStaffFactCache.id
INNER JOIN persis.AcademicMedicalDegree ON AcademicMedicalDegree.id =
PostgraduateEducation.AcademicMedicalDegree_id
INNER JOIN persis.Speciality ON Speciality.id = PostgraduateEducation.Speciality_id
INNER JOIN persis.EducationInstitution ON EducationInstitution.id =
PostgraduateEducation.EducationInstitution_id
left join nsi.SpecialityAspirantScienceLink ON
SpecialityAspirantScienceLink.SpecialityAspirant_id =
PostgraduateEducation.SpecialityAspirant_id
WHERE
Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='postgraduate')
and v\_MedStaffFactCache.id IS NOT NULL
AND PostgraduateEducation.DiplomaSeries IS NOT NULL
AND PostgraduateEducation.DiplomaNumber IS NOT NULL
AND ISNULL(Speciality.code,PostgraduateEducation.SpecialityAspirant_id) IS NOT null
AND Person_Snils IS NOT NULL

```

-----ext

```

SELECT TOP 50
Person_Snils AS snils,
EducationInstitution.code AS institutionId,
'1' as profEducationKindId,
QualificationImprovementCourse.hourscount AS hoursCount,
DocumentSeries AS docSerial,
DocumentNumber AS docNumber,
QualificationImprovementCourse.DocumentRecieveDate AS docDate,
--Speciality.code,
DiplomaSpeciality.frmr_id AS code,
CASE WHEN CHARINDEX('/',Speciality.name)-1>0 THEN
SUBSTRING(Speciality.name,0,CHARINDEX('/',Speciality.name)-1) ELSE Speciality.name
END as name
FROM persis.MedWorker v_MedStaffFactCache (nolock)
OUTER APPLY
(
SELECT TOP 1 MedWorker_id,staff.Lpu_id FROM

```

```

persis.WorkPlace (nolock)
INNER JOIN persis.v_Staff staff (nolock) ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
) AS staff
INNER JOIN dbo.v_PersonState (nolock) ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
LEFT JOIN persis.SpecialityDiploma (nolock) ON SpecialityDiploma.MedWorker_id =
v\_MedStaffFactCache.id AND EducationType_id IN (3,1,6,7)
INNER JOIN persis.DiplomaSpeciality (nolock) ON DiplomaSpeciality.id =
SpecialityDiploma.DiplomaSpeciality_id
INNER JOIN persis.EducationType (nolock) ON EducationType.id =
SpecialityDiploma.EducationType_id
INNER JOIN persis.QualificationImprovementCourse (nolock) AS
QualificationImprovementCourse ON QualificationImprovementCourse.MedWorker_id =
v\_MedStaffFactCache.id
LEFT join persis.Speciality (nolock) AS Speciality ON Speciality.id =
QualificationImprovementCourse.Speciality_id
INNER JOIN persis.EducationInstitution ON EducationInstitution.id =
QualificationImprovementCourse.EducationInstitution_id
WHERE
v\_MedStaffFactCache.id IS NOT NULL
and DocumentSeries IS NOT NULL
and DocumentNumber IS NOT NULL
AND Person_Snils IS NOT NULL
AND QualificationImprovementCourse.DocumentRecieveDate IS NOT NULL
and Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='ext')

UNION

SELECT TOP 50
Person_Snils AS snils,
EducationInstitution.code AS institutionId,
'3' as profEducationKindId,
RetrainingCourse.hourscount AS hoursCount,
DocumentSeries AS docSerial,
DocumentNumber AS docNumber,
RetrainingCourse.DocumentRecieveDate AS docDate,
--Speciality.code,
DiplomaSpeciality.frmr_id AS code,
CASE WHEN CHARINDEX('/',Speciality.name)-1>0 THEN
SUBSTRING(Speciality.name,0,CHARINDEX('/',Speciality.name)-1) ELSE Speciality.name
END as name
FROM persis.MedWorker v_MedStaffFactCache (nolock)
OUTER APPLY
(
SELECT TOP 1 MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace (nolock)
INNER JOIN persis.v_Staff staff (nolock) ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
) AS staff
INNER JOIN dbo.v_PersonState (nolock) ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
LEFT JOIN persis.SpecialityDiploma (nolock) ON SpecialityDiploma.MedWorker_id =
v\_MedStaffFactCache.id AND EducationType_id IN (3,1,6,7)
INNER JOIN persis.DiplomaSpeciality (nolock) ON DiplomaSpeciality.id =
SpecialityDiploma.DiplomaSpeciality_id
INNER JOIN persis.EducationType (nolock) ON EducationType.id =
SpecialityDiploma.EducationType_id
INNER JOIN persis.RetainingCourse (nolock) ON RetainingCourse.MedWorker_id =
v\_MedStaffFactCache.id

```

```

LEFT join persis.Speciality (nolock) ON Speciality.id = RetrainingCourse.Speciality_id
INNER JOIN persis.EducationInstitution ON EducationInstitution.id =
RetrainingCourse.EducationInstitution_id
WHERE
v\_MedStaffFactCache.id IS NOT NULL
AND DocumentSeries IS NOT NULL
AND DocumentNumber IS NOT null
AND Person_Snils IS NOT NULL
AND RetrainingCourse.DocumentRecieveDate IS NOT NULL
and Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='ext')

```

-----cert

```

SELECT TOP 50
Certificate.updDT,
Person_Snils,
EducationInstitution.code AS institutionId,
Speciality.code AS specId,
Certificate_ExamDate as examDate,
CertificateReceipDate AS passDate,
CertificateSeries as certSerial,
CertificateNumber AS certNumber
FROM persis.MedWorker v_MedStaffFactCache (nolock)
OUTER APPLY
(
SELECT TOP 1 MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace (nolock)
INNER JOIN persis.v_Staff staff (nolock)ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
) AS staff
INNER JOIN dbo.v_PersonState (nolock)ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
INNER JOIN persis.Certificate (nolock) ON Certificate.MedWorker_id =
v\_MedStaffFactCache.id
LEFT join persis.Speciality (nolock) AS Speciality ON Speciality.id = Certificate.Speciality_id
INNER JOIN persis.EducationInstitution ON EducationInstitution.id =
Certificate.EducationInstitution_id
WHERE
--v_PersonState.Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='cert')
Person_Snils IN (--указать СНИЛС выгружаемых врачей)
-- and CertificateNumber IS NOT NULL
--AND Certificate_ExamDate IS NOT NULL
--AND Person_Snils IS NOT NULL
--AND EducationInstitution_id IS NOT NULL
--AND Speciality.code IS NOT NULL

```

-----qualification

```

SELECT TOP 50
Person_Snils AS snils,
Category.code AS qualifyCategoryId,
AssignmentDate AS beginDate,
Speciality.code AS specId
FROM persis.MedWorker v_MedStaffFactCache (nolock)
OUTER APPLY
(
SELECT TOP 1 MedWorker_id,staff.Lpu_id FROM

```

```

persis.WorkPlace (nolock)
INNER JOIN persis.v_Staff staff (nolock) ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
) AS staff
INNER JOIN dbo.v_PersonState (nolock) ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
INNER JOIN persis.QualificationCategory ON QualificationCategory.MedWorker_id =
v\_MedStaffFactCache.id
INNER JOIN persis.Category ON Category.id = QualificationCategory.Category_id
INNER JOIN persis.Speciality ON Speciality.id = QualificationCategory.Speciality_id
WHERE
Person_Snils IS NOT NULL AND Category.code IS NOT NULL AND Speciality.code IS NOT
NULL
AND AssigmentDate IS NOT NULL
and Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='qualification')

```

-----card

```

SELECT TOP 50
MedStaffFact_id,
Person_Snils AS snils,
PassportToken_tid AS organizationId,
'false' AS targeted,
LpuUnit_FRMOUnitID AS nrPmuDepartId,
LpuSection_FRMOBuildingOid AS nrPmuDepartHospitalSubdivisionId,
Frmr_id AS postId,
CASE WHEN PostOccupationType_id=3 THEN null ELSE CAST(MedStaffFact_Stavka AS
decimal(16,4)) END AS rate,
v_MedStaffFactCache.WorkData_begDate beginDate,
v_MedStaffFactCache.WorkData_endDate endDate,
CASE WHEN DismissalReasonType_id IN (2,3) THEN 2 WHEN DismissalReasonType_id NOT
IN (2,3) THEN 1 ELSE null END AS endTypeld,
DismissalReasonType_id AS fireReasonId,
PostOccupationType.code AS positionTypeld
FROM persis.v_MedWorker (nolock)
INNER JOIN dbo.v_MedStaffFactCache ON v_MedStaffFactCache.Person_id =
v_MedWorker.Person_id
INNER JOIN fed.PassportToken ON PassportToken.Lpu_id = v_MedStaffFactCache.Lpu_id
--отделение
INNER JOIN dbo.v_LpuSection ON v_LpuSection.LpuSection_id =
v_MedStaffFactCache.LpuSection_id
--подразделение
INNER JOIN v_lpuunit ON v_LpuUnit.LpuUnit_id = v_MedStaffFactCache.LpuUnit_id
inner join persis.v_Post Post with (nolock) on v_MedStaffFactCache.Post_id = Post.id
INNER JOIN persis.FRMPPost ON Post.frmppEntry_id= FRMPPost.id
INNER JOIN persis.PostOccupationType ON PostOccupationType.id =
v_MedStaffFactCache.PostOccupationType_id
where
LpuUnit_FRMOUnitID IS NOT NULL
and
Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='card')
AND CAST(MedStaffFact_Stavka AS decimal(16,4)) != 0.0000
AND CAST(MedStaffFact_Stavka AS decimal(16,4)) != 0.0000

```

-----nomination

```

SELECT TOP 50
name AS nomination,
Person_Snils AS Snils,

```

```

FRMPNomination_id AS nominationId,
number AS nomNumber,
date AS nomDate
FROM persis.MedWorker v_MedStaffFactCache (nolock)
OUTER APPLY
(
SELECT TOP 1 MedWorker_id,staff.Lpu_id FROM
persis.WorkPlace (nolock)
INNER JOIN persis.v_Staff staff (nolock)ON WorkPlace.Staff_id = staff.id
WHERE WorkPlace.MedWorker_id=v\_MedStaffFactCache.id
) AS staff
INNER JOIN dbo.v_PersonState (nolock)ON v_PersonState.Person_id =
v_MedStaffFactCache.Person_id
INNER JOIN persis.Reward (nolock) ON Reward.MedWorker_id = staff.MedWorker_id
WHERE
Person_Snils IN (SELECT snils_id FROM tmp.FRMRUpload where
FRMRUpload_Method='nomination')
and name IS NOT NULL
AND Person_Snils IS NOT NULL
AND FRMPNomination_id IS NOT NULL
AND number IS NOT NULL
AND date IS NOT NULL

```















## 14 ФСС (ЛВН)

### 14.1 Настройка сервиса

Сервис настраивается через докер.

Адрес докера: [d-repo.rtmis.ru/jcpcipher:latest](https://d-repo.rtmis.ru/jcpcipher:latest)

Авторизация доменная. могу открыть для определенных групп. Дал в домене rtlabs на пулл группе g\_sysadmin.

При запуске можно использовать следующие переменные (в примере они указаны по умолчанию):

```
"CATALINA_OPTS=-Xms512M -Xmx1536M -server -XX:+UseParallelGC -
Dfile.encoding=UTF-8"
JCP_KEY_LICENSE=''
JCP_COMPANY_NAME=''
```

#### Code Block 98 environments

Порт внутри контейнера используется 8080

Полные логи сервиса располагаются внутри контейнера по следующему пути:  
/opt/tomee/logs

JCP\_KEY\_LICENSE и JCP\_COMPANY\_NAME - указывается для настройки лицензии КриптоПро JCP версии 2.0. Если оставить данные пустыми - то запустится с пробной версией КриптоПро на 3 месяца с неограниченным количеством ядер.

Для примера запуска прилагаю файл docker-compose.yml

```
version: '2'
services:
  jcpCipher:
    image: d-repo.rtmis.ru/jcpcipher:latest
    container_name: jcpcipher
    environment:
      - "CATALINA_OPTS=-Xms512M -Xmx1536M -server -XX:+UseParallelGC
-Dfile.encoding=UTF-8"
      - JCP_KEY_LICENSE=''
      - JCP_COMPANY_NAME=''
    ports:
      - "8080:8080"
```

#### Code Block 99 docker-compose.yml

### 14.2 Настройка ЕЦП.МИС

Настройка осуществляется в файле promed.php

```

$config['EvnStickServiceEncryption'] = true;
$config['EvnStickServiceEncryptionCert'] =
'documents/FSS_PROD_CERT_2020_GOST2012.pem';
if ($config['EvnStickServiceEncryption']) {
    $config['EncryptionServiceUrl'] =
'http:<IP>:<PORT>/jcpCipher/CipherServiceImplService?WSDL';
    $config['EvnStickServiceUrl'] =
'https://docs.fss.ru/WSLnCryptoV11/FileOperationsLnPort?WSDL';
} else {
    $config['EvnStickServiceUrl'] =
'http://193.148.44.24/FSSWSLn/FileOperationsLnPort?WSDL';
}
$config['OPENSSL_PATH'] = '/usr/bin/openssl';
$config['OPENSSL_CONF'] = '/etc/ssl/openssl.cnf';

```

#### Code Block 100 promed.php (Подключение к рабочей ФСС)

```

$config['EvnStickServiceEncryption'] = true;
$config['EvnStickServiceEncryptionCert'] =
'documents/FSS_TEST_CERT_2020_GOST2012.pem';
if ($config['EvnStickServiceEncryption']) {
    $config['EncryptionServiceUrl'] =
'http:<IP>:<PORT>/jcpCipher/CipherServiceImplService?WSDL';
    $config['EvnStickServiceUrl'] = 'https://docs-
test.fss.ru/WSLnCryptoV11/FileOperationsLnPort?WSDL';
} else {
    $config['EvnStickServiceUrl'] =
'http://193.148.44.24/FSSWSLn/FileOperationsLnPort?WSDL';
}
$config['OPENSSL_PATH'] = '/usr/bin/openssl';
$config['OPENSSL_CONF'] = '/etc/ssl/openssl.cnf';

```

#### Code Block 101 promed.php (Подключение к тестовой ФСС)

Переменная `$config['EvnStickServiceEncryption']` отвечает за переключение между сервисом ФСС с шифрованием и без шифрования.

`<IP>:<PORT>` Нужно заменить на ip и порт сервиса, что вы развернули выше.

По поводу openssl писал ранее при установке Промеда. Необходима версия 1.1.0 и выше с поддержкой ГОСТ 2012

Описано в следующих инструкциях: [Установка веб сервера на linux](#), [Важные настройки ЕЦП.МИС](#)

## 14.3 Настройка cron

Для запуска сервиса необходимо обратиться к ЕЦП.МИС через curl. Запускается раз в 3 минуты.

```

#отправка реестров в ФСС
*/3 * * * *

```

```
/usr/bin/curl 'https://<address>:<port>/?c=RegistryES&m=sendRegistryES  
ToFss&swtoken=<token>'
```

#проверка реестров в ФЛК

```
*/3 * * * *
```

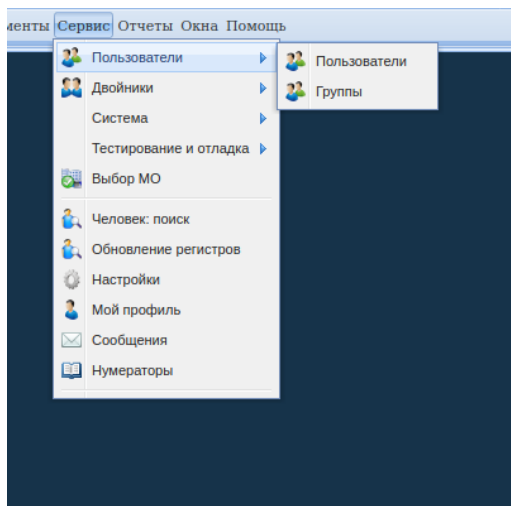
```
/usr/bin/curl 'https://<address>:<port>/?c=RegistryES&m=doFLKControlF  
orAll&swtoken=<token>'
```

[<address>:<port>](#) - Необходимо заменить на адрес и порт ЕЦП сервера либо на его FQDN  
<token>

### 14.3.1 Получение токена авторизации

Для создания пользователя необходима учетная запись с правами Супер Администратора СВАН.

Переходим к пользователям.



Проверяем есть ли пользователь scheduler, если пользователя нет то нажимаем добавить. Если пользователь существует, заходим в него.

A screenshot of the user management interface. At the top, there is a section for adding or editing a user with fields for 'Логин' (Login), 'Группа' (Group), 'Организация' (Organization), 'Описание' (Description), 'Фамилия' (Surname), 'Имя' (Name), 'Тип организации' (Organization type), and 'Заблокирован' (Blocked). Below this is a table titled 'Пользователи' (Users) with columns for 'Логин', 'Имя', 'Фамилия', 'Группы', 'Организации', and 'Описание'. The table contains one entry for the user 'scheduler' with the group 'SuperAdmin'.

Логин	Имя	Фамилия	Группы	Организации	Описание
scheduler	scheduler	scheduler	SuperAdmin		

Для добавления пользователя вводим логин и пароль для этого пользователя. Добавляем в пользователю группа SuperAdmin (СуперАдминистратор СВАН)

Для генерации токена нажимаем кнопку "Сгенерировать", вводим дату меньше 19.01.2038 года. О проблеме можно почитать [тут](#)

Пользователь: Редактирование

1. Основное 2. Доступ к АРМ

Организация: [Поиск] [Добавить] [Удалить]

Наименование [Тип]

Название не определено

Логин: scheduler  Заблокирован Идент. МАРШа: [Поиск]

Временный пароль: [Маскированный]

Токен: 907b3509d125c47d23fea9f6f434c40b60154c3c До: 28.02.2022 [Сгенерировать]

[Сертификаты] [Сертификаты РЭМД]

Сотрудник

Сотрудник: [Выбор]

Фамилия: scheduler Полное имя: scheduler scheduler

Имя: scheduler Эл. почта: [Поиск]

Отчество: [Поиск] Описание: [Поиск]

Группы: [Выбор] [Добавить] [Удалить]

Группа	Описание
SuperAdmin	Супер Администратор СВАН

Количество параллельных сеансов: [Поиск]

[Сохранить] [Помощь] [Отмена]

Сохраняем пользователя.

Сгенерированный токен необходим для запуска сервисов

## 14.4 Настройка cron на БД

Без настройки cron на БД сотрудниками ОЭСУБД сервис не заработает.

## 15 Голосовой помощник

### 15.1 Требования к полигону развертывания

<b>№</b>	<b>Лицензия / иные документы, ссылка на условия лицензии</b>	<b>Наименование библиотеки/ программы и ссылка на размещение (при наличии)<sup>1</sup></b>	<b>Фактический способ использования<sup>2</sup></b>
<b>1</b>	PostgreSQL License	PostgreSQL	Database
<b>4</b>	Apache License 2.0 license	gRPC	Implement gRPC protocol
<b>5</b>	BSD License	Protocol Buffers	Part of gRPC
<b>6</b>	Apache License 2.0	Google Cloud Speech API client library	For speech recognition and synthesis



<b>7</b>	Apache License 2.0	Requests	For sending HTTP <i>requests</i>
<b>8</b>	Apache License 2.0	aiohttp	Async http framework
<b>9</b>	BSD or MIT License	Redis	Database
<b>10</b>	BSD license	Python	General program language
<b>11</b>	BSD license	Flask	Http framework
<b>12</b>	MIT license	SQLAlchemy	SQL toolkit and Object Relational Mapper
<b>13</b>	BSD license	Jinja2	Templating engine

<b>14</b>	BSD 3-Clause license	Werkzeug	Part of Flask
<b>15</b>	BSD 3-Clause license	Celery	Task queue implementation for Python
<b>17</b>	MIT license	pydub	Audio processing library
<b>18</b>	MIT license	xlutils	Package for working with Excel files
<b>19</b>	BSD license	lxml	Package for parse xml
<b>20</b>	BSD license	passlib	Password hashing library
<b>21</b>	MIT license	pickle	For serialize and deserialize objects

<b>22</b>	BSD license	scipy	For wav file read
<b>23</b>	BSD license	sklearn	For learn Gaussian Mixture Models
<b>24</b>	MIT license	python_speech_features	Speech processing
<b>25</b>	BSD license	numpy	Computation framework
<b>26</b>	MIT license	bootstrap	front-end open source toolkit for develop html pages
<b>27</b>	MIT license	jquery	JavaScript library for HTML DOM tree traversal and manipulation
<b>28</b>	MIT license	datatables	Plug-in for the <i>jQuery</i> Javascript library, for create tables

<b>30</b>	MIT license	Redis	Database
<b>31</b>	MIT license	urllib	For sending HTTP <i>requests</i>
<b>32</b>	Apache License 2.0	tensorflow	For collection of workflows to develop and train models using <i>Python</i>
<b>33</b>	MIT license	tflearn	Transparent deep learning library built on top of Tensorflow
<b>34</b>	MIT license	Keras	Deep learning framework
<b>35</b>	Apache License 2.0	PlaidML	Deep learning framework
<b>36</b>	ISC license	Librosa	For music and audio analysis

## 15.1.1 2Требования к техническим и программным средствам

### Функциональные характеристики

№ п.	Характеристика
1.	Наличие графического интерфейса на русском и английском языках
2.	Возможность обзванивать абонентов по предоставленным спискам. В списках помимо номера телефона могут быть дополнительные параметры, используемые в скрипте
3.	Возможность установки разрешенного времени суток, интервалов и дней недели для исходящего звонка с учетом часовых поясов
4.	Возможность установки для исходящих вызовов количества попыток дозвона каждому контакту
5.	Возможность определения успешного дозвона до абонента, регистрация статуса звонка и запуск сценария только при дозвоне
6.	Возможность определения автоответчика или голосовой почты
7.	Возможность настройки перезвонов абоненту в настраиваемый промежуток времени с настраиваемым количеством попыток дозвона
8.	Возможность реагировать с минимальной задержкой, соответствующей естественной задержки человеческой речи
9.	Возможность записывать аудиозапись и транскрипцию всех разговоров (в т.ч. незавершенных) в нужном Заказчику стерео-формате и отправлять в согласованное хранилище
10.	Возможность хранения записей
11.	Возможность формирования статистики по всем проведенным звонкам в формате xls или csv и отправлять ее согласованным способом в согласованное время
12.	Возможность определения просьбы абонента перезвонить в определенный промежуток времени и перезванивать абоненту в этот промежуток времени
13.	Возможность сохранять историю контакта с абонентом по сценарию и начать диалог с места обрыва при перезвоне или поступлении входящего звонка от абонента, если это предусматривается сценарием
14.	Возможность перебивания (как в целом, так и на конкретные фразы)
15.	Возможность переводить звонок на оператора, если это предусмотрено сценарием
16.	Возможность определения пола абонента
17.	Наличие встроенного визуального конструктора диалогов
18.	Наличие в сценариях тегов, которые в диалоге заменяются на информацию из базы данных

19.	Возможность для Абонента передавать данные голосом и в DTMF-режиме
20.	Поддержка нелинейной логики с возможностью переключения между ветками диалога без потери контекста
21.	Использование фоновых шумов во время диалога
22.	Использование активного слушания во время диалога
23.	Наличие собственного 2-х уровневого решения NLU (Natural Language Understanding) для обработки текста на естественном языке с целью выделения сущностей и пользовательских намерений, работающего на паттернах на первом уровне и на алгоритмах глубокого обучения – второй уровень NLU
24.	Возможность генерации SMS
25.	Возможность генерации e-mail
26.	Программные интерфейсы для интеграции посредством HTTP API
27.	Поддержка ASR / TTS от Google, Yandex и ЦРТ
28.	Функционал выбора для бота тембра и пола голоса
29.	Возможность распознавать многозначные числа (номера лицевых счетов или показания), которые абонент произносит, произвольно группируя цифры (например, сто тридцать два, пятнадцать, два нуля, шесть)
30.	Возможность распознавать даты, дни недели
31.	Возможность автоматической обработки адресов
32.	Возможность автоматической обработки ФИО
33.	Журналирование и логирование, позволяющее быстро и четко диагностировать возникающие проблемы.
34.	Эксплуатация в режиме 24x7
35.	Возможность установки в изолированном сетевом сегменте
36.	Наличие интерфейса для мониторинга и контроля
37.	Возможность интеграции с Whatsapp
38.	Возможность интеграции с Viber
39.	Возможность интеграции с Telegram
40.	Возможность интеграции с FB Messenger
41.	Возможность интеграции с Skype

### Технические особенности

№ п.	Характеристика
1.	Микросервисная архитектура
2.	СУБД PostgreSQL v.11, нереляционная СУБД MongoDB v.4.2, файловое хранилище S3
3.	Платформа располагается на виртуальном облачном хостинге Яндекс.Облако
4.	Собственное решение медиа-части, которое обеспечивает масштабируемость, надежность и более широкий функционал по сравнению со стандартным UniMRCP, голосовая платформа собственной разработки (использование FreeSwitch только в качестве SBC), NLU собственной разработки
5.	Отказоустойчивость за счет использования виртуального хостинга и резервирования компонент в разных зонах доступности
6.	Хранение и передача данных в защищенном виде
7.	Механизмы обеспечения защиты данных при обмене: HTTPS, IPsec и др.
8.	Возможность горизонтального масштабирования, использование контейнерных технологий
9.	Опыт интеграции с Genesys, Mango Office, Kazahtelecom, Avaya и др. Возможна интеграция по API с любой системой клиента через слой middleware. По интеграции с телефонией по SIP-протоколу мы можем интегрироваться с любой системой
10.	Соответствие требованиям федерального закона № 152-ФЗ «О персональных данных»
11.	Выполнение мер по защите персональных данных согласно Постановлению № 1119 и 21 приказу ФСТЭК в соответствии с требованиями к 3-му уровню защищенности (УЗ-3)

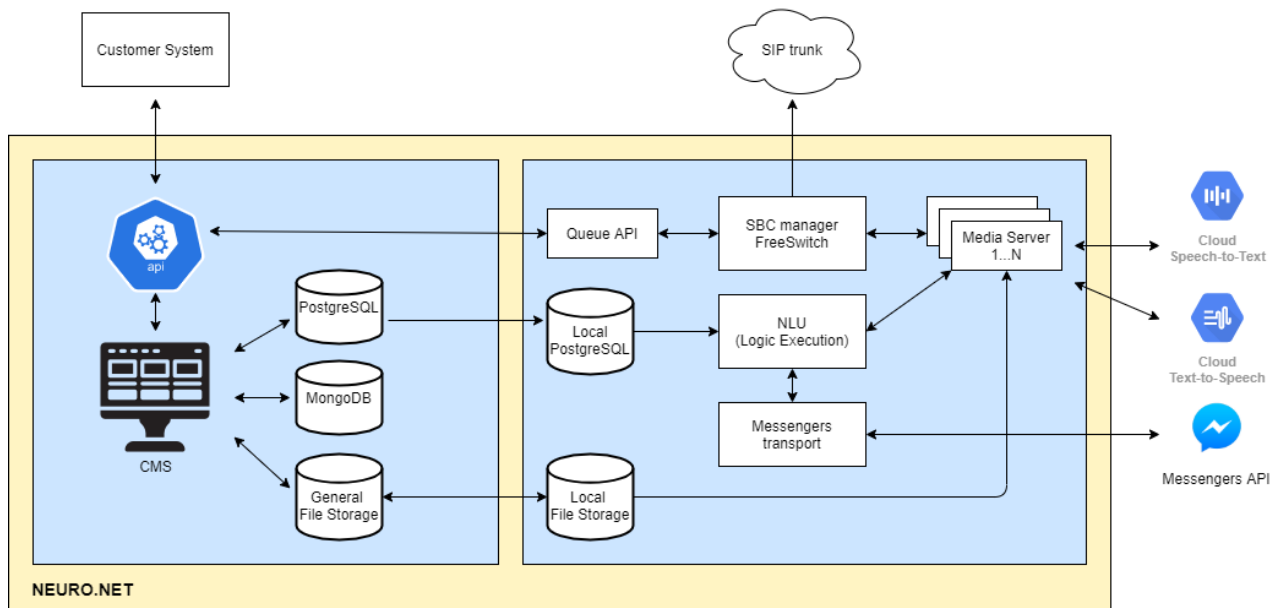
### Конфигурация оборудования

В минимальной конфигурации до 20 каналов нужно 4 сервера:

- DB - 4xCPU, 4 Gb RAM, 20 HDD
- Pool API - 4xCPU, 4 Gb RAM, 20 HDD
- Media Server - 4xCPU, 4 Gb RAM
- SBC - 2xCPU, 4Gb RAM

## Описание процессов, обеспечивающих поддержание жизненного цикла ПО

### High Level Design



### Backend Unit

Сервисы для настроек, сбора и отображения статистики по звонкам:

- CMS
- API
- PostgreSQL
- MongoDB
- General File Storage

Может устанавливаться в закрытый контур.

CMS - пользовательский UI:

- управление Conversational flow
- управление агентами
- управление пользователями
- дашбоард
- очереди
- загрузка звонков
- управление аудиофайлами и записями сущностей, диалогами, контактами и т.д.

HTTP API – для бэкенда и интеграции с внешними партнерами.

General File Storage – файлы промптов, записи звонков.



PostgreSQL – база для хранения метаданных: пользователи, агенты, компании. Содержит загруженные звонки, которые мигрируют в пулы, скрипты и прочие пользовательские данные.

MongoDB – хранит статистику, контакты и т.д.

### **Pool Unit**

Это набор сервисов для реализации диалога посредством голоса и текста:

- SBC Manager
- Media Servers
- NLU
- Messengers Transport
- Local File Storage

Может устанавливаться в закрытый контур.

Local PostgreSQL – очередь звонков, статистика звонков до миграции в MongoDB.

Local File Storage – файлы промптов, записи звонков.

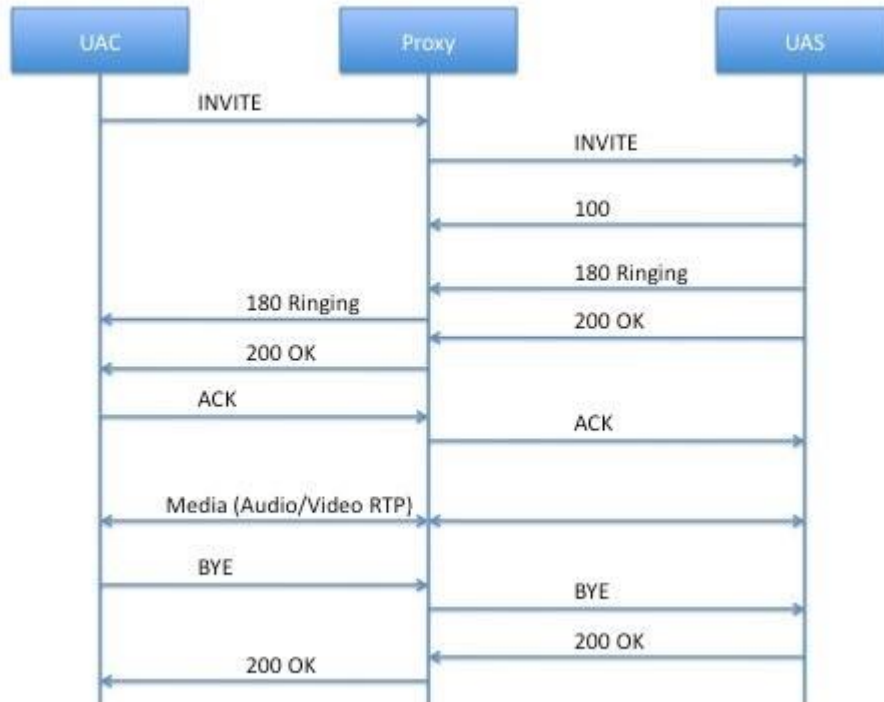
SBC-manager – работает с FreeSWITCH по протоколу ESL, отдает ему команды и получает информацию о каналах.

Media Server – приложение, которое непосредственно работает с SIP и RTP, играет промпты, слушает абонента и интегрируется с ASR.

Logic Executor – выполняет всю логику.

Pool API – работает с очередью звонков, загружает их из БД в память и отдает по запросу.

Call Flow



## 15.2 Core

Модуль	Описание
<i>Frontend Angular JS</i>	UI системы, написанный на AngularJS.
<i>Backend REST API</i>	REST API для Frontend ( <a href="#">ссылка на Swagger</a> ), а также external API ( <a href="#">ссылка на external API</a> ) для клиентов, написанная на Python с использованием фреймворка Flask.
<i>File storage GlusterFS</i>	Файловое хранилище на основе распределённой, параллельной, линейно масштабируемой файловой системы GlusterFS. Служит для хранения аудиозаписей фраз, сущностей и т. д.
<i>MongoDB</i>	Служит для хранения статистики диалогов, биллинга и различных логов.
<i>DB SQL Postgres</i>	Операционная SQL БД, служит для хранения пользователей, паролей, прав пользователей, скриптов, фраз и т.д.

<i>RabbitMQ</i>	Содержит в себе загружаемую через API очередь диалогов. Посредством <i>Dialog distributor</i> распределяет диалоги по пулам, в базу PostgreSQL пула.
<i>Redis</i>	Хранит JWT-токены пользователей.
<i>Middleware</i>	Связующее программное обеспечение для реализации нестандартных сценариев клиента. Имеет свою базу данных ( <i>DB SQL Postgres</i> ).

### 15.3 Pool Unit

Модуль	Описание
<i>PostgreSQL</i>	Хранит в себе операционные данные пула, состояние диалогов, временную статистику по диалогам и биллингу, а также несколько логически реплицируемых таблиц (с Postgres Core'a) с информацией по скриптам и аудиозаписям и т.д.
<i>SBC FreeSWITCH</i>	Session border controller (SBC), используется для сокрытия топологии сети, перекодирования при обработке вызовов на медиа-сервера.
<i>SBC manager</i>	Служит для управления входящими вызовами. Определяет, на какой медиа-сервер можно перевести входящий вызов с <i>SBC FreeSWITCH</i> .
<i>Media server</i>	Управляет медиа-поток, отправляет данные на распознавание, может останавливать звонок, переводить на оператора, делает записи разговоров.
<i>logic_executor_online</i>	Процесс, который обрабатывает онлайн-часть диалогов (звонков).
<i>logic_executor_offline</i>	Процесс, который обрабатывает оффлайн-часть диалогов.

<i>Caller</i>	Запускает звонки (проверка лимитов TCL, временных промежутков, проверка доступности положений).
<i>GlusterFS partition</i>	Является разделом файловой системы сервера с аудиозаписями фраз, сущностей и т. д.
<i>S3_migrator</i>	Скидывает с каждого медиа-сервера записи разговоров в S3 хранилища.
<i>Pool Queue API</i>	Приложение для управления очередью диалогов и звонков. Занимается их приоритизацией.
<i>NLU Engine</i>	Интент-классификатор на основе регулярных выражений. Либо может использовать <i>NLU Model API</i> для классификации на основе нейролингвистической модели.
<i>Message transport</i>	Собирательное понятие из приложений, которые занимаются отправкой и получением сообщений.
<i>ASR/TTS</i>	Сервисы для синтеза и распознавания голоса.

## 15.4 Требования по обеспечению безопасности

Система должна выполнять требования законодательства РФ и подзаконных актов в части защиты персональных данных и тайны связи.

Система должна поддерживать интеграцию с имеющимися системами, и не вводить новые системы, дублирующие функционал имеющихся.

Программный продукт должен иметь механизмы проведения идентификации и аутентификации пользователей.

Система должна обеспечивать уровень протоколирования действий сотрудников и клиентов достаточный для разрешения конфликтных ситуаций. Глубина протоколирования должна определяться действующим законодательством и практикой разрешения споров.

Система поддерживает ролевую модель доступа.

Система поддерживает шифрование данных при взаимодействии с другими приложениями или системами.

## 15.5 Инструкция по установке пользователя на виртуальную машину

На сервере настроен автозапуск компонентов через systemd

ip адрес сервера настраивается автоматически через dhcp

login: root

password: 1q2w3e4r5t

### CMS

1. Создать файл

*/etc/neuro-cms/86a317cb419b.json* с ключами доступа к google speech service

```
"private_key_id": ""
```

```
"private_key": ""
```

```
"client_id" : ""
```

```
"client_email": ""
```

*/etc/neuro-cms/config.py*

Добавляем ключи доступа к облачным сервисам.

```
SECRET_KEY = «»
```

```
#S3 keys
```

```
AWS_ACCESS_KEY_ID=""
```

```
AWS_SECRET_ACCESS_KEY=""
```

```
YANDEX_OAUTH_TOKEN = "
```

Распознавание речи STC, указываем следующие параметры.

```
STC_DOMAIN_ID = "
```

```
STC_USERNAME = "
```

```
STC_PASSWORD = "
```

```
MEGAFON_COMPANY_ID = "
```

2. `systemctl start cms.service`

### API

1. Создать файл с конфигами и прописать ключи доступа  
*/etc/api-neuro/config.py*

```
SECRET_KEY = "  
#S3 keys  
AWS_ACCESS_KEY_ID="  
AWS_SECRET_ACCESS_KEY="  
YANDEX_OAUTH_TOKEN = "  
STC_USERNAME = "  
STC_PASSWORD = "
```

2. `systemctl start api.service`

### PostgreSQL

Инстансы PostgreSQL запускаются на портах 5432 и 5433

```
systemctl start postgresql-10.service  
systemctl start postgresql-10-5433.service
```

### MongoDB

Запускается на порту 27017

```
systemctl start mongod.service
```

### Redis

Запускается как сервис на порту 6379

```
systemctl status redis.service
```

### MediaServer

1. Caller

```
systemctl start caller.service
```

Конфиги настроены и не требуют дополнительных настроек.

*/opt/aio-ivr/config.py*

*/opt/aio-ivr/config.ini*

## 2. Ivr

```
systemctl start ivr.service
```

Конфиги настроены и не требуют дополнительных настроек.

```
/opt/aio-ivr/config.py
```

```
opt/aio-ivr/config.ini
```

## 3. mrcp-server-google-synth

```
systemctl start mrcp-server-google-synth.service
```

В файле `/opt/config/neuronet-credentials.json` указываем параметры подключения google synth

```
"type": "service_account",
"project_id": "",
"private_key_id": "",
"private_key": "-----BEGIN PRIVATE KEY-----<you key>-----END PRIVATE KEY-----\n",
"client_email": "",
"client_id": "",
"auth_uri": "https://accounts.google.com/o/oauth2/auth",
"token_uri": "https://oauth2.googleapis.com/token",
"auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
"client_x509_cert_url": ""
```

## 4. mrcp-server-yandex-synth

В файле `/opt/mrcp-server-yandex-synth/config.ini` указываем параметры подключения к yandex synth

```
FOLDER_ID = <>>
```

```
yandex_api_key = <>>
```

```
OauthToken+OAuthUrl = <>>
```

## 5. mrcp-server

В файле `/opt/mrcp-server/config.ini` указываем параметры подключения к yandex

```
FOLDER_ID = <>>
```

```
yandex_api_key = «»  
OAuthToken+OAuthUrl = «»
```

## 6. mrcp-server-2

```
systemctl start mrcp-server-2.service
```

В файле /opt/config/neuronet-credentials.json указываем параметры подключения google synth

```
"type": "service_account",  
"project_id": "",  
"private_key_id": "",  
"private_key": "-----BEGIN PRIVATE KEY-----<you key>-----END PRIVATE KEY-  
-----\n",  
"client_email": "",  
"client_id": "",  
"auth_uri": "https://accounts.google.com/o/oauth2/auth",  
"token_uri": "https://oauth2.googleapis.com/token",  
"auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",  
"client_x509_cert_url": ""
```

## Freeswitch

1. В /etc/freeswitch/sip\_profiles/external.xml добавляем транк до абонента.
2. В /etc/freeswitch/dialplan/dialplan.xml необходимо настроить bridge до абонента.

```
<context name="internal">  
  <extension name="trunk_1">  
    <condition field="${sip_h_X-Via-Trunk}" expression="^trunk_1$">  
      <action application="log" data="INFO ->${sip_h_X-Via-Trunk}"/>  
      <action application="export" data="nolocal:absolute_codec_string=PCMA"/>  
      <action application="bridge"  
data="sofia/gateway/trunk_1/${destination_number}"/>  
    </condition>  
  </extension>  
</context>
```

queue\_api



queue\_api сконфигурирован и не требует настроек

call\_migrator

call\_migrator сконфигурирован и не требует дополнительных настроек