

Акционерное общество «Башкирский регистр социальных карт»

Техническая документация к ИС «Регистр карт жителя»
Руководство по установке, развертыванию и администрированию модуля

Оглавление

Определения и сокращения.....	3
1 Общая характеристика программы	4
2 Требования к персоналу	5
Требования к системному администратору и администратору БД.....	5
Требования к разработчикам.....	5
3 Системно-технические характеристики	6
3.1 Технические требования.....	6
4 Установка и настройка ПО.....	7
4.1 Сервер БД.....	7
4.1.1 Подготовка системы	7
4.1.2 Установка PostgreSQL 14.....	7
4.1.3 Создание БД	7
4.2 Сервер сервисов.....	7
4.2.1 Подготовка системы	7
4.2.2 Настройка проектов.....	8
4.3 Развертывание сервисов	10
4.3.1 Подготовка проектов	10
4.3.2 Настройка конфигураций Systemd.....	10
4.4 Мониторинг обновления журнала ошибок через Zabbix	11
4.4.1 Установка Zabbix	11
4.4.2 Настройка Zabbix	12
4.4.3 Удаление Zabbix.....	13

Определения и сокращения

Термин, аббревиатура, сокращение, обозначение	Пояснение, определение, расшифровка
АИС	Автоматизированная информационная система
АРМ	Автоматизированное рабочее место
БД	База данных
ИС, ИС «Регистр карт»	Информационная система «Регистр карт жителя»
ПО	Программное обеспечение – совокупность программ систем обработки информации
Программа, Система	Информационная система «Регистр карт жителя»
Разработчик	Подразделение/ Служба, отвечающая за разработку, адаптацию, доработку, внедрение программного обеспечения и сопровождение Системы
Системный администратор	Подразделение/ Служба, отвечающая за системное администрирование и администрирование БД.
API	application programming interface – программный интерфейс приложения.

1 Общая характеристика программы

Информационная система «Регистр карт жителя» (далее - ИС «Регистр карт») представляет собой АРМ. Вход в ИС «Регистр карт» осуществляется посредством web-браузера по адресу <https://sc.brsc.ru/>.

Каждому пользователю ИС назначается Роль с определенным набором прав для работы в Системе. Права доступа предоставляются индивидуально для каждого Пользователя. Пользователи Системы работают в ИС «Регистр карт» в соответствии с Руководством пользователя. Пользовательский функционал Системы представляет собой набор личных кабинетов для Ролей:

- Администратор,
- Оператор ППЗ,
- Оператор информационного обмена,
- Оператор ППЗ/ПВК,
- Печать отчетов с просмотром заявлений,
- Оператор ЭП,
- Оператор сортировки Банка,
- Печать отчетов без просмотра заявлений,
- Оператор СПК,
- Оператор ППВ и сортировки (Банк),
- Оператор ПВК – выдача и получение карты в ППВ,
- Оператор ППЗ/ПВК – реестровая загрузка,
- Просмотр заявлений,
- Оператор ППЗ-реестровая загрузка-все отделения банка.

Функционал Системы позволяет:

- осуществлять поиск клиента в Системе: для проверки наличия у него оформленного в Системе Заявления или Карты; для изменения учетных данных клиента;
- работать с Заявлениями: поиск, просмотр Заявлений; формирование Заявлений на открытие счета и выпуск/перевыпуск карты, загрузка фотографии клиента или фотографирование (при необходимости); просмотр статуса Заявления/Карты; печать Заявлений, реквизитов карты, расписки в получении карты/ПИН-конверта; реестровое оформление Заявлений в рамках зарплатных проектов Банка; фиксация в ИС в режиме online фактов выдачи клиентам карт и ПИН-конвертов, Конвертов с документами по ЭП;
- создавать квалифицированный сертификат ключа проверки ЭП;

- формировать отчеты;
- осуществлять информационный обмен между Системой и Банком;
- производить администрирование системы: настройка ролей пользователей, карточных продуктов, справочников и т.д.; регистрация ППВ; предоставление/блокировка доступов в Систему.

2 Требования к персоналу

Требования к системному администратору и администратору БД

Знания и опыт практической работы:

- ОС Linux,
- Уверенные знания работы с сетью, HTTP/HTTPS, DNS, DHCP, IPv4/IPv6, VPN, NAT, Firewall и др.,
- Понимание архитектуры современных вычислительных систем: процессоры, память, шины обмена данными, накопители,
- PostgreSQL 14 и выше:
 - а) знание и опыт администрирования СУБД PostgreSQL Server,
 - б) знание стандартов и опыт написания SQL-запросов,
 - в) резервное копирование БД.

Требования к разработчикам

Разработчик должен обладать следующими навыками:

- Знания: C#, ОПП, Entity Framework Core - EF Core, PostgreSQL, HTML, JavaScript, TypeScript, Python, Node.js, PostgreSQL, Visual Studio Code (VS Code), Git, Gitlab CI/CD, Node.JS, NexusJs, PrismaORM, VUE2;
- Опыт работы с HTML, jQuery, ASP NET MVC 3, Visual Studio, MSSQL Server, PostgreSQL;
- Опыт разработки API для веб приложений;
- Отличные знания NET, JavaScript, C#, Visual Studio, MSSQL Server, PostgreSQL;
- Опыт работы с базами данных (MongoDB, Microsoft SQL Server, PostgreSQL);
- Глубокое понимание принципов устройства баз данных «изнутри» (индексы, материализованные представления, оптимизация запросов, транзакции, процедуры, работа с планировщиком запросов и пр.);
- Умение писать UNIT-тесты;
- Уровень английского для чтения технической документации;

- Умение работать с системой контроля версий (Git);
- Понимание жизненного цикла разработки ПО.

3 Системно-технические характеристики

- технические средства для работы: Visual Studio 2022 версии 17.0 или более поздней, Dotnet SDK 6.0.412 (.NET Runtime 6.0.20, ASP.NET Core Runtime 6.0.20, C# 10.0);
- технические средства для frontend: HTML, JQuery, CSS;
- технические средства для framework: ASP NET MVC 6;
- язык программирования: C#;
- СУБД PostgreSQL 14;
- Zabbix – система мониторинга серверных метрик, с открытым исходным кодом.

3.1 Технические требования

Технические требования к серверам представлены в таблице 1.

Таблица 1

БД	Требования
ЦП	4
ОЗУ	2 Гб
Память	55 Гб
СУБД	PostgreSQL 14
Доступы	– доступ к фото и данным о клиенте для печати на карту
Сервер сервисов	Требования
ЦП	4
ОЗУ	8 Гб
Память	100 Гб
Доступы	– доступ к dadata.ru – доступ к транспортному API для экспорта карт – доступ к БД

4 Установка и настройка ПО

4.1 Сервер БД

4.1.1 Подготовка системы

Установить утилиты sudo:

```
apt install sudo
```

Обновить имеющиеся пакеты:

```
sudo apt update && sudo apt upgrade -y
```

Установить необходимые пакеты для возможности установки и работы Postgresql 14:

```
sudo apt -y install gnupg2 wget vim make gcc
```

4.1.2 Установка PostgreSQL 14

Добавить репозиторий PostgreSQL 14 в пакетный менеджер apt:

```
sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt $(lsb_release -cs)-pgdg main" > /etc/apt/sources.list.d/pgdg.list'
```

Импорт gpg ключа для добавленного репозитория:

```
wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -
```

Обновить индексации пакетов apt:

```
sudo apt -y update
```

Установить Postgresql 14:

```
sudo apt install postgresql-14
```

Установить дополнительные библиотеки:

```
apt install postgresql-server-dev-14
```

4.1.3 Создание БД

Инициализация базы данных происходит в момент запуска основного проект OPM.

4.2 Сервер сервисов

4.2.1 Подготовка системы

4.2.1.1 Основные операции в Linux

Установить утилиты sudo:

```
apt install sudo
```

Обновить имеющиеся пакеты:

```
sudo apt update && sudo apt upgrade -y
```

4.2.1.2 Установка .NET Core в Linux

Установить пакеты SDK для .NET Core 6.0

```
sudo apt install dotnet-sdk-6.0
```

4.2.1.3 Установка Nginx

Установить сервис nginx:

```
Sudo apt install install nginx
```

4.2.2 Настройка проектов

4.2.2.1 Настройка файлов конфигурации APM

Конфигурация файла ./Brsc.Scb.Mvc/appsettings.Production.json

1. Объект ConnectionStrings – строки подключения
 - а) PostgreSQL - строка подключения к СУБД PostgreSQL 14
2. Объект Project – настройки проекта
 - а) SearchResultsMaxCount - максимальное количества возможных записей в результате поиска "Заявления" и "Реестр";
 - б) BankProducts - идентификаторы банковских карточных продуктов, перечисленные через Пробел;
 - в) SalaryProducts - идентификаторы зарплатных банковских карточных продуктов, перечисленные через Пробел;
 - г) ClientPhotosDirectory - директория хранения загруженных фото клиентов;
 - д) PersonalizationDirectory - директория хранения фото клиентов для центра персонализации карт;
 - е) Api - адрес расположения сервиса АПИ. Прим. <http://localhost:5000/>;
 - ж) MfcActNoBankTemplateFile - путь к шаблону акта передачи карт без банка в МФЦ;
 - з) MfcActNoBankTemplateFile - директория хранения временных сформированных актов передачи карт в МФЦ;
 - и) ReceiptFileDirectory - Директория хранения сканов чеков об оплате карты без банка;
 - к) MfcPersoIssueCities - Список городов для фильтрации выборки отделений пунктов выдачи карт без банка;
 - л) TestApp - признак тестовой обработки запросов на выпуск карты через TIBCO, 0 - боевое обращение на TIBCO, 1 - тестовое обращение на заглушку в части отправки заявки в Банк;
 - м) Api - адрес сервиса АПИ.

4.2.2.2 Конфигурация проекта API

Конфигурация файла ./Brsc.Scb.Api/appsettings.Production.json

1. Объект ConnectionStrings - строки подключения:
 - a) PostgreSQL - строка подключения к СУБД PostgreSQL 14.
2. Kestrel:Endpoints: Http:Url - адрес привязки к порту;
3. Origins – список доменов для прохождения CORS;
4. Объект Services – конфигурация внешних сервисов:
 - a) Dadata – конфигурация интеграции с сервисом Дадаты:
 1. Api – адрес сервиса Дадаты;
 2. ApiKey – ключ АПИ.

```
{
  "ConnectionStrings": {
    "PostgreSQL": "Строка подключения"
  },
  "Kestrel": {
    "EndPoints": {
      "Http": {
        "Url": "http://0.0.0.0:9980"
      }
    }
  },
  "Origins": [
    "http://localhost:9980"
  ],
  "Services": {
    "Dadata": {
      "Api": "https://suggestions.dadata.ru/suggestions/api/4_1/rs/",
      "ApiKey": "${API_KEY}"
    }
  },
  "Authentication": {
    "Jwt": {
      "Issuer": "https://localhost:9980",
      "Audience": "https://localhost",
      "SecretKey": "JRuff1Zmqo4DWMyqNP7nm1yb1likBv04n",
      "LifeTime": 3600
    }
  },
  "AllowedHosts": "*",
  "Swagger": {
    "Endpoints": {
      "Brsc.Scb.Api v1": "/api/swagger/v1/swagger.json"
    }
  },
  "DistributedCache": {
    "Store": "InMemory",
    "SlidingExpirationInSeconds": 900,
    "SqlServer": {
      "ConnectionString": "ConnectionString",
      "SchemaName": "Schema",
      "TableName": "Table",
      "EnsureCacheCleared": false
    },
    "Redis": {
      "ConnectionString": "ConnectionString",
      "InstanceName": "Instance"
    }
  },
  "Prints": {
```

```

    "CardIssuance": {
      "1" : "Шаблон документов для выдачи карты.docx"
    }
  }
}

```

4.3 Развертывание сервисов

4.3.1 Подготовка проектов

Перейдите в папку проекта, затем выполните команду

```
Sudo dotnet publish
```

Затем, для переноса файлов АРМ, выполните команду:

```
sudo cp -a ./Brsc.Scb.Mvc/bin/Debug/net6.0/publish/ /var/scb/scb-service
```

Затем, для переноса файлов API, выполните команду:

```
sudo cp -a ./Brsc.Scb.Api/bin/Debug/net6.0/publish/ /var/scb/scb-api-service
```

4.3.2 Настройка конфигураций Systemd

Перейдите в папку конфигураций управления служб, выполните команду:

```
Cd /etc/systemd/system
```

4.3.2.1 Конфигурация АРМ

Создайте файл конфигурации scb.service, со следующим содержанием:

```

[Unit]
Description=АРМ
[Service]
WorkingDirectory=/var/scb/scb-service
ExecStart=/usr/share/dotnet/dotnet /var/scb/scb-service/Brsc.Scb.Mvc.dll
Restart=always
RestartSec=3
SyslogIdentifier=scb.service
User=root
Environment=ASPNETCORE_ENVIRONMENT=Production
ExecStop=/bin/kill -s TERM $ MAINPID
[Install]
WantedBy=multi-user.target

```

Запустите сервис, выполните команду:

```

Systemctl enable scb.service
Systemctl start scb.service

```

4.3.2.2 Конфигурация API

Создайте файл конфигурации scb-api.service, со следующим содержанием:

```

[Unit]
Description=АРМ АПИ
[Service]
WorkingDirectory=/var/scb/scb-api-service
ExecStart=/usr/share/dotnet/dotnet /var/scb/scb-api-service/Brsc.Scb.Api.dll
Restart=always
RestartSec=3
SyslogIdentifier=scb.service
User=root

```

```
Environment=ASPNETCORE_ENVIRONMENT=Production
ExecStop=/bin/kill -s TERM $ MAINPID
[Install]
WantedBy=multi-user.target
```

Запустите сервис, выполните команду:

```
Systemctl enable scb-api.service
Systemctl start scb-api.service
```

4.4 Мониторинг обновления журнала ошибок через Zabbix

Каждый сервис формирует журнал событий и журнал ошибок. Журнал ошибок содержит сведения о возникших исключительных ситуациях и часто требует внимания разработчиков. Поэтому для ряда сервисов настроен мониторинг обновления журнала ошибок через Zabbix, а также отправка сообщений об ошибках соответствующим разработчикам в систему мгновенного обмена сообщениями Telegram и в электронную почту.

Пример сообщения:

```
«ПРОБЛЕМА!!!: Error_dolg
Проблема началась в 04:03:13 on 2023.05.25
Название Проблемы: Error_dolg
Host: scb.it.brsc.ru
Важность: High
Эксплуатационные данные: [25.05.2023, 04:02:13] pg: connect ETIMEDOUT 10.3.26.121:5432
ID Проблемы: 12423500
```

Zabbix-new»

Настройка осуществляется системным администратором по заявке разработчика или аналитика.

4.4.1 Установка Zabbix

Debian:

```
wget http://repo.zabbix.com/zabbix/5.0/debian/pool/main/z/zabbix-release/zabbix-release\_5.0-1+stretch\_all.deb //Debian
dpkg -i zabbix-release_5.0-1+stretch_all.deb
```

Ubuntu:

```
wget https://repo.zabbix.com/zabbix/5.0/ubuntu/pool/main/z/zabbix-release/zabbix-release\_5.0-1+xenial\_all.deb //Ubuntu22.04
dpkg -i zabbix-release_5.0-1+xenial_all.deb
```

```
wget https://repo.zabbix.com/zabbix/5.0/ubuntu/pool/main/z/zabbix-release/zabbix-release\_5.0-1+focal\_all.deb //Ubuntu20.04  
dpkg -i zabbix-release_5.0-1+focal_all.deb
```

```
sudo apt-get update  
sudo apt-get install zabbix-agent
```

```
systemctl status zabbix-agent  
systemctl stop zabbix-agent  
systemctl is-enabled zabbix-agent  
systemctl enable zabbix-agent  
systemctl restart zabbix-agent  
systemctl status zabbix-agent  
service zabbix-agent restart
```

wget стандартно отправляет файл в корневой каталог пользователя. Если это пользователь root -> Каталог root.

Если wget не скачивает файл – необходимо скачать в браузере и через WinSCP перенести вручную на сервер в каталог пользователя. И дальше продолжать с команды dpkg.

4.4.2 Настройка Zabbix

На сервере, который добавляем на Zabbix, необходимо сделать следующее:

1. Зайти на /etc/zabbix/zabbix_agentd.conf
2. Добавить в пунктах и ServerActive, адрес сервера Zabbix:

Server=ИП_АДРЕС,

ServerActive=ИП_АДРЕС,

Hostname=ИМЯ_ХОСТА,

3. На веб-интерфейсе Zabbix необходимо:

- Перейти в Configuration -> Hosts, в правом верхнем углу нажать Create host.
- В разделе Host указать:

Host name: ИМЯ ХОСТА,

Groups: нажать Select и выбрать к какой группе будет относиться host,

Interfaces: прописать IP Address или DNS. Выбрать один из двух.

- В разделе Templates:

нажать Select, выбрать темплейт (Template OS Linux by Zabbix agent) -> Select.

4.4.3 Удаление Zabbix

```
sudo apt-get remove zabbix-agent
```

```
sudo apt-get purge --auto-remove zabbix-agent
```

```
rm -rf /etc/zabbix/*
```