### **README**

### Сервис для управления пользователями и безопасности операций с использованием OTP-кодов

### **Описание сервиса**

### Проект реализует веб-приложение для управления пользователями и безопасного выполнения операций с использованием одноразовых паролей (OTP).Предоставляет интерфейс для администрирования, регистрацию и авторизацию пользователей, управление защищенными операциями, доставку OTP-кодов разными методами (почта, SMS, Telegram, файл).

### **Установка и настройка базы данных**

### Для корректной работы приложения **необходимо** установить базу данных локально.База данных должна быть реализована с помощью **PostgreSQL версии 17**, поскольку взаимодействие с базой данных реализовано через JDBC.

### **Структура базы данных**

### Структура базы данных включает три таблицы:

### -- Таблица пользователей

### CREATE TABLE users (

### id SERIAL PRIMARY KEY,

### username VARCHAR(255) NOT NULL UNIQUE,

### password\_hash VARCHAR(255) NOT NULL,

### role VARCHAR(50) NOT NULL CHECK (role IN ('ADMIN', 'USER'))

### );

### -- Таблица конфигурации OTP-кодов

### CREATE TABLE otp\_config (

### id SERIAL PRIMARY KEY,

### codelength INTEGER NOT NULL,

### lifetimeinminutes INTEGER NOT NULL

### );

INSERT INTO otp\_config (codelength, lifetimeinminutes)

VALUES (5, 7); - – создадим запись, которую потом будем менять

### -- Таблица одноразовых кодов

### CREATE TABLE otp\_codes (

### id SERIAL PRIMARY KEY,

### user\_id BIGINT NOT NULL, -- Внешний ключ на таблицу users

### operation\_id INT,

### otp\_code VARCHAR(10) NOT NULL,

### status VARCHAR(20) NOT NULL DEFAULT 'ACTIVE' CHECK (status IN ('ACTIVE', 'EXPIRED', 'USED')),

### created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

### expires\_at TIMESTAMP,

### description TEXT NOT NULL,

### CONSTRAINT fk\_user FOREIGN KEY (user\_id) REFERENCES users(id) ON DELETE CASCADE

### );

Конфигурация подключения к базе данных расположена в пакете package org.example.util в файле DbConnection. Настройте её под свою среду разработки, указав нужные значения (адрес сервера, порт, название базы данных, имя пользователя и пароль).

### **Архитектура сервиса**

### Сервис состоит из нескольких слоев:

**API**

UserApi — Контроль регистрации и входа пользователей, инициация защищенных операций и проверка OTP-кодов.

AdminApi — Управление конфигурацией OTP-кодов и просмотр списка пользователей, удаление пользователей.

**Сервисы**

UserService — Управление пользователями (регистрация, авторизация, управление ролями).

OtpService — Генерация, отправка и проверка OTP-кодов, управление защитными операциями.

**DAO (Data Access Object)**

UserDao — Интеграция с таблицей пользователей.

OtpDao — Хранение и управление информацией об OTP-кодах.

**Модели данных**

User — Представление сущности пользователя с уникальными идентификаторами и хэшированными паролями.

OtpCode — Информация об одноразовом коде и связанной операции.

**Утилиты**

EmailNotificationService — Отправка уведомлений по электронной почте.

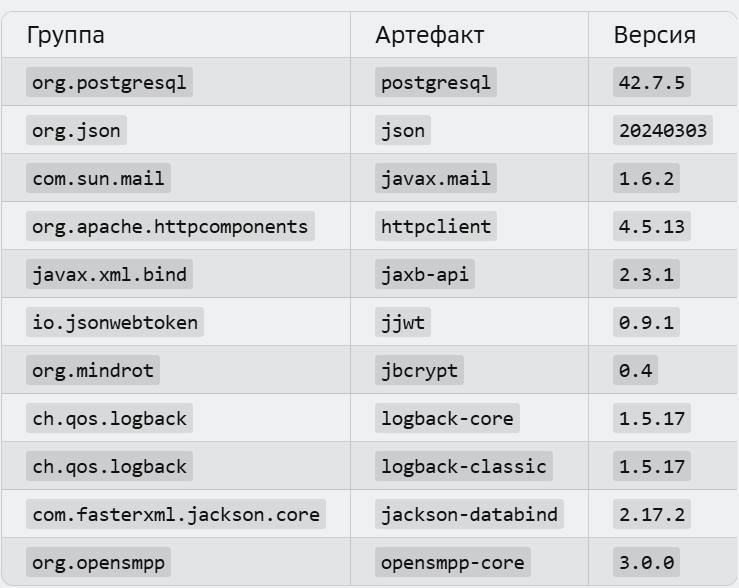
SmsSender — Эмулятор отправки SMS-сообщений.

TelegramBot — Бот для интеграции с Telegram. Название бота - @cvbsdjlsslabot необходимо использовать свой chatId для настройки.

FileSaver — Сохранение OTP-кодов в файле.

### **Необходимые библиотеки**

### Установка с помощью Maven, все библиотеки содержаться в pom.xml



### **Команды сервиса**

### Сервис предлагает следующий набор команд:

### **Зарегистрироваться** — Создание новой учетной записи.

### **Войти** — Аутентификация пользователя.

### **Получить список пользователей** — Доступно только пользователям с ролью администратора.

### **Удалить пользователя** — Доступно только пользователям с ролью администратора.

### **Изменить конфигурацию OTP-кодов (время жизни и количество знаков в коде).** — Доступно только пользователям с ролью администратора.

### **Инициировать операцию и отправить OTP-код** — Инициация безопасной операции с последующей отправкой одноразового пароля (OTP).

### **Проверить OTP-код** — Подтверждение одноразового пароля.

**Дополнительно реализовано:**

реализован механизм обработки устаревших OTP-кодов, который каждый час проверяет наличие активных OTP-кодов и помечает просроченные коды статусом EXPIRED.

### **Проблемы и поддержка**

### При возникновении сложностей обращайтесь к документации библиотек или свяжитесь с разработчиком **Давлетшина Алла +7 (915 ) 088 76 21**.