**Министерство науки и высшего образования Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО”  
(УНИВЕРСИТЕТ ИТМО)**

**ЦЕНТР АВТОРИЗОВАННОГО ОБУЧЕНИЯ ИНФОРМАЦИОННЫМ ТЕХНОЛОГИЯМ**

**ИТОГОВАЯ АТТЕСТАЦИОННАЯ РАБОТА**

«Система автоматизации работы медицинского учреждения»

Автор: Воронина Наталья Вячеславовна \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия Имя Отчество) (Подпись)

**Центр авторизованного обучения информационным технологиям**

Наименование программы **«Java-разработчик»**

Руководитель Лудов Д.Ю. \_\_\_\_\_\_\_\_\_\_\_\_

(Подпись)

**К защите допустить**

|  |  |  |
| --- | --- | --- |
| *Заместитель директора ЦАО ИТ,*  *к.т.н.* |  | / *Т.В. Зудилова*/ |

Санкт-Петербург, 2022г.

Обучающийся\_\_­­­­­\_\_\_\_\_Воронина Н.В.\_\_\_\_ Группа 124/12

(Фамилия, И. О.)

Работа принята «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2022г.

Работа выполнена с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2022г.

Секретарь ИАК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, И. О.) (подпись)

Листов хранения \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Демонстрационных материалов \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ОГЛАВЛЕНИЕ**

[Введение 4](#_Toc82990730)

[Цель 5](#_Toc82990731)

[Описание системы 6](#_Toc82990732)

[Функции системы 7](#_Toc82990733)

[Структура базы данных 19](#_Toc82990734)

[Заключение 20](#_Toc82990735)

[Литература 21](#_Toc82990736)

# Введение

Данная итоговая аттестационная работа предназначена для автоматизации процесса записи на прием пациентов к врачам медицинского учреждения.

Потребность в автоматизации процесса обоснована неудобством ведения ручного документооборота и координации времени работы врачей в подобных организациях. Пациенты также испытывают неудобства из-за необходимости оформлять запись на прием через обращение личное или по телефону в организацию.

Чтобы удовлетворить потребности пациентов и врачей требуется предоставить инструмент, позволяющий формировать запись на прием онлайн.

# 

# Цель работы

Основываясь на изученном материале, целью моего дипломного проекта является разработка серверной части приложения для записи пациентов на прием к врачу внутри одной медицинской организации.

Задачи, которые предстоит решить для достижения поставленной цели:

* Проработка требований к системе
* Выбор технологического стека
* Первичная конфигурация проекта
* Обеспечение авторизации/аутентификации пользователей
* Разработка необходимых для работы приложения методов API пациента, врача, администратора
* Тестирование приложения
* Настройка автоматической документации доступных методов API
* Настройка сборки и деплоя приложения

# Описание системы

Система написана на языке программирования Java с использованием программных средств, таких как :

- Java версии 17 - в качестве языка разработки серверной части приложения

- Maven - в качестве инструмента сборки зависимостей проекта

- Intellij IDEA Ultimate Edition - среда разработки

- Фреймворк Spring

- СУБД PostgreSQL 11 - реляционное хранилище данных

- Git - система контроля версий

- Docker - открытая платформа для разработки, доставки и эксплуатации приложений

- Swagger – графический интерфейс, позволяющий поддерживать документацию доступных методов API в актуальном состоянии

- Keyckoak – готовое решение с открытым кодом для реализации SSO (single sign on)

# Функции системы

1. **Пациент (patient)**

Сущность пациент имеет следующие функции:

* Создание нового пациента
* Просмотр пациентов по врачу
* Обновить информацию о пациенте
* Получить информацию по одному пациенту

Описание API в Swagger

Изображение выглядит как текст

Автоматически созданное описание

1. **Врач (doctor)**

Сущность врач имеет следующие функции:

* Создание нового врача
* Просмотр списка врачей

Описание API в Swagger

**Изображение выглядит как текст

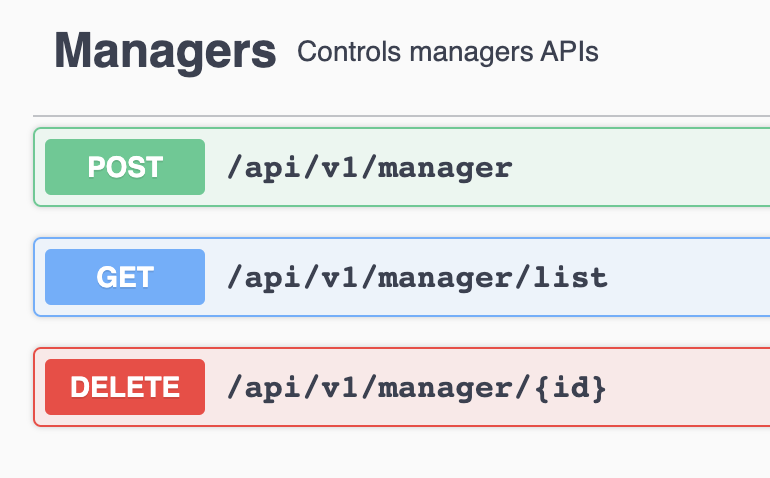
Автоматически созданное описание**

1. **Сотрудник регистратуры (manager)**

Сущность администратора имеет следующие функции:

* Создание нового администратора
* Просмотр списка администраторов
* Удаление администратора из системы

Описание API в Swagger

****

1. **Специализации (specialization)**

Сущность специализаций врача имеет следующие функции:

* Создание новой специализации в системе
* Просмотр списка доступных специализаций

Описание API в Swagger

**Изображение выглядит как текст

Автоматически созданное описание**

1. **Услуги (service)**

Сущность услуги имеет следующие функции:

* Создание новой услуги в системе
* Просмотр списка доступных услуг

Описание API в Swagger

**Изображение выглядит как текст

Автоматически созданное описание**

1. **Расписание врача (service)**

Сущность расписание врача имеет следующие функции:

* Заполнить расписание
* Удалить слот из расписания
* Получить расписание по врачу

Описание API в Swagger

**Изображение выглядит как текст

Автоматически созданное описание**

1. **Запись на прием (appointment)**

Сущность услуги имеет следующие функции:

* Создать запись на прием
* Удалить запись на прием
* Перенести запись
* Удалить запись на прием к врачу

Описание API в Swagger

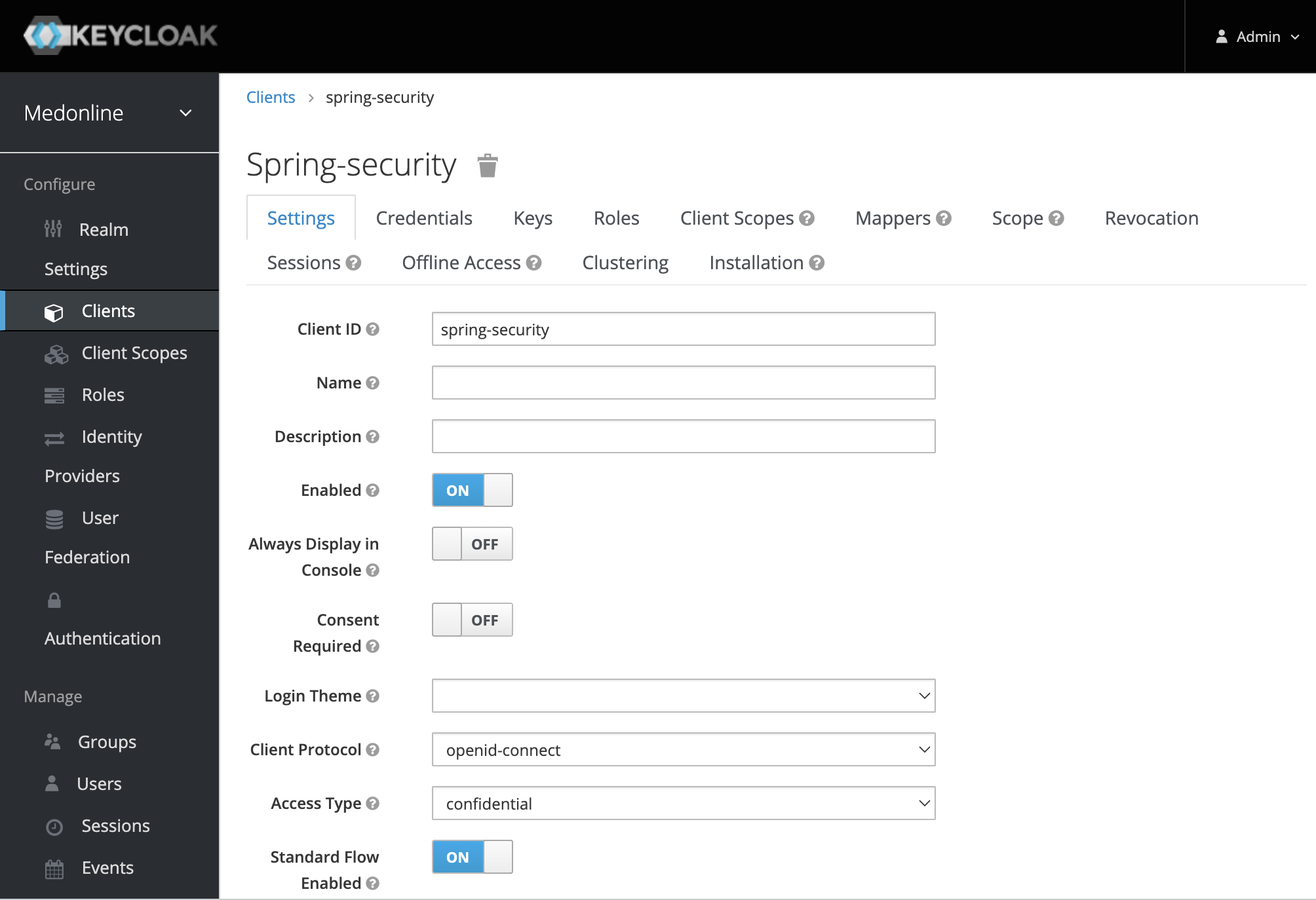
**Изображение выглядит как текст

Автоматически созданное описание**

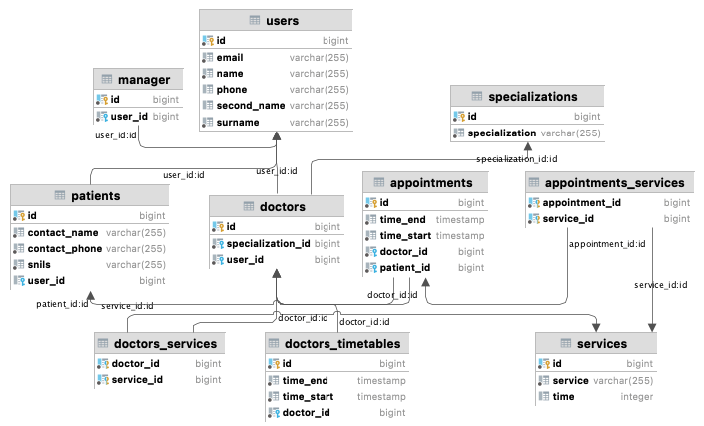
В системе реализована авторизация и аутентификация на базе KeyCloak.

KeyCloak позволяет гибко настраивать правила авторизации, управлять сессиями пользователя, добавлять служебную информацию в JWT токен пользователя, настраивать политику сброса паролей по регламентному времени, а также реализует множество других функций связанных с безопасностью приложения и пользовательских данных.

Ниже приведен один из разделов панели администратора KeyCloak.



# Структура базы данных



# Заключение

В процессе обучения бело реализовано WEB-приложение «Система автоматизации работы медицинского учреждения». Были выполнены все поставленные задачи, а именно:

* Проработаны базовые требования к функционалу проекта
* Разработаны необходимые методы API для работы системы автоматизации записи на прием к врачу
* Код покрыт тестами более 80% и проверен статическим анализатором
* Подготовлена документация в виде описания методов в Swagger
* Настроена система авторизации и аутентификации пользователей
* Настроен CI/CD с использованием Docker и Gitlab

Поскольку приложение находится на начальной стадии разработки, имеется возможность для расширения функционала. Например, благодаря заложенному инструментарию возможно реализовать ролевую модель доступа к API. Приложение можно доработать для возможности использования несколькими клиниками или же внедрить систему платности предоставляемых услуг.

# 

# Литература

«Docker Hub» [Электронный ресурс]: официальное руководство – Режим доступа: <https://hub.docker.com/>

«Keycloak documentation» [Электронный ресурс]: официальное руководство – Режим доступа: <https://www.keycloak.org/documentation>

«Hibernate ORM 5.5.7. Final User Guide» [Электронный ресурс]: официальное руководство пользователя Hibernate. – Режим доступа: <https://docs.jboss.org/hibernate/orm/5.5/userguide/html_single/Hibernate_User_Guide.html>

«Spring Framework Documentation» [Электронный ресурс]: официальная документация фреймворка Spring. – Режим доступа: <https://docs.spring.io/spring-framework/docs/current/reference/html/>

«Spring Guides» [Электронный ресурс]: официальное руководство для фреймворка Spring. – Режим доступа: <https://spring.io/guides>

«Overview of JPA/Hibernate Cascade types» [Электронный ресурс]: портал для разработчиков. – Режим доступа: <https://www.baeldung.com/jpa-cascade-types>

«StackOverflow.com» [Электронный ресурс]: портал для разработчиков. – Режим доступа: <https://stackoverflow.com/>