****

**Московский авиационный институт**

**(Национальный исследовательский университет)**

**Институт № 3**

**Кафедра 311**

**Проектирование информационных систем**

**Лабораторная работа № 4**

**Выполнил студент  
Плотников Антон Сергеевич**

**Группа М3З-201-БК**

**Дата 12.04.2021 г.**

**Принял преподаватель  
Смирнов Владимир Юрьевич**

**Содержание**

[Цель лабораторной работы 3](#_Toc69150911)

[Глава 1. Авторизация пользователей 4](#_Toc69150912)

[Глава 2. Отображение списка сотрудников 6](#_Toc69150913)

[Глава 3. Используемое ПО и схема БД 7](#_Toc69150914)

[Вывод 8](#_Toc69150915)

[Список литературы 9](#_Toc69150916)

# Цель лабораторной работы

Изучить методы ограничения доступа к базам данных на основе авторизации пользователей. Научиться создавать приложения с разграничением доступа.

Необходимо реализовать 4 набора прав по следующим критериям:

1. Директор. Есть все права (любые изменения списка сотрудников: добавление, изменение, удаление). Вход по паролю.
2. Заместитель директора. Есть все права, кроме удаления и добавления сотрудников. Вход по паролю.
3. Секретарь. Права на просмотр всех данных, нет прав на изменение любых данных. Вход по паролю.
4. Гость. Просмотр только фамилии, имени, отчества каждого сотрудника, должности, рабочего телефона, без возможности просмотра адреса и личного телефона. Вход без пароля.

# Авторизация пользователей

Для выполнения поставленной задачи было разработано вэб-приложение, позволяющее пользователям авторизоваться и в зависимости от их роли, получить права на работу со списком сотрудников (Рисунок 1.1).

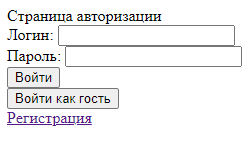


Рисунок 1.1 – Страница авторизации

Также была реализована возможность регистрации новых пользователей по нажатию ссылки «Регистрация» (Рисунок 1.2).

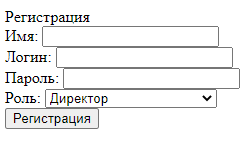


Рисунок 1.2 – Страница регистрации нового пользователя

Во время регистрации пользователю присваивается роль, выбранная в выпадающем списке «Роль» (Рисунок 1.3)

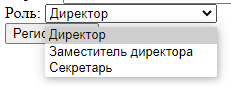


Рисунок 1.3 – Выпадающий список ролей

Помимо регистрации существует возможность авторизации как Гостя. Для этого необходимо нажать на кнопку «Войти как гость».

После авторизации, осуществляется перевод на страницу со списком сотрудников, где в заголовке отображается имя пользователя и его роль (Рисунок 1.4)

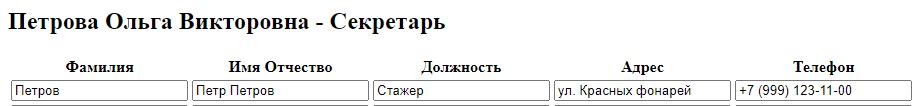


Рисунок 1.4 – Страница со списком сотрудников

# Отображение списка сотрудников

Работа со списком сотрудников ведётся в зависимости от роли пользователя.

1. Секретарь имеет права на просмотр всех данных, но не имеет прав на редактирование (Рисунок 2.1);

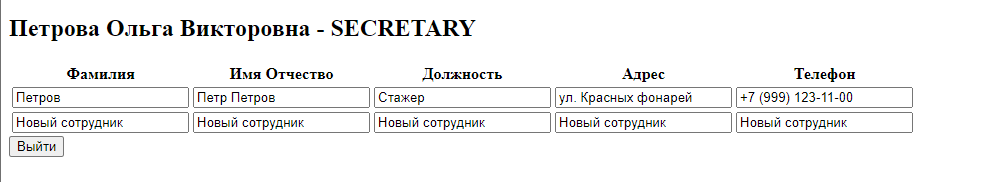


Рисунок 2.1 – Страница списка сотрудников под ролью секретаря

1. Заместитель директора имеет права на просмотр всех данных и редактирование существующего списка (Рисунок 2.2);

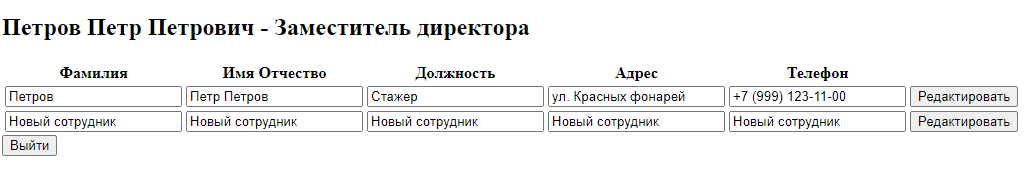


Рисунок 2.2 - Страница списка сотрудников под ролью заместителя директора

1. Директор имеет права на просмотр всех данных, редактирование, удаление и добавление новых сотрудников (Рисунок 2.3);

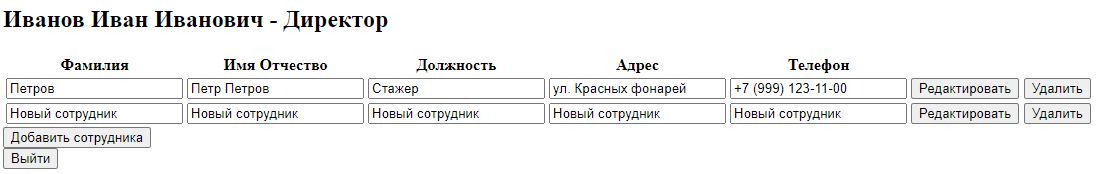


Рисунок 2.3 – Страница списка сотрудников под ролью директора

1. Гость имеет права на просмотр Фамилии, Имени, Отчества и Должности сотрудника (Рисунок 2.4).

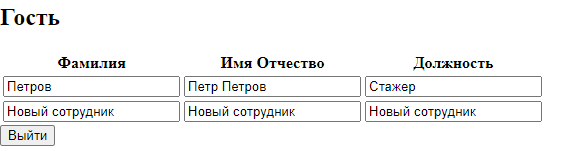


Рисунок 2.4 – Страница списка сотрудников под ролью гостя

# Используемое ПО и схема БД

Интерфейс написан на HTML5 с использованием фреймворка Apache Freemarker.

Серверная часть написана на языке Java 1.8 с использованием фреймворка Spring Boot для работы вэб-приложения.

Используемая база данных – MySQL 8.0.23

* Таблица table\_user – хранит информацию о пользователях

create table table\_user  
(  
 id bigint auto\_increment  
 primary key,  
 name varchar(255) not null,  
 password varchar(255) not null,  
 username varchar(255) not null  
);

* Таблица table\_user\_role – хранит информацию о ролях пользователей

create table table\_user\_role  
(  
 user\_id bigint null,  
 roles varchar(255) not null,  
 constraint table\_user\_role\_table\_user\_id\_fk  
 foreign key (user\_id) references table\_user (id)  
 on update cascade on delete cascade  
);

* Таблица table\_employee – хранит информацию о сотрудниках

create table table\_employee  
(  
 id bigint auto\_increment  
 primary key,  
 last\_name varchar(255) null,  
 first\_middle\_name varchar(255) null,  
 position varchar(255) null,  
 address varchar(255) null,  
 phone varchar(255) null  
);

# Вывод

Мы научились создавать приложения, позволяющие предоставлять ограниченный доступ к данным, основанный на роли пользователя

# Список литературы

* <https://dev.w3.org/html5/html-author/> - Документация HTML5;
* <https://docs.oracle.com/javase/8/docs/> - Документация Java 8;
* <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/> - Документация Spring Boot;