

Министерство науки и высшего образования Российской Федерации федеральное
государственное бюджетное образовательное учреждение высшего образования
«Рыбинский государственный авиационный технический университет имени П. А.
Соловьева»

Факультет радиоэлектроники и информатики Кафедра вычислительных систем

КУРСОВАЯ РАБОТА

по дисциплине: "Промышленная разработка приложений на основе java технологий"
на тему: "Сервис по поиску свободных кабинетов"

Студент гр ИВМ-19

Каленов А. С.

Преподаватель

Волков М. Л.

Рыбинск 2020

Содержание

Введение	3
1 Поставновка задачи	4
2 Разработка структуры классов	5
3 Разработка фронтэнда	7
4 Разработка бэкэнда	10
Заключение	11
Список использованных источников	12

Введение

Многие предприятия используются программное обеспечение являющиеся специфическим и используется только на них. Из-за этого создается благоприятная среда для развития бизнеса по производству программного обеспечения.

Среди многообразия языков программирования язык "Java" занимает позицию пусть и не самого высокопроизводительного языка, но самого безопасного и универсального.

Целью данного курсового проекта является создание ВЕБ-приложения позволяющее посмотреть расписание университета для студента, преподавателя и кабинета, а также дает возможность преподавателям отменять и назначать пары

1 Постановка задачи

Рассмотрим потребности целевой аудитории. Студентам нужно узнать расписание. Преподавателям узнать расписание и, возможно, изменить

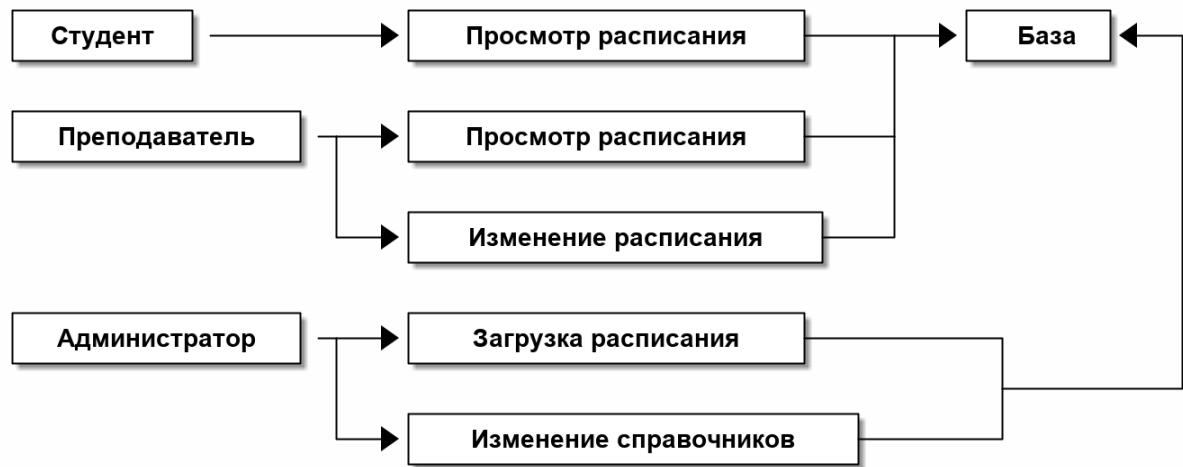


Рисунок 1. Взаимодействие сервиса с клиентами

2 Разработка структуры классов

Основной частью проекта является расписание, поэтому основной класс хранимый в базе данных и будет это самое расписание, которое состоит из элементов типа "Пара".

Пара из себя представляет встречу преподавателя, группы и студентов в одном кабинете, что создает еще три класса "Кабинет", "Группа", "Преподаватель".

Кабинеты объединяются в корпуса, а группы состоят из студентов. Студенты, преподаватели и администратор объединяют в "Пользователя".

Как было рассмотрено ранее, в приложении будет три роли : "Студент", "Преподаватель", "Администратор". Чтобы определить пользователя необходимо произвести вход в систему. Также необходима регистрация новых пользователей. Для этих целей создаем первую страничку. В итоге получаем структуру классов изображенную на рисунке 2.

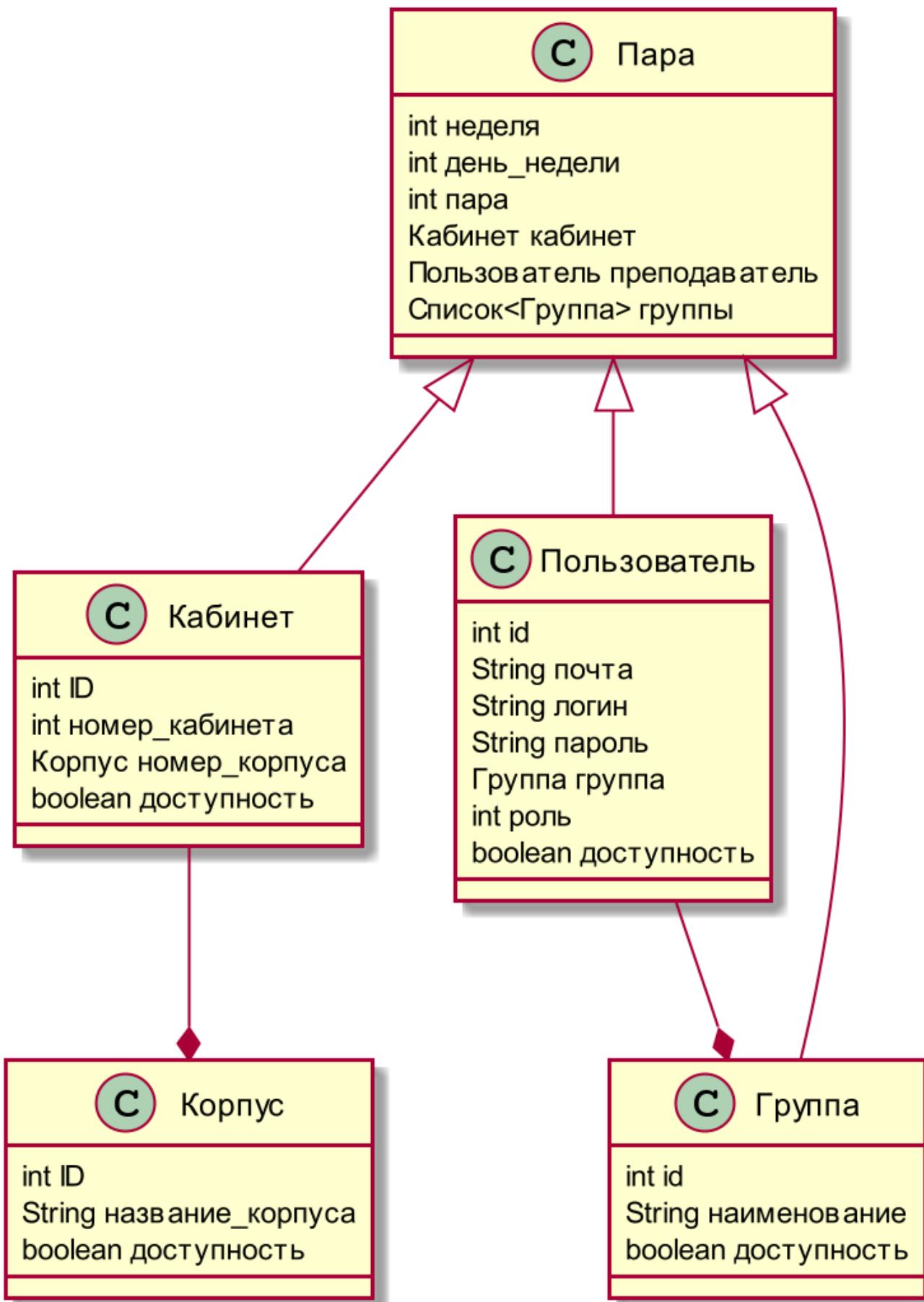
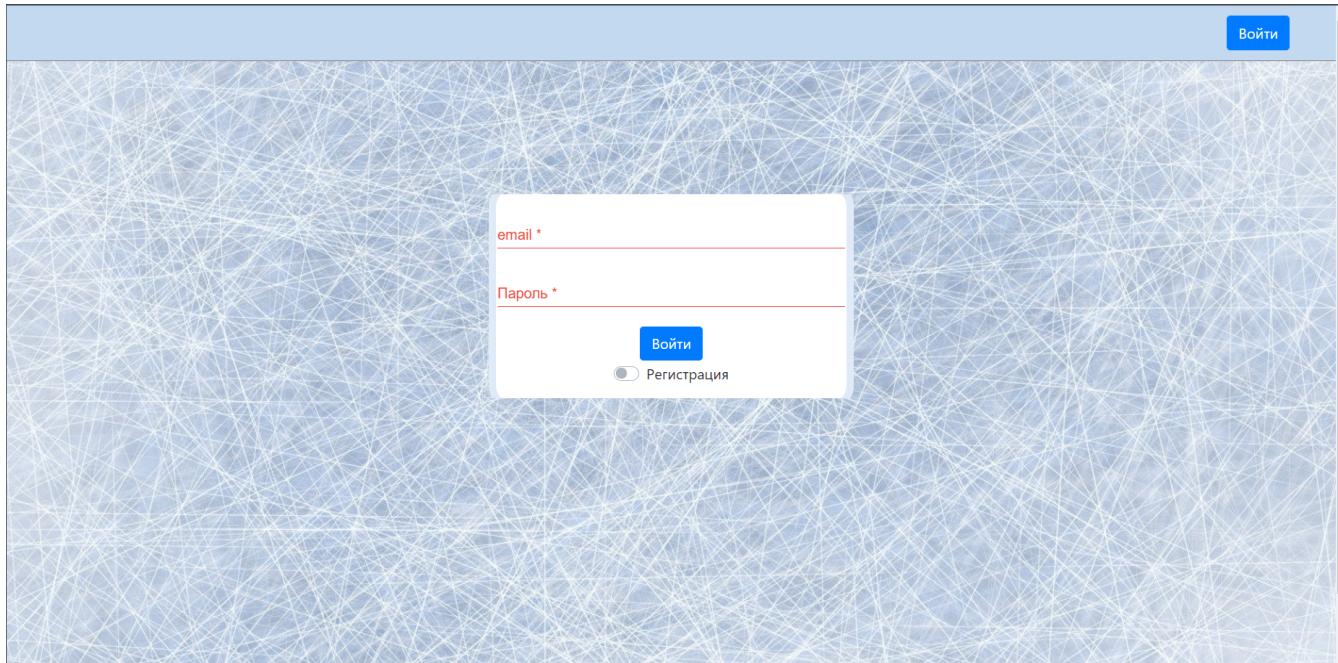


Рисунок 2. Диаграмма классов

3 Разработка фронтэнда

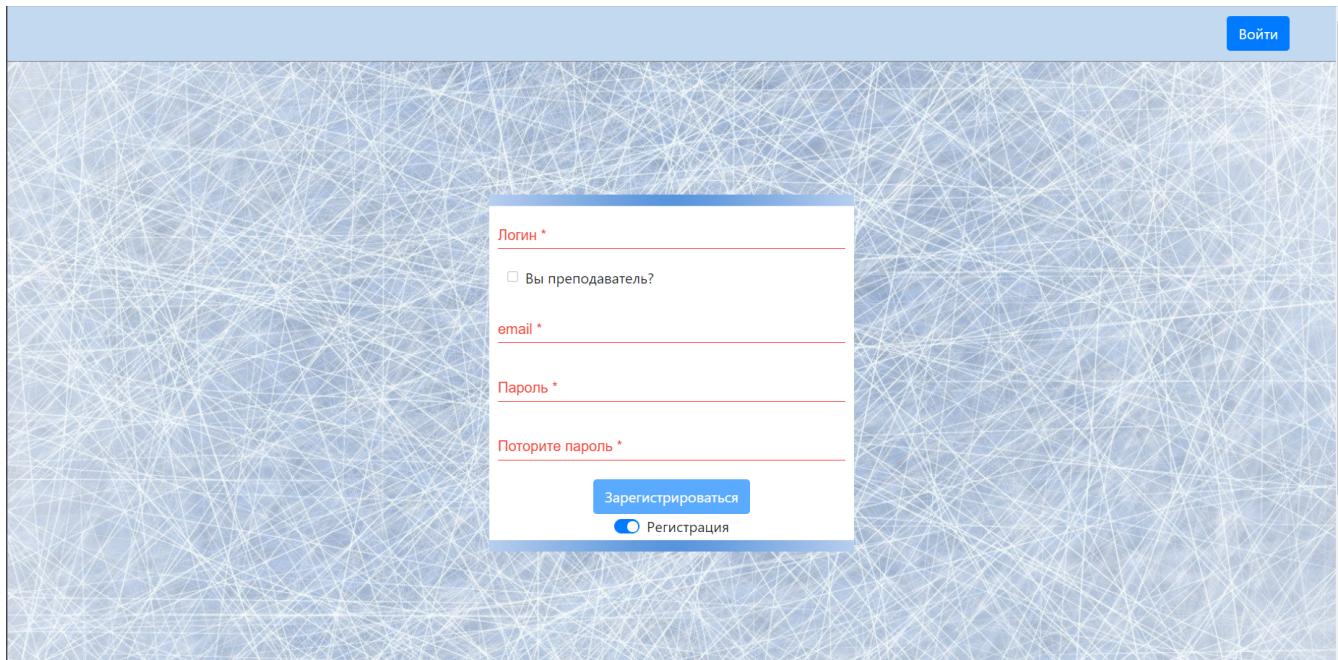
Разработку клиентской части будем производить а основе технологии "Angular".

Как было рассмотрено ранее, в приложении будет три роли: "Студент", "Преподаватель", "Администратор". Чтобы определить пользователя необходимо произвести вход в систему. Также необходима регистрация новых пользователей. Для этих целей создаем первую страничку.



The screenshot shows a login form on a light blue background. At the top right is a blue button labeled "Войти" (Login). The form itself is white and contains two input fields: "email *" and "Пароль *". Below these fields is a blue "Войти" button. Underneath the button is a small text "Регистрация" next to a radio button.

Рисунок 3. форма входа



The screenshot shows a registration form on a light blue background. At the top right is a blue button labeled "Войти" (Login). The form is white and includes several fields: "Логин *", a checkbox for "Вы преподаватель?", "email *", "Пароль *", and "Повторите пароль *". Below these fields are two buttons: a blue "Зарегистрироваться" (Register) button and a smaller "Регистрация" button next to a radio button.

Рисунок 4. форма регистрации

После отправки логина и пароля или данных форм регистрации от сервера идет ответ состоящий из логина и токена, эти данные хранятся в сессионном хранилище и позволяют получить доступ к рабочим местам пользователя. За защиту от несанкционированного доступа к этим страницам отвечает сервис AuthGuard. Если неопознанный пользователь

попытается попасть на рабочее место, его отправит на страницу входа в систему.

Далее рассмотрим макеты рабочих мест администратора, студента и преподавателя. Студенту необходимо лишь отображение таблицы расписания с некоторыми фильтрами, преподавателю необходимо тоже самое и дополнительно пару кнопок "Создать пару" и "Отменить пару".

The screenshot shows a web-based application interface for teachers. At the top, there are two input fields: 'Преподаватель' (NoName) and 'Группа' (NoName). Below these are two buttons: 'Четные' and 'Нечетные'. Underneath are two more buttons: 'Все' and 'Никакие'. A list of numbers from 1 to 18 is displayed, each with a checkbox. Some checkboxes are checked (e.g., 1, 3, 5, 7, 9, 13, 15). Below this are three dropdown menus: 'День' (Day), 'Пара' (Pair), and 'Корпус' (Building). The 'Корпус' dropdown shows '101'. At the bottom left is a 'Кабинет' section with '101' and two checkboxes: 'Свободные' and 'С компьютером'. On the right is a large table titled 'Show 10 entries' with columns: Неделя (Week), День (Day), Пара (Pair), Кабинет (Cabinet), Комп (Comp), Спр (Spr), Занят (Occupied), Парау (Pairu), and Дополнительно (Additional). The table contains 15 rows of data. At the bottom of the table are navigation links: 'Showing 1 to 10 of 150 entries', 'Previous', '1' (highlighted), '2', '3', '4', '5', '...', '15', and 'Next'. The top right corner of the window has the text 'xaleonn'.

Рисунок 5. рабочее место преподавателя и студента

The screenshot shows a web-based application interface for administrators. At the top, it says 'Вы вошли как: Войтибыстро'. There are two buttons: 'Административное место' (Administrative place) and 'Рабочее место' (Working place). The main area has four blue horizontal buttons: 'Корпуса' (Buildings), 'Группы' (Groups), 'Пользователи' (Users), and 'Кабинеты' (Cabinets). Below these is a form with a 'Выберите файл' (Select file) button, a text input 'Поиск свободных кабинетов.html', a 'Browse' button, and a green 'Отправить' (Send) button.

Рисунок 6. рабочее место администратора

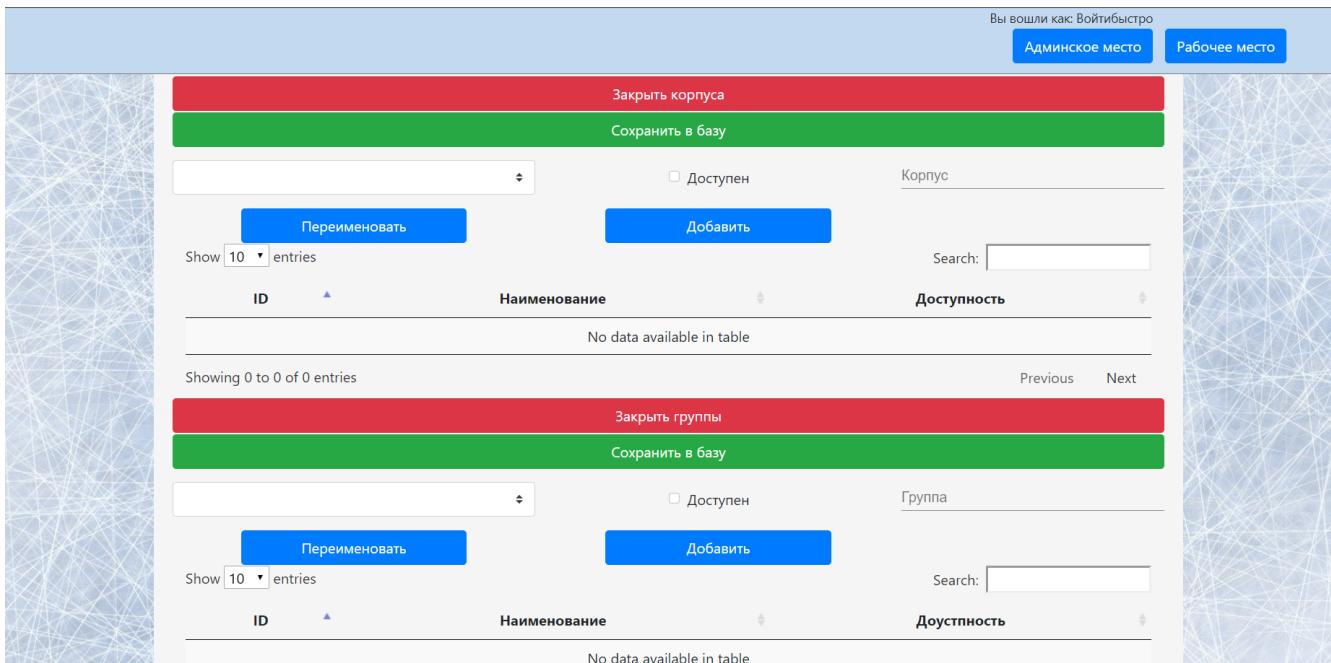


Рисунок 7. рабочее место администратора

В качестве элементов, из которых собираем страницу берем элементы из библиотеки mdb и дополнительно таблицы Angular DataTables. Данные элементы имеют привлекательный внешний вид и дополнительные возможности, не предусмотренные в стандартных HTML компонентах.

4 Разработка бэкэнда

Серверная часть будет разрабатываться на языке Java с использованием технологий Quarkus и Hibernate.

Технология Quarkus позволяет с легкостью запустить REST сервис, который будет отвечать на запросы со стороны клиента. После старта REST сервис готов получать и отправлять запросы. Одним из первых запросов будет POST запрос ссылающийся на .xls файл, который содержит расписание. Далее на основании этого файла на серверной части создаются записи в базе данных. В последствии эти записи смогут отредактировать администратор.

Технология Hibernate позволит с легкостью обмениваться данными с базой данных. В нашем случае в базе данных хранятся классы описанные ранее.

Заключение

В процессе работы над данным курсовым проектом мы использовали такие технологии как Angular, Hibernate, Quarkus. Написание пояснительной записи выполняли на языке разметки ASCIIDOC и в последующем конвертировали в файл .pdf.

Список использованных источников

1. Material Design for Bootstrap [Электронный ресурс]: Электрон., текстовые и граф. дан. – URL: <https://mdbootstrap.com/>
2. ASCIIDOC User Manual [Электронный ресурс]: Электрон., текстовые и граф. дан. – URL: <https://asciidoc.org/docs/user-manual/#user-biblio>
3. Angular Datatables [Электронный ресурс]: Электрон., текстовые и граф. дан. – URL: <https://www.datatables.net/>