

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

гр ПИМ-19

Преподаватель

Каленов А. С.

Головкин Д. М.

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

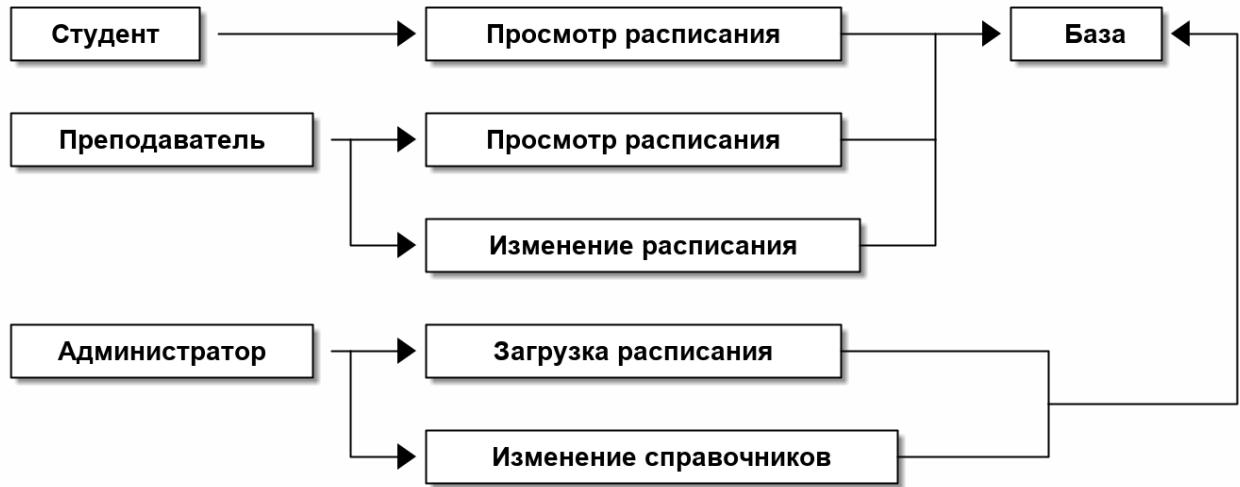


Рисунок 1. Взаимодействие сервиса с клиентами

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 объединяются 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.

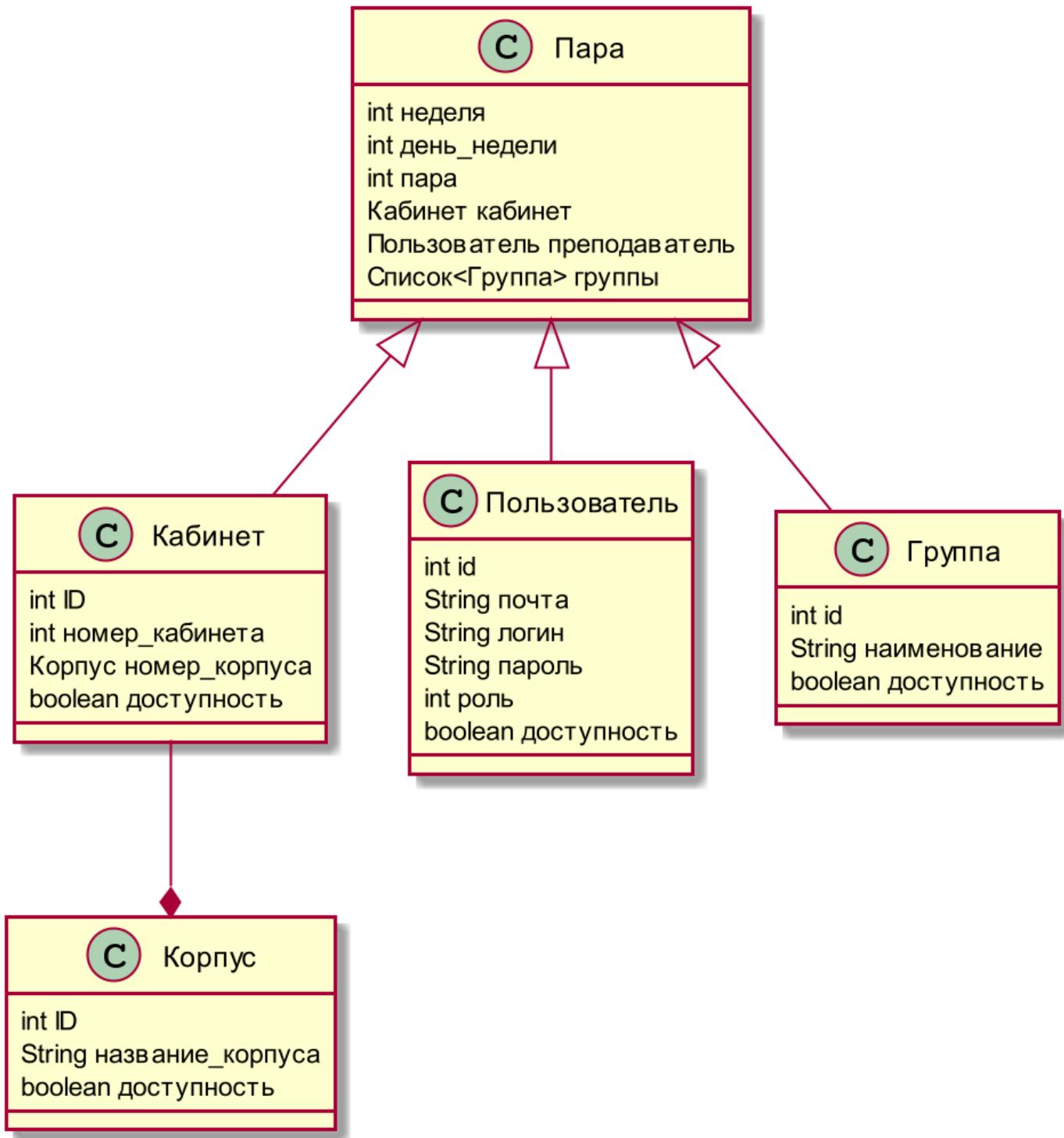
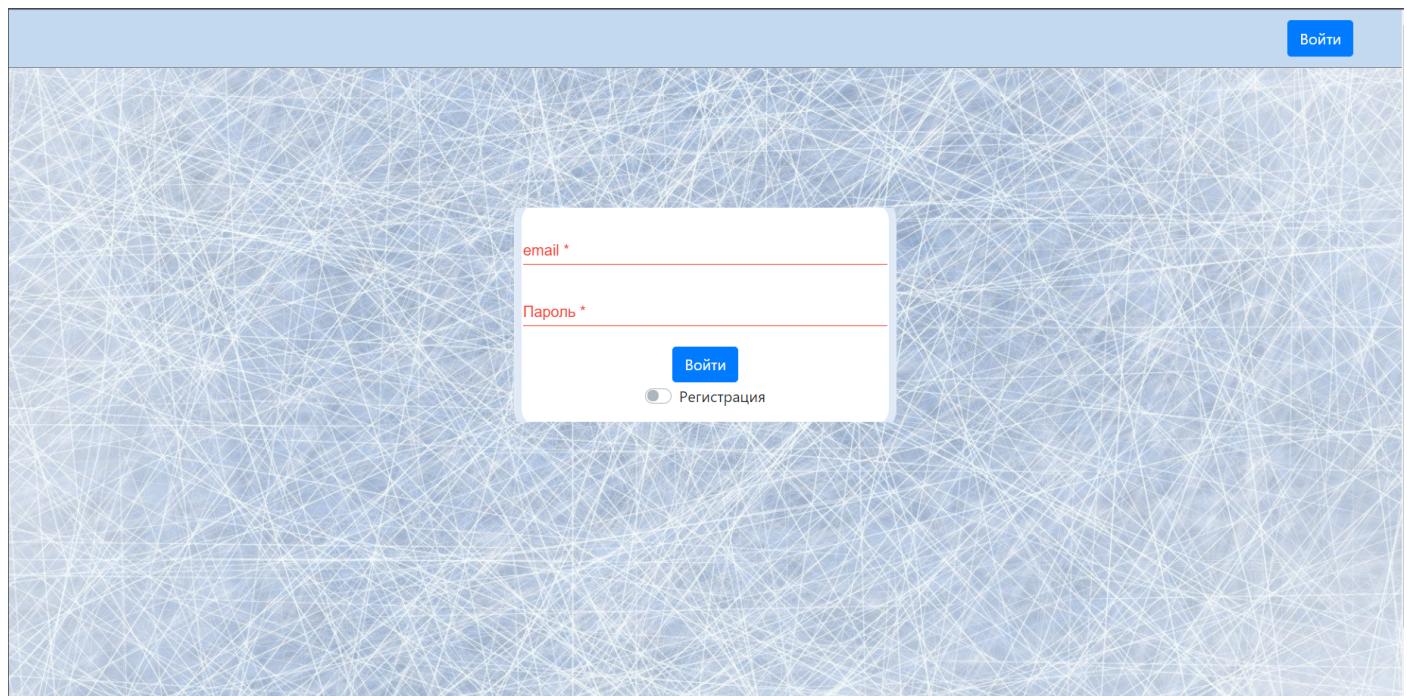


Рисунок 2. Диаграмма классов

3 Разработка фронтэнда

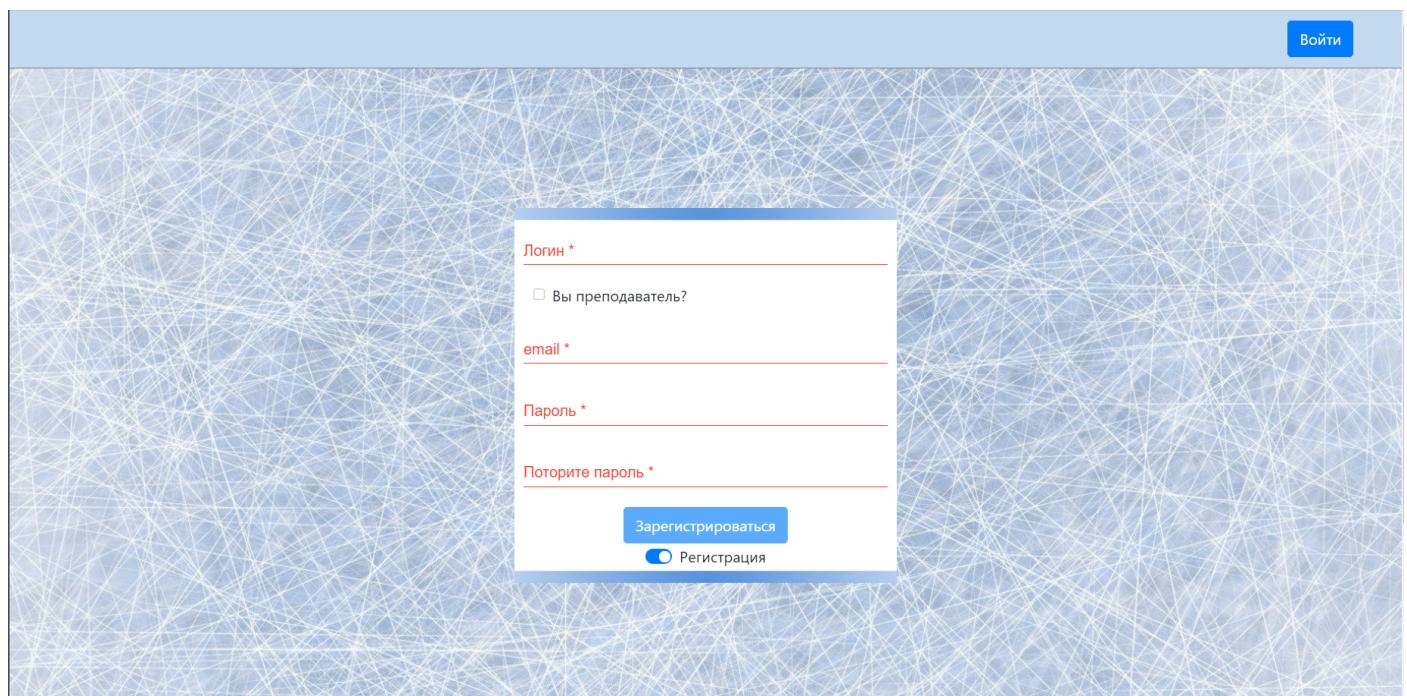
Разработку клиентской части будем производить а основе технологии "Angular".

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 image 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 is a blue "Войти" button. Underneath the button is a small link "Регистрация".

Рисунок 3. форма входа



The image 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 "Потопите пароль *". Below these fields are two buttons: a blue "Зарегистрироваться" (Register) button and a smaller "Регистрация" link.

Рисунок 4. форма регистрации

После отправки логина и порала 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 отправят 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".

xaleonn

Педодатель
NoName

Группа
NoName

1 2 3
 4 5 6
 7 8 9
 10 11 12
 13 14 15
 16 17 18

Кабинет
101

Свободные
 С компьютером

Show 10 entries

Search:

Неделя	День	Пара	Кабинет	Комп	Спр	Занят	Пару	Дополнительно
1	1	1	0	false	false	false	0	0
15	1	1	14	false	false	false	14	14
11	1	1	28	false	false	false	28	28
7	1	1	42	false	false	false	42	42
3	1	1	56	false	false	false	56	56
17	1	1	70	false	false	false	70	70
13	1	1	84	false	false	false	84	84
9	1	1	98	false	false	false	98	98
5	1	1	112	false	false	false	112	112
1	1	1	126	false	false	false	126	126

Showing 1 to 10 of 150 entries

Previous 1 2 3 4 5 ... 15 Next

Рисунок 5. рабочее место преподавателя и студента

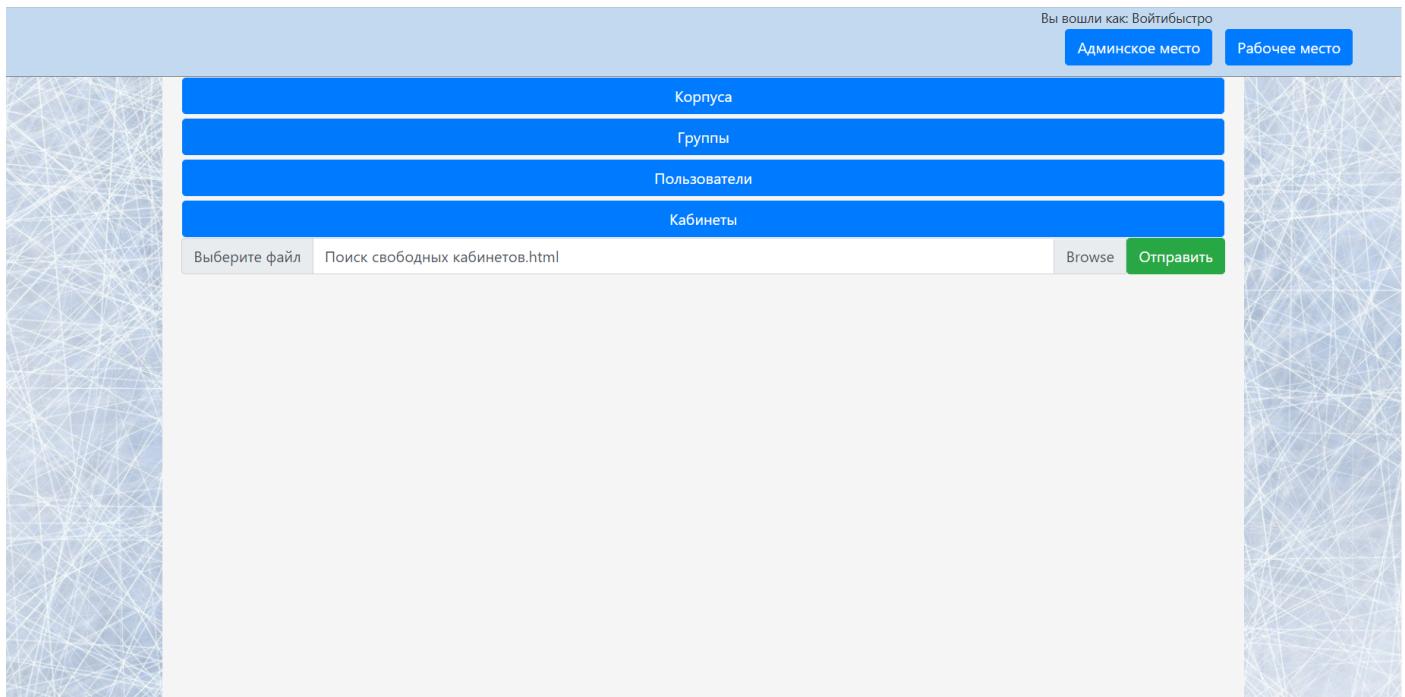


Рисунок 6. рабочее место администратора

Закрыть корпуса
Сохранить в базу

Корпус

Переименовать Добавить

Show 10 entries Search:

ID	Наименование	Доступность
No data available in table		

Showing 0 to 0 of 0 entries Previous Next

Закрыть группы
Сохранить в базу

Группа

Переименовать Добавить

Show 10 entries Search:

ID	Наименование	Доступность
No data available in table		

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 сможет 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/>