

УТВЕРЖДАЮ

директор АНПОО «Образовательные технологии  
«Скилбокс (Коробка навыков)»

Д. Р. Халилов

---

## **РАБОЧАЯ ПРОГРАММА ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ**

по профессиональному модулю

### **ПМ.08 Разработка дизайна веб-приложений**

образовательной программы «Веб-разработка на Java» среднего  
профессионального образования – программы подготовки  
специалистов среднего звена по специальности 09.02.07  
Информационные системы и программирование

Квалификация: разработчик веб и мультимедийных приложений

Форма обучения: очно-заочная

## **СОДЕРЖАНИЕ**

1. ПАСПОРТ ПРОГРАММЫ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ
2. СТРУКТУРА И СОДЕРЖАНИЕ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ
3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ

# **1. ПАСПОРТ ПРОГРАММЫ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ**

## **1.1. Область применения программы производственной практики**

Программа производственной практики профессионального модуля ПМ.08 Разработка дизайна веб-приложений является составной частью программы подготовки специалистов среднего звена по специальности СПО в соответствии с ФГОС по специальности 09.02.07 Информационные системы и программирование в части освоения основного вида профессиональной деятельности “Разработка дизайна веб-приложений”.

## **1.2. Цели и задачи программы производственной практики — требования к результатам освоения программы производственной практики**

В результате прохождения производственной практики студент должен освоить основной вид деятельности “Разработка дизайна веб-приложений” и соответствующие ему профессиональные компетенции.

### **1.2.1 Перечень профессиональных компетенций:**

ПК-8.1. Разрабатывать дизайн-концепции веб-приложений в соответствии с корпоративным стилем заказчика.

ПК-8.2. Формировать требования к дизайну веб-приложений на основе анализа предметной области и целевой аудитории.

ПК-8.3. Осуществлять разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки.

### **1.2.2. Результаты прохождения производственной практики**

С целью овладения указанным видом профессиональной деятельности и соответствующими профессиональными компетенциями обучающийся в ходе освоения программы производственной практики должен:

**1. В соответствии с минимальными требованиями к результатам освоения вида деятельности “Разработка дизайна веб-приложений” по ФГОС:**

а. **Приобрести практический опыт** в разработке дизайна веб-приложений в соответствии со стандартами и требованиями заказчика; создании, использовании и оптимизировании изображений для веб-приложений; разработке интерфейса пользователя для веб-приложений с использованием современных стандартов.

б. **Уметь** создавать, использовать и оптимизировать изображения для веб-приложений; выбирать наиболее подходящее для целевого рынка дизайнерское решение; создавать дизайн с применением промежуточных эскизов, требований к эргономике и технической эстетике; разрабатывать интерфейс пользователя для веб-приложений с использованием современных стандартов.

с. **Знать** нормы и правила выбора стилистических решений; современные методики разработки графического интерфейса; требования и нормы подготовки и использования изображений в информационно-телекоммуникационной сети "Интернет" (далее - сеть Интернет); государственные стандарты и требования к разработке дизайна веб-приложений.

2. **В соответствии с Перечнем знаний, умений, навыков в соответствии со спецификацией стандарта демонстрационного экзамена КОД 1.4 раздел 3 "Графический дизайн веб-страниц", специалист должен знать и понимать:**

структуру и общепринятые элементы веб-страниц различных видов и назначений; вопросы, связанные с когнитивными, социальными, культурными, технологическими и экономическими условиями при разработке дизайна; основные принципы создания дизайна по предоставляемым инструкциям и спецификациям; основные принципы организации контента веб-приложения; основные правила выбора цвета, работы с типографикой и композицией; принципы и методы создания и адаптации графики для использования ее на веб-сайтах; правила поддержания фирменного стиля, бренда и стиливых инструкций; ограничения, которые накладывают мобильные устройства и разрешения экранов при использовании их для просмотра веб-сайтов; принципы построения эстетичного и креативного дизайна; современные стили и тенденции дизайна.

**1.3. Количество часов на освоение программы производственной практики**

Программа производственной практики ПМ.08 Разработка дизайна веб-приложений рассчитана на прохождение студентами практики в объеме 100 часов.

Распределение разделов и тем по часам приведено в тематическом плане.

## 2. СТРУКТУРА И СОДЕРЖАНИЕ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ

Практическая подготовка осуществляется в «ОТ «Скилбокс» на образовательной платформе <https://go.skillbox.ru/> исключительно с применением дистанционных образовательных технологий и электронного обучения.

### 2.1. Объем производственной практики и виды производственной работы

Виды работ	Объем часов	из них в форме практической подготовки
<b>Производственная практика (всего)</b>	<b>125</b>	<b>125</b>
в том числе:		
• ознакомительная деятельность	8	8
• практическая деятельность	109	109
• оформление отчетной документации	8	8
Промежуточная аттестация в форме <b>дифференцированного зачета</b>		

## 2.2. Тематический план и содержание производственной практики

Наименование разделов, виды деятельности	Содержание производственной практики	Кол-во часов
Раздел 1. Ознакомительная часть практики	Цель и задачи практики. Выдача заданий на практику. Характеристика и общие сведения о проекте.	<b>4</b>
	Инструктаж обучающихся по ознакомлению с требованиями охраны труда, техники безопасности, пожарной безопасности, а также правилами внутреннего трудового распорядка.	<b>4</b>
Раздел 2. Практическая деятельность	<p><b>Виды работ.</b></p> <ol style="list-style-type: none"> <li>1. Ознакомление с комплексным заданием в формате кейса (Образец задания приведен в <a href="#">Приложении 1</a>)</li> <li>2. Выделение в комплексном задании видов деятельности и задач входящих в текущую производственную практику (Разработка дизайна веб-приложений)</li> <li>3. Выполнение выделенных из комплексного задания задач.</li> <li>4. Защита результатов выполнения задач.</li> </ol>	<b>109</b>
Раздел 3. Оформление отчетной	Обобщение материалов практики, подготовка отчета по практике (в т.ч. Дневник)	<b>6</b>
	Сдача отчета по практике. Дифференцированный зачет.	<b>2</b>

документ ации по практике		
		<b>125</b>

В соответствии с положением о практической подготовке обучающиеся:

- направляются на практику в соответствии с распорядительным актом образовательной организации;
- при направлении на практику обучающиеся получают установочные данные: цели и задачи практики, а также индивидуальное или групповое задание по освоению компетенций, соответствующих виду деятельности “Разработка дизайна веб-приложений”;
- проходят инструктаж обучающихся по ознакомлению с требованиями охраны труда, техники безопасности, пожарной безопасности, а также правилами внутреннего трудового распорядка;
- в период практики выполняют полученное задание и оформляют отчет (форму отчета и конкретное задание обучающиеся получают при направлении на практику);
- по окончании практики обучающиеся защищают отчет о прохождении практики преподавателю.

### **3. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ**

#### **3.1. Требования к минимальному материально-техническому обеспечению**

Производственная практика проводится в АНПОО «Образовательные технологии «Скилбокс» (Коробка навыков)», а также в организациях по профилю специальности 09.02.07 Информационные системы и программирование на основе договоров, включая сетевые, заключаемых между «ОТ «Скилбокс» и этими организациями.

Практическая подготовка реализуется на основании договора о практической подготовке обучающихся № 14062023 от 14 июня 2023 года с Обществом с ограниченной ответственностью «ВК».

Производственную практику можно проводить как концентрированно, так и рассредоточено.

Оборудование организаций/предприятий должно включать индивидуальные персональные компьютеры.

1. Операционная система Windows, Linux, MacOS официально поддерживаемая разработчиком на момент прохождения практики
2. Браузер Google Chrome, версии последней на момент прохождения практики
3. Учетная запись Google
4. Офисный пакет Google Workspace

#### **3.2. Информационное обеспечение обучения, организации и проведения производственной практики**

##### **3.2.1 Нормативные акты:**

- Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»

- Приказ Министерства науки и высшего образования Российской Федерации № 885, Министерства просвещения Российской Федерации № от 390 05.08.2020 "О практической подготовке обучающихся". (Зарегистрирован в Минюсте Российской Федерации 11.09.2020 № 59778)

- федеральный государственный образовательный стандарт среднего профессионального образования по профессии 09.02.07 Информационные системы и программирование.



- Положение о формах, периодичности и порядке проведения текущего контроля успеваемости и промежуточной аттестации обучающихся «ОТ «Скилбокс».

### **3.2.2 Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы.**

#### 3.2.2.1. Основные электронные издания (электронные ресурсы)

1. Немцова, Т. И. Компьютерная графика и web-дизайн : учебное пособие / Т.И. Немцова, Т.В. Казанкова, А.В. Шнякин ; под ред. Л.Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2023. — 400 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-8199-0790-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1905248> (дата обращения: 07.04.2023). – Режим доступа: по подписке.
2. Немцова, Т. И. Практикум по информатике. Компьютерная графика и web-дизайн : учебное пособие / Т.И. Немцова, Ю.В. Назарова ; под ред. Л.Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2023. — 288 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-8199-0800-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1908342> (дата обращения: 07.04.2023). – Режим доступа: по подписке.

#### 3.2.2.2. Дополнительные источники

1. Дадьянова И.Б. Современные технологии анимации в веб-дизайне // Культура и искусство. 2022. №2. URL: <https://cyberleninka.ru/article/n/sovremennye-tehnologii-animatsii-v-veb-dizayne> (дата обращения: 07.04.2023).
2. Исмойилов Х.Б. Стили сайтов в веб-дизайне // Современные материалы, техника и технологии. 2018. №2 (17). URL: <https://cyberleninka.ru/article/n/stili-saytov-v-veb-dizayne> (дата обращения: 07.04.2023).
3. Калугина Ю.В., Кондакова А.А., Михайлов А.С., Стрельникова С.В. Роль цвета в веб-дизайне // Решетневские чтения. 2018. №. URL: <https://cyberleninka.ru/article/n/rol-tsveta-v-veb-dizayne> (дата обращения: 07.04.2023).

4. Китаевская Т.Ю. Альтернативные стили в веб-дизайне // Вестник российских университетов. Математика. 2014. №2. URL: <https://cyberleninka.ru/article/n/alternativnyye-stili-v-veb-dizayne> (дата обращения: 07.04.2023).
5. Морозов М.Д., Романов В.В. Роль эстетики в веб-дизайне // Вестник АГТУ. 2019. №2 (68). URL: <https://cyberleninka.ru/article/n/rol-estetiki-v-veb-dizayne> (дата обращения: 07.04.2023).

#### 4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ

Контроль и оценка результатов освоения производственной практики осуществляется руководителем практики от предприятия и руководителем практики от образовательной организации.

<b>Код и наименование профессиональных и общих компетенций, формируемых в рамках производственной практики</b>	<b>Основные показатели оценки результата</b>	<b>Формы и методы контроля и оценки</b>
ПК-8.1. Разрабатывать дизайн-концепции веб-приложений в соответствии с корпоративным стилем заказчика.	<b>Оценка “отлично”:</b> <ul style="list-style-type: none"><li>● Разработаны дизайн-концепции веб-приложений в соответствии с корпоративным стилем заказчика.</li><li>● Через защиту результатов и отчет о практике продемонстрирована полная сформированность требуемых знаний, умений, практического опыта:</li><li>● Цели, задачи и результаты производственной практики полностью выполнены.</li></ul> <b>Оценка “хорошо”:</b> <ul style="list-style-type: none"><li>● Разработаны дизайн-концепции веб-приложений в соответствии с корпоративным стилем заказчика.</li><li>● Через защиту результатов и отчет о практике</li></ul>	Защита результатов выполнения задания; отчет о практике; интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики.

	<p>продемонстрировано сформированность большей части требуемых знаний, умений, практического опыта.</p> <ul style="list-style-type: none"> <li>● Цели, задачи и результаты производственной практики в целом или с незначительными ошибками выполнены.</li> </ul> <p><b>Оценка “удовлетворительно”:</b></p> <ul style="list-style-type: none"> <li>● Разработаны дизайн-концепции веб-приложений в соответствии с корпоративным стилем заказчика.</li> <li>● Через защиту результатов и отчет о практике продемонстрировано частичная сформированность требуемых знаний, умений, практического опыта.</li> <li>● Цели, задачи и результаты производственной практики частично или со значительными ошибками выполнены.</li> </ul>	
<p>ПК-8.2. Формировать требования к дизайну веб-приложений на основе анализа предметной области и целевой аудитории.</p>	<p><b>Оценка “отлично”:</b></p> <ul style="list-style-type: none"> <li>● Сформированы требования к дизайну веб-приложений на основе анализа предметной области и целевой аудитории.</li> <li>● Через защиту результатов и отчет о практике продемонстрирована полная сформированность требуемых знаний, умений, практического опыта:</li> <li>● Цели, задачи и результаты учебной практики полностью выполнены.</li> </ul> <p><b>Оценка “хорошо”:</b></p> <ul style="list-style-type: none"> <li>● Сформированы требования к дизайну веб-приложений на основе анализа предметной области и целевой</li> </ul>	<p>Защита результатов выполнения задания; отчет о практике; интерпретация результатов наблюдений за деятельностью обучающегося в процессе практики.</p>

	<p>аудитории.</p> <ul style="list-style-type: none"> <li>● Через защиту результатов и отчет о практике продемонстрировано сформированность большей части требуемых знаний, умений, практического опыта.</li> <li>● Цели, задачи и результаты учебной практики в целом или с незначительными ошибками выполнены.</li> </ul> <p><b>Оценка “удовлетворительно”:</b></p> <ul style="list-style-type: none"> <li>● Сформированы требования к дизайну веб-приложений на основе анализа предметной области и целевой аудитории.</li> <li>● Через защиту результатов и отчет о практике продемонстрировано частичная сформированность требуемых знаний, умений, практического опыта.</li> <li>● Цели, задачи и результаты учебной практики частично или со значительными ошибками выполнены.</li> </ul>	
<p>ПК-8.3. Осуществлять разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки.</p>	<p><b>Оценка “отлично”:</b></p> <ul style="list-style-type: none"> <li>● Осуществлена разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки.</li> <li>● Через защиту результатов и отчет о практике продемонстрирована полная сформированность требуемых знаний, умений, практического опыта:</li> <li>● Цели, задачи и результаты производственной практики полностью выполнены.</li> </ul> <p><b>Оценка “хорошо”:</b></p>	<p>Защита результатов выполнения задания; отчет о практике; интерпретация результатов наблюдений за деятельностью обучающегося в</p>

	<ul style="list-style-type: none"><li>● Осуществлена разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки.</li><li>● Через защиту результатов и отчет о практике продемонстрировано сформированность большей части требуемых знаний, умений, практического опыта.</li><li>● Цели, задачи и результаты производственной практики в целом или с незначительными ошибками выполнены.</li></ul> <p><b>Оценка “удовлетворительно”:</b></p> <ul style="list-style-type: none"><li>● Осуществлена разработку дизайна веб-приложения с учетом современных тенденций в области веб-разработки.</li><li>● Через защиту результатов и отчет о практике продемонстрировано частичная сформированность требуемых знаний, умений, практического опыта.</li><li>● Цели, задачи и результаты производственной практики частично или со значительными ошибками выполнены.</li></ul>	процессе практики.
--	---	--------------------

## Образец задания

### Над чем предстоит работать

Вам нужно разработать backend-составляющую системы бронирования отелей с возможностью управления контентом через административную панель CMS. Задача максимально приближена к реальной.

В ходе работы над проектом вы:

- подготовите окружение для разработки приложения;
- создадите базовые интерфейсы для работы с сущностями;
- создадите базовые сервисы для работы с репозиториями;
- настроите Spring Security в приложении;
- создадите DTO для HTTP-запроса и HTTP-ответа;
- создадите мапперы для сущностей из БД в DTO-объекты;
- обработаете ошибки на уровне контроллеров;
- создадите REST API для CRUD-операций;
- создадите методы;
- реализуете запросы;
- создадите слой, отвечающий за сбор статистики в приложении;
- создадите административную панель;
- настроите выгрузку статистических данных приложения.

### Какие материалы можно использовать

Для реализации этого проекта достаточно знаний, которые вы получили в рамках курса, и использовать внешние источники вам не придётся.

При выполнении финальной работы опирайтесь на техническое задание, в котором подробно описаны все задачи и этапы.

## **О проекте**

Вам предстоит разработать бэкенд-составляющую сервиса бронирования отелей с возможностью управлять контентом через административную панель CMS. Главная задача — обеспечить пользователям возможность забронировать понравившийся отель на определённый период. Помимо этого, нужно сделать поиск отелей по критериям и рейтингам и систему выставления оценок в приложении (от 1 до 5). Сервис должен позволять администраторам выгружать статистику по работе в формате CSV-файла.

### **Задание 1. Подготовка окружения**

- Создайте проект с необходимыми зависимостями для разработки приложения. Вам понадобится Spring, Spring Web MVC, Spring Data, Mupstruct.
- Создайте новую базу данных для приложения.
- Опишите файл `docker-compose`, который будет содержать контейнер с вашим приложением и контейнер с PostgreSQL.

### **Задание 2. Работа с отелями**

- Опишите сущность «Отель». У каждого объекта должно быть название, заголовок объявления, город, в котором расположен отель, адрес, расстояние от центра города, рейтинг (от 1 до 5) и количество оценок, на основе которых был рассчитан рейтинг.



- Создайте в базе данных таблицу для хранения информации об отелях.
- Настройте отображение данных из таблицы на поля в сущности в Java.
- Создайте интерфейс для работы с сущностью «Отель», используя Spring Data JPA. Интерфейс должен предоставлять возможность выполнять CRUD-операции.
  - Создайте сервис для работы с отелями. Сервис должен предоставлять возможность выполнять CRUD-операции.
  - Опишите объекты передачи данных DTO (Data Transfer Object), которые будут применяться в контроллерах при запросе пользователя и ответах ему.
    - Опишите DTO HTTP-запроса, который будет использоваться в слое контроллеров при запросе на создание или редактирование сущности «Отель». Обратите внимание, что рейтинг отеля и количество поставленных оценок нельзя изменить при помощи HTTP-запросов.
    - Опишите DTO HTTP-ответа, который будет использоваться в слое контроллеров для возвращения сущности «Отель».
    - Опишите DTO HTTP-ответа, который будет использоваться в слое контроллеров для возвращения списка объектов (других DTO) с пагинацией.
  - С помощью библиотеки Mapstruct создайте сервисы для маппинга сущностей в DTO и DTO в сущности:
    - из DTO на создание или редактирование в сущность «Отель»;
    - из сущности «Отель» в DTO на редактирование сущности;
    - из сущности «Отель» в DTO, описывающее сущность «Отель» (в этой DTO должен быть рейтинг отеля и количество поставленных оценок).
  - Создайте контроллер выполнения для CRUD-операций над отелями с помощью REST API. Контроллер должен содержать следующие методы:

- метод поиска по ID конкретного отеля;
- создание отеля;
- редактирование отеля;
- удаление отеля;
- получение списка всех имеющихся отелей.

### **Задание 3. Обработка ошибок**

- Опишите DTO, который вернётся в случае ошибок в сервисе.
- Опишите `advice`, который обрабатывает ошибки и вернёт их пользователям DTO с сообщением, а также корректный HTTP-код ответа с указанием:
  - на отсутствие сущности в БД — возвращение кода 404;
  - клиентские ошибки (неправильный ввод, некорректные данные) — возвращение кода 400;
  - необработанные ошибки — возвращение кода 500.

### **Задание 4. Работа с комнатами**

- Опишите сущность «Комната». Для каждого объекта указываются: название, описание, номер, цена, максимальное количество людей, которое можно разместить, и даты, когда комната недоступна. Каждая комната находится в конкретном отеле.
  - Создайте в базе данных таблицу для хранения информации о комнатах.
  - Настройте отображение данных из таблицы на поля в сущности в Java.
  - Создайте интерфейс для работы с сущностью «Комната», используя Spring Data JPA. Интерфейс должен предоставлять возможность выполнять CRUD-операции.

- Создайте сервис для работы с комнатами. Сервис должен предоставлять возможность выполнять CRUD-операции.

- Опишите объекты передачи данных DTO (Data Transfer Object), которые будут применяться в контроллерах при запросе пользователя и ответах ему.

- Опишите DTO HTTP-запроса, который будет использоваться в слое контроллеров при запросе на создание или редактирование сущности «Комната».

- Опишите DTO HTTP-ответа, который будет использоваться в слое контроллеров для возвращения сущности «Комната».

- С помощью библиотеки Mapstruct создайте сервисы для маппинга сущностей в DTO и DTO в сущности:

- из DTO на создание или редактирование в сущность «Комната»;

- из сущности «Комната» в DTO на редактирование сущности;

- из сущности «Комната» в DTO, которое описывает сущность «Комната».

- Создайте контроллер выполнения для CRUD-операций над комнатами с помощью REST API.

Контроллер должен содержать следующие методы:

- метод поиска по ID конкретной комнаты;

- создание комнаты;

- редактирование комнаты;

- удаление комнаты.

### **Задание 5. Работа с пользователями**

- Опишите сущность «Пользователь». У каждого пользователя есть уникальное имя, пароль, электронная почта и роль (пользователь или администратор).

- Создайте в базе данных таблицу для хранения информации о пользователях.
- Настройте отображение данных из таблицы на поля в сущности в Java.
- Создайте интерфейс для работы с сущностью «Пользователь», используя Spring Data JPA.

Интерфейс должен предоставлять возможность:

- выполнять CRUD-операции;
- искать пользователя по его имени;
- проверять, существует ли пользователь с указанными именем и адресом электронной почты.
- Создайте сервис для работы с пользователями.

Сервис должен предоставлять возможность:

- выполнять CRUD-операции;
- создавать нового пользователя;
- искать пользователя по имени.

- Опишите объекты передачи данных DTO (Data Transfer Object), которые будут применяться в контроллерах при запросе пользователя и ответах ему.

- Опишите DTO HTTP-запроса, который будет использоваться в слое контроллеров при запросе на создание или редактирование сущности «Пользователь».

- Опишите DTO HTTP-ответа, который будет использоваться в слое контроллеров для возвращения сущности «Пользователь».

- С помощью библиотеки Mapstruct создайте сервисы для маппинга сущностей в DTO и DTO в сущности:

- из DTO на создание или редактирование в сущность «Пользователь»;
  - из сущности «Пользователь» в DTO на редактирование сущности;

- из сущности «Пользователь» в DTO, которое описывает сущность «Пользователь».

- Создайте контроллер выполнения для CRUD-операций над пользователями посредством REST API. Контроллер должен содержать метод создания нового пользователя с указанной ролью (роль принимается как один из параметров при создании пользователя). Перед созданием нового пользователя требуется проверка по логину и почте на отсутствие уже зарегистрированного аккаунта с такими же данными.

### **Задание 6. Работа с бронированием**

- Опишите сущность «Бронирование». Она включает даты заезда и выезда, информацию о забронированной комнате и пользователе, который осуществляет бронирование.

- Создайте в базе данных таблицу для хранения информации о бронированиях.

- Настройте отображение данных из таблицы на поля в сущности в Java.

- Создайте интерфейс для работы с сущностью «Бронирование», используя Spring Data JPA. Интерфейс должен предоставлять возможность выполнять CRUD-операции.

- Создайте сервис бронирования.

Сервис должен предоставлять возможность:

- Пользователям бронировать комнаты на указанные даты. Обратите внимание, что бронируется именно комната. Забронировать комнату могут разные пользователи и несколько раз, однако только на свободные даты.

- Получать список всех оформленных броней.

- Опишите объекты передачи данных DTO (Data Transfer Object), которые будут применяться в контроллерах при запросе пользователя и ответах ему.

- Опишите DTO HTTP-запроса, который будет использоваться в слое контроллеров при запросе на бронирование.

- Опишите DTO HTTP-ответа, который будет использоваться в слое контроллеров для возвращения результата операции бронирования.

- Создайте контроллер для бронирования.

Контроллер должен содержать следующие методы:

- Для бронирования комнаты.
- Получения всех оформленных броней.

### **Задание 7. Настройка Spring Security в приложении**

Опишите все необходимые сервисы и конфигурации для настройки Spring Security в вашем приложении. Для разработки сервиса используйте Basic Auth.

- Опишите объект UserDetails, который будет содержать нужные поля и роли, а также сервис считывания пользователей из базы данных — UserDetailsService.

- Опишите базовую конфигурацию Spring Security. В вашем сервисе доступ без авторизации пользователя будет возможен только для эндпоинта регистрации.

- Измените созданные контроллеры:

- чтобы регистрация пользователя была доступна без авторизации;
- создание, редактирование и удаление отелей было доступно только администратору;
- создание, редактирование и удаление комнат было доступно только администратору;
- получение списка броней было доступно только администратору;
- все остальные методы были доступны и пользователю, и администратору, но только при авторизации.

### **Задание 8. Создание метода изменения рейтинга отеля**

Создайте метод в контроллере сущности «Отель», который позволит менять рейтинг отеля (по шкале оценивания от 1 до 5). Контроллер должен быть доступен обычному пользователю и администратору. Изменение рейтинга отеля будет происходить по следующей формуле: Если:

- rating — текущий рейтинг;
- newMark — новая оценка;
- totalRating — сумма всех оценок;
- numberOfRating — общее количество оценок.

Тогда:

- Сначала вычисляется сумма всех оценок (totalRating) путём умножения текущего рейтинга (rating) на общее количество оценок (numberOfRating):  $totalRating = rating \times numberOfRating$

- Затем вычитается текущий рейтинг (rating) из полученной суммы (totalRating) и добавляется новая оценка (newMark):  $totalRating = totalRating - rating + newMark$

- Далее вычисляется новый средний рейтинг (новое значение rating) путём деления суммы всех оценок (totalRating) на общее количество оценок (numberOfRating) с округлением до одной десятой (один знак после запятой):  $rating = totalRating / numberOfRating$  (округление до одного знака после запятой)

- Далее общее количество оценок (numberOfRating) увеличивается на единицу, так как добавилась одна новая оценка.  $numberOfRating = numberOfRating + 1$

## **Задание 9. Разработка метода, отдающего постраничную информацию об отелях с учётом фильтрации от пользователя**

- Опишите DTO, который будет отдавать список объектов с числом всех имеющихся записей в БД. Этот DTO должен использоваться при работе с пагинацией.

- Добавьте метод в контроллер для сущности «Отель» с возможностью фильтрации по следующим категориям:

- ID отеля;
- название отеля;
- заголовок объявления;
- город;
- адрес;
- расстояние до центра города;
- рейтинг и число рейтингов.

- Добавьте возможность задавать параметры пагинации на запрос по отелям (количество отелей на странице и номер страницы). Для написания фильтрации используйте JPA Specification.

### **Задание 10. Разработка метода, отдающего постраничную информацию о комнатах с учётом фильтрации от пользователя**

- Добавьте метод в контроллер для сущности «Комната» с возможностью фильтрации по следующим категориям:

- ID комнаты;
- заголовок;
- минимальная и максимальная цена;
- количество гостей в комнате;
- дата заезда и дата выезда;
- ID отеля.



- Необходимо добавить возможность задавать параметры пагинации на запрос по комнатам (количество комнат на странице и номер страницы). Для написания фильтрации используйте JPA Specification.

- Фильтрация по датам заезда и выезда должна учитывать оба поля. Если заполнено только одно из полей, фильтрация не срабатывает. При выборе дат заезда и выезда нужно показывать только те номера, которые свободны в этом временном диапазоне.

- Измените метод получения информации об оформленных бронированиях таким образом, чтобы он поддерживал пагинацию.

Пояснение.

В отеле есть номер, который забронирован на следующие даты:

- 01.01.23–10.01.23;
- 20.01.23–30.01.23.

Если в полях фильтра были указаны даты с 03.01.23 (заезд) по 11.01.23 (выезд), то номера отеля, забронированные на даты 01.01.23–10.01.23, не попадут в выборку. Если были выбраны даты с 11.01.23 (заезд) по 19.01.23 (выезд), в выборку попадут номера отеля, забронированные на даты:

- 01.01.23–10.01.23;
- 20.01.23–30.01.23.

### **Задание 11. Создание слоя, отвечающего за сбор статистики в приложении**

Создайте слой статистики, который будет работать с Kafka (получение статистической информации) и MongoDB (хранение информации). Напишите сервис внутри этого слоя с методом выгрузки статистических данных в CSV-файл. Всего в сервисе должно быть два события, которые будут обрабатываться этим слоем, — регистрация пользователя и бронирование комнаты.

- В событии регистрации должны храниться данные о созданном пользователе (его ID).
- В событии бронирования комнаты должны храниться данные о пользователе, который оформил бронь (его ID), а также о датах заезда и выезда. Данные по этим событиям нужно хранить в MongoDB в формате JSON.
- Опишите базовые события в сервисе. Создайте модели для отправки и обработки событий.
- Опишите настройки для Kafka. Добавьте сервисы Kafka и ZooKeeper в docker-compose.yml.
- Создайте сущность, репозиторий и сервис для работы со статистикой.
- Создайте сервис для выгрузки всех статистических данных в CSV-файл.
- Добавьте отправку событий при регистрации нового пользователя и бронировании комнаты.
- Добавьте контроллер для работы со статистическими данными. Контроллер должен предоставлять метод для скачивания CSV-файла с накопленными статистическими данными. Метод должен быть доступен только администратору.

### **Формат сдачи материалов и оценивание**

1. Готовую финальную работу и её промежуточные итерации (отдельно выполненные задания) принимаются в формате Git-репозитория с историей коммитов.
2. Промежуточные итерации можно обсуждать с проверяющим куратором, если у вас появятся вопросы.

### **Рекомендации по выполнению**

1. При написании сервиса (основной бизнес-логики) используйте Spring MVC и PostgreSQL.
2. При написании слоя статистики используйте MongoDB.
3. Ниже указаны связи сущностей в БД. Изображение: предоставлено Павлом Вавиловым

4. При разработке проекта опирайтесь на схему выше.
5. Полностью соблюдайте требования для сервиса, которые подробно описаны в каждом задании.

### **Как отправить финальную работу на проверку**

Отправьте финальную работу и её промежуточные итерации (отдельно выполненные задания) в формате Git-репозитория с историей коммитов через форму на платформе.