

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,
МЕХАНИКИ И ОПТИКИ”

Факультет ИКТ

Образовательная программа Интеллектуальные системы в гуманитарной сфере

Направление подготовки (специальность) Интеллектуальные системы в гуманитарной
сфере (45.03.04)

О Т Ч Е Т

по курсовой работе

Тема задания: РЕАЛИЗАЦИЯ WEB-СЕРВИСОВ СРЕДСТВАМИ Django REST framework.

Обучающийся: Ларионова Мария Владиславовна К3343

Руководитель: Говоров А. И.

Подписи членов комиссии:

_____ **Ф.И.О.**
(подпись)

_____ **Ф.И.О.**
(подпись)

_____ **Ф.И.О.**
(подпись)

Дата _____

Санкт-Петербург
2020

1. СОДЕРЖАНИЕ

1.	СОДЕРЖАНИЕ.....	2
2.	ВВЕДЕНИЕ	3
1.	АРХИТЕКТУРА	4
1.1.	Архитектура веб-сервиса	4
1.2.	Архитектура базы данных.....	6
2.	РЕАЛИЗАЦИЯ WEB-СЕРВИСА	7
2.1.	Общая информация.....	7
2.2.	Реализация интерфейса “Главная страница”	8
2.3.	Реализация интерфейса “Страница авторизации”	9
2.4.	Реализация интерфейса “Кабинет администратора”	10
2.5.	Реализация интерфейса “Кабинет служащего”	11
2.6.	Реализация интерфейса “Кабинет претендента”	12
2.7.	Реализация интерфейса “Бронирование номера”	13
2.8.	Реализация интерфейса «Подача заявки на работу»	15
2.9.	Реализация запросов.....	17
3.	ЗАКЛЮЧЕНИЕ.....	18
4.	СПИСОК ЛИТЕРАТУРЫ	19

2. ВВЕДЕНИЕ

В последнее время сфера туризма становится все более популярной, привлекая людей со всего света открывать новые города и страны. Существует множество различных сервисов, которые предлагают туристам остановиться в том или ином городе – будь то гостиницы, отели, хостелы или апартаменты. Но всех их объединяет то, что управляющим этих средств размещения нужно получать актуальную информацию о гостях и номерах, чтобы мочь отследить заполненность отеля/гостиницы, а также контролировать въезд, выезд и оплату гостей.

Цель данной курсовой работы заключалась в реализации веб-сервиса, который бы позволял администратору отеля/гостиницы/хостела отслеживать все изменения, касающиеся состояния номеров, въезда и выезда посетителей, оплаты номеров, а также самостоятельно вносить необходимые изменения.

Бэкенд сервиса был реализован при помощи Django Rest. Фронтенд был реализован при помощи Vue.js. Также в ходе работы применялась база данных PostgreSQL.

Функционал веб-сервиса может быть описан следующими возможностями:

- возможность для потенциального посетителя отеля забронировать один из свободных номеров
- возможность для администратора отслеживать изменения в статусе номеров (занят, не занят)
- возможность для администратора получать сведения об актуальном доходе отеля с занятых номеров
- возможность для администратора просматривать расписание рабочего персонала
- возможность отправить заявку на рассмотрение заявления о принятии на работу в отель
- возможность для администратора принимать или отклонить заявления о приёме на работу
- возможность для рабочего персонала просматривать график рабочих смен.

1. АРХИТЕКТУРА

1.1. Архитектура веб-сервиса

Для работы с веб-сервисом администрирования отеля был выбран Django Rest Framework. Эта библиотека работает со стандартными моделями Django и помогает создать API. С её помощью удалось реализовать отображение данных, возможность выборки запрашиваемой информации, добавление какой-либо новой информации и исполнение запросов, отвечающих требованиям функционала данного веб-сервиса.

Для работы с фронтендом был выбран Vue.js с UI-фреймворком Vuetify. Это было сделано из соображений увеличить скорость сбора данных.

PostgreSQL использовалась для работы с базами данных. Она позволила осуществить все необходимые настройки, требующиеся для базы данных.

Редактор PyCharm был задействован в основной части работы и сборки приложения. Также использовались контейнеры Docker.

Диаграмма архитектуры разрабатываемого веб-сервиса продемонстрирована на Рисунке 1.

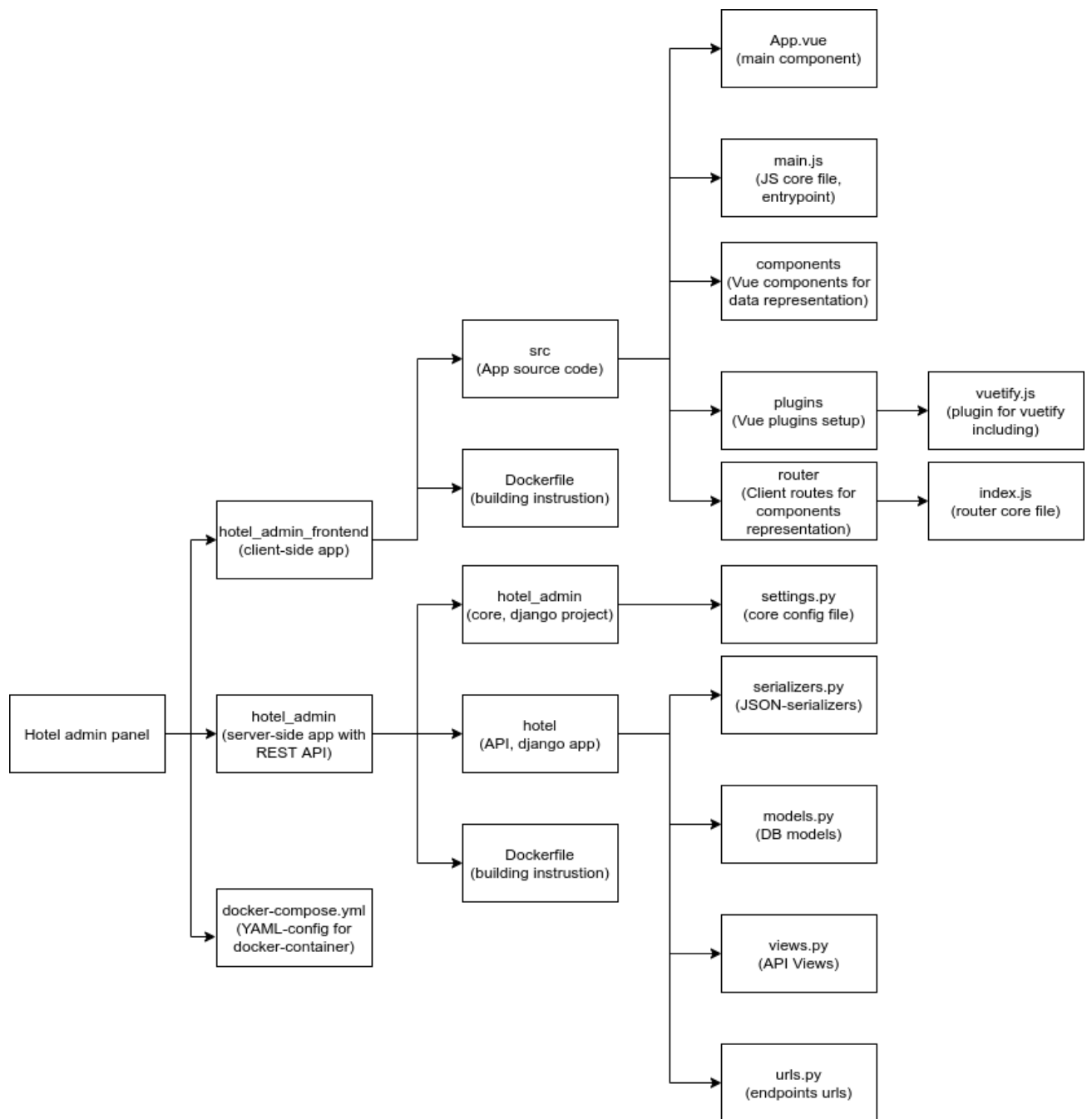


Рисунок 1 - Архитектура веб-сервиса

1.2. Архитектура базы данных

Для разрабатываемого веб-сервиса были созданы следующие сущности: клиент, номер, этаж, уборка, сотрудник, претендент, а также связывающие сущности: заселение, заявка. Сущность «заселение» хранит информацию о клиенте и занимаемом им гостиничном номере, а сущность заявка – о претендентах, находящихся среди сотрудников отеля, которые бы желали занять позицию администратора.

На Рисунке 2 продемонстрирован реализация настоящей базы данных, используемой в работе с веб-сервисом.

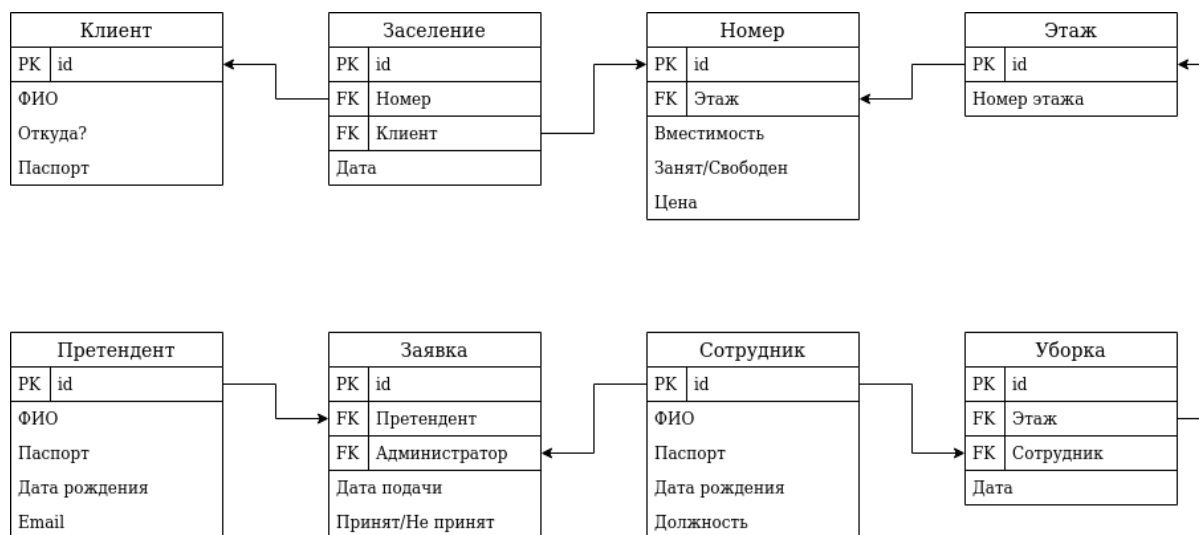


Рисунок 2 - Диаграмма базы данных веб-сервиса

2. РЕАЛИЗАЦИЯ WEB-СЕРВИСА

2.1. Общая информация

В веб-сервисе задействованы следующие интерфейсы: главная страница, страница авторизации, кабинет администратора, кабинет служащего, бронирование номера, подача заявления на работу. На главной странице предоставлена вся информация об отеле, его номерах, ценах и условиях. Кабинет администратора призван помочь администратору отеля быть в курсе обновлений, происходящих в отеле, а также просматривать график работы служащих. Кабинет служащего предназначен для рабочего персонала и позволяет работнику просмотреть необходимую информацию о его рабочей смене.

Реализованы две формы, которые необходимы для взаимодействия с клиентами и потенциальными сотрудниками. Первая форма – «бронирование номера» позволяет потенциальному путешественнику оставить свои данные и зарезервировать какой-либо номер в отеле. Форма подачи заявления на работу необходима для потенциальных сотрудников, которые хотели бы устроиться на работу в данный отель.

2.2. Реализация интерфейса “Главная страница”

Интерфейс «Главная страница» отображает информацию о названии отеля, свободных номерах, имеющихсся на данный момент, а также ценах на номера. Здесь же представлены фото-превью каждого номера, чтобы потенциальный путешественник смог сориентироваться, что из себя представляет тот или иной номер.

Hotel

Добро пожаловать в отель "Hotel"

У нас Вы проведёте своё время абсолютно незабываемо!

Свободные номера

Любой номер под Ваши нужды

Уютный номер для одного

Доступно: 2

Комфорт для двоих

Доступно: 2

Мама, папа, я! (Номер на 3)

Доступно: 2

Прайс

Отель "Hotel" - стабильно низкие цены

Тип номера	Стоимость
1 персона	600
2 персоны	800
3 персоны	1200

Арендовать номер

Просто заполните форму и мы с Вами свяжемся

Фамилия

Имя

Отчество

Паспорт

Откуда к нам приезжаете?

Когда приедете? (ГГГГ-ММ-ДД)

Номер

ОТПРАВИТЬ

ОЧИСТИТЬ

Как устроиться?

Всё просто! Заполните анкету:

Фамилия

Имя

Отчество

Паспорт

Email

Дата рождения (ГГГГ-ММ-ДД)

ОТПРАВИТЬ

ОЧИСТИТЬ

Рисунок 3 – Интерфейс главной страницы

2.3. Реализация интерфейса «Страница авторизации»

Интерфейс «Страница авторизации» содержит два поля, предназначенных для ввода логина и пароля авторизованного пользователя. Внизу находится кнопка «вход», при нажатии которой авторизованный пользователь заходит в систему. Интерфейс «страница авторизации» представлен на рисунке 3.

Страница авторизации может использоваться администратором отеля. При вводе логина и пароля он сможет получить доступ к кабинету администратора, где появится вся актуальная информация о занятых на данный момент номерах, доходе, а также о расписании рабочего персонала.

Также страница авторизации может быть использована служащим, то есть каким-либо членом рабочего персонала. При входе он сможет зайти в кабинет служащего и получить информацию о своем рабочем расписании, а также детали о его задачах на какую-либо рабочую смену.

Наконец, страница авторизации позволит потенциальному сотруднику, оставившему заявку о работе, зайти в кабинет претендента и посмотреть актуальную информацию о состоянии его заявки на данный момент. На рисунке 4 представлена реализация интерфейса «страница авторизации».



The image shows a web interface for a hotel system. At the top, there is an orange header bar containing a menu icon and the word "Hotel". Below the header, the login form is centered. It consists of two text input fields: the first is labeled "Логин" (Login) and the second is labeled "Пароль" (Password). Directly beneath the password field is an orange button with the white text "ВХОД" (Login).

Рисунок 4 - Реализация интерфейса «Страница авторизации»

2.4. Реализация интерфейса «Кабинет администратора»

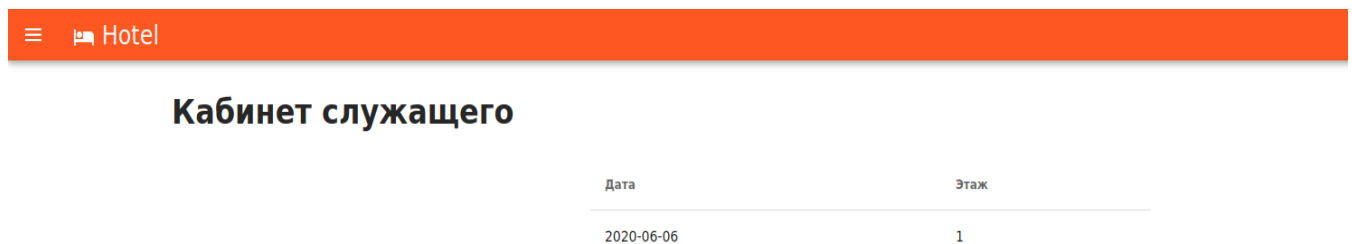
Интерфейс «Кабинет администратора» доступен администратору отеля, который будет получать всю необходимую информацию о постояльцах отеля, номерах и сотрудниках отеля. В данном интерфейсе представлена информация о количестве свободных номеров, разделенных по их вместительности (на 1 персону, на 2 персоны, на 3 персоны), информация о доходе, полученном с занятых номеров, а также информация о расписании смен служащих отеля, осуществляющих уборку номеров. В расписании смен служащих администратор отеля может увидеть фамилию, имя, отчество служащего персонала, а также этаж, на котором он будет работать и дату, в которую он будет работать. На рисунке 5 представлена реализация интерфейса «кабинет администратора».



Рисунок 5 – Реализация интерфейса «Кабинет администратора»

2.5. Реализация интерфейса “Кабинет служащего”

Интерфейс «Кабинет служащего» предназначен для рабочего персонала отеля, занимающегося уборкой номеров. Используя данный интерфейс, служащий увидит информацию об этаже, который закреплен к нему в тот или иной день для проведения там работ, а также дату, когда служащий должен произвести уборку этого этажа. На рисунке 6 представлена реализация интерфейса «Кабинет служащего».



Дата	Этаж
2020-06-06	1

Рисунок 6 - Реализация интерфейса “Кабинет служащего”

2.6. Реализация интерфейса “Кабинет претендента”

Кабинет претендента предназначен для потенциального работника отеля, который подал заявку на рассмотрение его резюме. Зайдя в личный кабинет, он увидит оставленную им информацию о заявке, а именно: его фамилию, имя, отчество, дату рождения, номер паспорта, должность, на которую он подал заявку. Также он увидит информацию о статусе его заявки (принят ли он или нет) и дату изменения статуса заявки. На рисунке 7 представлена реализация интерфейса «кабинет претендента».

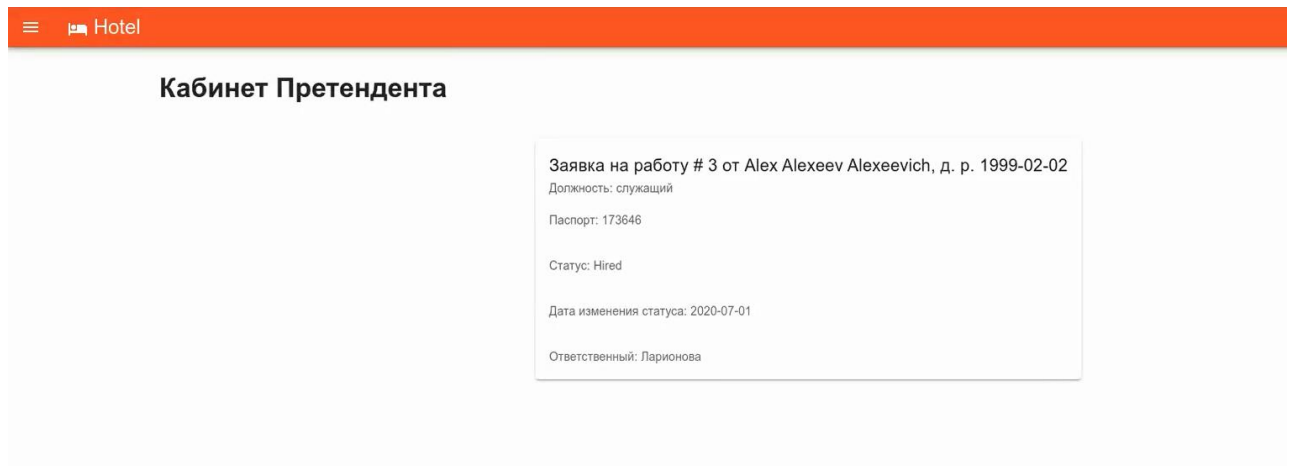


Рисунок 7 - Реализация интерфейса “Кабинет претендента”

2.7. Реализация интерфейса “Бронирование номера”

Один из самых главных интерфейсов веб-сервиса, позволяющий арендовать номер, называется «бронирование номера». Для того чтобы забронировать номер, путешественнику нужно заполнить форму, состоящую из следующих полей: фамилия, имя, отчество, паспорт, город, из которого путешественник приезжает, дату приезда. В последнем поле «номер» путешественник сможет выбрать понравившийся ему номер из предложенных на сайте. Ему будет отображен список номеров, которые свободны на данный момент, и он сможет выбрать один из них, после чего уже отправить свое бронирование. На рисунке 8 представлена реализация интерфейса «бронирование номера».

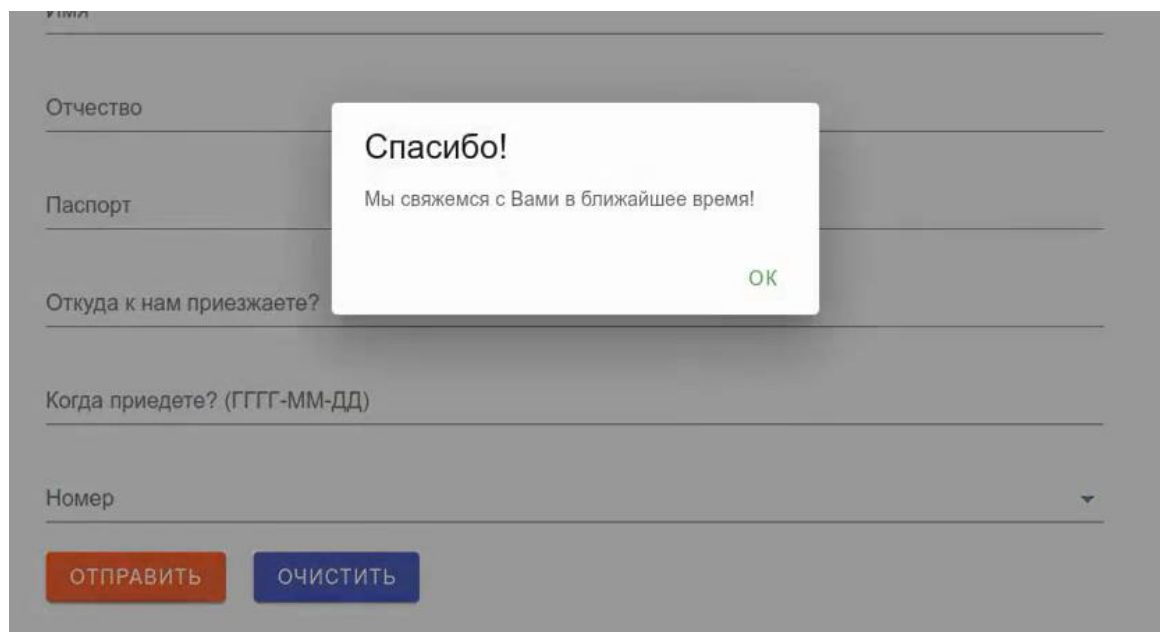
Арендовать номер

Просто заполните форму и мы с Вами свяжемся

The form consists of several input fields stacked vertically. The first field is labeled 'Фамилия' (Surname) in blue text. The second field is labeled 'Имя' (Name) in gray text. The third field is labeled 'Отчество' (Patronymic) in gray text. The fourth field is labeled 'Паспорт' (Passport) in gray text. The fifth field is labeled 'Откуда к нам приезжаете?' (Where are you coming from?) in gray text. The sixth field is labeled 'Когда приедете? (ГГГГ-ММ-ДД)' (When will you arrive? (YYYY-MM-DD)) in gray text. The seventh field is labeled 'Номер' (Room) in gray text and includes a dropdown arrow on the right. Below the fields are two buttons: an orange button labeled 'ОТПРАВИТЬ' (SEND) and a blue button labeled 'ОЧИСТИТЬ' (RESET).

Рисунок 8 - Реализация интерфейса “Бронирование номера”

После отправки формы бронирования номера пользователь увидит модальное окно, на котором будет написано, что его заявка на бронирование номера получена и что с ним свяжутся. Пример модального окна представлен на рисунке 9.

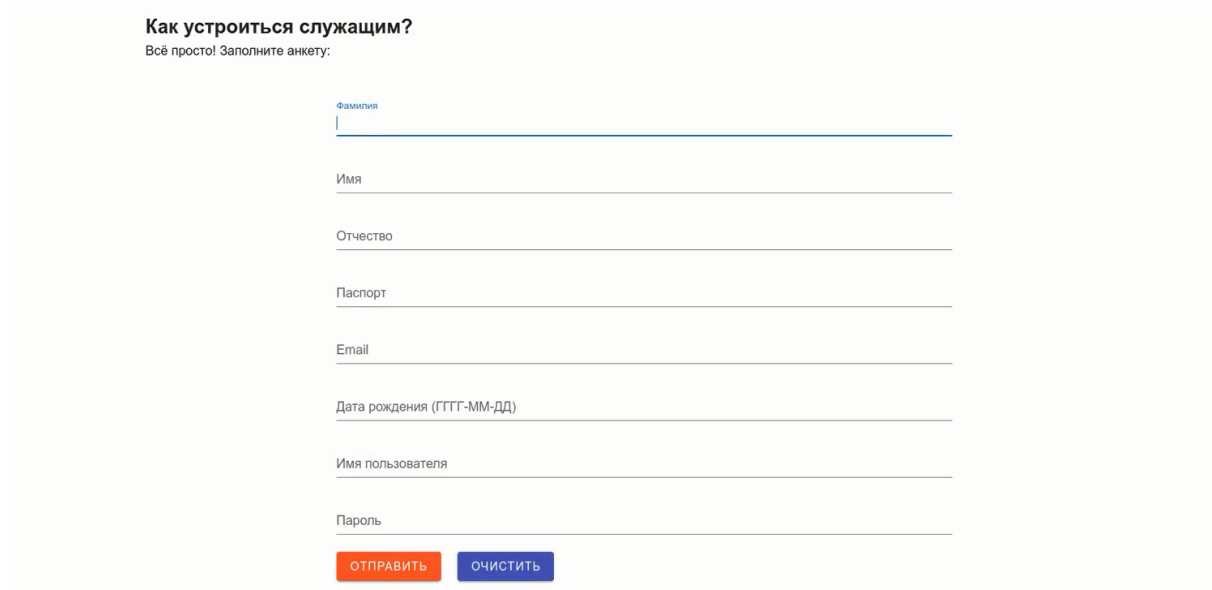


The image shows a web form for booking a room number. The form has several input fields: "Отчество" (Surname), "Паспорт" (Passport), "Откуда к нам приезжаете?" (Where are you coming from?), "Когда приедете? (ГГГГ-ММ-ДД)" (When will you arrive? (YYYY-MM-DD)), and "Номер" (Number) with a dropdown arrow. At the bottom of the form are two buttons: "ОТПРАВИТЬ" (SEND) in orange and "ОЧИСТИТЬ" (RESET) in blue. A white modal window is centered over the form, containing the text "Спасибо!" (Thank you!) and "Мы свяжемся с Вами в ближайшее время!" (We will contact you as soon as possible!). There is a green "ОК" button in the bottom right corner of the modal window.

Рисунок 9 – Модальное окно, всплывающее после отправки формы бронирования

2.8. Реализация интерфейса «Подача заявки на работу»

Пользователь, заинтересованный в работе в данном отеле, может заполнить форму подачи заявки на работу. Для этого ему нужно будет заполнить поля: фамилия, имя, отчество, номер паспорта, email, дата рождения, имя пользователя и пароль. Реализация интерфейса представлена на рисунке 10.



The screenshot shows a web form titled "Как устроиться служащим?" (How to get hired?). Below the title is a subtitle: "Всё просто! Заполните анкету:" (It's simple! Fill out the questionnaire:). The form consists of several input fields, each with a label above it: "Фамилия" (Surname), "Имя" (Name), "Отчество" (Patronymic), "Паспорт" (Passport), "Email", "Дата рождения (ГГГГ-ММ-ДД)" (Date of birth (YYYY-MM-DD)), "Имя пользователя" (Username), and "Пароль" (Password). At the bottom of the form are two buttons: "ОТПРАВИТЬ" (SEND) in orange and "ОЧИСТИТЬ" (RESET) in blue.

Рисунок 10 – Реализация интерфейса «Подача заявления на работу»

После заполнения формы пользователь увидит модальное окно, на котором будет написано, что с ним свяжутся в ближайшее время.

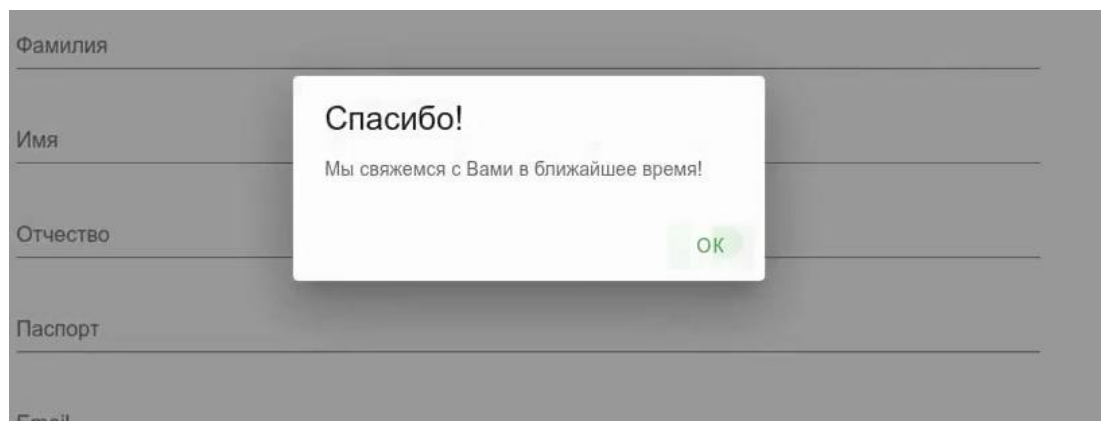


Рисунок 11 – Модальное окно, всплывающее после отправки формы заявления на работу

Все поданные заявления о принятии на работу будут доступны администратору для просмотра в Django Administration. Будет отображена вся оставленная потенциальным сотрудником информация, а также будет отображен статус заявки: принят ли человек на работу (hired) или не принят (not hired).

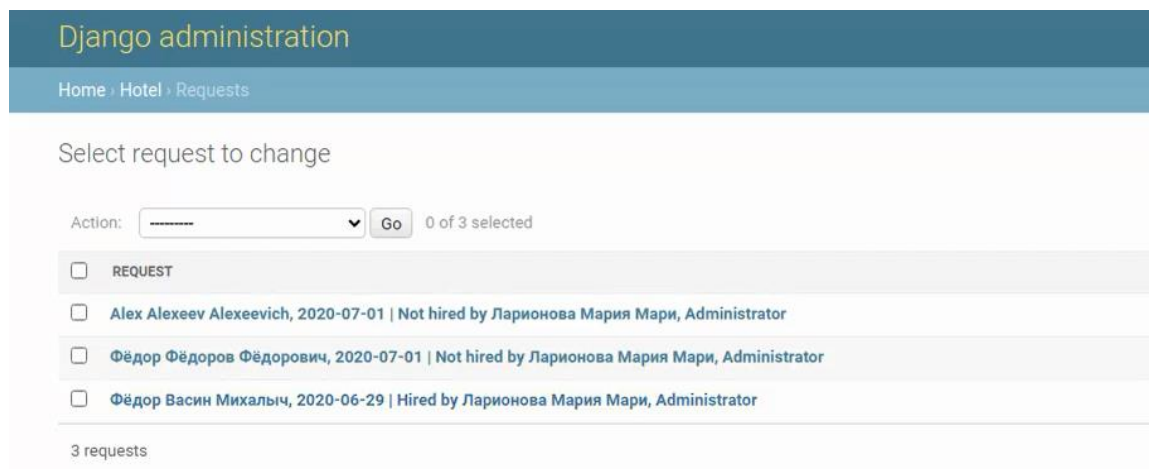


Рисунок 12 – Просмотр оставленных заявок в Django Administration

Администратор сможет просмотреть каждую заявку в отдельности, а также внести необходимые коррективы, например, назначить заявку на другого администратора или поменять статус заявку (принять человека на работу или, наоборот, отклонить его заявление).

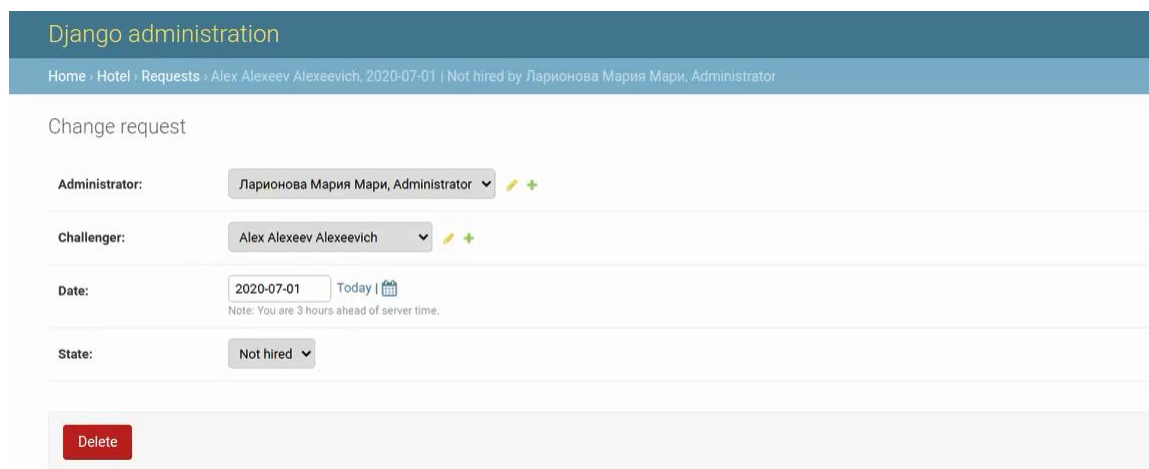


Рисунок 13 – Просмотр отдельно выбранной заявки в Django Administration

2.9. Реализация запросов

В качестве запросов, реализованных в данном проекте, можно привести в пример запрос на выдачу свободных номеров в отеле на данный момент. Когда путешественник заполняет форму, предназначенную для оставления заявки на бронирование номера, в последнем окне ему отображается список доступных на данный момент номеров. В этот список входят только те номера, которые свободны сейчас. Номера, которые заняты другими постояльцами, отображаться не будут, ведь заселиться в них не получится. Для реализации этого запроса использовался класс `APIView`. На рисунке 14 изображена реализация запроса «доступные номера».

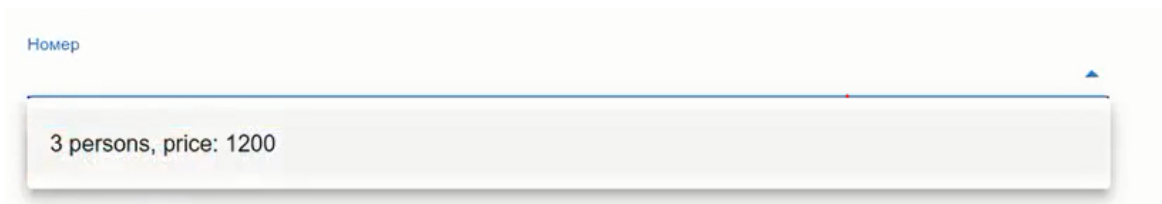


Рисунок 14 - Реализация запроса «доступные номера»

Еще одним примером запроса может быть доход отеля, который отображается в кабинете администратора.

Кабинет администратора

Статистика

ВСЕ ЗАЯВКИ НА РАБОТУ

Свободных номеров на 1 персону: 0

Свободных номеров на 2 персоны: 0

Свободных номеров на 3 персоны: 0

Актуальный доход с занятых номеров: 2600

Рисунок 15 - Реализация запроса «доход с занятых номеров»

3. ЗАКЛЮЧЕНИЕ

Целью курсовой работы было реализовать веб-сервис, который бы позволял сотрудникам отеля, таким как администратор и рабочий персонал, получать необходимые сведения о номерах отеля, постояльцах, графике уборки, а также вносить необходимые изменения в эти сведения.

Чтобы реализовать данный веб-сервис, были использованы следующие инструменты: Django Rest Framework, Vue.js, PostgreSQL, каждый из которых внес свой вклад в создание проекта. В ходе работы над проектом удалось выполнить все описанные выше задачи, которые являлись целью реализации данного веб-сервиса. Цель курсовой работы достигнута, а также получены новые знания, навыки и умения, необходимые для работы с аналогичными проектами.

Чтобы развить проект, его можно усовершенствовать, добавив некоторые качественные изменения. Можно расширить функционал, например, добавив кабинеты для таких работников как работники пищевого цеха, работники анимационной зоны и т.д. Можно детализировать дизайн сайта, добавив больше визуальной информации о номерах и их описании.

4. СПИСОК ЛИТЕРАТУРЫ

Django Rest Framework. Документация Django Rest Framework [Электронный ресурс]. URL: <https://www.django-rest-framework.org> (дата обращения: 30.06.2020).

WebDevBlog. Создание Django API используя Django Rest Framework [Электронный ресурс]. URL: <https://webdevblog.ru/sozдание-django-api-ispolzuya-django-rest-framework-apiview/> (дата обращения: 30.06.2020).

Evantotuts+. JWT Аутентификация в Django [Электронный ресурс] URL: <https://code.tutsplus.com/ru/tutorials/how-to-authenticate-with-jwt-in-django--cms-30460> (дата обращения: 30.06.2020).

Vue.js. Документация Vue.js [Электронный ресурс]. URL: <https://vuejs.org> (дата обращения: 30.06.2020).

Vuetify. Документация Vuetify [Электронный ресурс]. URL: <https://vuetifyjs.com/ru/> (дата обращения: 30.06.2020).