**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ**

**ФАКУЛЬТЕТ ИКТ**

Отчет

по лабораторной работе №2-3

по курсу «Основы Web-программирования»

Тема:

РЕАЛИЗАЦИЯ WEB-СЕРВИСОВ СРЕДСТВАМИ Django REST framework, Vue.js, Muse-UI

Вариант 5

Выполнила:

Богданова Е.Ю.

Студентка группы К3343

Проверил:

Говоров А. И.

Санкт-Петербург

2020

**Цель работы:** овладеть практическими навыками и умениями реализации web-сервисов средствами Django REST framework, Vue.js, Muse-UI.**Оборудование**: компьютерный класс.

**Программное обеспечение**: Python 3.6, Django REST framework, Vue.js, Muse-UI (или аналогичная библиотека), PostgreSQL \*.

**Практическое задание:**

Реализовать сайт используя вышеуказанные технологии, в соответствии с практическим заданием.

Задание 5.

Создать программную систему, предназначенную для работников приемной комиссии колледжа. Она должна обеспечивать хранение, просмотр и изменение сведений об абитуриентах. Результатом работы приемной комиссии должен быть список абитуриентов, зачисленных в колледж.

Секретарь приемной комиссии регистрирует абитуриентов. Для каждого абитуриента в базу данных заносятся следующие сведения: фамилия, имя, отчество, паспортные данные, какое учебное заведение, где и когда окончил, наличие золотой или серебряной медали, название специальности, на которые поступает абитуриент. При подаче заявления абитуриент указывает форму обучения (очная, очно-заочная (вечерняя), заочная), поступление на бюджет или контракт. Абитуриент может поступать вне конкурса (инвалиды, сироты). Также существуют абитуриенты-целевики, которые поступают по договорам с направляющими организациями, и обучаются на коммерческой основе.

Абитуриенты, поступающие на базе 9 классов, участвуют в конкурсе аттестатов. Для них указывается информация по 4-м профильным дисциплинам и средний балл по всем остальным дисциплинам аттестата. На основе этих данных строится рейтинг абитуриентов.

Абитуриенты, поступающие на базе 11 классов, предоставляют сертификаты ЕГЭ по 2 дисциплинам, на основе чего строится рейтинг абитуриентов.

Конкурс для абитуриентов на базе 9 и 11 классов раздельный, т.к. они поступают на разные курсы.

Абитуриент может не только подать, но и забрать документы, а также перевести их на другую специальность.

Известно количество мест на каждый факультет. Приемная комиссия по результатам экзаменов должна сформировать списки абитуриентов, зачисленных в колледж.

Секретарю приемной комиссии могут потребоваться следующие сведения:

• Список абитуриентов, подавших заявление на заданную специальность.

• Количество абитуриентов, подавших заявления на каждую специальность по каждой форме обучения на бюджет (или контракт).

• Количество абитуриентов на базе 9 и 11 классов, поступающих на бюджет (или контракт).

• Общее количество поданных заявлений ежедневно.

• Конкурс на каждую специальность по каждой форме обучения на бюджет.

Необходимо предусмотреть возможность получения документа, представляющего собой сгруппированный по заданной специальности список абитуриентов по заданной форме обучения, зачисленных в колледж, с указанием набранных ими баллов по аттестату. Отчет должен содержать проходной балл по специальности в целом, а также количество абитуриентов, поступающих на специальность.

**Ход работы:**

1. Разработка модели базы данных

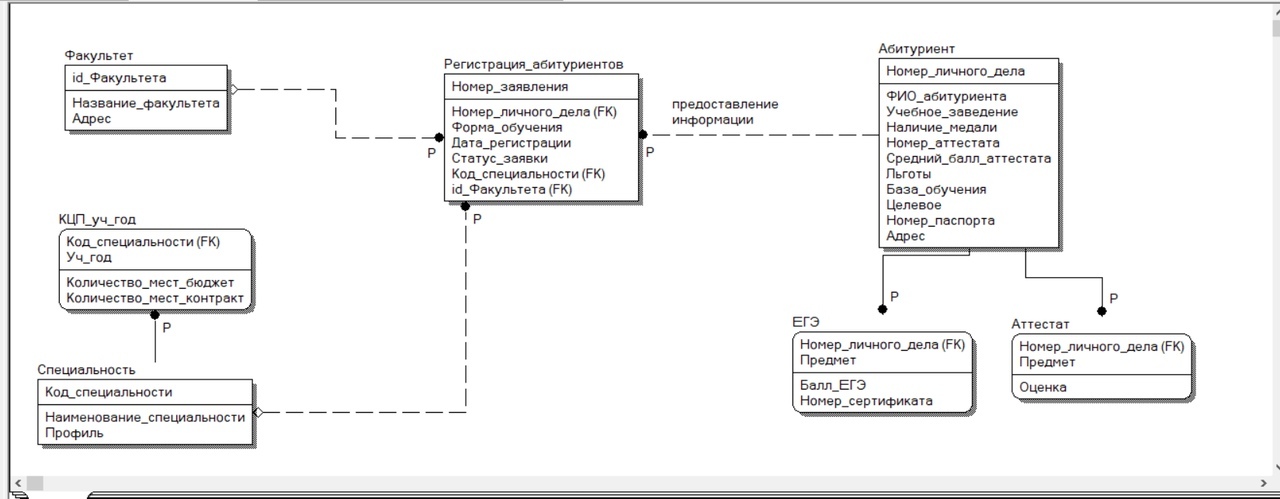
В соответствии с вариантом была разработана модель базы данных, представленная на рисунке 1.

Рисунок 1 – модель базы данных

1. Создание моделей

Согласно разработанной базе данных были созданы следующие модели:

* Faculty – Факультет. Модель содержит информацию о названии факультета и адрес его деканата.
* Specialty – Специальность. Модель содержит информацию о названии специальности, ее профиле и количестве контрактных и бюджетных мест.
* Enrollee –
* Application –
* EGE –
* EgeSubject –
* Attestat –
* AttestatSubject –

1. Создание отображений

Для создания отображений использовался класс ViewSet, который обладает встроенными атрибутами для последующего создания функций CRUD для модели. Согласно варианту, были созданы следующие отображения:

* class

1. Полученные интерфейсы в Django REST
   1. Авиакомпании

Выводятся данные обо всех авиакомпаниях, а также о самолетах и экипажах, которые к ним относятся.

Рисунок 2 – Авиакомпании в Django REST

* 1. Экипаж

Вывод заданного экипажа с подробной информацией о каждом его члене.

Рисунок 3 – заданный Экипаж в Django REST

* 1. Командир экипажа

Добавление командира экипажа. Аналогичные интерфейсы имеют второй пилот, штурман и бортпроводник.

Рисунок 4 – создание Командира экипажа в Django REST

* 1. Самолеты

Вывод информации обо всех самолетах.

Рисунок 5 – Самолеты в Django REST

* 1. Маршруты

Вывод информации обо всех маршрутах.

Рисунок 6 – Маршруты в Django REST

* 1. Рейсы

Вывод информации обо всех рейсах.

Рисунок 7 – Рейсы в Django REST

1. Полученные интерфейсы с помощью Vue.js
   1. Стартовая страница

Стартовая страница web-сервиса с верхним меню навигации и списком компаний-авиаперевозчиков.

Рисунок 8 – Стартовая страница web-сервиса

* 1. Вход

Страница авторизации пользователя имеет форму входа, а также ссылку на страницу регистрации.

Рисунок 9 – Страница входа

* 1. Регистрация новых пользователей

Страница регистрации нового пользователя представляет собой форму, которую необходимо заполнить для создания нового пользователя. В случае правильного заполнения всех полей и отсутствия пользователя с таким же username, пользователь будет зарегистрирован и перенаправлен на главную страницу. Скриншот представлен на рисунке 4.

Рисунок 10 – Страница регистрации нового пользователя

* 1. Просмотр определенной компании-авиаперевозчика

На данной странице показывается информация о выбранной авиакомпании. При нажатии на кнопку «Самолеты» появляется информация о самолетах, которые относятся к данной авиакомпании, при нажатии на кнопку «Экипажи» - об экипажах.

Рисунок 11 – Страница авиакомпании Аэрофлот

* 1. Просмотр всех маршрутов

Страница с таблицей маршрутов.

Рисунок 12 – Страница с таблицей маршрутов

* 1. Просмотр всех рейсов

Страница с таблицей с информацией об авиарейсах.

Рисунок 13 – Страница со всеми авиарейсами

* 1. Просмотр всех сотрудников аэропорта

Страница со списком сотрудников, при нажатии на каждую должность появляется таблица с ФИО сотрудника и кнопка, при нажатии на которую осуществляется переход на страницу выбранного сотрудника. Имеются также кнопки добавления сотрудника каждой должности, при нажатии на которые появляется форма добавления

Рисунок 14 – Страница со всеми сотрудниками аэропорта

Рисунок 15 – Добавление нового Штурмана

Рисунок 16 – Просмотр всех Бортпроводников

* 1. Просмотр всей информации о выбранном бортпроводнике

Страница с полной информацией о выбранном бортпроводнике. Имеется кнопка «Удалить» и «Изменить» с соответствующими функциями. Аналогичный интерфейс имеют страницы с подробной информацией о командире экипажа, втором пилоте и штурмане.

Рисунок 17 – Страница с подробной информацией о выбранном Бортпроводнике

* 1. Изменение информации о выбранном бортпроводнике

При нажатии на кнопку «Изменить» на рис.17 появляется форма с возможностью изменения данных (Фамилии, Имени, Отчества) у выбранного бортпроводника. Аналогичные интерфейсы созданы для командира экипажа, второго пилота и штурмана.

Рисунок 18 – Изменение информации о выбранном Бортпроводнике

**Вывод**:

В ходе выполнения лабораторной работы были получены навыки создания web-приложения с помощью web-фреймворка Django REST языка программирования Python, web-фреймворка Vue языка программирования JavaScript и JS Muse-UI.