Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский политехнический университет»

Кафедра «Инфокогнитивные технологии»

Образовательная программа «Веб-технологии»

Отчет по курсовому проекту

по дисциплине «Инженерное проектирование»

Тема: «Портфолио студентов»

**Выполнил:**

Студент группы 191-322

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Цалапов А.М.

подпись, дата

**Принял:**

Старший преподаватель

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Даньшина М.В.

подпись, дата

Москва 2020

Оглавление

[Введение 3](#_Toc61538640)

[Основная часть 4](#_Toc61538641)

[Анализ конкурентов 4](#_Toc61538642)

[i-exam 4](#_Toc61538643)

[4portfolio 4](#_Toc61538644)

[Проектирование 5](#_Toc61538645)

[Выбор функционала 5](#_Toc61538646)

[Кейсы использования приложения 5](#_Toc61538647)

[Этапы разработки приложения 6](#_Toc61538648)

[Инфологическая структура базы данных 7](#_Toc61538649)

[Физическая схема данных 9](#_Toc61538650)

[Тестирования “Postman” 10](#_Toc61538651)

[Заключение 11](#_Toc61538652)

# Введение

Для инженерного проекта по программе «Веб-технологии» был выдан проект по теме «Продажа билетов на мероприятия». Это веб-приложение с базой данных из 10 таблиц, разработан с использованием технологий HTML, CSS, Django, MYSQL, фреймворка Bootstrap, Rest API, Postman. Сайты аналогичных организаций:

* <https://4portfolio.ru/student.html>
* <https://i-exam.ru/node/291>
* <https://portfolio.hse.ru/?filter=1,1,N,10,N,2020,N,1,N,N,N>
* <https://www.canva.com/ru_ru/?irgwc=1&utm_medium=affiliate&utm_source=Indoleads.&clickId=wbPUbV26GxyLWLt0M-0V7wD0UkEUnmXplSq0Tg0>

# Основная часть

## Анализ конкурентов

### i-exam

Ссылка - <https://i-exam.ru/node/291>

Удобный, интуитивно понятный интерфейс, подробная информация о студентах, есть фильтры по статусу, дисциплинам, датам и т.п., что облегчает работу с интерфейсом. Подробная статистика с разными типами диаграмм, с её помощью можно узнать рейтинг, группу и т.п.

### 4portfolio

Ссылка - <https://4portfolio.ru/student.html>

Плюсы:

* Кастомизация
* Современный интерфейс
* Соответствие требованиям ФГОС

Минусы

* Узкий функционал

## Проектирование

### Выбор функционала

Веб-приложение для онлайн просмотра портфолио тысяч студентов. В приложении будет 2 типа пользователей – обычный пользователь и администратор. У администратора есть права ко всем таблицам, доступ к админ панели, он имеет полный контроль над сайтом. Также администратор имеет возможность импортировать/экспортировать данные в таблицы, что позволяет ему делать резервные копии или же переносить таблицы в другие сервисы

### Кейсы использования приложения

* Награждение лучших студентов по данным рейтинга.
* Работодатель может выбрать лучших студентов на стажировку
* Авторизация
* Посещение страницы
* Регистрация новых пользователей в админке
* Размещение студентов
* Размещение новых проектов
* Редактирование информации о проектах, студентах
* Импорт данных из таблиц
* Экспорт отредактированных данных

## Этапы разработки приложения

Веб-приложение состоит из страницы с предложениями, на ней же размещен вход в учетную запись.

1. Анализ аналогичных веб-сервисов. Рассмотрение функционала, дизайна и пользовательских сценариев с перечисленных аналогов. Обнаружение отличий между этими сервисами.
2. Проектирование базы данных, разработка десяти таблиц, связывание таблиц между собой. Документирование базы данных: создание инфологической модели и физической. Создание типовых обращений к базе данных.
3. Заполнение базы данных информацией.
4. Изучение документации и обучающей работе с Django, RestAPI, Postman.
5. Создание макета страницы для веб-приложения. В качестве готовых стилей – фреймворк bootstrap.
6. Создание серверной части веб-приложения.
7. Исправление ошибок, неполадок, выявленных в следствии тестирования получившегося веб-приложения. Размещение приложения на сервере fit.mospolitech.
8. Изучение информации о возможностях, этапах настройки административного интерфейса Django. Настройка административного интерфейса Django.
9. Реализация rest api, импорта-экспорта, admin-actions.
10. Установка и тестирование Postman.
11. Окончательное тестирование получившегося веб-приложения.
12. Разработка и документирование кейсов взаимодействия обычного пользователя или администратора с получившимся веб-приложением.
13. Написание Readme для репозитория, дозаполнение документации. Выгрузка окончательного набора документации и файлов веб-приложения в репозиторий.

## Инфологическая структура базы данных

Почти самые важные «стороны» сервиса по продажу билетов – её база данных, серверная и административная части. В области продажи билетов, да и в целом продажи чего-либо, важно иметь хорошую базу данных, приведенную к нормальной форме, чтобы можно было легче и лучше сделать фильтры, важные для такого вида сервисов.

Главная таблица в Базе Данных – «Students» и «Projects», на них и строились все остальные тиблицы. Они связаны с другими таблицами – Инженерные проекты, Места проведений мероприятий (связана с таблицей Типы мест проведения мероприятий), Типы мероприятий, Ограничения. Также таблица связана с таблицами, посвященными билетам: Кол-во билетов, Билеты, Проданные билеты. В них указаны id мероприятия, к которому относится билет.

Также есть таблица Пользователи – в нее записываются все зарегистрированные пользователи с отметкой об их правах.

Используется 10 таблиц. Вот главные из них:

(жирным – primary key)

*Students* – таблица с логинами, паролями всех пользователей веб-приложения.

|  |  |  |
| --- | --- | --- |
| **Id** | Int, auto increment | ID пользователя |
| name | text | Имя пользователя |
| Surname | text | Фамилия пользователя |
| Status | char | Статус студента, учится, отчислен, зачислен или восстановлен |
| Course | Int | Курс, на котором находится студент |

*Projects* – таблица со всеми проектами, которые выполняются студентами в обязательном порядке.

|  |  |  |
| --- | --- | --- |
| **Id** | Int, auto increment | ID проекта |
| name | text | Название проекта |
| description | text | Описание проекта |
| isDone | boolean | Завершен проект или нет |

*Lessons* – все предметы, которые преподаются студентам

|  |  |  |
| --- | --- | --- |
| **id** | Int, auto increment | Id предмета |
| Name | text | Название предмета |
| description | Text | Краткое описание предмета |

*Clients* – клиенты, которые заказывали проекты в качестве пд для студентов.

|  |  |  |
| --- | --- | --- |
| **id** | Int, auto increment | Id типа площадки |
| name | Text | Имя заказчика |
| surname | Text | Фамилия заказчика |

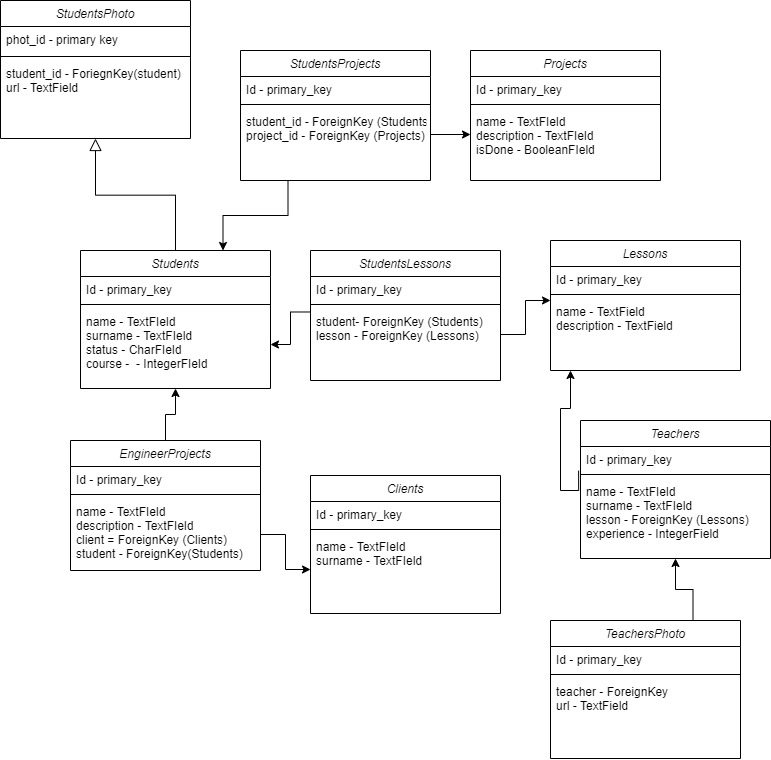
*EngineerProjects* – инженерные проекты студентов

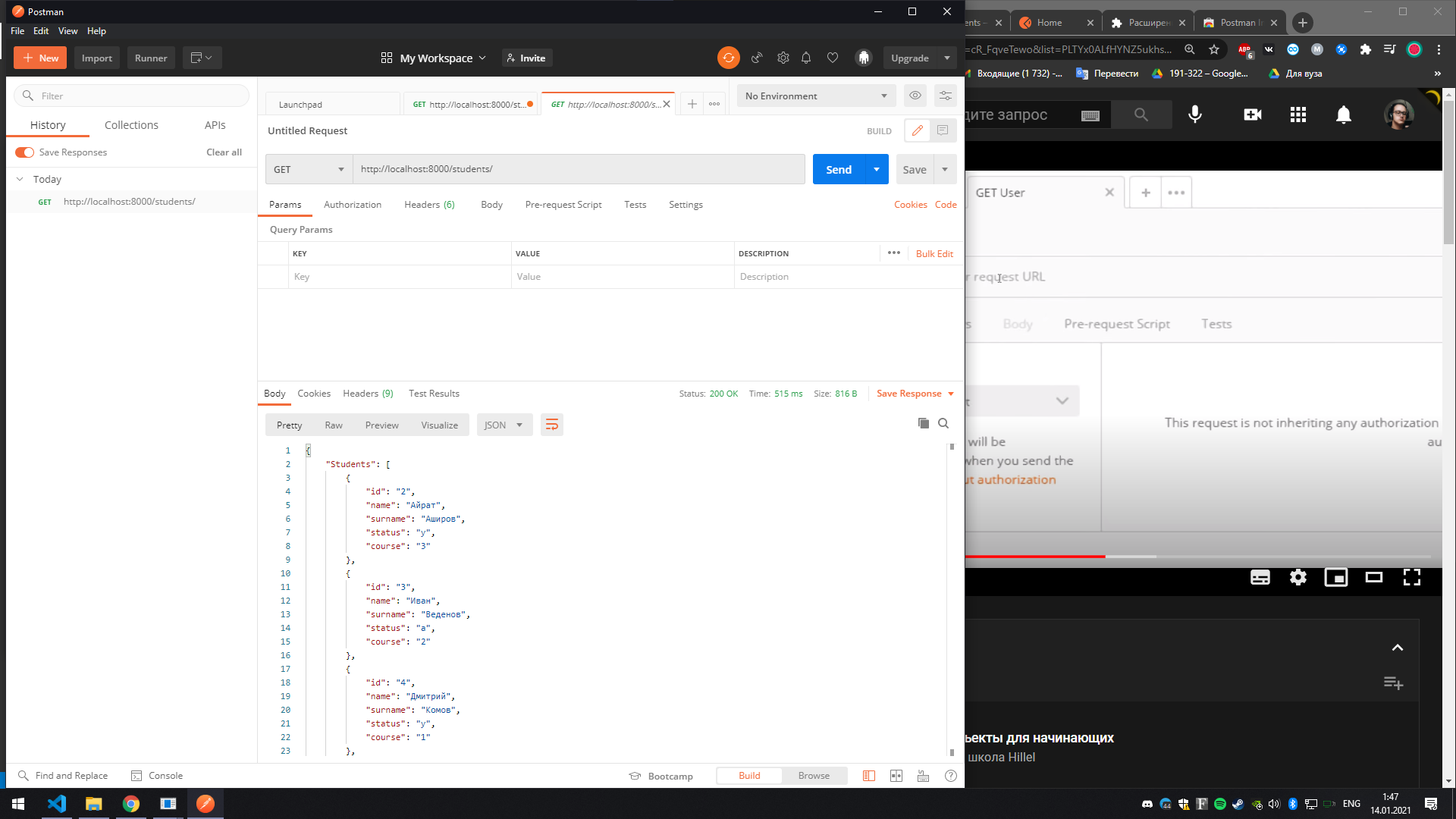
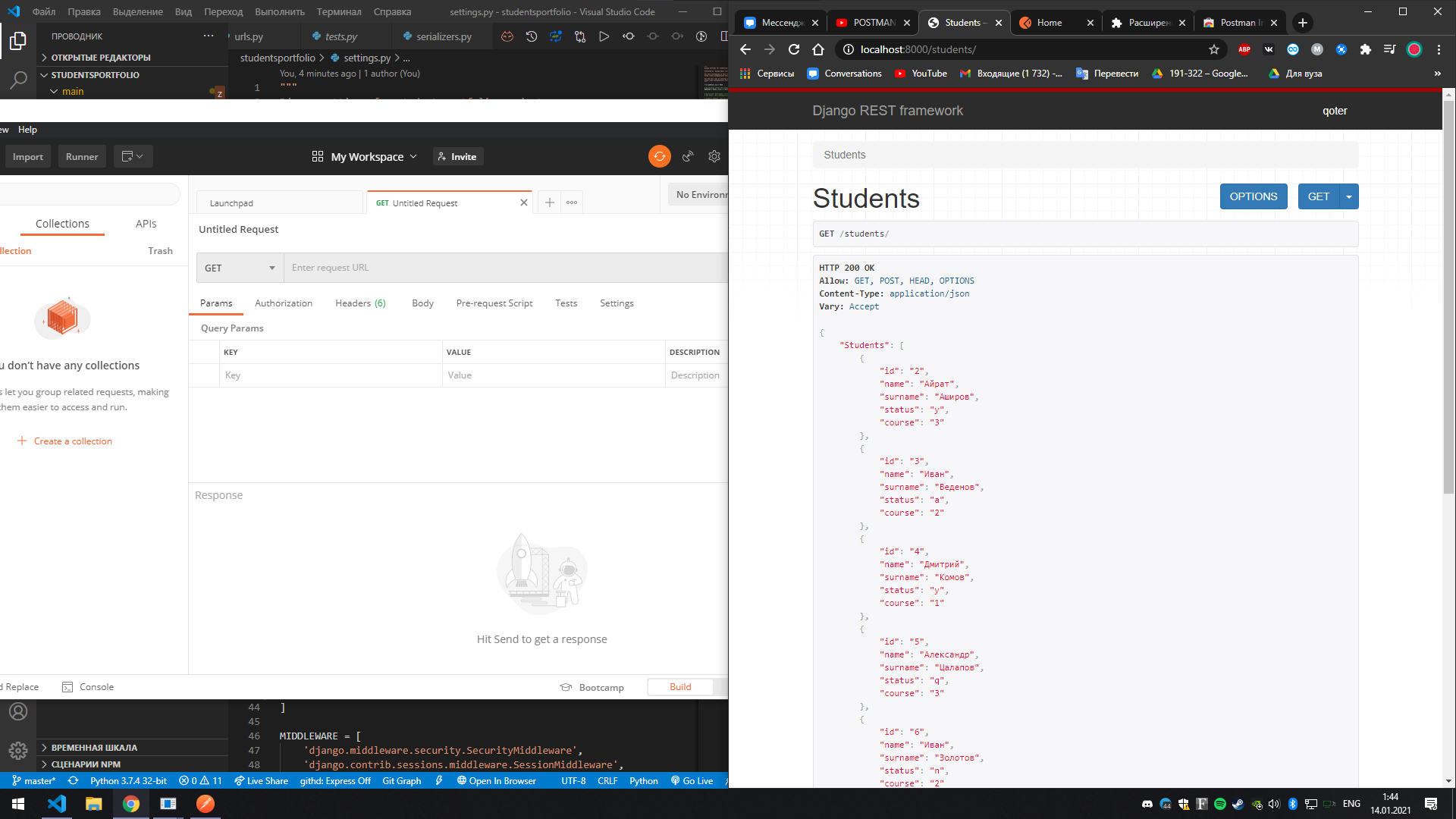
|  |  |  |
| --- | --- | --- |
| **Id** | Int, auto increment | ID проекта |
| name | text | Название проекта |
| description | text | Описание проекта |
| Client | Foreign | Клиент, заказавший данный проект |
| Student | Foreign | Студент, исполняющий заказ |

*Teachers* – Учителя, преподающие проект

|  |  |  |
| --- | --- | --- |
| **Id** | Int, auto increment | ID пользователя |
| name | text | Имя пользователя |
| Surname | text | Фамилия пользователя |
| lesson | foreign | Предмет, который преподает преподаватель |
| experience | int | Опыт преподавания |

## Физическая схема данных



Тестирования “Postman”  


## Заключение

В результате всей проделанной работы, получился готовый продукт, обладающий всеми нужными функциями для сосуществования и конкурирования со своими аналогами.

Сайт –

Landing page -

GITHUB репозиторий - <https://github.com/qoterwoter/django>