

# Постановка задачи приложения

## 1. Проблема

На данный момент у студентов отсутствует приложение или же веб-сайт, куда они могут спокойно размещать свои какие-либо выполненные задания или же аудио файлы, которыми они могут потом воспользоваться. Я решила создать не приложение, а веб-сайт электронной библиотеки, чтобы все (начиная от учеников, заканчивая студентами) могли загружать и пользоваться данными. Если спросите почему веб-сайт, то лично мне удобно пользоваться веб-сайтом, чем приложением.

## 2. Сценарий

Шодона, студентка, у которой вечно теряются выполненные домашние задания. Дело в том, что в систему универа не всегда можно добавлять свои выполненные задания. Она воспользовалась этим веб-сайтом, прошла регистрацию, загрузила свои домашки и ей стало намного удобнее хранить свои дз до сдачи всех работ.

## 3. Требования

В данном проекте должны быть учтены следующие функциональные требования (FR) и нефункциональные требования (NFR)

FR1: Пройти регистрацию: Студент может создать свой логин и пароль, через который он будет иметь доступ к веб-сайту.

FR2: Добавить свои данные: Студент может ввести свои данные, а конкретнее: Ф.И., домашний адрес, место работы или же учёбы и номер телефона.

FR3: Изменение своих персональных данных: Студент может ввести изменения в свои данные в любое удобное для него время.

FR4: Добавление материала: Студент может добавить книги, мультимедию, свои домашки, программные обеспечения, а также ссылки, и по-своему желая он может разметить их в публичной оферте, или же наоборот, скрыть от всех и пользоваться самим.

NFR1: Последние действия: Студент может смотреть последние действия всех тех, кто недавно что-то добавлял, но не имея доступа к ним.

NFR2: Добавление групп: Студент в разделе “Пользователи и группы” по желанию может указать свою группу и найти своих однокурсников.

NFR3: Быстрый поиск: Студент, чтобы не искал свои задания, он в поиске может ввести название или же код ISBN, чтобы быстро найти и скачать себе.

NFR4: Дата создания: Студент когда добавляет что-либо в веб-сайт, он может увидеть, когда он его добавил.

#### **4. Целевая среда**

Код веб-сайт должен быть создан на Python Django и продемонстрирован в браузере.

#### **5. Документация**

- Документ с анализом требований (RAD - Requirements Analysis Document)
- Проектный документ системы (SDD - System Design Document)
- Исходный код под контролем версий, включая документацию по исходному коду

#### **6. Критерии приемлемости клиента**

Веб-сайт должен демонстрировать, по крайней мере следующую функциональность: он при входе должен показывать и проходить по разделам: «Аутентификация», «Основные разделы» и «Пользователь и группы». Каждый раздел имеет подразделы. Студент на главной странице может увидеть разделы и последние действия всех остальных. Кроме этого, в веб-сайте он может ввести свои данные, вносить изменения или же удалять их. Так же он может добавлять какие-либо информации в определённых разделах, естественно добавляя их нужном формате. Веб-сайт взаимодействует с серверной системой и соответствует требованиям к удобству использования.