

ГУАП

КАФЕДРА № 41

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

Старший преподаватель

должность, уч. степень, звание

подпись, дата

Акопян Б.К.

инициалы, фамилия

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №5

ОРГАНИЗАЦИЯ ПРОЕКТНОЙ ДЕЯТЕЛЬНОСТИ

по курсу: ОСНОВЫ ПРОЕКТНОЙ ДЕЯТЕЛЬНОСТИ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ гр. №

4315

подпись, дата

М. А. Соколов

инициалы, фамилия

Санкт-Петербург 2024

Цель работы: приобрести умения и навыки по формированию технического задания, выбору методологии и фреймворка управления проектной деятельностью для группового или индивидуального учебного проекта.

1. Задание работы

Для выбранного учебного проекта необходимо сформировать техническое задание и согласовать его с заказчиком (ведущим преподавателем по дисциплине «Основы проектной деятельности») в часы проведения занятий по дисциплине.

2. Формирование подгруппы для выполнения задания.

Проект выполняется: Маркиянов Сергей Александрович, Лобанов Константин Евгеньевич, Соколов Марк Алексеевич, Дерябина Мария Алексеевна, Гуляева Арина Дмитриевна.

3. Тема проекта: Образовательное веб-приложение на языке программирования Python.

4. Результаты выполнения практического задания №1

Техническое задание – документ или комплект документов, определяющие цель, структуру, свойства и методы, применяемые при реализации проекта, и исключающие двусмысленное толкование различными исполнителями.

По результатам выполнения работы заполнено и согласовано с преподавателем техническое задание, представленное в таблице 1.

Таблица 1 – Техническое задание на проект «Образовательное веб-приложение на языке программирования Python.»

1. ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ
1.1. Список обозначений и сокращений
БД — база данных, хранилище информации, связанной с учебными материалами и данными пользователей.
API — интерфейс прикладного программирования, для интеграции модулей.
CRUD – Создавать, читать, Обновлять, Удалять.
MVC — Model-View-Controller, архитектурный шаблон проектирования.
HTML/CSS/JS — языки, используемые для разработки интерфейса веб-приложения
1.2. Список определений

Образовательное веб-приложение — программный продукт, доступный через веб-браузер и предназначенный для предоставления учебного контента и выполнения заданий в интерактивной форме.

Пользователь — лицо, использующее веб-приложение для просмотра и изучения учебного контента, выполнения заданий, прохождения тестов и отслеживания своего прогресса.

Администратор — пользователь с расширенными правами доступа, который может добавлять, редактировать и удалять учебный контент, управлять пользователями и модерировать взаимодействие в системе.

Личный кабинет — раздел веб-приложения, в котором пользователь может просматривать свои данные, отслеживать прогресс, видеть результаты тестов

Аутентификация — процесс проверки подлинности пользователя с помощью логина и пароля.

Авторизация — процесс предоставления пользователю доступа к различным функциям веб-приложения на основании его роли (например, обычного пользователя или администратора).

Итерация — повторяющийся этап разработки, на котором проект постепенно дорабатывается и улучшается в рамках Agile-методологии.

Фронтенд — часть веб-приложения, которая отвечает за отображение интерфейса и взаимодействие с пользователем в браузере.

Бэкенд — это часть веб-приложения, которая управляет логикой и взаимодействием с базой данных, а также обеспечивает работу API.

2. ВВЕДЕНИЕ

2.1 Организация-заказчик и исполнитель проекта

Организацией-заказчиком продукта является кафедра №41 Прикладной информатики ГУАП в лице ведущего преподавателя группы по дисциплине «Основы проектной деятельности» (далее Заказчик).

Исполнителем проекта является команда студентов второго курса направления 09.03.03 «Прикладная информатика» (далее Исполнитель).

2.1. Предмет разработки

Предмет разработки выражается в виде создания образовательного веб-приложения реализуемого в ходе проекта (далее Продукт).

2.2. Назначение и характеристика продукта

Для разработки продукта, предназначенного для образовательной деятельности, акцент сделан на поддержку студентов и начинающих программистов, которые хотят отслеживать собственный прогресс, делиться достижениями и взаимодействовать с другими пользователями, развивая навыки программирования. Основное предназначение продукта — создание комфортной учебной среды, способствующей обмену знаниями и поддержке участников сообщества, что способствует углубленному изучению основ программирования и мотивации в учебном процессе.

2.3. Цель и задачи работы

Цель проекта: разработать образовательное веб-приложение на языке программирования Python, которое будет содержать удобный, интуитивно понятный интерфейс и функционал для регистрации и отслеживания прогресса пользователей, до 23 декабря 2024 года, чтобы улучшить навыки программирования и получить зачёт по дисциплине «Основы программирования».

Задачи проекта:

- Разработка функционала для регистрации пользователей и создания личных профилей до 28 октября 2024 года;
- Внедрение системы публикации постов с успехами и достижениями пользователей до 20 ноября 2024 года;
- Реализация механизма взаимодействия между пользователями: возможность комментирования, лайков и подписок до 7 декабря 2024 года;
- Разработать структуру приложения и создать базу данных для хранения информации о пользователях и курсах до 18 октября 2024 года;
- Создание системы тегов и категорий для упрощения поиска контента до 30 ноября 2024 года.

2.4. Стадийность
<ul style="list-style-type: none"> – Разработка и создание базы данных — до 18 октября 2024 года. – Разработка системы регистрации и профилей — до 28 октября 2024 года. – Внедрение функционала публикации постов — до 20 ноября 2024 года. – Создание системы тегов и категорий — до 30 ноября 2024 года. – Реализация механизма взаимодействия между пользователями — до 7 декабря 2024 года.

3. ОПИСАНИЕ ПРОВОДИМЫХ РАБОТ
<p>–Разработка и создание базы данных</p> <p>Спроектировать и создать структуру базы данных для хранения информации о пользователях, профилях, постах и взаимодействиях. База данных должна поддерживать все основные функции приложения и обеспечивать безопасность данных.</p>
<p>–Разработка системы регистрации и профилей</p> <p>Реализовать функционал регистрации пользователей и создания профилей, где каждый пользователь сможет указывать данные, отслеживать прогресс и редактировать информацию, обеспечивая персонализированный опыт.</p>
<p>–Внедрение функционала публикации постов</p> <p>Создать систему публикации постов, чтобы пользователи могли делиться успехами и достижениями, формируя поддерживающую среду на платформе.</p>
<p>–Создание системы тегов и категорий</p> <p>Разработать систему тегов и категорий для удобной организации и поиска контента, что улучшит навигацию и удобство использования.</p>
<p>–Реализация механизма взаимодействия между пользователями</p> <p>Внедрить функции комментирования, лайков и подписок, чтобы пользователи могли общаться, поддерживать друг друга и создавать сообщество на платформе.</p>

4. ИСХОДНЫЕ ДАННЫЕ ДЛЯ ВЫПОЛНЕНИЯ РАБОТЫ

4.1 Требования к возможностям продукта

1. Функциональность регистрации и профилей

- Возможность регистрации нового пользователя с уникальными данными.
- Создание личного профиля с возможностью редактирования информации.
- Отображение прогресса пользователя.

2. Публикация постов

- Возможность пользователям создавать и публиковать посты с описанием достижений.
- Просмотр постов другими пользователями для поддержки и мотивации.
- Возможность редактирования и удаления постов.

3. Система тегов и категорий

- Присвоение тегов и категорий постам для упрощенного поиска контента.
- Фильтрация и сортировка постов по темам и популярности.

4. Взаимодействие между пользователями

- Возможность подписываться на других пользователей, комментировать и ставить лайки на посты.
- Уведомления о новых лайках, комментариях и подписчиках.

5. Интерфейс и удобство использования

- Удобный и интуитивный интерфейс, адаптированный под разные устройства.

Быстрая навигация по разделам и элементам приложения.

4.2 Допущения и ограничения

1. Технические ограничения

- Приложение разрабатывается на языке Python с использованием Django в качестве фреймворка.
- Используется SQLite для локального хранения данных (с возможностью перехода на другой SQL-бэкенд при необходимости).

2. Ограничения по безопасности

- Обязательное шифрование паролей пользователей и обеспечение базовых мер безопасности данных.
- Использование базового уровня аутентификации и ограничения доступа к данным через ролевую модель (например, ограничение доступа к редактированию чужих постов).

3. Ограничения по времени

- Завершение всех этапов разработки до 23 декабря 2024 года с промежуточными дедлайнами по основным функциям.
- Поддержка и тестирование функционала до окончания срока проекта, включая тестирование интерфейса и взаимодействий между пользователями.

4. Ограничения по объёму функционала

- Реализация только базового набора функций: регистрация, профили, публикация постов, взаимодействие пользователей.

Расширение функционала (например, расширенная аналитика) не предполагается в рамках текущего проекта.

5. РЕСУРСЫ ПРОЕКТА

5.1 Требования к команде проекта

Frontend-разработчик (1-2 человека)

- **Компетенции:** знание HTML, CSS, JavaScript, опыт работы с фреймворками для фронтенда
- **Задачи:** создание интерфейса веб-приложения, работа над UX/UI, интеграция с API.

Backend-разработчик (1-2 человека)

- **Компетенции:** опыт программирования на
- **Задачи:** разработка серверной части приложения, настройка базы данных, реализация бизнес-логики и API.

Дизайнер интерфейсов (1 человек)

- **Компетенции:** навыки UX/UI-дизайна, знание инструментов для создания макетов (Figma, Adobe XD), умение разрабатывать адаптивный дизайн.
- **Задачи:** создание визуального стиля, разработка макетов страниц, улучшение пользовательского опыта.

Тестировщик (QA-инженер) (1 человек)

- **Компетенции:** знание методик тестирования, навыки юнит- и интеграционного тестирования, опыт работы с системами отслеживания ошибок.
- **Задачи:** тестирование функционала, выявление и фиксация багов, проведение тестов производительности и юзабилити.

Администратор базы данных (1 человек)

- **Компетенции:** опыт работы с базами данных (PostgreSQL, MySQL), знание SQL, навыки настройки, резервного копирования и восстановления данных.
- **Задачи:** настройка и обслуживание базы данных, обеспечение её безопасности и производительности, создание резервных копий.

5.2 Требования к программно-аппаратному обеспечению

1. Средства разработки:

- **Python** (версия 3.9 или выше) — основной язык программирования для разработки приложения.

- **Фреймворки:** Flask или Django — для создания серверной части.
- **JavaScript-библиотеки/фреймворки:** React или Vue.js — для разработки интерактивного пользовательского интерфейса.
- **HTML, CSS** – для верстки и стилизации интерфейса.
- **Инструменты для работы с базой данных:** PostgreSQL или MySQL — для хранения данных.

2.Среда разработки:

- **IDE или текстовые редакторы:** Visual Studio Code, PyCharm или Sublime Text — для написания и отладки кода.
- **Git** — система контроля версий для командной работы и отслеживания изменений в проекте.
- **Платформа для управления задачами:** Mori, Kaiten или Jira — для организации работы команды и отслеживания прогресса.

3.Система управления проектом и дизайном:

- **Figma или Adobe XD** – для разработки макетов и визуального
- **Miro или аналог** – для командной работы и визуализации концепций.

РАЗДЕЛ 6. ТРЕБОВАНИЯ К ТЕХНИЧЕСКИМ РЕЗУЛЬТАТАМ

6.1 Основные требования к конечной реализации продукта

Основные требования к продукту включают создание образовательного веб-приложения с интуитивно понятным интерфейсом и функционалом для регистрации пользователей, создания профилей и отслеживания их прогресса. Продукт должен содержать следующие ключевые возможности:

1. **Система регистрации и профилей:** Возможность регистрации новых пользователей, создания и редактирования личных профилей, отображения личного прогресса.
2. **Публикация пользовательского контента:** Функционал для создания и публикации постов, позволяющий пользователям делиться достижениями, редактировать и удалять свои посты.
3. **Система тегов и категорий:** Возможность организации контента по темам с помощью тегов и категорий, чтобы упростить поиск информации и улучшить навигацию.

4. **Интерактивные возможности взаимодействия между пользователями:** Включение функций подписки, лайков и комментариев для создания взаимодействий между пользователями, поддержки и обмена опытом.
5. **Интуитивный интерфейс:** Продукт должен быть удобен в использовании и адаптирован для работы на различных устройствах.

Окончательные требования могут уточняться и согласовываться в процессе выполнения работ с Заказчиком.

6.2 Лицензирование продукта

Все права на Продукт принадлежат Исполнителю.

6.3 Используемая нормативная документация

1. ГОСТ Р ИСО/МЭК 12207 «Информационные технологии. Процессы жизненного цикла программных средств»
2. ГОСТ 34.602—2020 «Информационные технологии. Комплекс стандартов на автоматизированные системы»
3. ГОСТ 7.32—2017 «Отчет о научно-исследовательской работе. Структура и правила оформления»

7. КАЛЕНДАРНЫЙ ПЛАН

Содержание календарного плана приведено в таблице 7.1.

Табл. 7.1

Этап	Содержание работ	Сроки	Результат
1	Разработать структуру приложения и создать базу данных для хранения информации о пользователях и курсах.	01.09.24 - 18.10.24	Разработана и внедрена база данных для хранения информации о пользователях и контенте.
2	Разработка функционала для регистрации	19.10.24 – 28.10.24	Реализована система регистрации и профилей,

	пользователей и создания личных профилей		обеспечивающая доступ пользователей к персональным данным.
3	Внедрение системы публикации постов с успехами и достижениями пользователей	29.10.24 – 20.11.24	Внедрен функционал публикации постов, позволяющий пользователям делиться своими достижениями.
4	Создание системы тегов и категорий для упрощения поиска контента	21.11.24 – 30.11.24	Создана система тегов и категорий для удобного поиска и организации контента.
5	Реализация механизма взаимодействия между пользователями: возможность комментирования, лайков и подписок	01.12.24 – 07.12.24	Реализован механизм взаимодействия между пользователями, включающий комментарии, лайки и подписки.
Итого		07.12.24	Готовый продукт в виде образовательного веб-приложения для студентов

8. ТРЕБОВАНИЯ К КАЧЕСТВУ ВЫПОЛНЕНИЯ РАБОТ

Исполнителю при реализации проекта руководствоваться требованиями ГОСТ, перечисленных в разделе 6.2.

Заказчику при оценке результатов выполнения работ руководствоваться:

- стандартом организации ГУАП СМК 3.76 «Положение о текущем контроле успеваемости и промежуточной аттестации студентов и аспирантов ГУАП,

обучающихся по образовательным программам высшего образования» и «О модульно-рейтинговой системе оценки качества учебной работы студентов в ГУАП»;

- рабочей программой дисциплины «Основы проектной деятельности» для направления 09.03.03 «Прикладная информатика» очной формы обучения 2022 года приема.

9. ПОРЯДОК КОНТРОЛЯ ПРИЕМКИ

9.1 Требования к отчетной документации

1. Промежуточным отчетным документом является отчет о проведенной работе в формате pdf, оформленный в соответствии с ГОСТ 7.32-2017.
2. Итоговым отчетным документом является презентация в формате pptx или аналогичном ему, описывающая процесс проведения работ и конечный продукт, содержащая не менее 10 слайдов (далее Презентация).

9.2 Формат отчетной документации

Отчетная документация предоставляется в электронном виде путем загрузки в соответствующий слот в Личном кабинете ГУАП.

9.3 Порядок рассмотрения и приемки результатов работы

Результаты работы принимаются приемочной комиссией Заказчика в лице ведущего преподавателя группы по дисциплине «Основы проектной деятельности».

От Исполнителя требуется представить итоговые результаты работы на практическом занятии по дисциплине «Основы проектной деятельности» в осеннем семестре 2023/24 уч.г. в форме устного доклада, сопровождающегося Презентацией, длительностью не более 7 мин.

РАЗДЕЛ 10. ПЕРЕЧЕНЬ ПРИЛОЖЕНИЙ

Приложений нет.

Исполнитель:

Участники команды разработчика:

Соколов М. А.
Маркиянов С. А.
Лобанов К. Е.
Дерябина М. А.
Гуляева А. Д.

СОГЛАСОВАНО:

Ведущий преподаватель группы Б. К. Акопян
по дисциплине «Основы проектной деятельности»

Старший преподаватель кафедры №41

Б.К. Акопян

5. Результаты выполнения практического задания №2

Для проекта была выбрана следующая модель жизненного цикла:

Итеративная.

Была выбрана данная модель, потому что проект начиная от начального планирования неоднократно будет проходить этапы планирования, тестирования, дизайна, оценки перед запуском и поддержкой.

Также была выбрана методология проектирования:

Agile-методология.

Данная методология подразумевает гибкое управление проектом. В рамках этого подхода команда работает со списком задач, который можно менять. У сотрудников больше свободы действий, в процессе работы они могут убирать и добавлять новые задачи.

По результатам работы была сформирована доска в системе Kaiten, представленная на рисунке 1:

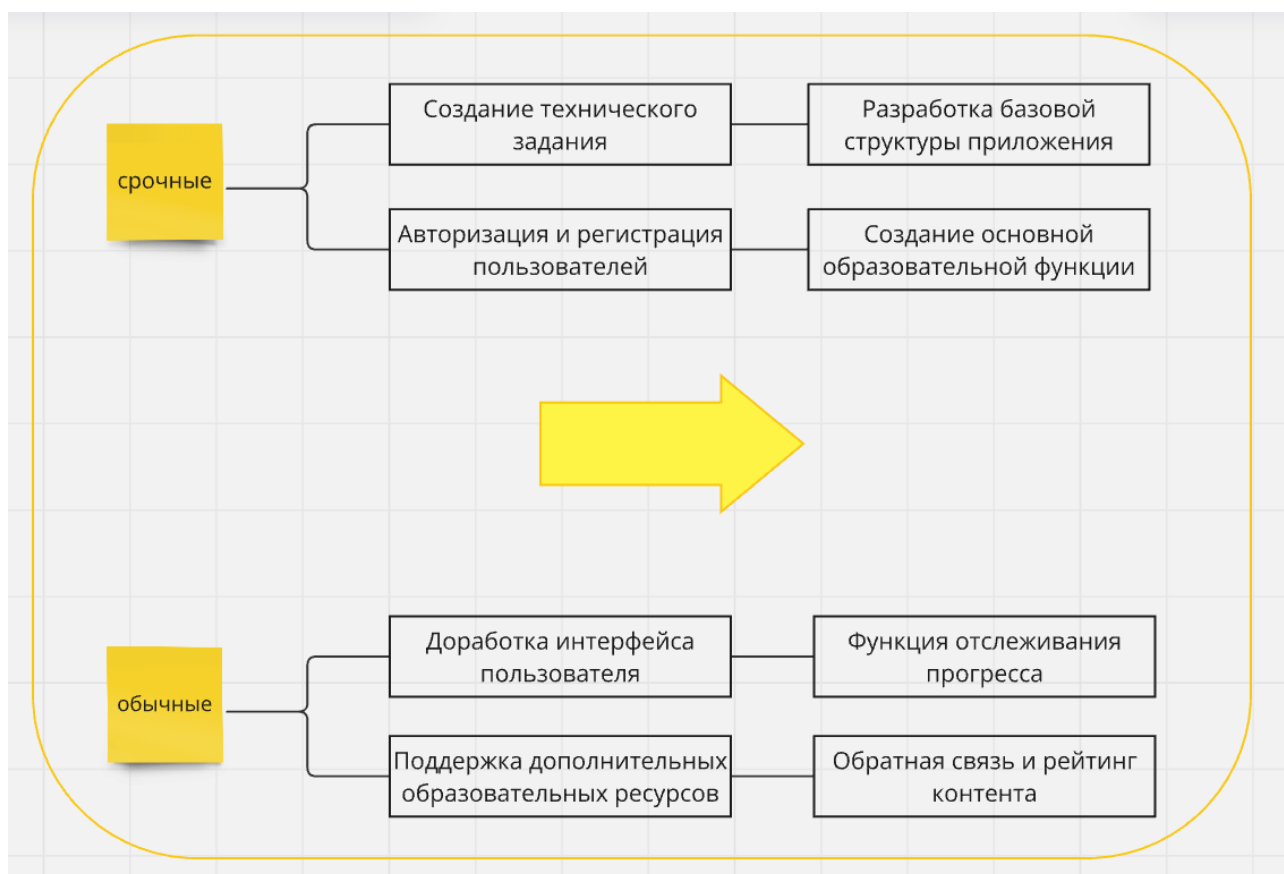


Рисунок 1 – Результаты формирования доски в Kaiten

Задачи:

1. Создание технического задания
2. Разработка базовой структуры приложения
3. Авторизация и регистрация пользователей
4. Создание основной образовательной функции
5. Доработка интерфейса пользователя
6. Функция отслеживания прогресса
7. Поддержка дополнительных образовательных ресурсов
8. Обратная связь и рейтинг контента

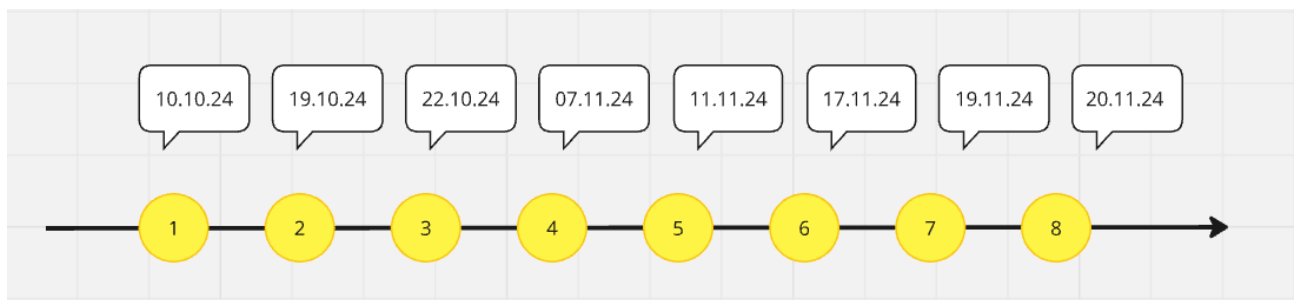


Рисунок 2 - Timeline (лента времени)

В Miro была создана доска для расстановки приоритетов в задачах проекта. В первую очередь были настроены блоки для разделения задач на срочные и обычные, что позволяет видеть ключевые этапы разработки и выделять наиболее важные задачи. Для каждой задачи был установлен временной интервал, что позволяет отслеживать прогресс и соблюдать сроки выполнения на всех этапах. Благодаря использованию доски с приоритетами и временной шкалой команда может эффективно распределять ресурсы, своевременно выполнять ключевые задачи и поддерживать высокую степень организованности в работе над проектом.

Выводы

Освоение навыков разработки учебных проектов включает в себя создание технического задания, выбор подходящей методологии и фреймворка для управления процессом. Для образовательного веб-приложения на Python оптимально использовать итеративную модель жизненного цикла: она позволяет постепенно дорабатывать проект с помощью повторяющихся циклов. Методология Agile идеально подходит для такого подхода, так как обеспечивает гибкость в планировании, регулярные проверки и возможность оперативно реагировать на изменения. Agile помогает как в групповой, так и в индивидуальной работе, создавая условия для эффективного взаимодействия между участниками и заказчиком. В итоге итеративный подход и Agile-методология способствуют повышению качества проекта, его соответствию требованиям и максимальной адаптации к образовательным целям.