**Министерство науки и высшего образования Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**(Университет ИТМО)**

Факультет **Инфокоммуникационных технологий**

Образовательная программа **Мобильные и сетевые технологии**

Направление подготовки (специальность) **09.03.03 Прикладная информатика**

**К У Р С О В О Й П Р О Е К Т**

**по дисциплине «Инфокоммуникационные системы и технологии»**

на тему: Социальная сеть университета ИТМО

Обучающийся Москалец Данила Алексеевич, группа К3141

**Работа сдана**

Дата 07.01.2024

Санкт-Петербург 2024

**СОДЕРЖАНИЕ**

[ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ 3](#_heading=h.30j0zll)

[ВВЕДЕНИЕ 4](#_heading=h.1fob9te)

[1 Суть проекта 5](#_heading=h.3znysh7)

[2 Процессы работы над проектом 6](#_heading=h.2et92p0)

[3 Решение задач 7](#_heading=h.tyjcwt)

[3.1 Проектирование базы данных 7](#_heading=h.3dy6vkm)

[3.2 Реализация базы данных 9](#_heading=h.1t3h5sf)

[3.3 Обеспечение безопасности и целостности данных 11](#_heading=h.4d34og8)

[3.4 Тестирование базы данных 12](#_heading=h.2s8eyo1)

[3.5 Написание документации 12](#_heading=h.17dp8vu)

[4 Анализ работы 13](#_heading=h.3rdcrjn)

[5 Взаимодействие команды, оценка 14](#_heading=h.26in1rg)

[ЗАКЛЮЧЕНИЕ 15](#_heading=h.lnxbz9)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 16](#_heading=h.35nkun2)

[ПРИЛОЖЕНИЕ А 17](#_heading=h.1ksv4uv)

**ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ**

В настоящем отчете о НИР применяют следующие сокращения и обозначения

iOS – iphone operational system, мобильная операционная система

СУБД – система управления базами данных

API - Application Programming Interface, программный интерфейс

БД – база данных

ПО – программное обеспечение

SQL - Structured Query Language, язык запросов

ТЗ – техническое задание

**ВВЕДЕНИЕ**

Целью данного проекта является создание мобильного приложения для социальной сети университета ИТМО, которое будет обеспечивать эффективное и удобное взаимодействие студентов, преподавателей и администрации университета. Приложение должно предоставлять возможность обмена информацией, обсуждения академических вопросов. Это поможет улучшить коммуникацию и сотрудничество между всеми участниками университетского сообщества, а также повысить уровень учебного процесса и студенческой жизни в целом.

Актуальность проекта заключается в том, что в современном обществе, практически каждый человек использует социальные сети, и это стало неотъемлемой частью жизни. Студенты университета ИТМО также пользуются различными мессенджерами и социальными сетями, однако они сталкиваются с проблемой недостатка коммуникации между собой. Им часто приходится пропускать информацию, которая затерялась среди множества сообщений в различных социальных сетях. Создание социальной сети, в которой зарегистрированы только люди, связанные с университетом, будет успешным решением. Все пользователи будут студентами, преподавателями и другим персоналом, и вся активность будет связана с жизнью университета.

Задачи проекта:

* Реализовать приложение под операционную систему iOS
* Протестировать приложение
* Проанализировать полученный результат

**1 Суть проекта**

Суть проекта заключается в создании приложения под операционную систему iOS, которая связывает всех студентов, преподавателей и сотрудников университета в одной сети, создавая удобное и эффективное пространство для общения, обмена информацией и учебных материалов, а также для организации различных событий и активностей.

iOS был выбран нашей командой, так как данная платформа является одной из самых популярных в мире и нашей команде было интересно поработать именно с данной операционной системой.

Основная функциональность социальной сети университета включает:

* Регистрация и вход в аккаунт
* Публикация постов, фото, видео и других материалов
* Возможность настройки своего аккаунта
* Общение с другими пользователями через комментарии к постам
* Просмотр ленты новостей с самыми популярными и интересными материалами
* Возможность оценивать и комментировать материалы других пользователей

Сочетание всех этих функций создает пространство для активного участия студентов, преподавателей и сотрудников в жизни университетского сообщества, способствует легкому обмену информацией, созданию новых связей, совместной работе и развитию академических и профессиональных навыков.

**2 Процессы работы над всем проектом**

В начале работы над проектом нашей командой были следующие этапы работы над проектом:

* Разработка технического задания. Определение основных требований и задач. Описание функциональности. Создание “дорожной карты”. Вы можете ознакомиться с ТЗ в приложении А
* Разработка дизайна. Создание макетов экранов, подбор палитры цветов, работа на типографикой, разработка иконок и изображений, корректировка макетов
* База данных. Проектирование базы данных, реализация, обеспечение безопасности и целостности данных.
* Разработка Backend. Проектирование и организация кода, разработка API
* Рекомендательная система. Подготовка и обработка данных, написание API метода
* Разработка iOS клиента. Проектирование и организация кода. Создание экранов приложения
* Защита проекта
* Написание индивидуальных отчетов

Моя зона ответственности – разработка базы данных, моими задачами были:

1. Проектирование базы данных
2. Реализация базы данных
3. Обеспечение безопасности и целостности данных
4. Тестирование базы данных
5. Написание документации для дальнейшего взаимодействия backend разработчика с базой данных

**3 Решение задач**

3.1 Проектирование базы данных.

В проектирование базы данных входит несколько пунктов [5]:

* Анализ требований. Проектирование сущностей и атрибутов

На данном этапе происходило изучение требований к базе данных, определение основных сущностей (таблиц) и атрибутов (столбцы), которые будут храниться в базе данных. Так же определение типов данных для каждого атрибута и их ограничения. Описание сущностей и атрибутов приведено в таблице 1 (страница 8)

* Определение основных связей

Этот этап включает определение связей между различными сущностями.

* Нормализация базы данных

Процесс структурирования базы данных таким образом, чтобы данные были сохранены в наиболее эффективном и организованном виде. [4]

* Создание схемы (см. Рисунок 1)

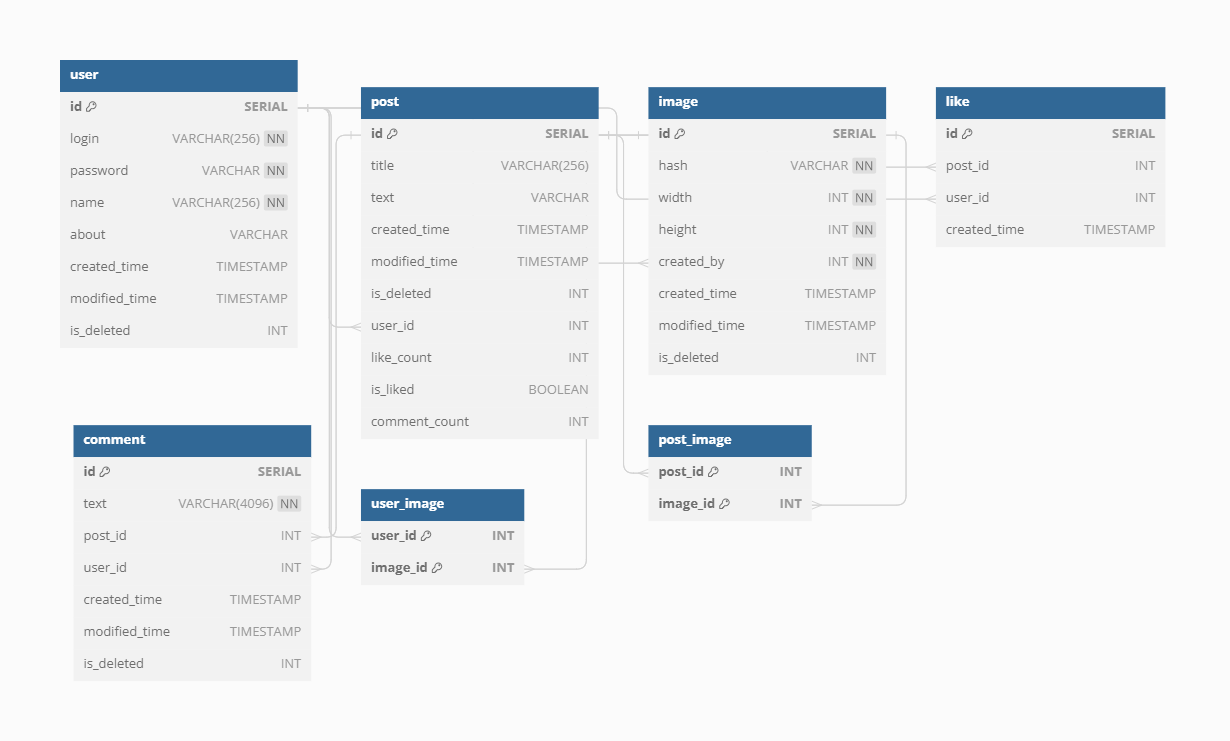


Рисунок 1 – схема базы данных

Таблица 1 – список таблиц в базе данных

| Сущности | Атрибуты | Описание таблицы |
| --- | --- | --- |
| user | id — уникальный номер записи  login — уникальный ник пользователя  password — зашифрованный пароль пользователя  name — имя пользователя  about — описание пользователя  created\_time — время создания записи  modified\_time — время изменения записи  is\_deleted — удалена ли запись | В этой таблице хранится список всех пользователей |
| post | id — уникальный номер записи  title — заголовок поста  text — текст поста  like\_count — количество отметок “нравится”  is\_liked — есть ли на посте отметки “нравится”  comment\_count — количество комментариев  created\_time — время создания записи  modified\_time — время изменения записи  is\_deleted — удалена ли запись | В таблице хранится список всех постов |
| image | id — уникальный номер записи  hash — хэш фотографии  width — ширина фотографии  height — высота фотографии  created\_by — кем было создано  created\_time — время создания записи  modified\_time — время изменения записи  is\_deleted — удалена ли запись | В таблице хранятся все изображения. Связь с другими many to many |
| user\_image | user\_id — уник. идентификатор пользователя  image\_id — уник. идентификатор изображения | Таблица для связывания таблицы пользователей и таблицы изображений |
| post\_image | post\_id — уник. идентификатор поста  image\_id — уник. идентификатор изображения | Таблица для связывания таблицы постов и таблицы изображений |
| comment | id — уникальный номер записи  text — текст комментария  post\_id — уник. идентификатор поста  user\_id — уник. идентификатор пользователя  created\_time — время создания записи  modified\_time — время изменения записи  is\_deleted — удалена ли запись | В таблице хранится список всех комментариев |
| like | id — уникальный номер записи  post\_id — уник. идентификатор поста  user\_id — уник. идентификатор пользователя  created\_time — время создания записи | В таблице хранится список всех отметок “нравится” |

3.2 Реализация базы данных

В качестве системы управления базами данных была выбрана PostgreSQL, так как у данной СУБД множество возможностей. Созданный с использованием объектно-реляционной модели, PostgreSQL поддерживает сложные структуры и широкий спектр встроенных и определяемых пользователем типов данных. Он обеспечивает расширенную емкость данных и заслужил доверие бережным отношением к целостности данных [1]. PostgreSQL имеет ряд преимуществ: надежность, масштабируемость, расширяемость, большое сообщество, бесплатное и открытое ПО. Изначально база данных была реализована с помощью SQL запрос на локальном сервере (см. Рисунок 2)

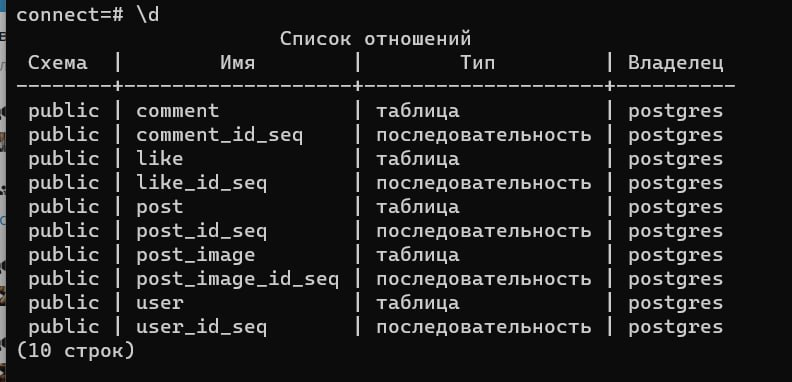


Рисунок 2 – База данных PostgreSQL

Для дальнейшего удобного связывания базы данных и backend части было решено использовать SQLAlchemy, так как данная технология позволяет работать с базой данных, используя объектно-ориентированный подход. Преимуществами данной технологии является кроссплатформенность, гибкость, безопасность, масштабируемость. [2]

Мне необходимо было реализовать сущности и запросы к ним. (см. Рисунок 3)



Рисунок 3 – пример реализации сущности

Необходимы запросы:

Для пользователя: добавить пользователя, изменить данные, получить всех пользователей, получить определенного пользователя по id

Для постов: добавить пост, добавить пост с изображением, получить все посты, получить пост по id, удалить пост по id

Для отметок “нравится”: добавить, удалить, получить количество

Для комментариев: добавить комментарий, удалить, изменить, получит количество всех комментариев определенного поста.

Пример запроса можно увидеть на рисунке 4

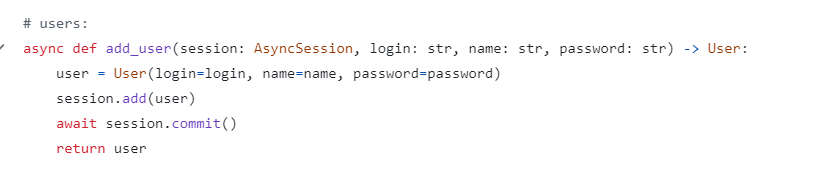


Рисунок 4 – пример запроса

3.3 Обеспечение безопасности и целостности данных

Настройка различных уровней безопасности для базы данных, такие как права доступа и шифрование данных. Добавление ограничения целостности данных для гарантировать, что данные в базе остаются согласованными и без повреждений. [3]

Методы обеспечения безопасности:

1. Хэширование паролей

2. Предотвращение от SQL-инъекций с помощью инструментов SQLAlchemy.

3. Использование JWT авторизации (на уровне backend)

* 1. Тестирование базы данных

Различные тесты для проверки функциональности базы данных, ее производительности и надежности. В тестировании включено тестирование запросов, проверка на возможность восстановления после сбоев и проверка безопасности.

* 1. Написание документации для дальнейшего взаимодействия backend разработчика с базой данных.

В документацию входит описание сущностей и атрибутов, описание запросов к базе данных.

Вы можете ознакомиться с документацией по ссылке: <https://docs.google.com/document/d/11-iWjyEqCbfQu_E_hBkUzmHCxYaow2WWLFflWsz95P8/edit?usp=sharing>

1. **Анализ работы**

За время выполнения проекта были выполнены поставленные задачи, как индивидуальные, так и всего проекта в целом. В течение работы над проектом у меня возникли следующие трудности:

* Возникали небольшие трудности с работой с SqlAlchemy, так как для меня это первый опыт
* При связывании базы данных с backend частью были небольшие недопонимания с коллегой

Несмотря на трудности, за время участия в проекте я научился:

* Проектировать базы данных
* Работать над проектом в команде, в частности взаимодействовать с базой данных вместе с коллегами
* Работать с технологией SqlAlchemy

1. **Взаимодействие в команде, оценка**

В ходе работы я взаимодействовал с дизайнером для понимания необходимости некоторых сущностей и атрибутов, с backend разработчиком. Довольно часто возникали ситуации, когда необходимо было переписать некоторые запросы, добавить новые атрибуты.

Во время работы над проектом наша команда взаимодействовала в общем чате Telegram, где решались вопросы по проекту.

Внутри команды у нас получилось выстроить хорошее взаимопонимание. Мы проводили регулярные онлайн встречи, на которых обсуждали прогресс и препятствия, чтобы убедиться, что каждый член команды понимает свою роль и задачи. Это помогло нам синхронизироваться и избежать возможных конфликтов. Благодаря хорошей коммуникации и распределению обязанностей, мы эффективно использовали ресурсы и сократили время выполнения проекта.

Всю нашу работу организовывал магистрант Алексей Красиков. Он каждую неделю проводил онлайн встречи, поддерживал нас, помогал разобраться со всеми трудностями. Поэтому я считаю, что наш руководитель полностью справился со своими обязанностями. Оценка 5 из 5.

**ЗАКЛЮЧЕНИЕ**

Задачи, поставленные передо мной полностью выполнены.

Команда успешно справилась со всеми поставленными задачами и достигла поставленной цели - создания социальной сети ИТМО. В ходе проекта были разработаны и реализованы все необходимые функциональности.

Команда успешно преодолела все трудности, возникшие на пути к достижению цели. Были поставлены четкие сроки и задачи, которые были выполнены вовремя и с высоким качеством. Каждый член команды был ответственен за свою область работы и внес значимый вклад в разработку проекта.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Чем PostgreSQL лучше других SQL баз данных с открытым исходным кодом. Часть 1 — URL: https://habr.com/ru/articles/282764/ (дата обращения: 17.11.2023)
2. Крадущийся тигр, затаившийся SQLAlchemy. Основы — URL: https://habr.com/ru/articles/470285/ (дата обращения: 19.11.2023)
3. Безопасность в базах данных — URL: https://habr.com/ru/companies/otus/articles/732850/ (дата обращения: 20.11.2023)
4. Нормализация отношений. Шесть нормальных форм — URL: https://habr.com/ru/articles/254773/ (дата обращения: 16.11.2023)
5. Основы правил проектирования базы данных — URL: https://habr.com/ru/articles/514364/ (дата обращения: 16.11.2023)

**ПРИЛОЖЕНИЕ А**

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

1. **Название проекта**

Социальная сеть университета ИТМО

1. **Цель (назначение)**

Создать социальную сеть университета ИТМО в виде мобильного приложения, которое будет обеспечивать эффективное и удобное взаимодействие студентов, преподавателей и администрации университета. Это поможет обеспечить коммуникацию и сотрудничество между всеми участниками университетского сообщества.

1. **Сроки выполнения**

Начало    01 ноября 2023 г. Окончание 20 декабря 2023 г.

1. **Исполнитель проекта (руководитель проекта)**

Красиков Алексей Александрович

1. **Термины и сокращения**

Лента – список публикаций социальной сети

REST API – это способ взаимодействия сайтов и веб-приложений с сервером

Ранжирование – сортировка публикаций

Git-репозиторий – это все файлы, находящиеся под контролем версий, вместе с историей их изменения и другой служебной информацией

Хостинг – сервис, который предоставляет услуги размещения сервера для удаленного доступа клиентов к приложению

1. **Технические требования**

Функциональные требования:

* приложение должно поддерживать создание аккаунтов пользователей;
* пользователи должны иметь возможность изменять описание своего профиля;
* приложение должно предоставлять функции для создания публикаций, включая возможность прикрепления фотографий и текста к публикациям;
* пользователи должны иметь возможность удалять свои публикации;
* приложение должно предоставлять функцию “нравится”, позволяющую пользователям ставить и снимать отметки “нравится” к публикациям;
* приложение должно предоставлять функции для написания комментариев к публикациям и чтения всех комментариев;
* приложение должно включать рекомендательную систему публикаций в ленте.

Нефункциональные требования:

* приложение должно быть совместимо с мобильной операционной системой iOS;
* приложение должно обеспечивать быстрый отклик и высокую производительность;
* приложение должно быть безопасным и защищать конфиденциальность пользователей;
* приложение должно быть простым в использовании и иметь интуитивно понятный интерфейс.

Требования к поддержке и обслуживанию:

* приложение должно обновляться регулярно для исправления ошибок и добавления новых функций;
* пользователи должны иметь возможность получать поддержку в случае возникновения проблем или вопросов.

1. **Содержание работы (этапы по срокам, можно в таблицу)**

| № | Этапы проекта | Сроки выполнения этапов | Ответственный за этап | Вид представления результатов этапа |
| --- | --- | --- | --- | --- |
| 1. | Разработка технического задания | 1.11 – 10.11 | Красиков А. А. | Техническое задание |
| 2. | Мониторинг и контроль выполнения | 10.11 – 20.12 | Красиков А. А. | Приложение |
| 3. | Разработка дизайна | 13.11 – 20.11 | Майстренко А. Н. | Прототип Figma |
|  | Создание макетов экранов | 13.11 – 15.11 | Майстренко А. Н. | Прототипы экранов в Figma |
|  | Подбор палитры цветов | 16.11 | Майстренко А. Н. | Палитра цветов в Figma |
|  | Работа над типографикой | 17.11 | Майстренко А. Н. | Визуальное оформление текста в приложении в Figma |
|  | Разработка иконок и изображений | 20.11 | Майстренко А. Н. | Иконки и изображении на макетах Figma |
|  | Корректировка макетов | 20.11 | Майстренко А. Н. | Макеты экранов в Figma |
| 4. | База данных | 16.11 – 20.11 | Москалец Д. А. | База данных |
|  | Проектирование базы данных | 16.11 | Москалец Д. А. | Схема хранения данных |
|  | Реализация базы данных | 17.11 – 20.11 | Москалец Д. А. | База данных, готовая для создания, чтения, обновления и удаления данных |
|  | Обеспечение безопасности и целостности данных | 20.11 | Москалец Д. А. | Реализация шифрования и защиты данных |
| 5. | Разработка Backend | 20.11 – 30.11 | Владзиевский Е. Д. | Серверная часть, обрабатывающая запросы iOS клиента |
|  | Проектирование и организация кода | 20.11 | Владзиевский Е. Д. | Git-репозиторий с готовой структурой проекта |
|  | Разработка API метода регистрации и аутентификации | 21.11 | Владзиевский Е. Д. | API метод регистрации и аутентификации |
|  | Разработка API метода изменения описания профиля пользователя | 22.11 | Владзиевский Е. Д. | API метод изменения описания профиля пользователя |
|  | Разработка API методов создания и удаления публикаций | 22.11 | Владзиевский Е. Д. | API методы создания и удаления публикаций |
|  | Разработка API метода получения всех публикаций (хронологический порядок) | 23.11 – 27.11 | Владзиевский Е. Д. | API метод получения всех публикаций в хронологическом порядке |
|  | Разработка API метода выставления отметки «Нравится» | 27.11 | Владзиевский Е. Д. | API метод выставления отметки «Нравится» |
|  | Разработка API метода просмотра всех комментариев к публикации | 28.11 – 29.11 | Владзиевский Е. Д. | API метод просмотра всех комментариев к публикации |
|  | Разработка API методов создания и удаления комментариев к публикации | 30.11 | Владзиевский Е. Д. | API методы создания и удаления комментариев к публикации |
| 6. | Разработка рекомендательной системы | 20.11 – 5.12 | Гоголева В. П. | API метод ранжирования публикаций |
|  | Подготовка и обработка данных | 20.11 – 21.11 | Гоголева В. П. | Схема хранения данных |
|  | Разработка API метода получения всех публикаций (умная сортировка) | 23.11 – 28.11 | Гоголева В. П. | API метод получения всех публикаций в рекомендательном порядке |
|  | Исследование возможности использования машинного обучения для улучшения системы рекомендаций в будущем | 29.11 – 5.12 | Гоголева В. П. | Документ с исследованием |
| 7. | Разработка iOS клиента | 24.11 – 11.12 | Кошкарев К. П  Зотеев М. Е. Майстренко А. Н. | Мобильное приложение под iOS |
|  | Проектирование и организация кода | 24.11 | Кошкарев К. П.  Зотеев М. Е. Майстренко А. Н. | Git-репозиторий с готовой структурой проекта |
|  | Разработка экрана регистрации и авторизации | 27.11 – 1.12 | Кошкарев К.П. | Прототип соответствующего макету экрана регистрации и авторизации |
|  | Разработка экрана профиля пользователя | 27.11 – 5.12 | Майстренко А. Н. | Прототип соответствующего макету экрана профиля пользователя |
|  | Разработка экрана создания публикации | 27.11 – 1.12 | Зотеев М. Е. | Прототип соответствующего макету экрана создания публикации |
|  | Разработка экрана просмотра ленты публикаций | 4.12 – 11.12 | Кошкарев К. П. | Прототип соответствующего макету экрана ленты |
|  | Разработка экрана комментариев к публикации | 4.12 – 11.12 | Зотеев М. Е. | Прототип соответствующего макету экрана комментариев |
| 8. | Написание отчетов | 10.12 – 19.12 | Красиков А. А.  Майстренко А. Н.  Москалец Д. А.  Владзиевский Е. Д.  Гоголева В. П.  Кошкарев К. П.  Зотеев М. Е. | Отчеты по областям ответственностей |
| 9. | Защита проекта | 20.12 | Красиков А. А. | Доклад и презентация |

1. **Основные результаты работы и формы их представления**

Результатом работы является приложение под платформу iOS, которое можно запустить на мобильном устройстве пользователя или симуляторе приложений Apple. Оно должно взаимодействовать с разработанным и запущенном на хостинге сервером и выполнять требования, описанные в п. 6.