**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 облачные 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 ИТМО

Обучающийся Владзиевский Е.Д., группа К3141

**Работа сдана**

Дата 07.01.2023

Санкт-Петербург 2023

СОДЕРЖАНИЕ

ВВЕДЕНИЕ 3

1 Основные аспекты проекта 4

1.1 Суть проекта 4

1.2 Методы и процессы работы над проектом 4

2 Решение поставленных задач 7

2.1 Описание моих персональных задач 7

2.2 Ход и методы решения 7

3 Анализирование и взаимодействие с участниками проекта 12

3.1 Анализ проделанной работы 12

3.2 Взаимодействие с командой и руководителем 12

3.3 Оценка работы руководителя 12

ЗАКЛЮЧЕНИЕ 14

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 16

ПРИЛОЖЕНИЕ A 17

ВВЕДЕНИЕ

В современном мире информационных технологий образование и наука становятся все более важными сферами, требующими активного взаимодействия и обмена знаний. Однако существующие образовательные учреждения часто сталкиваются с вызовами, такими как недостаток эффективного взаимодействия между студентами, потеря интересной информации по учебной и научной тематике, а также ограниченные учебные и профессиональные возможности для их участников.

В ответ на эти вызовы, предлагается создание социальной сети, ориентированной на университетское сообщество ИТМО. Эта платформа объединит студентов и преподавателей в виртуальном пространстве, предоставляя им уникальные возможности для обмена научными достижениями, учебными материалами и поиском партнеров для сотрудничества.

Целью данного проекта является улучшение взаимодействия и обмена информацией внутри университетского сообщества, а также создание эффективной среды для научного и профессионального развития студентов и преподавателей ИТМО. Платформа позволит каждому участнику выделить свои научные труды, демонстрировать достижения и находить единомышленников для совместной работы.

**1 Основные аспекты проекта**

**1.1 Суть проекта**

Проект "​​Социальная сеть ITMO" являет собой разработку удобного приложения, предоставляющего пользователям возможность взаимодействия с социальной сетью через устройства Apple. Клиент обеспечивает функциональность, такую как публикации записей, участие в обсуждениях, поиск коллег, обеспечивая мобильный доступ к академической среде университета ИТМО.

**1.2 Методы и процессы работы над проектом**

Для выполнения проекта был выбран метод "гибкой разработки", который эффективно адаптировался к новым задачам и изменениям в процессе работы. Этот подход также способствовал независимой реализации задач участниками проекта. Управление проектом и поддержание постоянной связи с командой осуществлялись с использованием современных инструментов и платформ, таких как GitHub, Telegram, Discord.

В ходе работы над проектом было 4 основных процесса: проектирование дизайна, создание базы данных для хранения информации, разработка IOS клиента, создание API

Следующая таблица дает исчерпывающее описание всех этапов работы:

Таблица 1– Этапы работы

|  |  |
| --- | --- |
| Название этапа | Описание этапа |
| ​​Подготовительный этап | Перед тем как приступить к активной фазе работы над проектом, команда провела несколько встреч с целью выявления основных проблем. На этих совещаниях были установлены основные направления для разработки функционала, сформулированы требования к социальной сети, а также распределены задачи между участниками проекта. |
| UX/UI  проектирование | На данном этапе команда определила основные требования к интерфейсу, после чего был создан макет приложения в Figma |
| Проектирование, разработка БД | Была выбрана PostgreSQL в качестве СУБД, учитывая требования проекта и тип данных, с которыми необходимо было работать. Для написания запросов к СУБД использовалась ORM SQLAlchemy |
| Разработка API | Для приложения необходим API, и для его разработки был использован FastAPI. Для удобства развертывания и изоляции приложения применялся Docker. |

Продолжение таблицы 1

|  |  |
| --- | --- |
| Разработка IOS приложения | Разработка iOS-приложения включала в себя планирование, выбор технологий, создание бэкенда, интеграцию с API, реализацию функциональности, тестирование |

**2 Решение поставленных задач**

**2.1** **Описание моих персональных задач**

Передо мной стояла задача разработать API для социальной сети, включающее ряд ключевых функций:

* Регистрация и авторизация пользователей: Создание механизма для регистрации новых пользователей и обеспечение безопасной авторизации.
* Получение и изменение профиля пользователя: Разработка функционала, который позволит пользователям получать информацию о своем профиле и вносить необходимые изменения.
* Получение всех публикаций с хронологической сортировкой: Создание механизма, позволяющего пользователям просматривать все публикации в хронологическом порядке.
* Создание и удаление публикаций: Разработка функций для публикации нового контента и удаления существующих записей.
* Комментарии и отметки "Нравится" к публикациям: Внедрение возможности оставлять комментарии к публикациям и выражать свое одобрение с помощью отметок "Нравится".

**2.2 Ход работы и методы решения**

Начиная работу над поставленными задачами, критически важно выделить и тщательно обсудить все ключевые требования, учитывая мнения и потребности конечных пользователей. Этот подход не только способствует активному вовлечению всех сторон в процесс, но и создает прочную основу для успешного развития и реализации проекта в долгосрочной перспективе.

На начальном этапе я занялся проектированием структуры API, схем приема и отдачи данных с учетом мнения и потребностей IOS разработчиков.

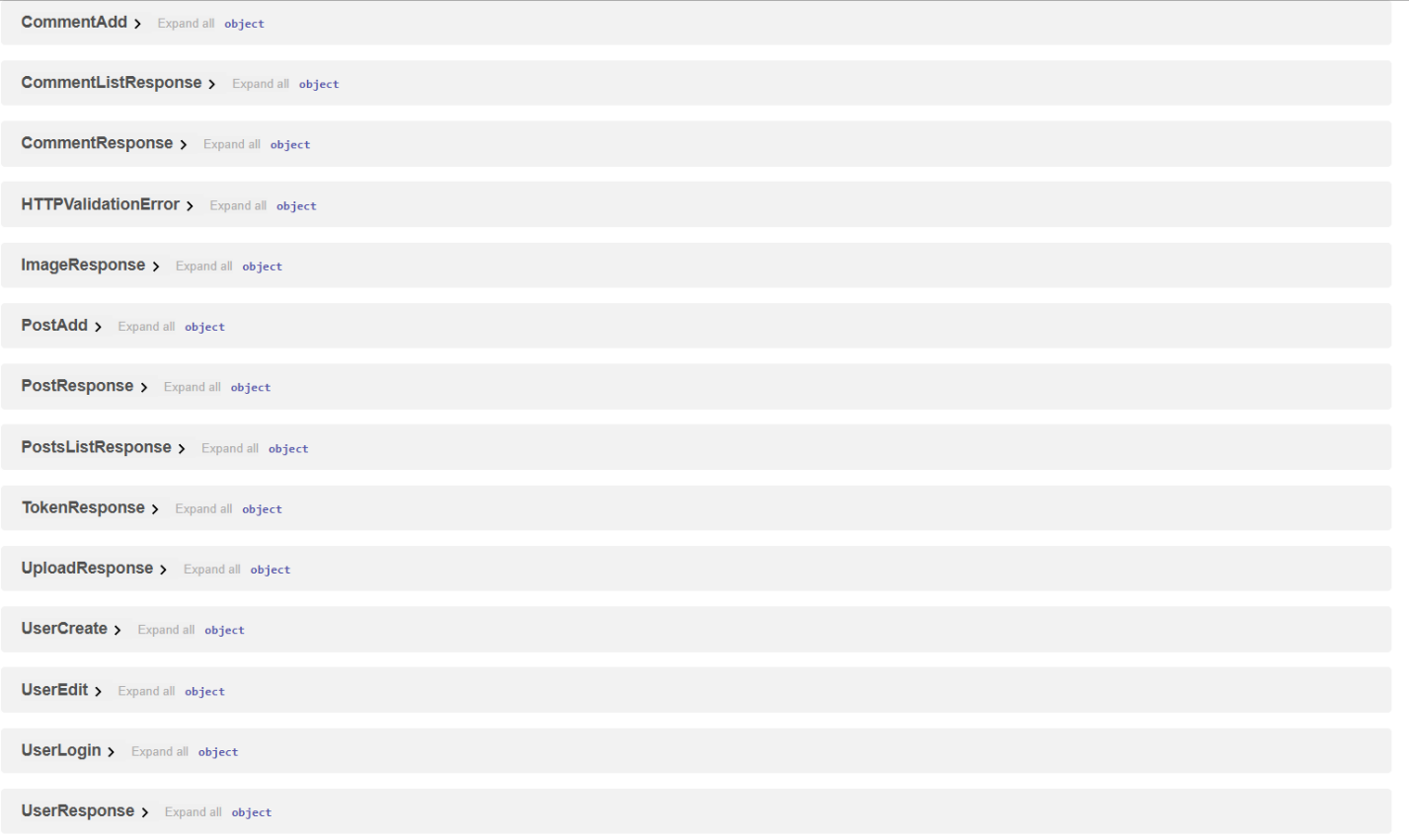


Рисунок 1 – Схемы API

Перед приступлением к написанию бэкенд-кода мы приняли стратегическое решение использовать Docker контейнер для упаковки всего серверного функционала и его зависимостей. Это позволяет нам обеспечить изоляцию приложения и создать консистентную среду выполнения на всех этапах разработки и внедрения.

Создание Docker контейнера включает в себя описание всех необходимых зависимостей и настроек в Dockerfile, обеспечивая таким образом воспроизводимость окружения. Это особенно важно для поддержания согласованности между различными средами, такими как разработка, тестирование и продакшн.

Дополнительно для упрощения развертывания и настройки всей инфраструктуры на серверах был создан файл docker-compose. В этом файле описаны все компоненты системы, включая контейнеры приложения и базы данных, а также настройки сети и переменные окружения. Docker-compose обеспечивает удобный способ управления всей инфраструктурой как единым целым, что упрощает процесс развертывания на различных средах.

Этот подход не только обеспечивает гибкость в управлении приложением, но и упрощает процесс масштабирования, а также минимизирует потенциальные проблемы, связанные с различиями в окружении между разработкой и продакшеном.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 2 – Docker compose файл

После этого начал реализовывать необходимые конечные точки, сначала был создан механизм безопасной регистрации и авторизации с использованием JWT (JSON Web Tokens). При успешной аутентификации пользователю выдаются два типа токенов: access token и refresh token. Access token используется для аутентификации пользователя в системе и имеет ограниченный срок действия, что повышает безопасность. Refresh token, в свою очередь, служит для обновления access token без необходимости повторной авторизации пользователя. Этот механизм обеспечивает эффективное управление безопасностью и доступом к функционалу социальной сети.

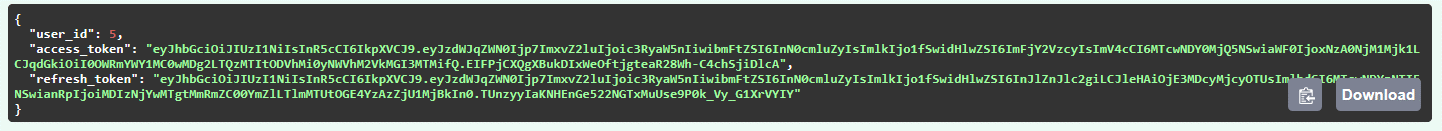


Рисунок 2 – Ответ сервера при успешной авторизации

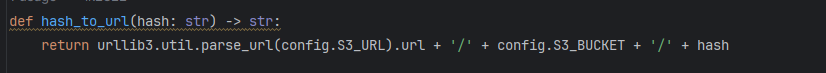
В связи с возможностью пользователей прикреплять фотографии к публикациям в социальной сети, потребовалась эффективная и удобная система хранения изображений. Для этой цели было принято решение использовать хранилище, совместимое с S3, а именно MinIO. При загрузке изображений клиент получает хэш, который может быть использован при добавлении изображений к постам. Когда же серверу нужно отдать пост с картинкой, то хэш из базы данных на этапе валидации ответа от сервера с помощью Pydantic конвертируется в ссылку к MinIO, что позволяет уменьшить количество запросов к API.

Рисунок 3 – Функция конвертации хэша в ссылку

В ходе тестирования также была выявлена проблема с перепутанными значениями ширины и высоты в базе данных. После дополнительного исследования стало ясно, что камера в устройствах iOS сохраняет информацию о вращении изображения в метаданных EXIF, вместо того чтобы изменять физические размеры (ширину и высоту) изображения. В связи с чем пришлось с помощью библиотеки Pillow разворачивать изображение и менять местами ширину и высоту.

В заключительной стадии разработки основное внимание было уделено реализации функций добавления постов, комментариев и отметок "Нравится". Пользователи получили возможность создавать и публиковать посты с текстовым и медийным контентом, оставлять комментарии и выражать свои предпочтения с помощью отметок "Нравится". Также стоит отметить, что при разработке API были применены передовые практики написания REST-сервисов с использованием FastAPI, включая использование механизма внедрения зависимостей и структурирование функциональности с помощью "роутеров".

**3 Анализ работы и взаимодействие участниками проекта**

**3.1 Анализ проделанной работы**

Разработка бэкенда и iOS-приложения осуществлялась синхронно, что способствовало оперативному получению обратной связи по внедряемому функционалу. В ходе работы были успешно реализованы все определенные в техническом задании задачи. Тем не менее, в связи с ограниченной компетенцией одного из участников команды возникла необходимость в предоставлении дополнительной поддержки и содействии в выполнении его задач.

**3.2 Взаимодействие с командой и руководителем**

Процесс работы над проектом включал в себя активное взаимодействие с членами команды и руководителем с целью эффективного выполнения задач и достижения общих целей. Регулярные онлайн-совещания в Zoom использовались для распределения ролей в проекте, обмена статусом по задачам и обеспечения обратной связи. Для постоянной связи с руководителем были созданы чаты в Telegram. Поддержка от других участников команды и консультации с руководителем по техническим вопросам способствовали более качественному выполнению поставленных задач.

**3.3 Оценка работы руководителя**

Оценивая действия и результаты команды на текущем этапе, следует подчеркнуть высокий уровень согласованности и эффективности всех участников проекта, а также отметить четкое и профессиональное руководство со стороны Красикова А. А. На протяжении всего процесса работы над проектом команда успешно соблюдала установленные дедлайны, демонстрируя отличную организацию рабочего процесса и сильный командный дух.

Каждая персональная задача была ясно сформулирована и внедрена в доску проекта на сайте Odoo, что способствовало прозрачности и легкости отслеживания хода выполнения задач. Эффективное использование таких инструментов управления проектом свидетельствует о высокой степени профессионализма команды.

Руководитель проявлял постоянную готовность оказать помощь по техническим вопросам и предоставлял ценные консультации в области разработки. Этот активный и поддерживающий подход руководителя способствовал разрешению технических трудностей и содействовал успешному продвижению проекта.

Учитывая вышеописанные факторы, команда проекта "Социальная сеть ИТМО" единогласно приняла решение поставить Красикову А. А. высший балл - 5 из 5. Эта высокая оценка отражает не только результативность руководства, но и успешность всей команды в достижении общих целей проекта.

ЗАКЛЮЧЕНИЕ

В заключении хочется сказать, что команды достигла всех поставленных целей. Процесс разработки продемонстрировал высокий уровень координации и эффективности внутри команды. Руководитель проекта, Красиков А. А., проявил ясное руководство, что сыграло ключевую роль в достижении поставленных целей. Регулярные онлайн-совещания и использование современных инструментов управления проектом способствовали хорошей организации рабочего процесса.

Техническая реализация проекта также заслуживает внимания. Разделение проекта на бэкенд и iOS-приложение позволило эффективно координировать разработку обеих частей. Использование современных технологий, таких как Docker, MinIO, FastAPI и Swift, говорит о технической компетентности команды и осознанном подходе к выбору инструментов разработки.

Все вышеперечисленные факторы в совокупности дали возможность команде успешно справиться с поставленными задачами, поддерживая высокий уровень качества и соблюдая сроки.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 Документация FastAPI – URL: https://fastapi.tiangolo.com/

2 Лучшие практики при работе с FastAPI – URL: https://github.com/zhanymkanov/fastapi-best-practices

3 Документация PostgreSQL – URL: https://www.postgresql.org/docs/

4 Документация Docker – URL: https://docs.docker.com/

5 МЕЖГОСУДАРСТВЕННЫЙ СТАНДАРТ. Система стандартов по информации, библиотечному и издательскому делу. ОТЧЕТ О НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ. Структура и правила оформления (ГОСТ 7.32–2017) / Стандартинформ - М., 2017. – 32 с. – URL: https://cs.msu.ru/sites/cmc/files/docs/2021-11gost\_7.32-2017.pdf (дата обращения 2023-12-19)

# ПРИЛОЖЕНИЕ А. ТЕХНИЧЕСКОЕ ЗАДАНИЕ К ПРОЕКТУ.

**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.