

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ
Школа бакалавриата

ОТЧЕТ

По проекту
«Разработка цифрового помощника для адаптации студентов»
по дисциплине «Проектный практикум»

Заказчик: Куклин И.Э.

Куратор: Хрушков А.Е.

Студенты команды «kukushiki»

Метелева Д.А.

Суркина М.О.

Федорова А.В.

Троянов Д.А.

Екатеринбург, 2025

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Аналитика	5
1.1 Анализ целевой аудитории	6
1.2 Определение проблемы.....	7
1.3 Подходы к решению проблемы	8
1.4 Анализ конкурентов	9
2 Разработка.....	12
2.1 Сценарии использования	12
2.2 Требования к продукту и MVP	16
2.2.1 Нефункциональные требования.....	16
2.2.2 Функциональные требования.....	16
2.3 Технологический стек	17
2.4 Прототипирование.....	18
2.5 Проектирование и разработка системы.....	18
3 Индивидуальные отчеты каждого из участников.....	20
3.1 Метелева Дарья, тимлид-разработчик.....	20
3.2 Федорова Алина, аналитик	22
3.3 Суркина Мария, frontend-разработчик	23
3.4 Троянов Дмитрий, backend-разработчик.....	26
ЗАКЛЮЧЕНИЕ	27
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	28
ПРИЛОЖЕНИЕ А Wireframes и макеты из Figma	30
ПРИЛОЖЕНИЕ В Календарный план	33

ВВЕДЕНИЕ

Целью проекта является разработка цифрового помощника в формате Telegram-бота, предназначенного для адаптации студентов к образовательной среде вуза. Основными задачами являются реализация безопасной авторизации и входа в личный кабинет обучающегося на сайте вуза, интеграция интеллектуального модуля для предоставления ответов на часто задаваемые вопросы на основе базы знаний, а также внедрение сервиса автоматического создания конспектов лекций путём обработки аудиофайлов (транскрибация и резюмирование).

Актуальность проекта обусловлена необходимостью повышения качества информационной поддержки студентов, особенно в период начала обучения. Большое количество организационной информации, различия в уровне подготовки и отсутствие опыта взаимодействия с вузовскими системами создают сложности для первокурсников. Применение цифрового помощника, использующего алгоритмы искусственного интеллекта и работающего в привычной для студентов среде (мессенджере Telegram), позволяет обеспечить постоянный доступ к необходимым данным, снизить нагрузку на учебные подразделения и повысить эффективность адаптационных процессов.

Область применения программного продукта охватывает образовательные организации, в которых актуальны задачи сопровождения обучающихся и автоматизации типовых информационных запросов. Разработка может быть использована как в рамках внутренней информационной системы вуза, так и в качестве отдельного инструмента поддержки студентов.

Ожидаемыми результатами проекта являются создание полнофункционального Telegram-бота с возможностью подключения к личному кабинету студента, реализация интеллектуального диалога на основе базы знаний, а также разработка сервиса автоматической транскрибации и

генерации конспектов лекций. По завершении проекта предполагается проведение тестирования, сбор обратной связи и подготовка прототипа для последующего внедрения в учебную среду.

1 Аналитика

Разработка программного продукта началась с анализа потребностей целевой аудитории — студентов образовательного учреждения. На данном этапе были проведены мероприятия по сбору требований от потенциальных пользователей, в том числе анкетирование, целью которого было выявление типичных затруднений студентов в процессе адаптации к обучению и поиску информации. Кроме того, был проведён анализ существующих решений, включая как сторонние образовательные Telegram-боты, так и внутренние системы вузов. Особое внимание при этом уделялось функциональности, удобству использования, доступности, а также качеству ответов, предоставляемых такими системами.

На основе собранных данных были сформулированы как функциональные, так и нефункциональные требования к системе. Среди них: интеграция с личным кабинетом обучающегося через защищённый интерфейс, наличие базы знаний о вузе в структурированном формате (JSON) для генерации ответов, встроенный ИИ-модуль для обработки естественного языка и ведения диалогов, возможность загрузки аудио- и видеофайлов для автоматической транскрибации и создания кратких конспектов, удобный и интуитивно понятный интерфейс взаимодействия через Telegram, а также высокий уровень безопасности персональных данных.

На основании проведённого анализа был сформирован план разработки, включающий следующие этапы:

- 1) разработка базы знаний с вопросами и ответами, специфичными для конкретного вуза;
- 2) подключение и настройка языковой модели для генерации диалогов в рамках заданной тематики;
- 3) реализация авторизации через Telegram с последующим входом в личный кабинет обучающегося;

- 4) внедрение функциональности транскрибации аудиофайлов и автоматического резюмирования лекций;
- 5) разработка пользовательского интерфейса для административной панели на основе принципов UX/UI;
- 6) проведение тестирования с участием студентов для оценки точности, полноты и удобства работы сервиса;
- 7) подготовка технической и пользовательской документации для последующей защиты проекта.

1.1 Анализ целевой аудитории

Для анализа целевой аудитории проекта была использована методика 5W Марка Шеррингтона. Данный подход предполагает рассмотрение пяти ключевых вопросов: «Что?», «Кто?», «Где?», «Когда?» и «Почему?». В результате анализа были получены данные, приведенные ниже.

Что? Разрабатываемый продукт представляет собой Telegram-бота — цифрового помощника для студентов университета. Функциональность бота охватывает доступ к личному кабинету обучающегося, расписанию занятий, навигации по корпусам вуза, а также содержит интеллектуальный модуль взаимодействия с пользователем и систему автоматической генерации конспектов лекций на основе загруженных аудиофайлов.

Кто? Целевую аудиторию составляют студенты университета, прежде всего первокурсники и иногородние обучающиеся. Данная категория пользователей наиболее остро сталкивается с трудностями адаптации в новой образовательной и социальной среде. Также потенциальными пользователями могут быть старшекурсники, использующие сервисы бота в повседневной образовательной деятельности.

Где? Telegram-бот доступен в любом месте при наличии подключения к сети Интернет и установленного клиента Telegram. Наибольшую актуальность

система приобретает на территории университета — при необходимости сориентироваться в корпусах, уточнить расписание или получить доступ к внутренним сервисам.

Когда? Наиболее значимым периодом использования является начало учебного года, когда студенты сталкиваются с максимальным количеством организационных вопросов. Тем не менее, в течение всего периода обучения бот сохраняет актуальность как универсальный информационный инструмент.

Почему? Проект решает ключевые проблемы студентов — недостаток информации, сложности навигации по вузовской инфраструктуре, недоступность некоторых онлайн-сервисов с мобильных устройств, а также необходимость оперативного получения справочной информации. Использование единого цифрового помощника позволяет снизить уровень стресса у студентов и повысить их вовлечённость в образовательный процесс.

По результатам анализа было установлено, что основными потребителями функционала бота выступают молодые люди в возрасте от 17 до 23 лет, активно использующие мессенджеры и цифровые сервисы в повседневной жизни. Проект ориентирован на мобильность, доступность и простоту взаимодействия, что соответствует ожиданиям указанной целевой аудитории.

1.2 Определение проблемы

Были выделены несколько ключевых проблем, решение которых является основной целью данного проекта и направлено на повышение эффективности адаптации студентов и улучшение доступа к важной информации и сервисам вуза:

- 1) ограниченный доступ студентов к необходимой информации и сервисам вуза, что затрудняет процесс адаптации и вызывает стресс;

- 2) необходимость постоянного обращения к разным источникам для получения расписания, финансовых данных и навигации по корпусам, что снижает удобство и эффективность;
- 3) отсутствие централизованного и удобного инструмента для общения с университетской системой и получения оперативных ответов на часто задаваемые вопросы;
- 4) сложности с обработкой и систематизацией учебных материалов, в том числе конспектов лекций, особенно при необходимости трансформации аудио- и видеозаписей в текст;
- 5) недостаточная персонализация и интерактивность существующих решений, что снижает мотивацию студентов к использованию доступных ресурсов.

Проект направлен на решение этих проблем за счёт создания интегрированного цифрового помощника с функциями автоматизации, удобного интерфейса и встроенного ИИ.

1.3 Подходы к решению проблемы

По результатам проведённого анализа потребностей целевой аудитории и изучения существующих решений были выделены ключевые требования к телеграм-боту — цифровому помощнику для адаптации студентов. К основным требованиям относятся удобство интерфейса, высокая точность и оперативность предоставляемой информации, интеграция с личным кабинетом университета, обеспечение конфиденциальности данных, а также поддержка различных типов контента, включая аудиофайлы и видеофайлы для создания конспектов.

Основная идея проекта заключается в разработке многофункционального телеграм-бота, который объединяет доступ к

расписанию, навигации по корпусам и встроенному ИИ-помощнику. Бот позволяет студентам эффективно получать ответы на вопросы, связанные с вузом, а также загружать лекционные материалы для автоматической транскрибации и создания кратких конспектов, что значительно упрощает процесс адаптации и обучения.

1.4 Анализ конкурентов

Для проведения анализа конкурентов были определены основные трудности, с которыми сталкиваются пользователи при использовании учебных сервисов, включая взаимодействие с тьюторами и различными образовательными платформами.

Таблица 1 – Сравнение тьютора и Telegram-бота

Характеристика	Тьютор	Telegram-бот
Ответы на вопросы	Да, но может быть долго	Да, мгновенно через ИИ
Доступность	Ограничена (рабочее время)	Не ограничена
Эмоциональная поддержка	Да, человеческое общение	Да, но частично (дружественный тон ИИ)
Скорость и автоматизация	Медленно, вручную	Быстро

Таблица 2 – Сравнение БРС и Telegram-бота

Характеристика	БРС	Telegram-бот
Просмотр оценок	Да, официально и точно	Да

Продолжение таблицы 2

Характеристика	БРС	Telegram-бот
Ответы на вопросы	Да, но может быть долго	Да, мгновенно через ИИ
Удобство интерфейса	Часто неудобный, устаревший	Удобный Telegram-интерфейс
Доступ с телефона	Да, но вход может быть долгим и затрудненным	Да, приложение Telegram всегда по рукой
Дополнительные функции (вопросы, расписание)	Нет	Да

Таблица 3 – Сравнение сервиса Modeus и Telegram-бота

Характеристика	Modeus	Telegram-бот
Расписание занятий	Да, официально	Да
Интерфейс	Сложный, часто долго грузит	Простой (по кнопке «Расписание»)
Поиск аудитории/корпуса	Нет	Да (по кнопке «Навигация»)
Оповещение и удобство	Да	Да, напоминания
Ответы на вопросы	Нет	Да, через ИИ

Таблица 4 – Анализ косвенных конкурентов

Сервис/приложение	Недостатки	Решение Telegram-бота
БРС (сайт)	Долгий вход в систему, сбои и недоступность	Быстрый доступ к оценкам

Продолжение таблицы 4

Сервис/приложение	Недостатки	Решение Telegram-бота
Modeus (сайт)	Долгий вход в систему, сбои и большая нагрузка	Мгновенный доступ к расписанию
Приложение «УрФУ.Учеба»	Незаметность онлайн-курсов в основном интерфейсе, нерегулярное посещение приложения пользователями	Полное расписание, чёткое меню, удобство за счет мессенджера

После анализа конкурентов можно выделить ключевые преимущества нашего Telegram-бота, такие как объединение всех нужных сервисов в одном месте (вопросы, расписание), высокая скорость работы, интуитивно понятное меню с кнопками, быстрые и точные ответы благодаря встроенному ИИ-помощнику, а также возможность получить официальную поддержку через обращение к тьютору.

Таким образом, наш бот ориентирован на студентов, которым важно быстро находить нужную информацию, получать помощь и эффективно адаптироваться к учебной среде.

2 Разработка

Процесс разработки Telegram-бота — цифрового помощника для студентов — включает несколько ключевых этапов, направленных на создание функционального, надёжного и удобного в использовании продукта. Особое внимание уделяется реализации логики взаимодействия с пользователем, разработке интуитивного интерфейса с быстрым доступом к расписанию и оценкам. Важным этапом является тестирование работы бота в различных сценариях, чтобы обеспечить стабильность, актуальность ответов и соответствие ожиданиям целевой аудитории.

2.1 Сценарии использования

Таблица 5 – Запуск Telegram-бота

Действующее лицо	Студент – пользователь Telegram
Триггер	Пользователь запускает Telegram-бот для получения информации
Предусловие	Telegram установлен, бот найден и открыт пользователем
Результат	Открывается главное меню бота с кнопками функционала
Описание	1) пользователь открывает Telegram и запускает бота; 2) система приветствует пользователя и отображает главное меню;

Продолжение таблицы 5

	<p>3) главное меню включает кнопки: «Личный кабинет», «ИИ - помощник», «Навигация», «Конспекты», «Поддержка», «Помощь», «Язык»;</p> <p>4) пользователь может выбрать нужный раздел для продолжения работы.</p>
--	--

Таблица 6 – Создание конспекта по лекции

Действующее лицо	Студент – пользователь Telegram
Триггер	Пользователь хочет получить скомпонованный конспект лекции
Предусловие	Пользователь открыл главное меню и выбрал раздел «Конспекты»
Результат	Система обрабатывает файл, проводит транскрибацию и отправляет созданный конспект
Описание	<p>1) пользователь нажимает кнопку «Конспект лекции»;</p> <p>2) система направляет пользователя в окно загрузки файла;</p> <p>3) пользователь загружает нужный файл;</p>

Продолжение таблицы 6

	<p>4) после завершения транскрибации ИИ создает краткий текстовый конспект;</p> <p>5) пользователь получает готовый текст-конспект в сообщении от бота.</p>
--	---

Таблица 7 – Просмотр расписания

Действующее лицо	Студент – пользователь Telegram
Триггер	Пользователь хочет просмотреть расписание занятий
Предусловие	Пользователь открыл главное меню и выбрал раздел «Личный кабинет»
Результат	Пользователь переходит в свой личный кабинет, получая возможность проверить расписание
Описание	<p>1) пользователь нажимает кнопку «Личный кабинет»;</p> <p>2) пользователь успешно проходит аутентификацию для входа в личный кабинет;</p> <p>3) пользователь переходит в интересующие его разделы;</p> <p>4) пользователь проверяет расписание.</p>

Аналогично сценарию, представленному в таблице 7, пользователь может просмотреть карту аудиторий/корпусов и обратиться за помощью к тьютору, выбрав в Telegram-боте разделы «Навигация» и «Поддержка» соответственно.

2.2 Требования к продукту и MVP

2.2.1 Нефункциональные требования

В процессе работы над проектом удалось выделить ряд нефункциональных требований, которые важно учесть при разработке:

- 1) надежность – бот должен корректно работать при большом количестве пользователей;
- 2) скорость ответа – время отклика бота не более 2 секунд на стандартный запрос;
- 3) многоязычность – возможность выбрать язык;
- 4) удобство интерфейса – логичная навигация и минимум текста на экране;
- 5) масштабируемость – возможность в будущем добавлять новые модули.

2.2.2 Функциональные требования

На основе потребностей пользователей также были сформулированы ключевые функциональные возможности Telegram-бота:

- 1) выбор языка – возможность выбрать язык взаимодействия (русский/английский);
- 2) главное меню:
 - а) личный кабинет – просмотр расписания;
 - б) ИИ-помощник – возможность задать свободный вопрос и получить ответ;
 - в) навигация – просмотр карт корпусов;
 - г) поддержка – информация о способах связи с тьютором.

- 3) управление через команды и кнопки – структурированная работа через инлайн- и replay-кнопки;
- 4) переход в главное меню – возможность вернуться к стартовому выбору команды в любое время.

2.3 Технологический стек

В процессе разработки Telegram-бота для взаимодействия студентов с образовательной системой используется современный технологический стек, обеспечивающий стабильность, надёжность и масштабируемость проекта. Основу серверной части составляет язык программирования Java в связке с фреймворком Spring Boot, что позволяет организовать гибкую архитектуру и упростить реализацию бизнес-логики. Для хранения и обработки данных применяется реляционная база данных PostgreSQL, обеспечивающая быстрый доступ к информации о пользователях, расписании и других сущностях. Взаимодействие с пользователями осуществляется через Telegram с использованием Telegram Bot API, что позволяет реализовать удобный и привычный формат общения. Для проектирования интерфейсов и визуального представления админ-панели применяется Figma, обеспечивающая наглядность и согласование дизайна. Также в процессе разработки используется система контроля версий Git совместно с платформой GitHub, что обеспечивает удобную командную работу и отслеживание изменений. Такой технологический стек позволяет создать функциональное, удобное и легко расширяемое решение, соответствующее задачам проекта.

В дополнение к основным технологиям для создания админ-панели использовались HTML, CSS и JavaScript, что позволило разработать удобный и адаптивный интерфейс для управления ботом. Для хранения и обработки вопросов применялся формат JSON, обеспечивающий гибкость и простоту работы с данными. Кроме того, для интеграции с внешними сервисами

использовался токен Deepseek, что позволило реализовать расширенный функционал поиска и анализа информации.

2.4 Прототипирование

Цель команды — разработать прототип Telegram-бота, который обеспечит удобное и эффективное взаимодействие студентов с образовательной системой через функции, востребованные пользователями. Прототип должен включать возможность быстрого доступа к основным сервисам, интуитивно понятный интерфейс с навигацией по кнопкам и интеграцию с ИИ-помощником для получения быстрых ответов на вопросы. Основные элементы прототипа — это главное меню с разделами «Личный кабинет», «ИИ-помощник», «Навигация», «Конспекты», «Поддержка», «Помощь», «Язык», реализованные в виде удобных кнопок, а также обработка и отображение результатов взаимодействия с ботом в реальном времени. Важным аспектом является обеспечение надежной работы с аудиофайлами для создания конспектов и точной транскрибации лекций, а также локальная обработка данных для защиты конфиденциальности пользователей. После тестирования прототипа планируется собрать отзывы пользователей для улучшения интерфейса, повысить точность и скорость работы ИИ-помощника, расширить функционал за счет поддержки дополнительных форматов файлов и возможностей редактирования транскрибированного текста.

2.5 Проектирование и разработка системы

Главным элементом Telegram-бота является главное меню, через которое пользователь получает быстрый доступ к основным функциям бота — таким как личный кабинет, ИИ-помощник, навигация, создание конспектов и поддержка. При выборе раздела «Конспекты» пользователь может загрузить

аудио- или видеофайл для последующей транскрибации лекций с помощью встроенного ИИ. После завершения обработки бот выводит готовый текст конспекта с возможностью его копирования или сохранения, а также тайм-код исходного файла.

Для удобства предусмотрены дополнительные функции — навигация по корпусам через карты ВУЗа, а также быстрая связь с тьютором. Проектирование интерфейса и логики работы бота ориентировано на интуитивное управление с помощью кнопок, что облегчает освоение сервиса пользователями разного уровня подготовки. Важным аспектом разработки является обеспечение безопасности данных и своевременная обработка запросов для высокой скорости работы бота.

После тестирования прототипа планируется собрать обратную связь пользователей для улучшения навигации и расширения функционала, а также оптимизировать работу ИИ-помощника для повышения точности и скорости транскрибации.

3 Индивидуальные отчеты каждого из участников

В процессе работы над проектом каждый из участников команды выполнял конкретные обязанности, которые способствовали успешной реализации общих целей. Каждый участник вносил свой вклад в различные этапы работы, обеспечивая эффективное взаимодействие и выполнение задач. В данном разделе представлен индивидуальный отчет каждого участника, в котором отражены его основные роли и достижения в рамках проекта.

3.1 Метелева Дарья, тимлид-разработчик

В рамках выполнения проектной деятельности по созданию Telegram-бота для адаптации студентов была организована командная работа, обеспечивающая планомерную реализацию всех этапов проекта. Основной задачей стало координирование деятельности участников, постановка задач и контроль за соблюдением сроков выполнения. Поддерживалась постоянная связь с куратором для получения обратной связи и оперативного решения возникающих вопросов.

На старте проекта был сформирован и согласован календарный план, отражённый в диаграмме Ганта, включающей основные спринты, контрольные точки и этапы разработки. Для чёткого распределения ответственности между участниками использовалась доска задач, на которой регулярно обновлялись карточки с указанием ответственных и сроков выполнения (см. рисунки 1 и 2).

Была оказана помощь аналитику в составлении функциональных и нефункциональных требований, в анализе целевой аудитории и конкурентов. Также была оказана помощь в вёрстке сайта администрирования, поиске вопросов для базы знаний.

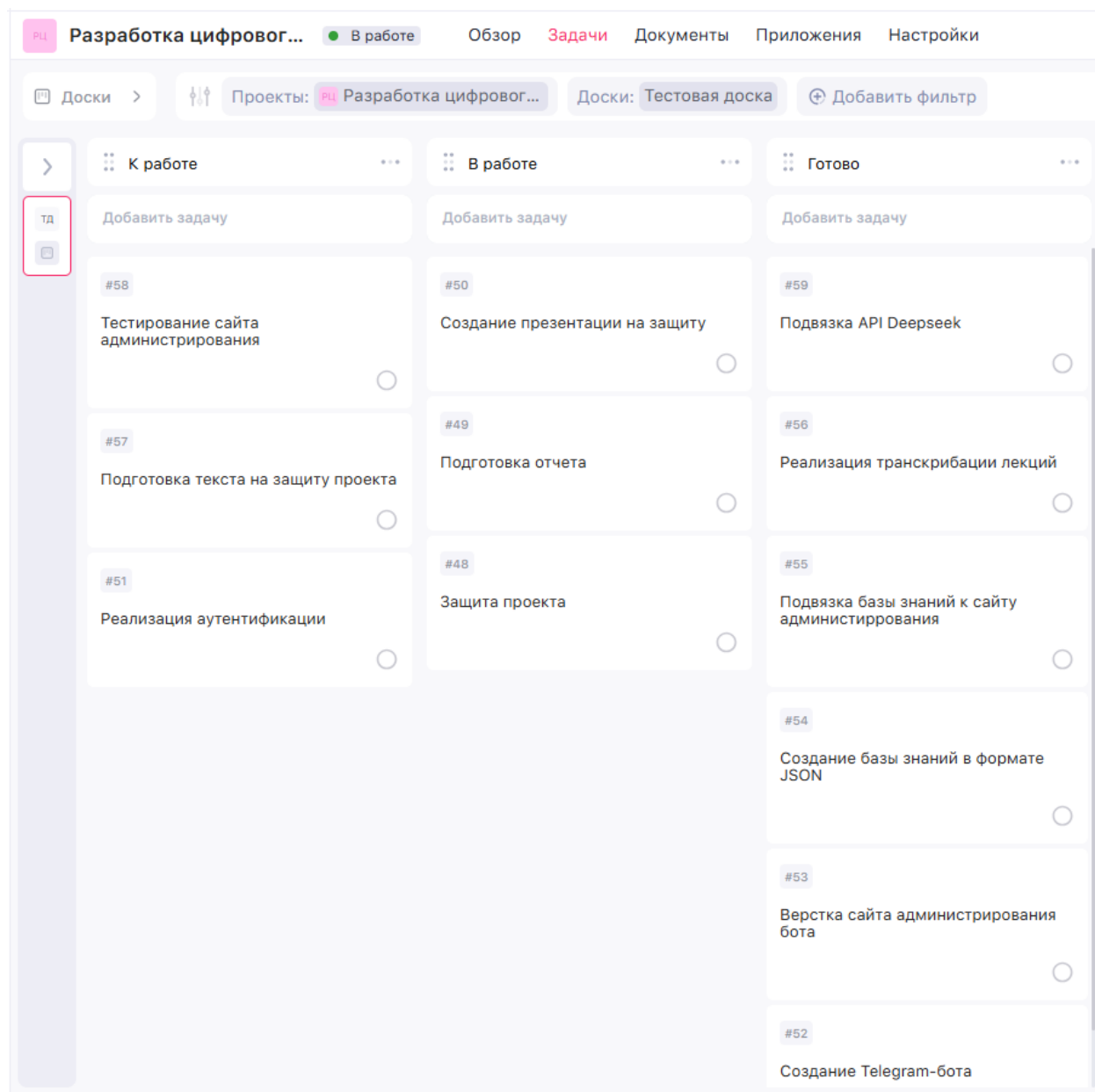


Рисунок 1 – Доска Week 1

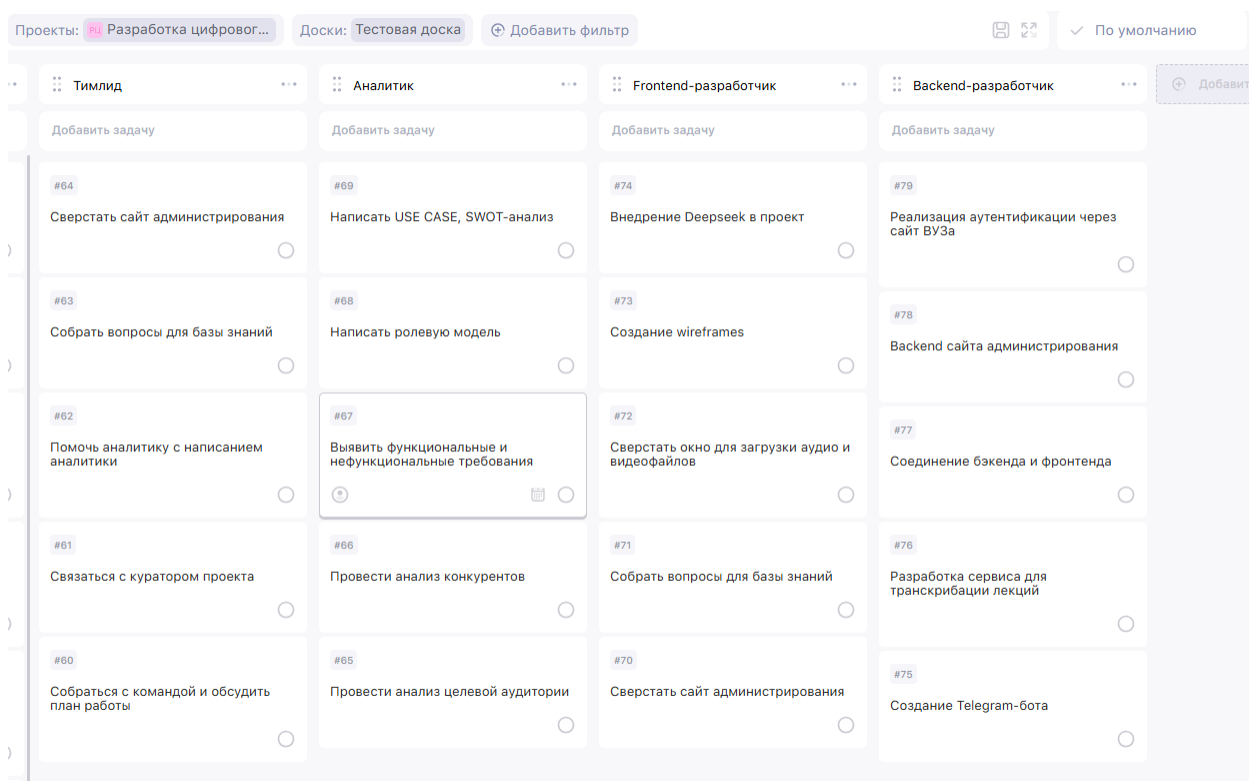


Рисунок 2 – Доска Week 2

Особое внимание уделялось командной коммуникации — регулярно проводились созвоны, обсуждения прогресса, а также оказание поддержки участникам при возникновении технических или организационных затруднений. Благодаря эффективному управлению команда смогла успешно пройти ключевые этапы реализации и выйти на уровень полноценного прототипа.

3.2 Федорова Алина, аналитик

В рамках выполнения работы над проектом был осуществлён комплекс аналитических работ, направленных на изучение целевой аудитории, рынка аналогов и формализацию функциональных требований к разрабатываемому Telegram-боту.

На начальном этапе реализации проекта был проведён анализ целевой аудитории с применением методики 5V Марка Шеррингтона. Полученные результаты позволили выявить приоритетный пользовательский сегмент, на который необходимо ориентироваться при проектировании системы.

Был выполнен анализ существующих решений и аналогов на рынке, обладающих схожей функциональностью. На основании сравнительного анализа определены основные недостатки и преимущества аналогичных сервисов, что позволило сформировать уникальные преимущества разрабатываемого продукта и выделить ключевые направления дальнейшей разработки. Результаты анализа целевой аудитории и конкурентов представлены в соответствующем разделе отчёта.

Также были представлены ролевая модель, флоу взаимодействия с ботом и SWOT-анализ.

Для формализации функциональных требований были разработаны USE CASE (сценарии использования), включающие в себя описание действующего лица, триггера, предусловия, результата и пошагового описания процессов. Эти сценарии представлены в соответствующем разделе отчёта и использовались в качестве основы для проектирования архитектуры системы и взаимодействия между модулями.

3.3 Суркина Мария, frontend-разработчик

В рамках работы над проектом были выполнены ключевые задачи, связанные с разработкой пользовательского интерфейса, интеграцией внешних инструментов и подготовкой контента для базы знаний Telegram-бота.

Для визуального проектирования структуры и логики взаимодействия с ботом были разработаны wireframes в графическом редакторе Figma (см.

рисунок 3). Макеты позволили определить расположение элементов интерфейса, спланировать пользовательский сценарий и согласовать концепцию с остальными членами команды.

В целях расширения функциональности Telegram-бота была осуществлена интеграция языковой модели DeepSeek, что позволило реализовать интеллектуальную генерацию ответов на вопросы студентов и создание конспектов на основе транскрибированных лекций. Данная интеграция обеспечила обработку сложных пользовательских запросов и повысила адаптивность бота к различным формулировкам.

С целью обеспечения удобного управления содержанием базы знаний был разработан и сверстан веб-интерфейс для администрирования (см. рисунок 4). Интерфейс включает базовые функции добавления и удаления вопросов, а также предоставление текстовых ответов. При создании сайта использовались HTML, CSS и JavaScript. Структура данных реализована с применением формата JSON для хранения и обработки информационных блоков, используемых ботом.

Кроме того, была проведена работа по поиску и отбору вопросов, актуальных для первокурсников, с целью наполнения базы знаний Telegram-бота.

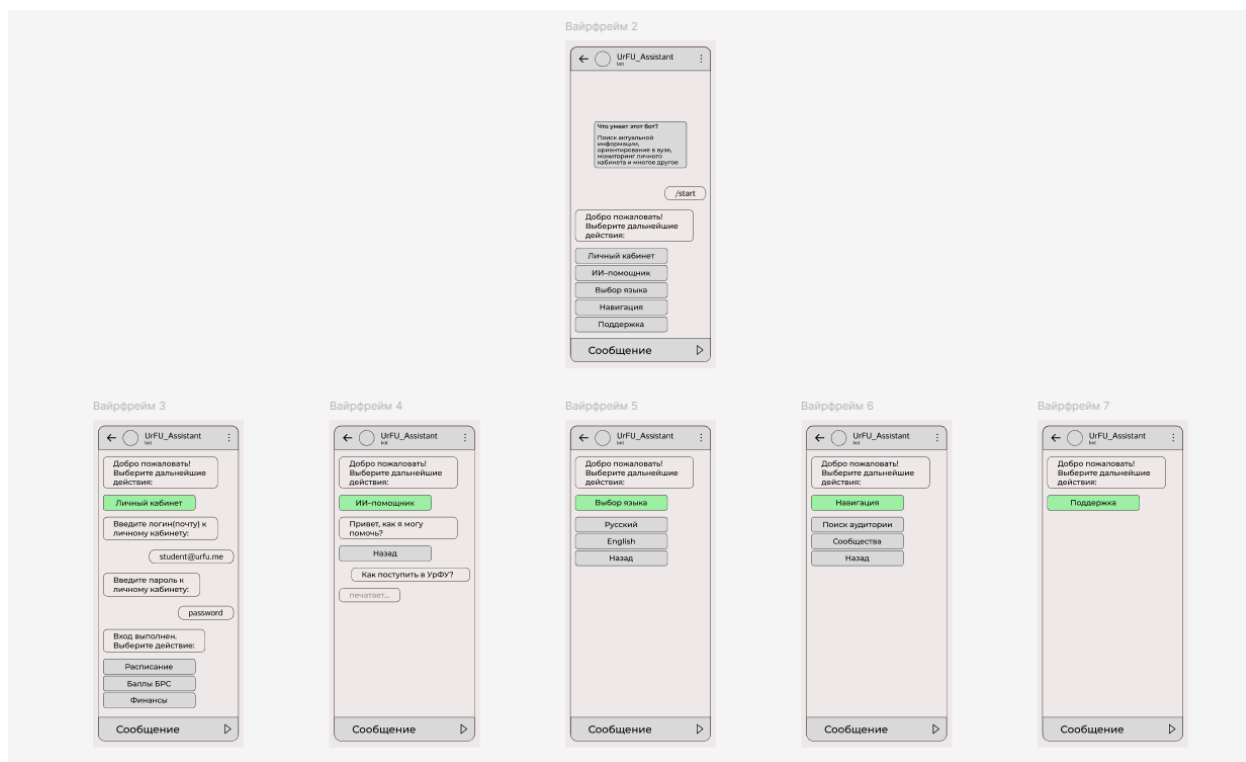


Рисунок 3 – Создание wireframes

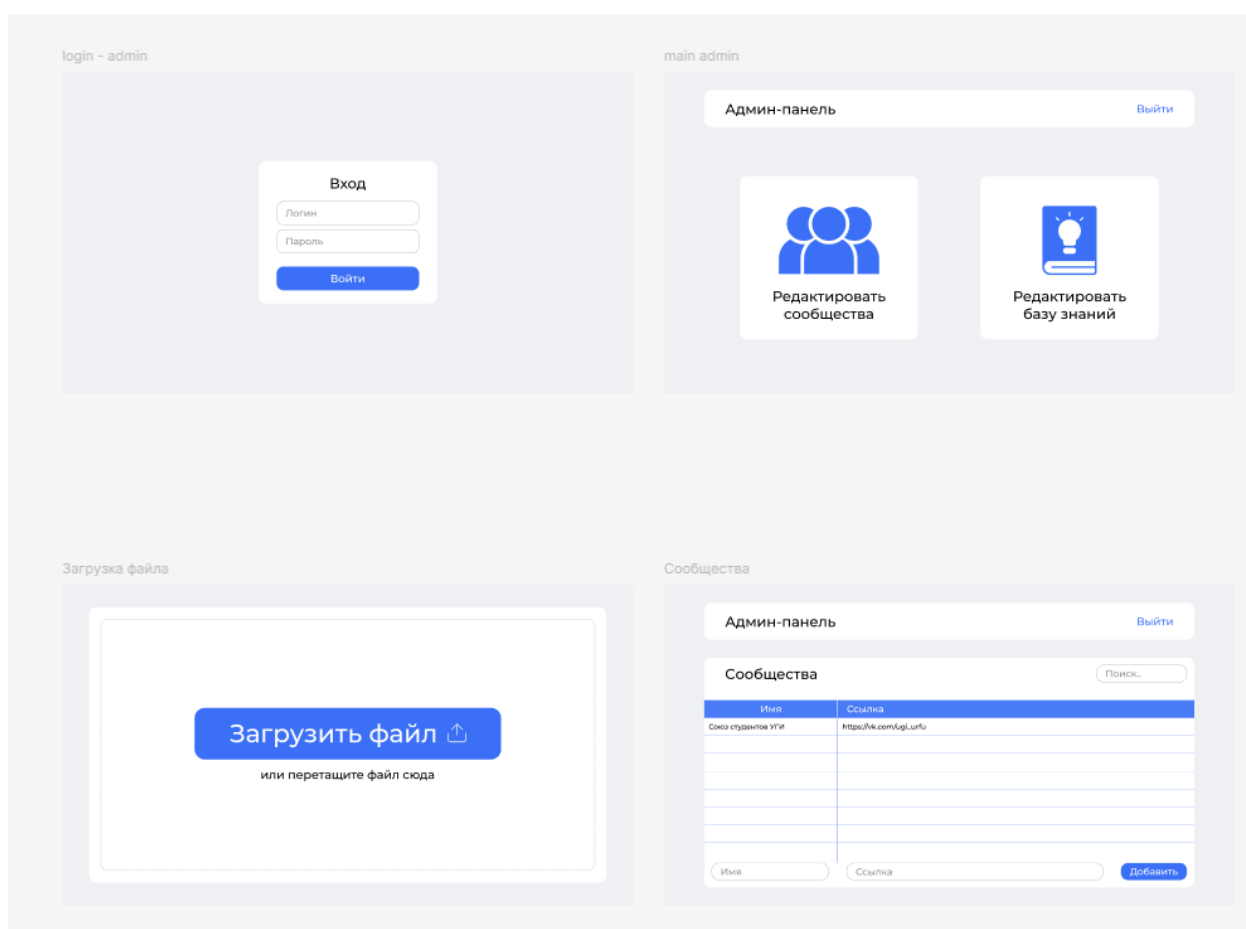


Рисунок 4 – Дизайн сайта администрирования и окна загрузки файлов

3.4 Троянов Дмитрий, backend-разработчик

В ходе работы над проектом были выполнены основные задачи по созданию и развитию серверной части системы, обеспечивающей функциональность Telegram-бота и веб-сервиса для транскрибации лекций.

Был разработан Telegram-бот, включающий основные возможности взаимодействия с пользователями, такие как прием запросов, обработка данных и выдача ответов. Особое внимание уделялось обеспечению стабильной работы и корректной интеграции с внешними сервисами.

Для реализации функционала транскрибации лекций был создан отдельный веб-сервис, обеспечивающий загрузку аудио- и видеофайлов, их обработку с помощью библиотеки распознавания речи и преобразование в текстовый формат. Сервис был спроектирован с учетом необходимости точной и быстрой обработки.

Кроме того, была реализована серверная часть административного сайта, предоставляющая инструменты для управления контентом. В рамках этой работы обеспечивалась логика взаимодействия с базой знаний и поддержка API для фронтенд-части.

Для обеспечения полноценной работы продукта была выполнена интеграция между backend и frontend компонентами, что позволило реализовать слаженную и эффективную работу всех модулей системы.

ЗАКЛЮЧЕНИЕ

В результате реализации проекта был создан программный продукт — Telegram-бот с функцией транскрибации и веб-интерфейсом для администрирования. Разработка соответствует требованиям пользователей и заказчика: реализована удобная загрузка файлов, выбор параметров обработки, получение текста, а также управление ИИ-помощником через сайт.

Все модули Telegram-бота функционируют корректно, взаимодействие между компонентами реализовано надёжно. Сервис транскрибации стабильно обрабатывает аудио и возвращает текст с удовлетворительной точностью. На основе проведённого тестирования серьёзных сбоев в работе системы не выявлено.

В дальнейшем возможна реализация дополнительных модулей, расширение функциональности транскрибирования — таких как автоматическая пунктуация, поддержка новых языков и диалектов, а также интеграция с внешними источниками аудиоданных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Садов И. В., Смирнов П. С. Современные подходы к обработке аудиоданных. – М. : Техносфера, 2019. – 256 с.
2. Бауэр Р. Обработка сигналов: основы и приложения / пер. с англ. – 2-е изд. – М. : Мир, 2018. – 312 с.
3. Гаврилов А. И. Технологии машинного обучения в обработке речи. – СПб. : Питер, 2020. – 384 с.
4. Чернышев В. В. Алгоритмы распознавания речи: учебное пособие. – Новосибирск : НГУ, 2017. – 180 с.
5. Коровин Е. А., Петров Д. Н. Введение в нейронные сети и глубокое обучение. – Екатеринбург : УГТУ, 2021. – 298 с.
6. Вайсберг М. Л. Автоматическая обработка естественного языка. – М. : Наука, 2015. – 428 с.
7. TensorFlow: официальный сайт [Электронный ресурс]. – Режим доступа: <https://www.tensorflow.org> (дата обращения: 21.03.2025).
8. Kaldi Speech Recognition Toolkit [Электронный ресурс]. – Режим доступа: <https://kaldi-asr.org> (дата обращения: 26.03.2025).
9. Исследование технологий ASR (Automatic Speech Recognition): науч. обзор / под ред. И. П. Соколова. – СПб. : СПбГУ, 2023. – Режим доступа: <https://asr-research.spbu.ru> (дата обращения: 05.04.2025).
10. DeepSeek: официальный сайт [Электронный ресурс]. – Режим доступа: <https://deepseek.ai> (дата обращения: 19.03.2025).
11. DeepSeek: интеграция и возможности [Электронный ресурс] / Tech Insights. – Режим доступа: <https://techinsights.com/deepseek-integration> (дата обращения: 12.05.2025).
12. DeepSeek в обработке речи: обзор технологий и решений / А. Козлов. – М. : Технопресс, 2024. – 154 с.
13. Telegram Bot API [Электронный ресурс]. – Режим доступа: <https://core.telegram.org/bots/api> (дата обращения: 18.04.2025).

14. Павлов А. Ю. Telegram-боты на Python: создание, настройка и развёртывание. – М. : ДМК Пресс, 2022. – 192 с.
15. Иванов С. М. Безопасность Telegram-ботов: защита данных и взаимодействие с API. – М. : Инфра-М, 2023. – 176 с.
16. Морозов П. В. Интеграция Telegram-ботов в цифровые экосистемы: практическое руководство. – СПб. : БХВ-Петербург, 2022. – 204 с.
17. Telegram для разработчиков [Электронный ресурс] // Хабр. – Режим доступа: <https://habr.com/ru/company/ruvds/blog/460847> (дата обращения: 22.05.2025).
18. Витушинский А. С. Локальные веб-сервисы: разработка и эксплуатация. – Томск : Томский политехнический университет, 2022. – 240 с.
19. Интеграция DeepSeek с Telegram-ботами: практические кейсы / Е. Смирнова // Dev Journal. – 2025. – №4. – С. 32–38.
20. Обзор современных сервисов транскрибации и их применение в чат-ботах / В. Лебедев. – М. : Информационные технологии, 2024. – 198 с.

ПРИЛОЖЕНИЕ А

Wireframes и макеты из Figma



Рисунок А.1 – Wireframe 1

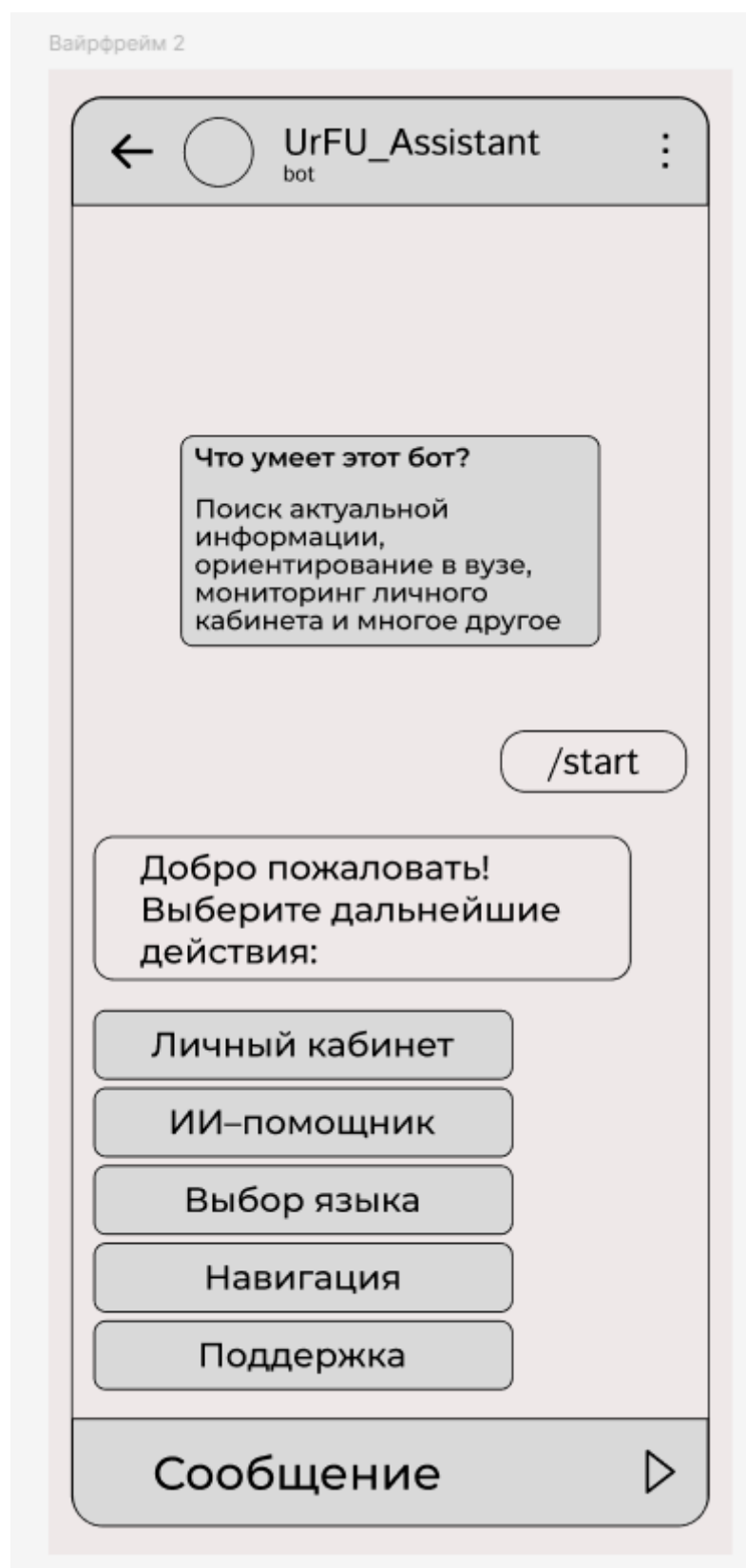


Рисунок А.2 – Wireframe 2

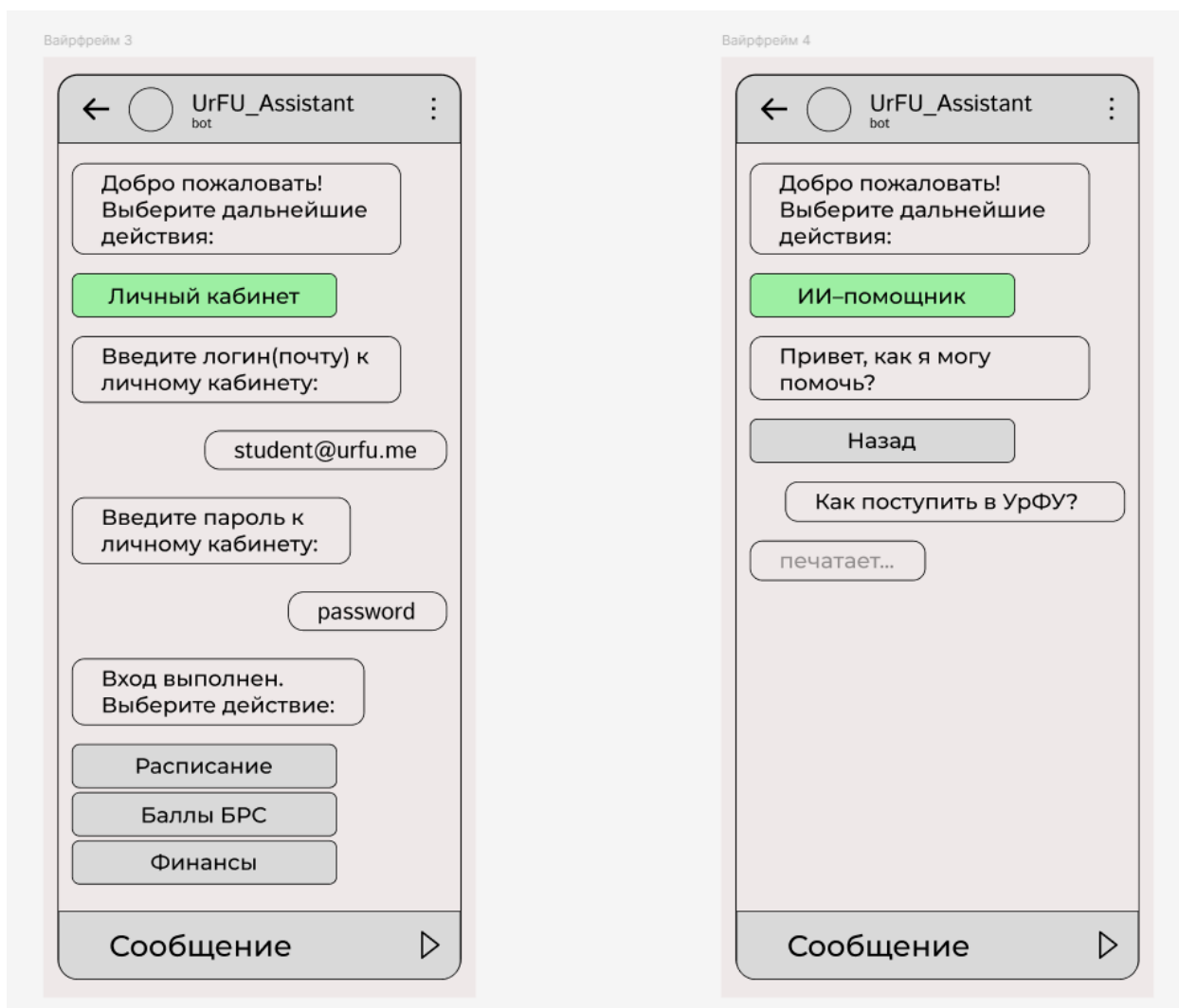


Рисунок А.3 – Wireframe 3 и Wireframe 4



Рисунок А.4 – Wireframe 5, Wireframe 6, Wireframe 7

ПРИЛОЖЕНИЕ В

Календарный план

Календарный план представлен в отдельном документе. Ознакомиться с ним можно по следующей ссылке:

<https://docs.google.com/document/d/1MrhRYpX1JFlpTD9mlsyavDfGQpgmfEJo/edit> (дата обращения: 26.05.2025).