МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

(ФГБОУ ВО «ВГУ»)

Факультет Компьютерных наук

Кафедра программирования и информационных технологий

Техническое задание

на разработку мобильного приложения

«AI Job Interview - ProTalk»

Исполнители

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.Н. Петров

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ У. С. Лихих

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ К.А. Бессонов

Заказчик

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.С. Тарасов

Воронеж 2025

СОДЕРЖАНИЕ

[Термины 5](#_Toc198587507)

[1. Общие сведения 8](#_Toc198587508)

[1.1 Полное наименование системы и название приложения 8](#_Toc198587509)

[1.2 Заказчик приложения 8](#_Toc198587510)

[1.3 Исполнитель 8](#_Toc198587511)

[1.4 Перечень документов, на основании которых создается система 8](#_Toc198587512)

[1.5 Плановые сроки начала и окончания работы по созданию системы 8](#_Toc198587513)

[1.6 Источники финансирования 9](#_Toc198587514)

[2. Назначение и цели создания 10](#_Toc198587515)

[2.1 Назначение проекта 10](#_Toc198587516)

[2.2 Цели проекта 10](#_Toc198587517)

[2.3 Требования к видам обеспечения 10](#_Toc198587518)

[2.4 Задачи проекта 10](#_Toc198587519)

[2.5 Предметная область 11](#_Toc198587520)

[3. Требования к приложению и программному обеспечению 12](#_Toc198587521)

[3.1 Требования к программному обеспечению приложения 12](#_Toc198587522)

[3.2 Общие требования к оформлению и верстке экранов 12](#_Toc198587523)

[4. Структура приложения 14](#_Toc198587524)

[4.1 Динамические страницы 14](#_Toc198587525)

[5. Языковые версии приложения 15](#_Toc198587526)

[6. Группы пользователей и их возможности 16](#_Toc198587527)

[6.1 Роли пользователей 16](#_Toc198587528)

[6.2 Функциональные возможности 16](#_Toc198587529)

[6.2.1. Неавторизованный пользователь 16](#_Toc198587530)

[6.2.2. Авторизированный пользователь 16](#_Toc198587531)

[6.3 Логика работы интерфейса 16](#_Toc198587532)

[6.3.1. Загрузка данных 17](#_Toc198587533)

[7. Общая логика работы страниц 18](#_Toc198587535)

[7.1 Загрузка данных 18](#_Toc198587536)

[7.2 Обновление страниц 18](#_Toc198587537)

[7.3 Работа кнопок 18](#_Toc198587538)

[7.4 Ошибки 18](#_Toc198587539)

[7.5 Ошибки 4xx 20](#_Toc198587540)

[7.6 Ошибка 404 20](#_Toc198587541)

[7.7 Ошибка 401 (неверный логин или пароль) 20](#_Toc198587542)

[7.8 Ошибка 401 (истёк срок действия токена) 20](#_Toc198587543)

[7.9 Ошибка 400 (пользователь с такой почтой уже существует) 20](#_Toc198587544)

[7.10 Ошибка 400 (на сервер пришли невалидные данные) 20](#_Toc198587545)

[7.11 Ошибка 403 (доступ запрещен): 21](#_Toc198587546)

[7.12 Ошибки 5xx 21](#_Toc198587547)

[7.13 Ошибка 504 (если ответ не приходит в течение 45-60 секунд) 21](#_Toc198587548)

[7.14 Нет подключения к интернету 21](#_Toc198587549)

[7.15 Пустое состояние 21](#_Toc198587550)

[8. Описание экранов приложения 22](#_Toc198587551)

[8.1 Загрузочный экран 22](#_Toc198587552)

[8.2 Приветственный экран 22](#_Toc198587553)

[8.3 Регистрация 23](#_Toc198587554)

[8.4 Авторизация 25](#_Toc198587555)

[8.5 Страница тестов (разделы) 26](#_Toc198587556)

[8.6 Страница тестов (темы) 27](#_Toc198587557)

[8.7 Страница тестов (вопросы) 28](#_Toc198587558)

[8.8 Страница вопроса 29](#_Toc198587559)

[8.9 Страница базы знаний (разделы) 30](#_Toc198587560)

[8.10 Страница базы знаний (темы) 31](#_Toc198587561)

[8.11 Страница базы знаний (статьи) 32](#_Toc198587562)

[8.12 Страница статьи 33](#_Toc198587563)

[8.13 Экран разделов собеседования 34](#_Toc198587564)

[8.14 Экран вопроса собеседования 35](#_Toc198587565)

[8.15 Экран результатов собеседования 36](#_Toc198587566)

[8.16 Экран результатов собеседования 37](#_Toc198587567)

[8.17 Экран профиля пользователя 38](#_Toc198587568)

[8.18 Экран профиля пользователя 39](#_Toc198587569)

[9. Дизайн приложения 41](#_Toc198587570)

[10. Общие требования к системе 42](#_Toc198587571)

[10.1 Функциональные требования 42](#_Toc198587572)

[10.2 Нефункциональные требования 42](#_Toc198587573)

[11. Порядок контроля и приемки работ 43](#_Toc198587574)

[12. Реквизиты и подписи сторон 44](#_Toc198587575)

[ПРИЛОЖЕНИЕ 1 45](#_Toc198587576)

Термины

В этом разделе приведены основные термины и определения, используемые в техническом задании для того, чтобы обеспечить единое понимание их значений. Информация о них приведена в таблице 1.

**Таблица 1 – Термины, используемые в техническом задании**

| **Термин** | **Значение** |
| --- | --- |
| API (Application Programming Interface) | Интерфейс программирования приложений, позволяющий взаимодействовать между клиентской и серверной частями системы. |
| Backend | Серверная часть приложения, отвечающая за логику, обработку данных и взаимодействие с базой данных. Пользователь не видит эту часть системы. |
| CI/CD(Continuous Integration/Continuous Deployment) | Практика автоматизации процессов интеграции кода и его развертывания. |
| FastAPI | Фреймворк для написания RESTful API с автоматической генерацией документации (Swagger/OpenAPI) и поддержкой асинхронности. |
| Figma | Инструмент для проектирования пользовательских интерфейсов. |
| Frontend | Клиентская сторона, включающая пользовательский интерфейс (UI) и визуальное представление данных. |
| GitHub | Платформа для хостинга IT-проектов, их совместной разработки и контроля версий. |
| Miro | Онлайн-платформа для совместной работы распределённых команд, интерактивная онлайн-доска. |
| ORM (Object-Relational Mapping) | Технология, позволяющая взаимодействовать с базой данных через объекты программирования, минуя SQL-запросы. |
| PostgreSQL | Реляционная СУБД с открытым исходным кодом, поддерживающая сложные запросы и транзакции. |
| Python | Язык программирования, широко применяемый в веб-разработке, анализе данных и машинном обучении. |
| REST API (REST) | Стандарт взаимодействия между клиентом и сервером через HTTP-запросы, основанный на принципах REST (Representational State Transfer). |
| SQLAlchemy | Инструмент для работы с БД в Python, который предоставляет ORM. |
| Swagger | Инструмент для документирования и тестирования API. |
| UI Kit Material UI | Набор готовых компонентов и стилей для React, основанный на дизайн-системе Google Material Design. |
| UML (Unified Modeling Language) | Унифицированный язык моделирования, используемый для визуализации архитектуры системы. |
| YouGile | Система управления проектами и задачами. |
| Клиент (клиентская сторона) | Приложение или веб-страница, через которую пользователь взаимодействует с системой. |
| Сервер (серверная часть) | Компьютер или программа, обрабатывающая запросы от клиентов, выполняющая вычисления и предоставляющая доступ к данным. |
| Фреймворк | Программная платформа, задающая архитектуру приложения и упрощающая разработку за счет готовых решений для объединения компонентов проекта. |

1. Общие сведения
   1. Полное наименование системы и название приложения

Полное наименование: «Разработка мобильного приложения AI Job Interview - ProTalk».

Название приложения: «AI Job Interview - ProTalk».

* 1. Заказчик приложения

Старший преподаватель Тарасов Вячеслав Сергеевич, Воронежский Государственный Университет, Факультет компьютерных наук, кафедра Программирования и Информационных Технологий.

* 1. Исполнитель

Разработчик: «13» команда группы «5».

Состав команды разработчиков:

* Петров Артем Николаевич;
* Бессонов Кирилл Алексеевич;
* Лихих Ульяна Сергеевна.
  1. Перечень документов, на основании которых создается система

Данное приложение будет создаваться на основании следующих документов:

* + - Учебный план по программе бакалавриата «Информационные системы и технологии 09.03.02»;
    - Распоряжение от 09.01.2025 №1600-62 «О сроках текущей и промежуточной аттестации».
  1. Плановые сроки начала и окончания работы по созданию системы

Начало работы по созданию системы: февраль 2025 года.

Окончание работы по созданию системы: июнь 2025 года.

* 1. Источники финансирования

Не предусмотрены.

1. Назначение и цели создания
   1. Назначение проекта

Приложение предназначено для подготовки пользователей к собеседованиям через персонализированные симуляции с учетом их специализации и уровня навыков. Оно должно помогать отрабатывать технические и поведенческие вопросы в реалистичных условиях, используя AI для анализа ответов и обратной связи. Система должна автоматически адаптировать обучение на основе статистики успехов и ошибок, предлагая индивидуальные рекомендации для улучшения результатов.

* 1. Цели проекта

Сервис будет предназначен для:

* + - Создания эффективного инструмента подготовки к собеседованиям с использованием AI;
    - Повышения конкурентоспособности пользователей за счёт персонализированных тренировок;
    - Снижения стресса перед реальными собеседованиями;
    - Оптимизации времени подготовки благодаря адаптивному обучению.
  1. Требования к видам обеспечения
     + Программное обеспечение: Flutter 3.7+, Python 3.10+, PostgreSQL 13;
     + Техническое: сервер с 8 ГБ ОЗУ, 4 ядра CPU;
     + Лингвистическое: поддержка русского языка.
  2. Задачи проекта

Основные задачи проекта:

* + - Персонализация обучения;
    - Автоматизированная подготовка к собеседованиям;
    - Отработка технических навыков;
    - Тренировка поведенческих компетенций;
    - Анализ прогресса;
    - Генерация рекомендаций.
  1. Предметная область

Приложение охватывает цифровую платформу для подготовки к профессиональным собеседованиям через AI-симуляцию технических, поведенческих и кейсовых интервью. Система анализирует ответы пользователей, предоставляет персонализированные рекомендации и отслеживает их прогресс. Поддерживает основные языки программирования и предлагает премиум-контент по подписке.

1. Требования к приложению и программному обеспечению
   * + Приложение должно иметь архитектуру, соответствующую шаблону Клиент-Серверного приложения и иметь разделение на back-end и front-end, связанных между собой REST API.
     + Система должна иметь возможность регистрации для новых пользователей;
     + Система должна хранить данные пользователя для автоматического входа в аккаунт;
     + Система должна сохранять результаты пользователя в истории;
     + Система должна предоставлять пользователю анализ результатов.
   1. Требования к программному обеспечению приложения

Экраны должны корректно отображаться на мобильных устройствах с операционными системами:

* + - iOS;
    - Android.

Для реализации программной части были выбраны следующие технологии:

* + - Язык программирования Python;
    - Фреймворк: FastAPI;
    - СУБД: PostgreSQL;
    - OpenAI API.

Для реализации клиентской части были выбраны следующие технологии:

* + - Фреймворк: Flutter;
    - Язык программирования Dart.
  1. Общие требования к оформлению и верстке экранов

Оформление экранов приложения должно соответствовать единой стилистике и выбранной цветовой палитре, обеспечивая удобочитаемость интерфейса.

При верстке экранов используется язык программирования Dart.

1. Структура приложения
   1. Динамические страницы
      * Экран авторизации;
      * Экран регистрации;
      * Главный экран.
2. Языковые версии приложения

Все страницы приложения должны быть реализованы с поддержкой русской языковой версии.

1. Группы пользователей и их возможности
   1. Роли пользователей

Система предусматривает две основные роли:

Неавторизованный пользователь;

Авторизованный пользователь.

* 1. Функциональные возможности
     1. Неавторизованный пользователь

Может выполнять следующие действия:

1. Проходить процедуру регистрации:

* ввод email;
* создание пароля (от 6 символов);
* подтверждение email.

1. Авторизоваться в системе.
   * 1. Авторизированный пользователь

Имеет доступ к следующим функциям:

1. Управление профилем:

* редактирование персональных данных;
* указание целевой должности;
* выбор уровня подготовки (Junior/Middle/Senior).

1. Работа с системой:

* прохождение тестовых собеседований;
* тренировка ответов на вопросы.

1. Аналитика:

* просмотр истории результатов;
* получение персональных рекомендаций.
  1. Логика работы интерфейса
     1. Загрузка данных

Индикация состояния:

* при действиях: блокировка кнопок с отображением индикатора загрузки «loader»;
* при загрузке: отображение shimmer-эффектов.

1. Общая логика работы страниц
   1. Загрузка данных

Происходит следующим образом:

* при отправке запроса на получение данных на экране отображаются shimmers (шиммеры) в местах загрузки элементов;
* если на экране присутствует кнопка, инициирующая запрос, при нажатии она становится неактивна на время отображения loader (лоадера) в ней.
  1. Обновление страниц

Происходит следующим образом:

* используется механизм Pull-to-Refresh (PTR) — при свайпе вниз происходит повторный запрос данных;
* в момент обновления на экране отображаются шиммеры для элементов;
* при обновлении страницы, если в текстовых полях была введена информация, она сохраняется;
* после успешного получения ответа шиммеры исчезают, и загруженные данные отображаются на экране.
  1. Работа кнопок

Описание:

* при нажатии на кнопку изменяется ее состояние нажатия (pressed state);
* при нажатии на кнопку “<” (стрелка) происходит переход на предыдущий экран;
* при нажатии на кнопку “X” (крестик) экран закрывается, и пользователь остается на предыдущем экране.
  1. Ошибки

Ошибки обрабатываются следующим образом:

* ошибка при загрузке данных (GET-запросы): в случае неудачного запроса отображается экран ошибки с соответствующим сообщением, если в ответ на запрос получен код 504 (таймаут), отображается экран таймаута;
* ошибка при отправке данных (POST, DELETE, PUT-запросы): ошибка отправки данных отображается в виде красного снека с текстом ошибки. Снек закрывается автоматически через 6 секунд или вручную – по тапу на крестик или свайпу вверх. Если в каком-либо поле введены некорректные данные, это поле подсвечивается красным цветом с соответствующим сообщением;
* ошибка соединения: при потере Интернет-соединения отображается экран с соответствующим сообщением;
* пустые массивы: если в ответ на запрос получен пустой массив, отображается экран пустого состояния;
* пустые поля: если поле является обязательным и остается пустым, оно подсвечивается красным, а под ним появляется текст с сообщением, что оно не может быть пустым “Поле не может быть пустым”;
* если в поле введены невалидные символы, оно также подсвечивается красным, а под ним отображается соответствующий текст о некорректном вводе "Введены некорректные символы”;
* если количество введенных символов меньше или больше допустимого значения, поле подсвечивается красным, а под ним появляется сообщение "Длина ввода должна быть от X до Y символов", в случае если задана одна из границ, показываем сообщение с данной границей “Длина ввода должна быть от Х символов” или “Длина ввода должна быть до У символов”.
* в последующих маркированных списках данного подпункта представлены тексты снэков.
  1. Ошибки 4xx

Описание:

* заголовок: «Ошибка запроса»;
* подзаголовок: «Перезагрузите страницу».
  1. Ошибка 404

Описание:

* заголовок: «Ничего не найдено»;
* подзаголовок: «Запрашиваемый ресурс недоступен или не существует».
  1. Ошибка 401 (неверный логин или пароль)

Описание:

* заголовок: «Неверный логин или пароль»;
* подзаголовок: «Попробуйте ещё раз».
  1. Ошибка 401 (истёк срок действия токена)

Описание:

* заголовок: «Сессия завершена»;
* подзаголовок: «Войдите снова, чтобы продолжить».
  1. Ошибка 400 (пользователь с такой почтой уже существует)

Описание:

* заголовок: «Пользователь уже существует»;
* подзаголовок: «Используйте другой адрес электронной почты или восстановите доступ».
  1. Ошибка 400 (на сервер пришли невалидные данные)

Описание:

* заголовок: «Некорректные данные»;
* подзаголовок: «Проверьте введённую информацию и попробуйте снова».
  1. Ошибка 403 (доступ запрещен):

Описание:

* заголовок: «Недостаточно прав»;
* подзаголовок: «У вас нет доступа к этому разделу».
  1. Ошибки 5xx

Описание:

* заголовок: «Ошибка сервера»;
* подзаголовок: «Что-то пошло не так. Попробуйте позже».
  1. Ошибка 504 (если ответ не приходит в течение 45-60 секунд)

Описание:

* заголовок: «Сервер не отвечает»;
* подзаголовок: «Запрос занял слишком много времени. Попробуйте позже».
  1. Нет подключения к интернету

Описание:

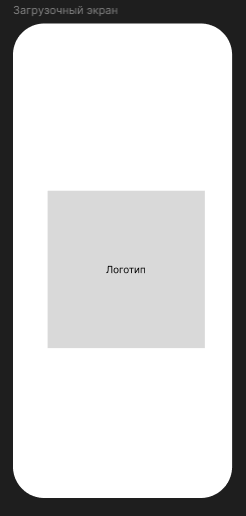
* заголовок: «Нет подключения к сети»;
* подзаголовок: «Проверьте соединение и попробуйте снова».
  1. Пустое состояние

Описание:

* заголовок: «Пока здесь пусто»;
* подзаголовок: «Возможно, это временно — загляните позже!».

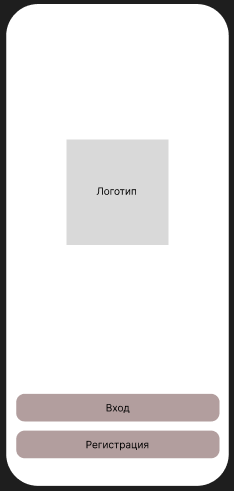
8. Описание экранов приложения
   1. Загрузочный экран

Страница доступна всем пользователям. В центре экрана находится логотип приложения «ProTalk».



1. Загрузочный экран
   1. Приветственный экран

Страница доступна всем пользователям. Помимо приветственного сообщения и логотипа приложения, на экране расположены две кнопки: для авторизации и регистрации, которые ведут на соответствующие страницы. На странице работает стандартная логика обработки ошибок сервера.



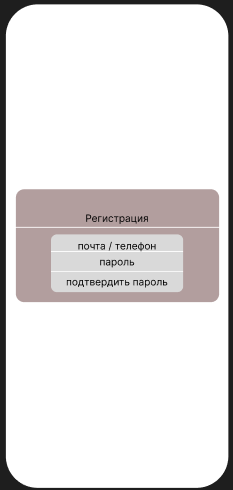
1. Приветственный экран
   1. Регистрация

Страница доступна неавторизованным пользователям. Помимо названия приложения, на экране расположены поля для ввода информации о будущем пользователе, а также ссылка на политику конфиденциальности и пользовательское соглашение, при нажатии на которые открываются страницы для ознакомления с данными документами. При закрытии данной страницы пользователь может продолжить изучение приложения в качестве гостя, без необходимости в авторизации.

Ограничения для полей ввода:

* Электронная почта: нет ограничения на количество вводимых символов, используется регулярное выражение для почты ([a-zA-Z0-9.\_-]+@[a-zA-Z0-9.\_-]+\.[a-zA-Z0-9\_-]+ ) ;
* Логин: допускаются от 2 до 20 вводимых символов, буквы любого регистра на любом языке, знак тире, исключаются цифры, специальные символы, эмоджи;
* Пароль: допускаются от 6 вводимых символов, исключаются только эмоджи;
* Незаполненные поля: если хотя бы одно поле не заполнено, кнопка "зарегистрироваться" является некликабельной;
* Недопустимые символы: символы отображаются на экране при вводе и вставке, однако при попытке сохранить данные, срабатывает стандартная логика обработки ошибок текстовых полей;
* Некорректная почта: если пользователь ввёл электронную почту не в соответствии с регулярным выражением, поле её ввода подсвечивается красным цветом, чуть ниже появляется снек “Некорректный формат почты”.

При корректно введённых данных, кнопка "зарегистрироваться" становится доступной, при нажатии на неё загружается домашняя страница (Страница тестов).



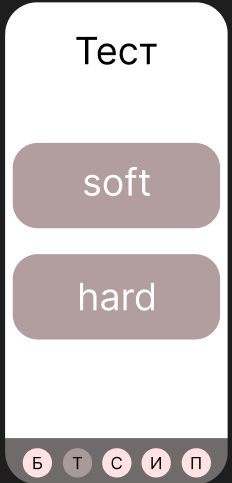
1. Макет регистрации
   1. Авторизация

Страница доступна неавторизованным пользователям. Помимо пояснительного сообщения и названия приложения, на экране расположены поля для ввода информации о пользователе, кнопки “войти”. Ограничения для полей ввода идентичны ограничениям в предыдущем подпункте, кнопка “войти” становится кликабельной только в случае корректно введённых данных, при нажатии на неё загружается страница профиля пользователя.



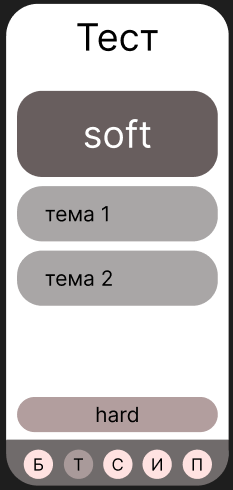
1. Страница авторизации
   1. Страница тестов (разделы)

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю доступен выбор между двумя разделами подготовки – Soft-навыки и Hard-навыки. При нажатии на кнопку раздела он разворачивается и пользователю становится доступен перечень тем (описывается в пункте 7.4.6). В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – тестирование, выделяется цветом.



1. Страница тестирования (разделы)
   1. Страница тестов (темы)

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю доступен выбор между темами подготовки. При нажатии на кнопку темы он разворачивается и пользователю становится доступен перечень вопросов (описывается в пункте 7.4.7). В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – тестирование, выделяется цветом.



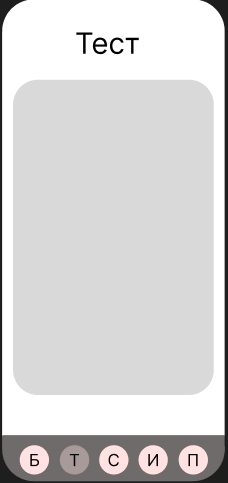
1. Страница тестирования (темы)
   1. Страница тестов (вопросы)

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю доступен выбор между вопросами для подготовки. При нажатии на кнопку вопроса, открывается страница вопроса (описывается в пункте 7.4.8). В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – тестирование, выделяется цветом.



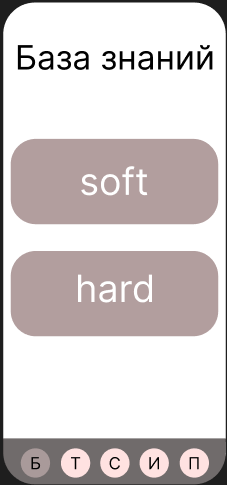
1. Страница тестирования (вопросы)
   1. Страница вопроса

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователь выбирает ответ на вопрос или вводит собственный, после чего получает результат ответа. В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – тестирование, выделяется цветом.



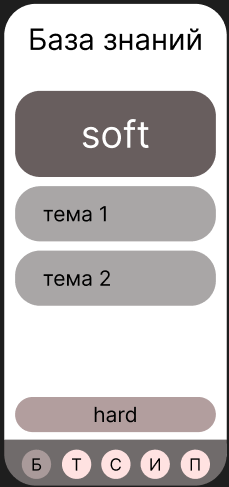
1. Страница вопроса
   1. Страница базы знаний (разделы)

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю доступен выбор между разделами подготовки. При нажатии на кнопку раздела он разворачивается и пользователю становится доступен перечень тем (описывается в пункте 7.4.10). В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – тестирование, выделяется цветом.



1. Экран разделов базы знаний
   1. Страница базы знаний (темы)

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю доступен выбор между темами подготовки. При нажатии на кнопку темы он разворачивается и пользователю становится доступен перечень тем (описывается в пункте 7.4.11). В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – база знаний, выделяется цветом.



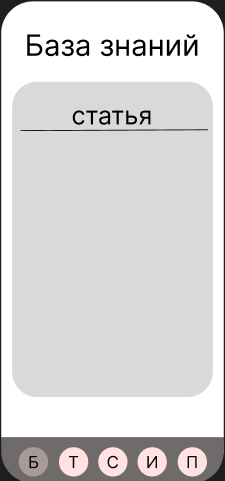
1. Экран тем базы знаний
   1. Страница базы знаний (статьи)

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю доступен выбор между статьями подготовки. При нажатии на кнопку статьи открывается соответствующее окно (описывается в пункте 7.4.12). В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – база знаний, выделяется цветом.



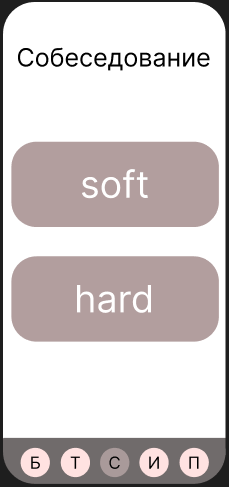
1. Экран статей базы знаний
   1. Страница статьи

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю доступна статья для ознакомления. В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – база знаний, выделяется цветом.



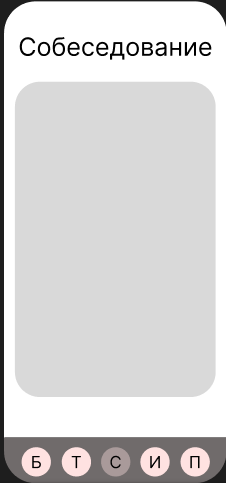
1. Экран статьи
   1. Экран разделов собеседования

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю доступны для выбора разделы. При выборе раздела открывается страница собеседования (описывается в пункте 7.4.14) В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – тестирование, выделяется цветом.



1. Экран разделов собеседования
   1. Экран вопроса собеседования

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю предоставляется вопрос и возможность дать на него ответ в текстовой или голосовой форме. После ответа на все вопросы, открывается экран результатов (описывается в пункте 7.4.15). В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – собеседование, выделяется цветом.



1. Экран вопроса собеседования
   1. Экран результатов собеседования

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю выводятся результаты собеседования, сильные и слабые стороны, рекомендации по исправлению. В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – собеседование, выделяется цветом.

* 1. Экран результатов собеседования

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю выводятся результаты собеседовании за последнюю попытку, сильные и слабые стороны, рекомендации по исправлению. В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – тестирование, выделяется цветом.



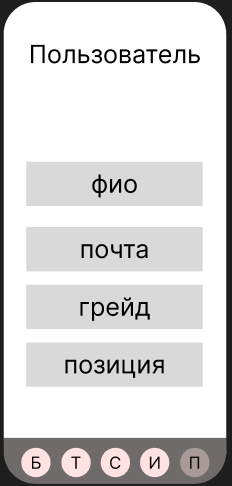
1. Экран результатов собеседований №1
   1. Экран профиля пользователя

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю выводятся результаты собеседовании за последнее время с возможностью подробного изучения статистики через экран результатов собеседования (7.4.16), сильные и слабые стороны, рекомендации по исправлению. В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – история, выделяется цветом.



1. Экран результатов собеседований №2
   1. Экран профиля пользователя

Страница доступна авторизованным пользователям. На странице работает стандартная логика обработки ошибок сервера. Пользователю доступна информация о своем профиле, он так же может изменить реквизиты аккаунта. В нижней части экрана пользователю доступно меню с выбором страниц. Выбранная страница, в данном случае – профиль, выделяется цветом.



1. Экран профиля пользователя
2. Дизайн приложения
   * + Все экраны должны быть выполнены в одном стиле;
     + Все экраны должны корректно отображаться;
     + Верстка экранов должна соответствовать ТЗ;
     + Все элементы управления приложением должны быть заметны и выделены на общем фоне;
     + Основное меню приложение должно находиться в нижней части каждого экрана.
3. Общие требования к системе
   1. Функциональные требования

Система должна:

* + - предоставлять форму регистрации нового пользователя;
    - предоставлять форму входа для существующих пользователей;
    - разрешать выход из аккаунта по запросу пользователя;
    - предоставлять возможность изменить данные профиля (имя, фото, контакты);
    - выбирать грейд (Junior/Middle/Senior);
    - позволять выбрать специализацию (Frontend, Backend, Data Science и т.д);
    - запускать симуляцию собеседования на основе выбранной специализации и грейда;
    - генерировать технические вопросы по выбранной теме и проверять ответы;
    - предоставлять оформление и отмену платной подписки.
  1. Нефункциональные требования

К основным требованиям относится безопасность данных пользователей, а именно:

* + - Шифрование передаваемых данных;
    - Модульная архитектура для легкого добавления новых функций;
    - Адаптивный интерфейс для разных разрешений экрана;
    - Возможность увеличения количества пользователей в 10 раз без изменения архитектуры.

1. Порядок контроля и приемки работ

Предварительные отчёты по работе будет проводиться вовремя

рубежных аттестаций:

* + - 1 аттестация (конец марта 2025) - создан репозиторий проекта на GitHub, распределены задачи проекта в YouGile, создан проект Miro с общей логикой системы, созданы вайерфреймы в Figma, подготовлено техническое задание с прилагающимися диаграммами проектирования, а также предоставлены API документация, Roadmap проекта, анализ целевой аудитории и локальных особенностей, планы по масштабированию и монетизации;
    - 2 аттестация (конец апреля 2025) - написана основополагающая часть кода приложения, реализована БД и ее взаимодействие с сервером, проведена отладка и доработка кода, проведено тестирование по работе системы;
    - 3 аттестация (конец мая 2025) - разработан курсовой проект, выполнены завершающие работы по доработке приложения, предоставлена готовая система.

Результатом работы является готовое веб-приложение со всей необходимой документацией предоставляется заказчику в срок. После заказчик осуществление проверки соответствия функциональности приложения согласно Техническому Заданию.

Для проверки исполнитель обязуется предоставить следующий комплект:

* + - техническое задание;
    - курсовой проект;
    - демонстрационная версия приложения со всеми основными сценариями;
    - исходный код приложения.

Все необходимые материалы будут хранится на Github.

1. Реквизиты и подписи сторон

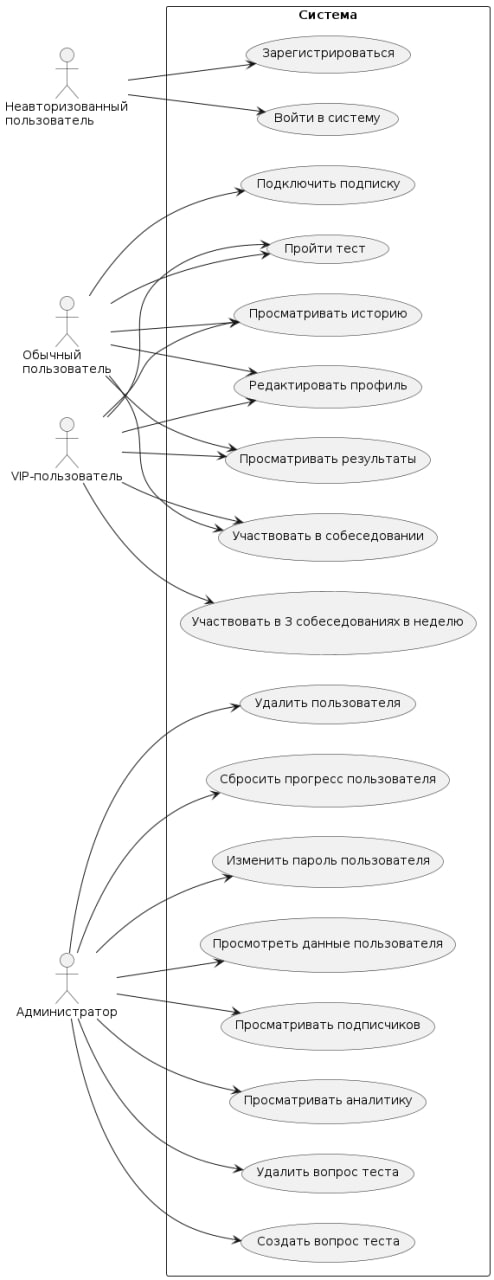
СОСТАВИЛИ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование  организации, предприятия | Должность исполнителя | Фамилия, имя, отчество | Подпись | Дата |
| Воронежский  Государственный Университет | Студент | Бессонов Кирилл Алексеевич |  | 02.04.2025 |
| Воронежский  Государственный Университет | Студент | Петров Артем Николаевич |  | 02.04.2025 |
| Воронежский  Государственный Университет | Студент | Лихих Ульяна Сергеевна |  | 02.04.2025 |

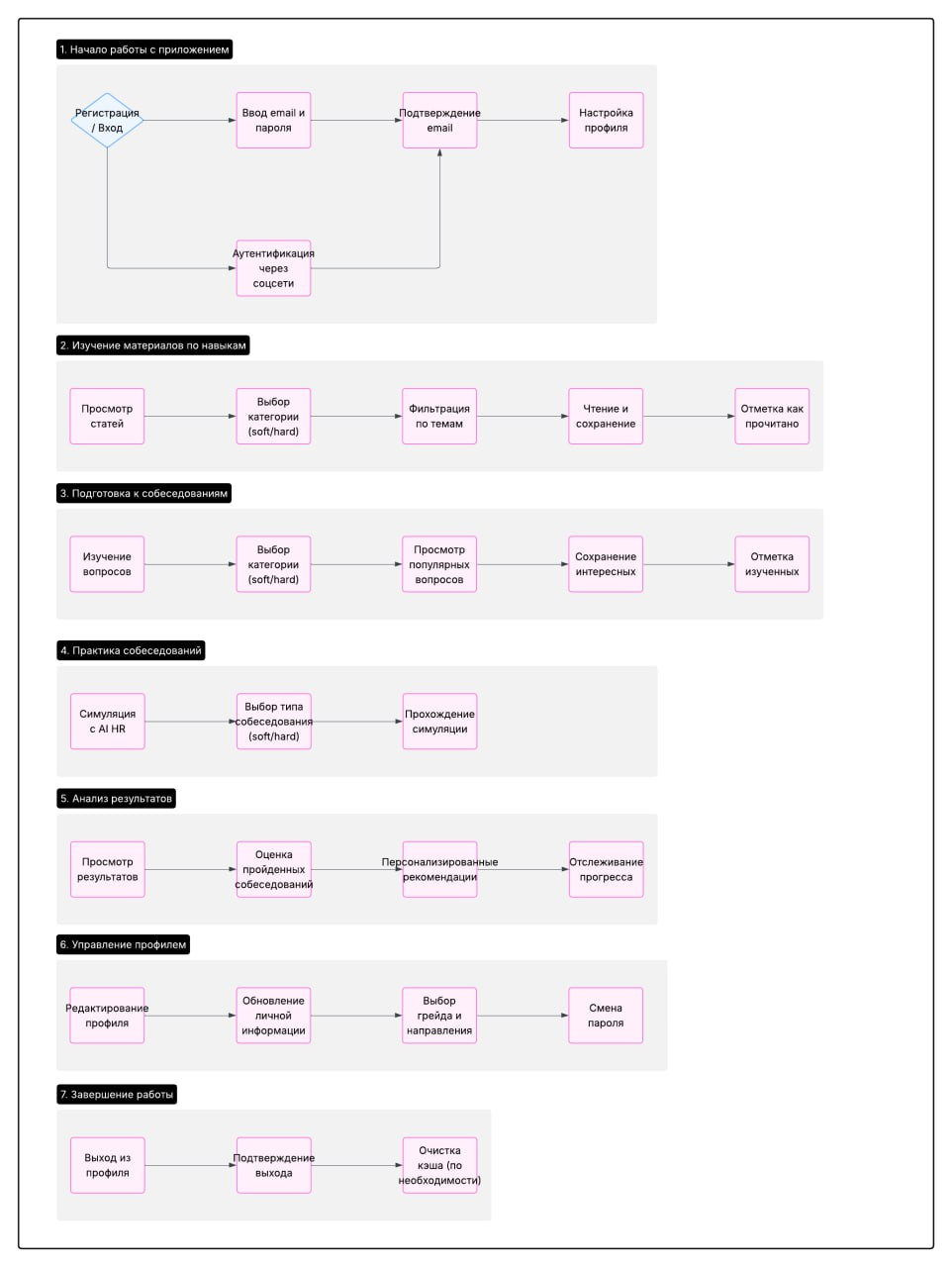
СОГЛАСОВАНО

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование  организации, предприятия | Должность | Фамилия, имя, отчество | Подпись | Дата |
| Воронежский  Государственный Университет | Старший преподаватель | Тарасов Вячеслав Сергеевич |  | 02.04.2025 |
| Воронежский Государственный  Университет | Преподаватель | Москаленко Андрей Вячеславович |  | 02.04.2025 |

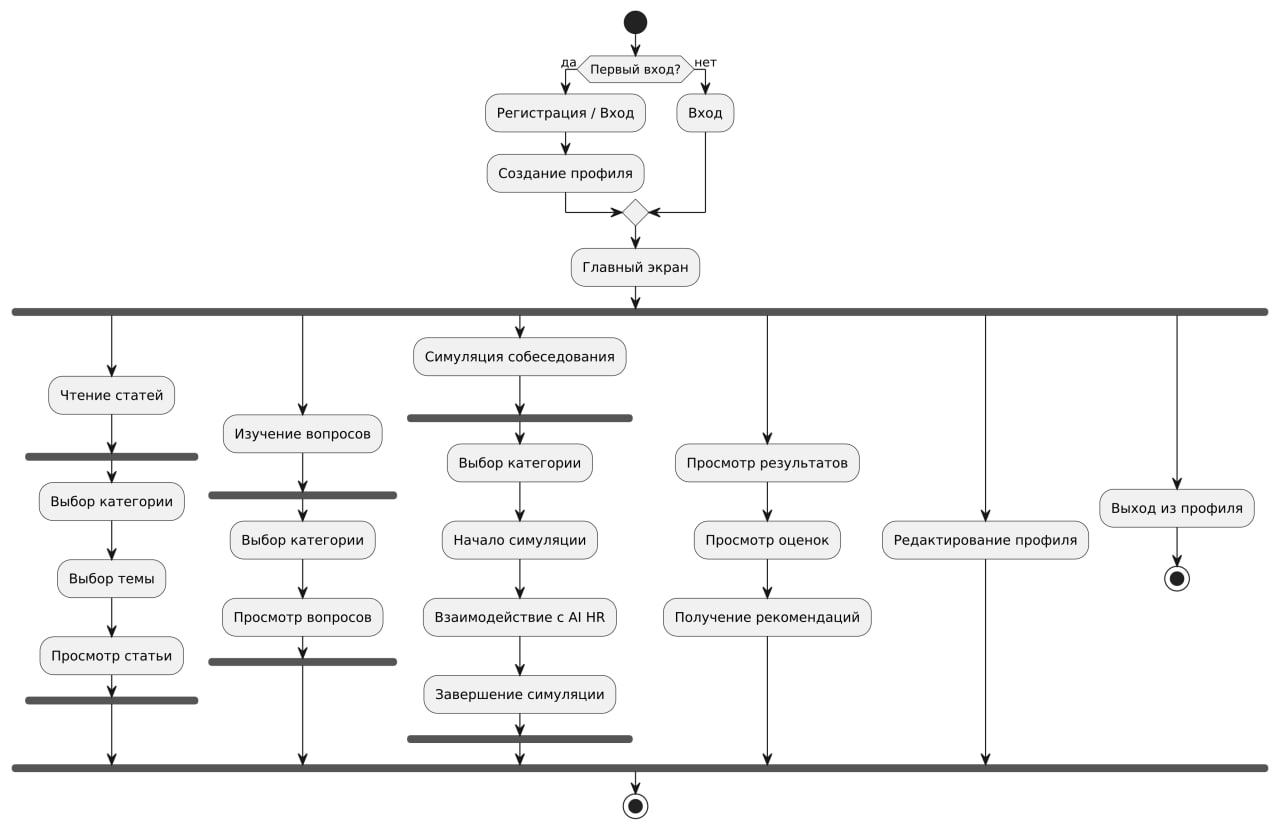
ПРИЛОЖЕНИЕ 1



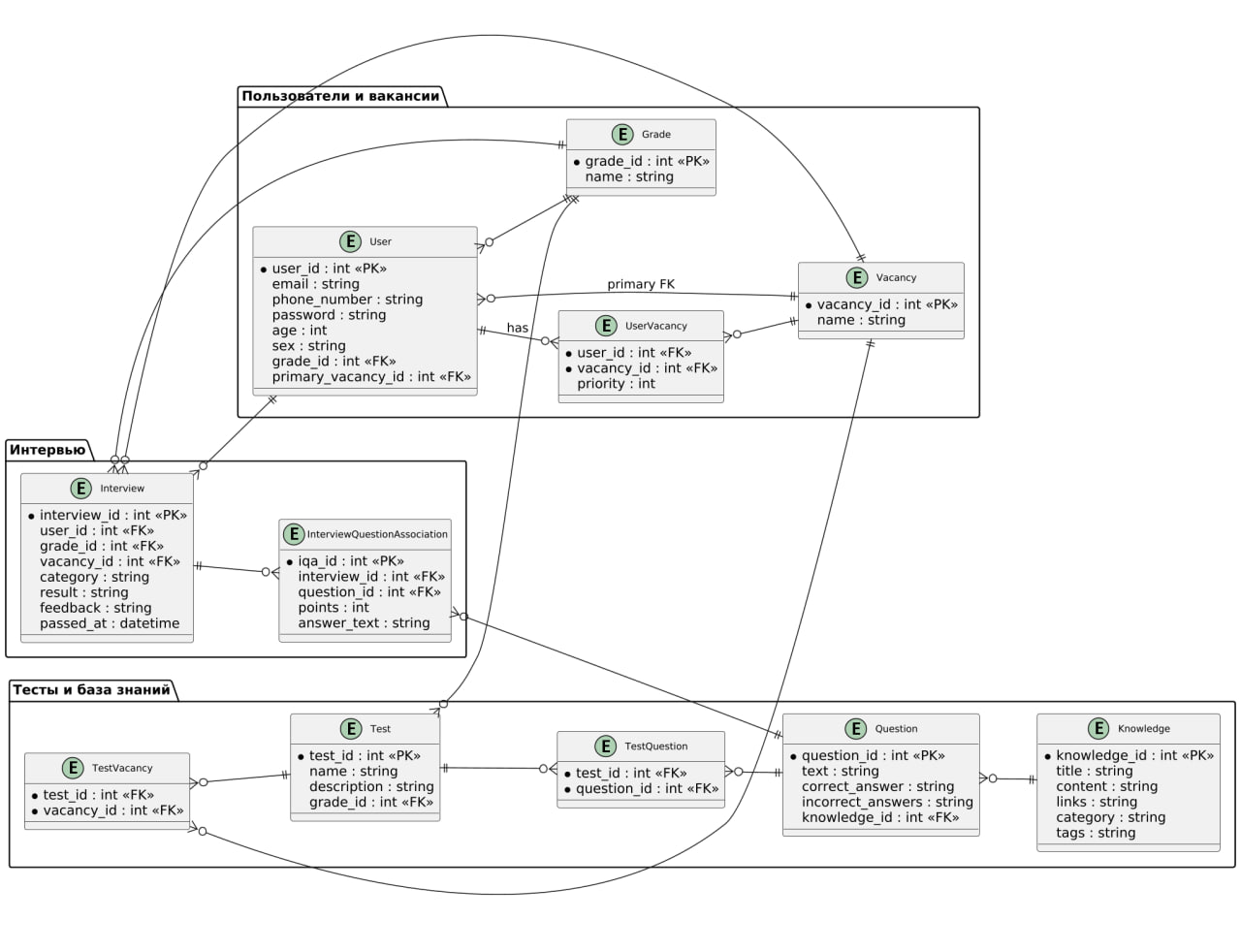
1. Диаграмма Use case



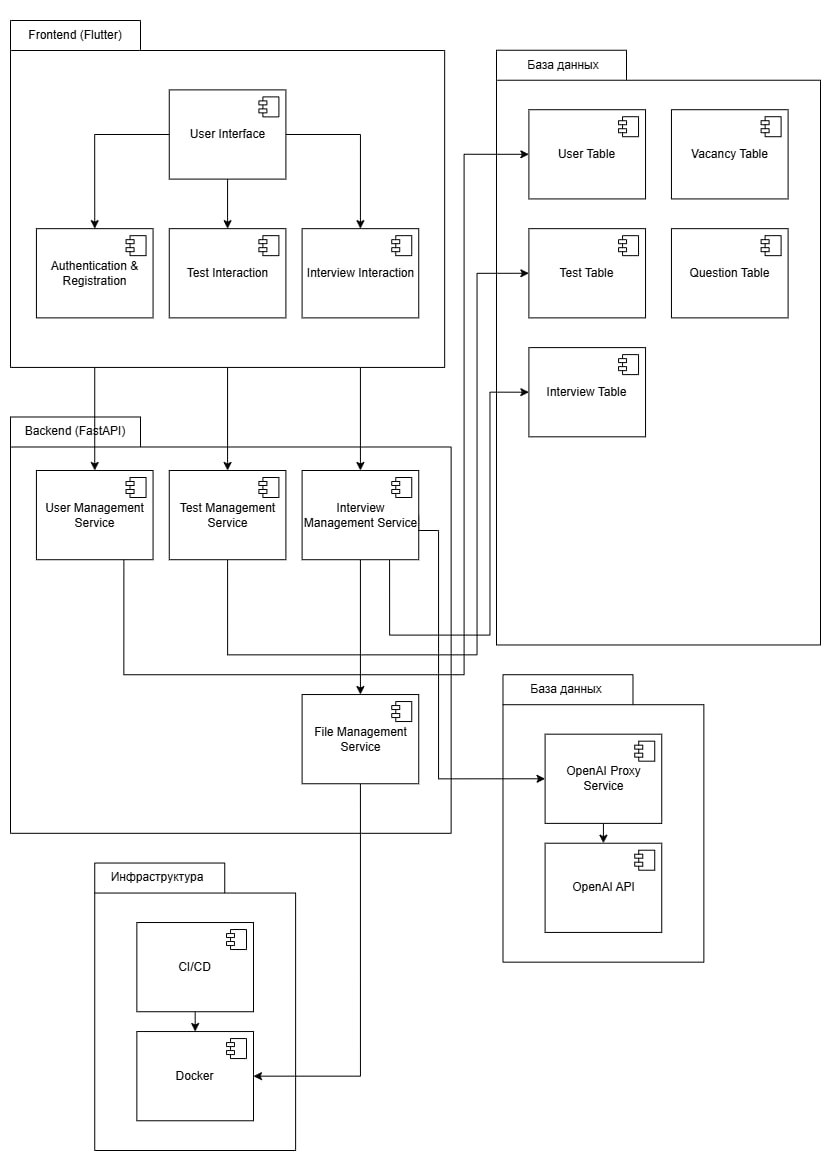
1. Диаграмма User story map



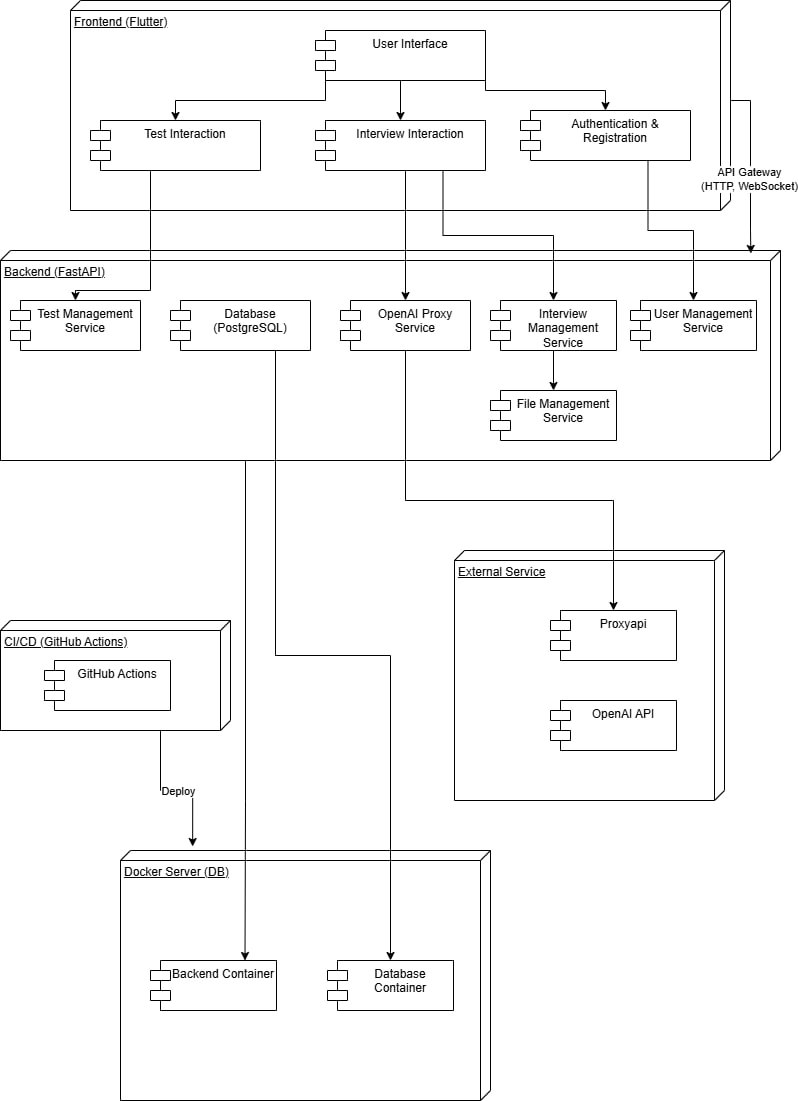
1. Диаграмма активностей



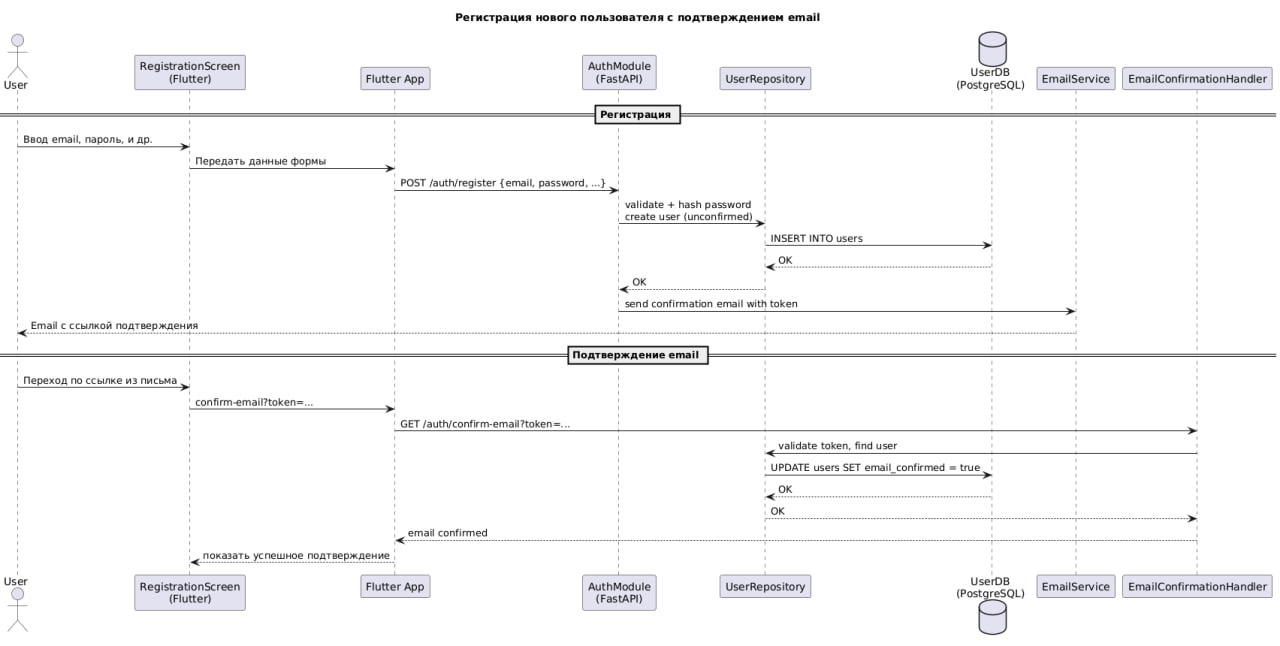
1. ERD



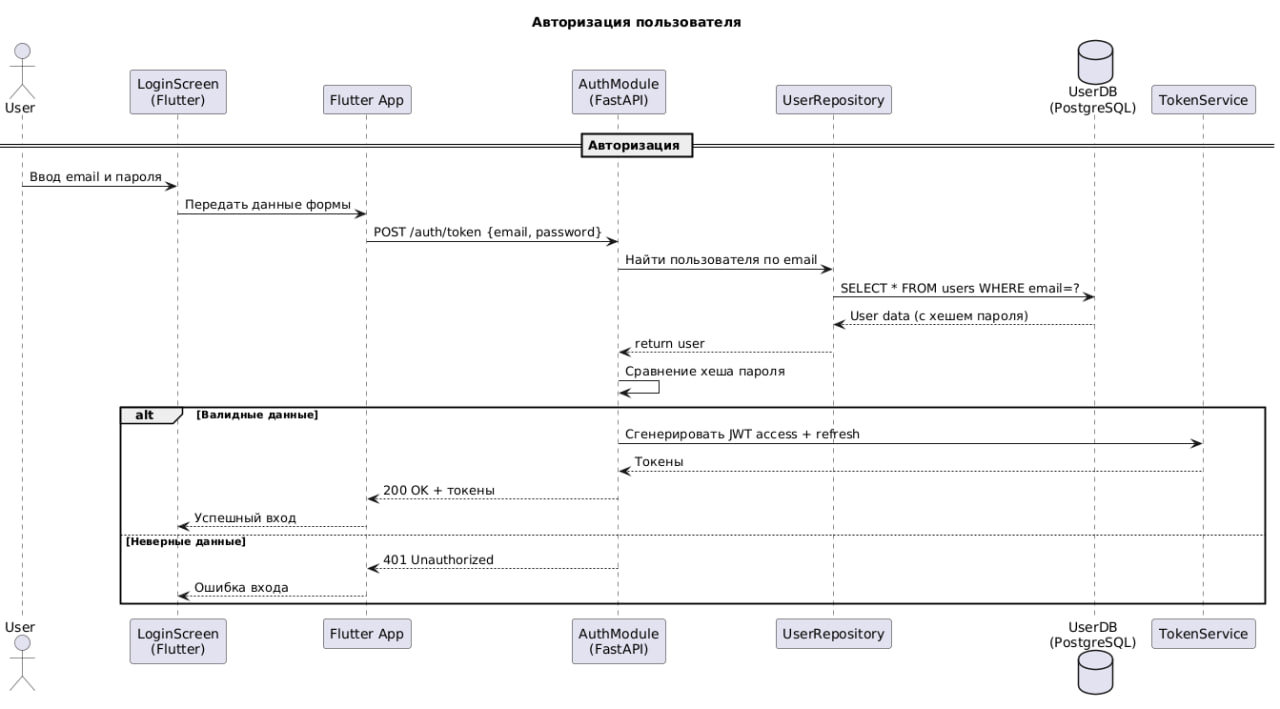
1. Диаграмма компонентов



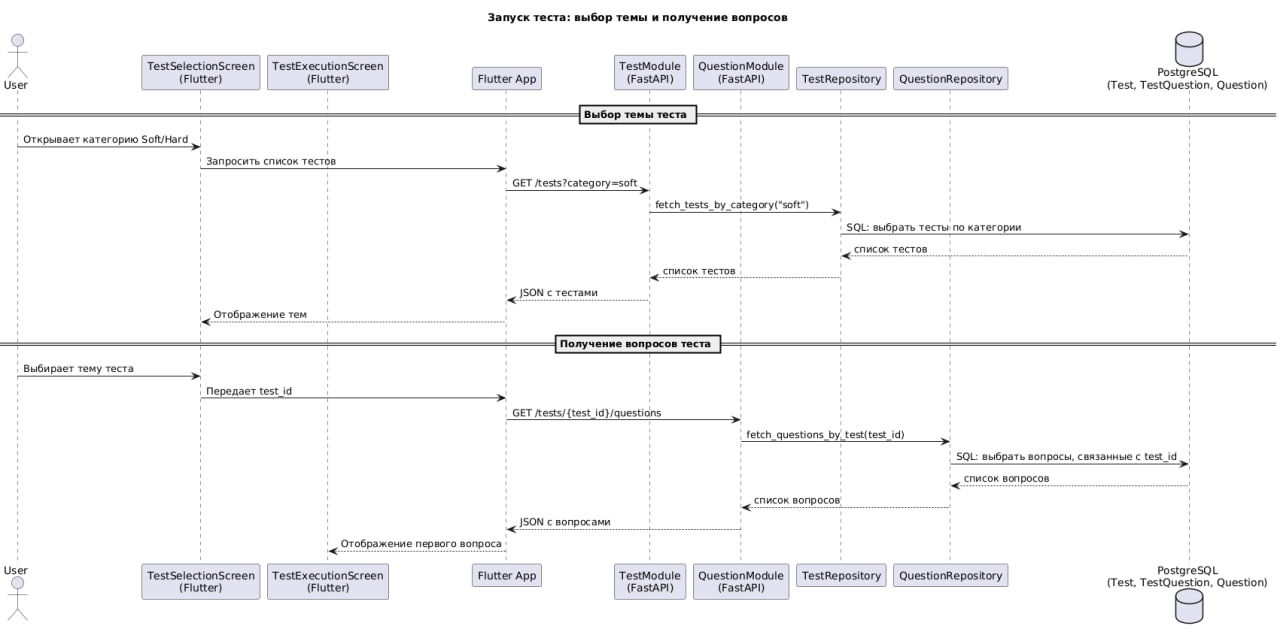
1. Диаграмма развертывания



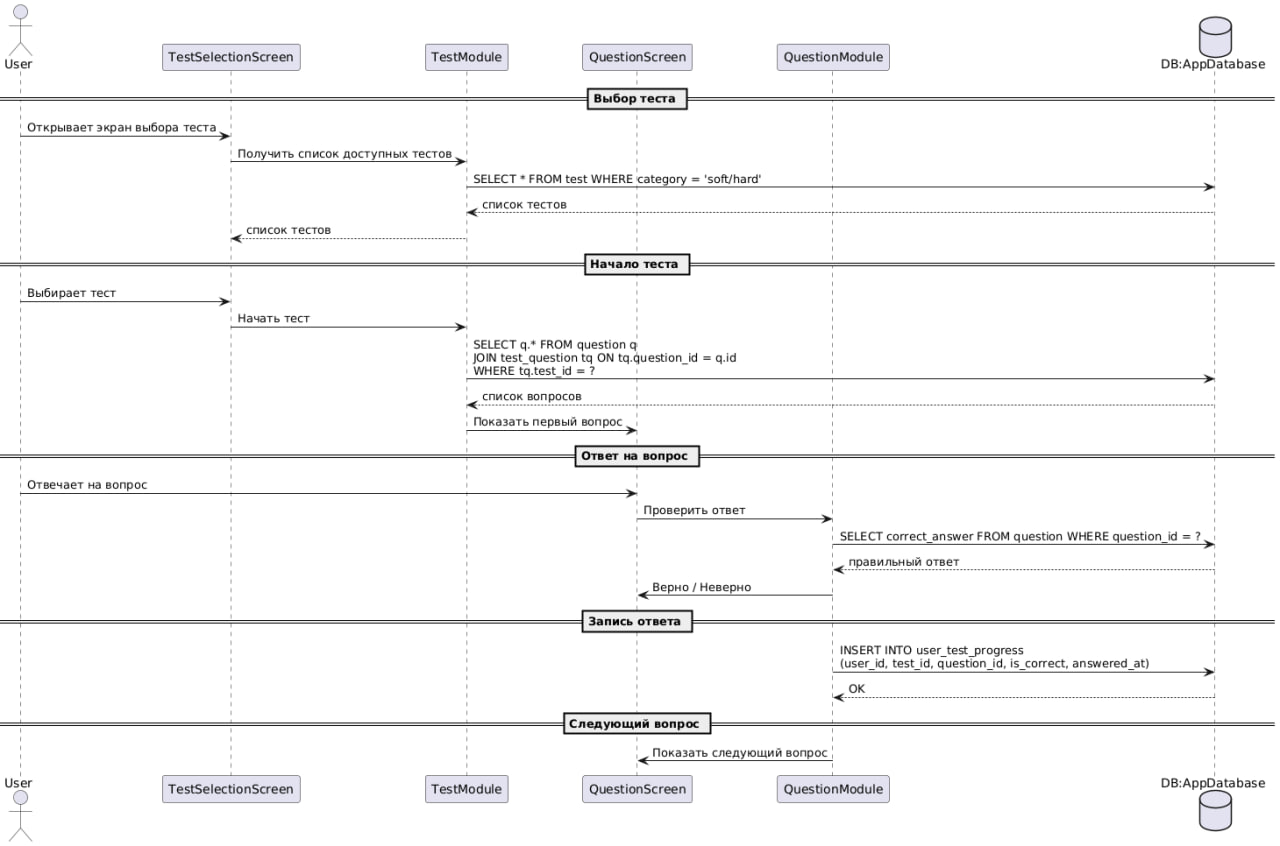
1. Диаграмма последовательностей 1



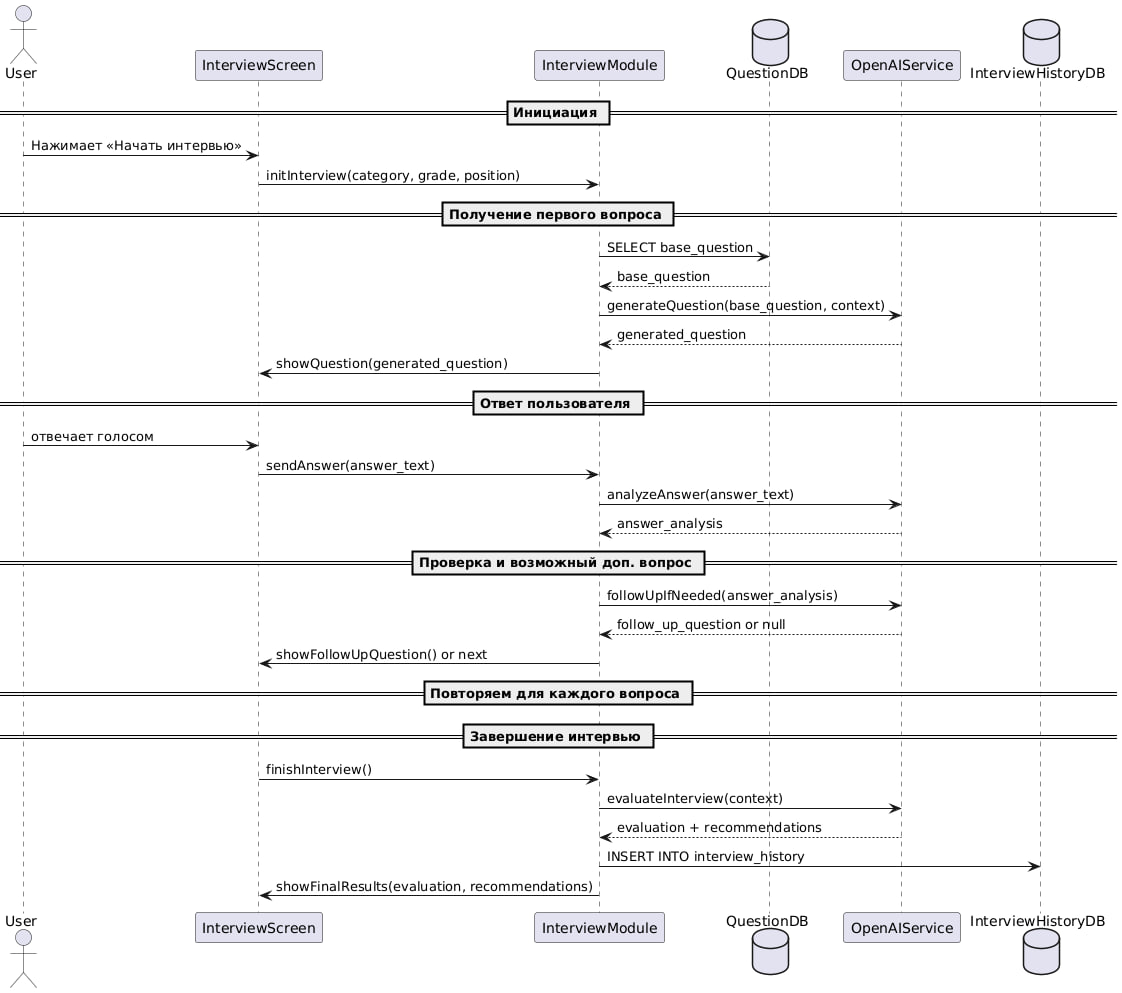
1. Диаграмма последовательностей 2



1. Диаграмма последовательностей 3



1. Диаграмма последовательностей 4



1. Диаграмма последовательностей 5