### РУКОВОДИТЕЛЬ

**ЗАЯВЛЕНИЕ ВИДЕНИЯ:** «Голос твоего текста — твой выбор!»

**ВИДЕНИЕ ПРОДУКТА:**

Цель: предоставить людям возможность персонализировать опыт чтения и прослушивания контента, давая им возможность выбирать голос, который лучше всего подходит для их потребностей: анонимности, улучшения восприятия голоса, помощи людям с ограниченными возможностями, развлечения и т.д.

Концепция: Концепция продукта основывается на идее свободы и креативности. Мы стремимся создать платформу, где каждый человек может найти подходящий голос для своих текстов и уникально выразить свои мысли и идеи. Для людей с ограниченными возможностями платформа позволит лучше передавать информацию собеседнику. Озвучивание текста голосом может помочь людям, предпочитающим слушать вместо чтения, а также желающим сохранить анонимность в общении, не раскрывая свою личность.

Мотив: расширение возможности самовыражения и улучшение взаимодействия людей с технологией.

План действий:

1. Разработка прототипа и концепции приложения, выбор технологии озвучивания.

2. Создание базы данных записей и личного кабинета для каждого пользователя.

3. Разработка кода, который будет преобразовывать написанный пользователем текст в озвучку выбранного голоса.

4. Создание удобного интерфейса, который позволит пользователям выбирать голоса, подходящие для озвучивания своих текстов.

5. Тестирование приложения и исправление возможных ошибок.

6. Запуск продукта.

7. Развитие и постоянное улучшение продукта, добавление новых функций.

**ФУНКЦИОНАЛЬНАЯ ДИАГРАММА:**

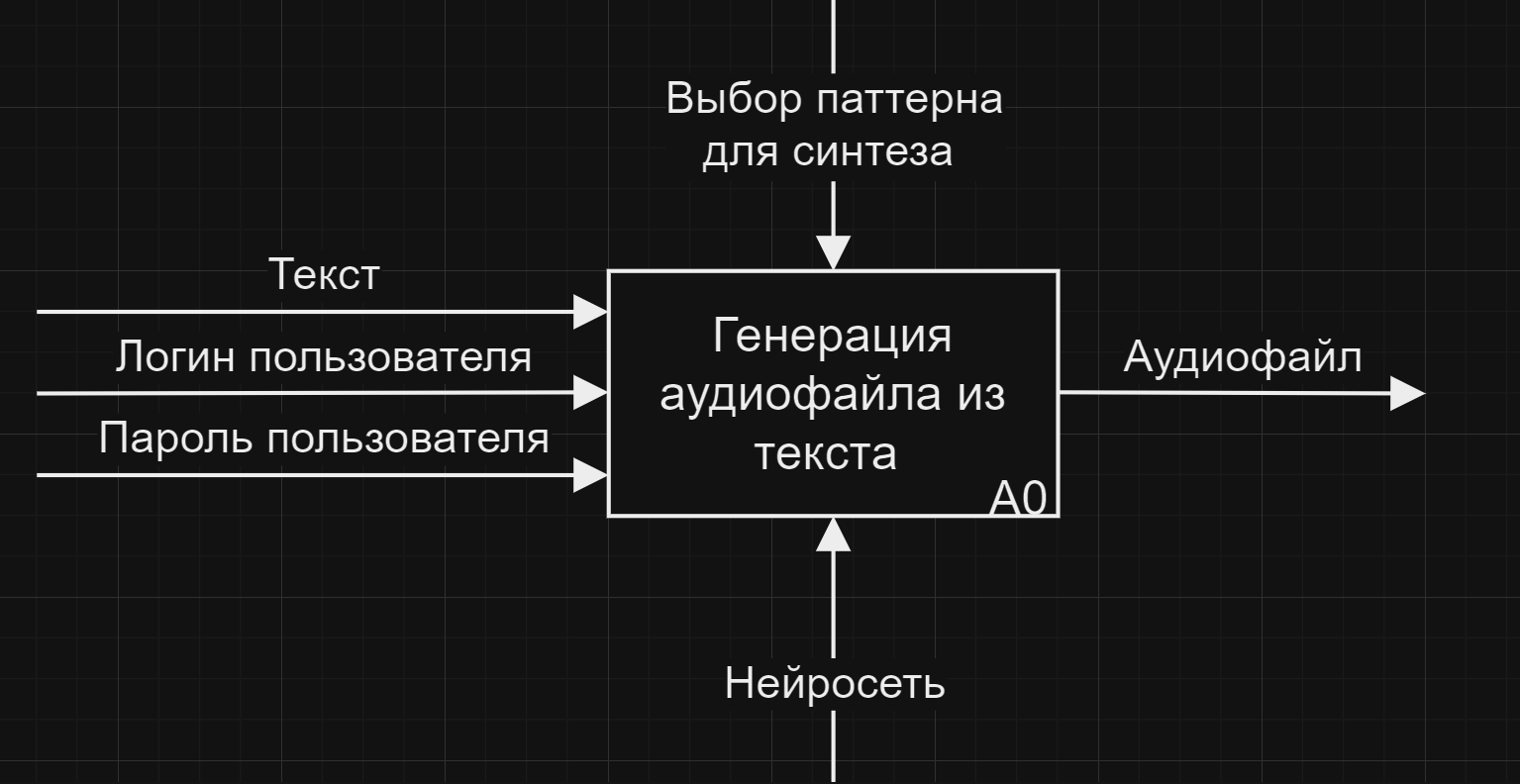
****

Рисунок 1 – исходная функциональная диаграмма

### ПРОЕКТИРОВЩИК

**АРХИТЕКТУРА ПРОДУКТА:**

1. Тело программы: включает в себя модули, отвечающие за взаимодействие с нейросетью, обучение моделей, преобразование массивов данных, связь с БД, выстраивание графической среды (построение GUI)

2. Хранилище данных: база данных для хранения голосовых и пользовательских данных

3. Пользовательский интерфейс (UI): часть приложения, с которой пользователь взаимодействует для ввода текста и выбора голоса

4. Аутентификация и управление учетными записями: компонент, предоставляющий безопасное управление учетными записями пользователей, аутентификацию и регистрацию.

**ТРЕБОВАНИЯ К СИСТЕМНОМУ ПО:**

Для обеспечения эффективной работы приложения важно запускать его на операционной системе Windows, так как разработка и тестирование осуществлялись на этой платформе. Особое внимание следует уделить ознакомлению с руководством по развертыванию нейросети на персональном компьютере, которое поможет правильно настроить объекты и предоставить необходимые ресурсы для работы приложения. Необходимый файл может быть найден и загружен на ресурсе GitHub, где также доступна подробная информация и инструкции по развертыванию и настройке.

**АРХИТЕКТУРНАЯ ДИАГРАММА:**

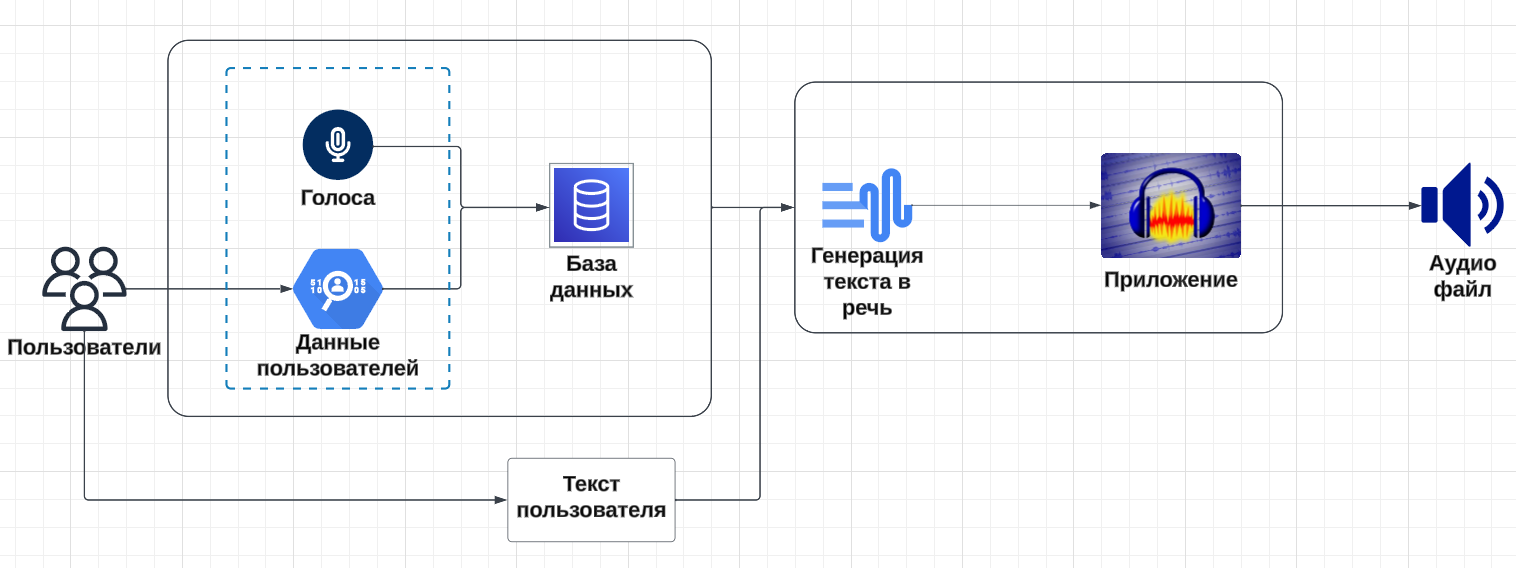
****

Рисунок 2 – архитектурная диаграмма

**ФУНКЦИОНАЛЬНАЯ ДИАГРАММА:**

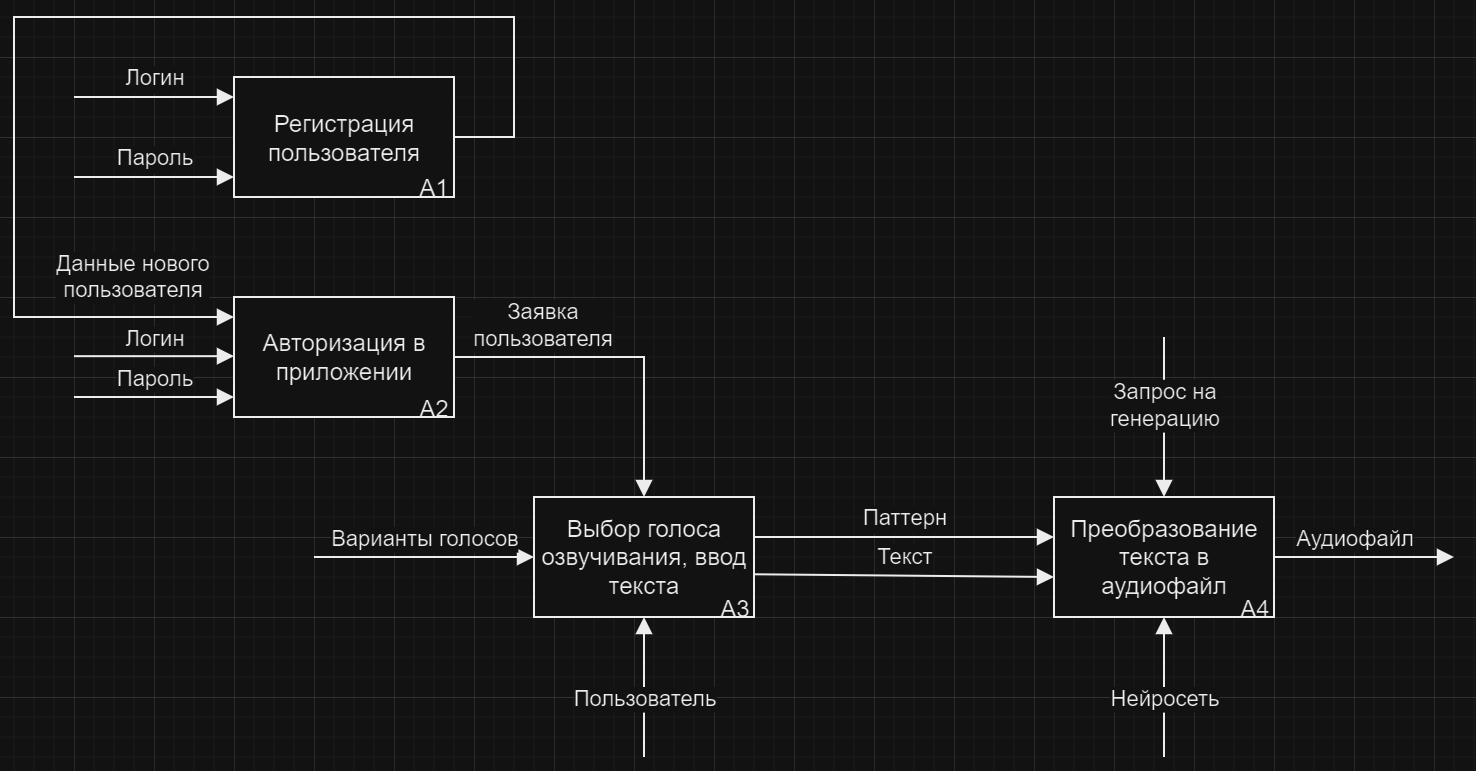


Рисунок 3 – фактическая детализированная функциональная диаграмма

**ДИАГРАММА ПОСЛЕДОВАТЕЛЬНОСТИ:**

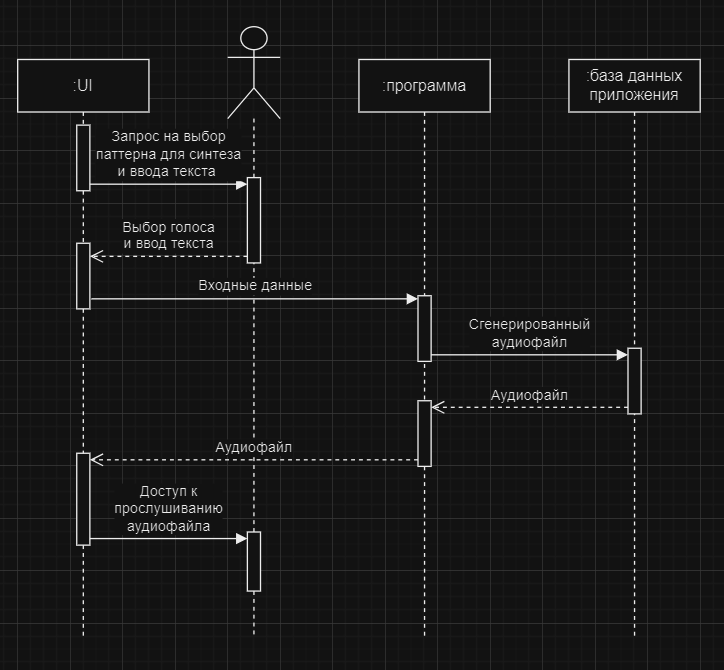


Рисунок 4 – регистрация в приложении

**КОНЦЕПТУАЛЬНАЯ МОДЕЛЬ БД:**

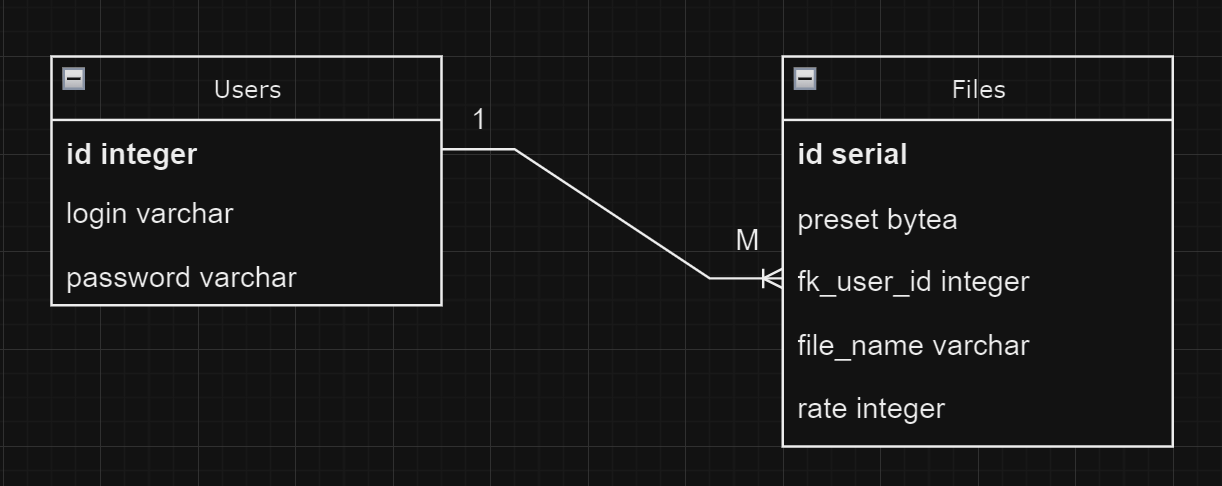
****

Рисунок 5 – концептуальная модель БД

### РАЗРАБОТЧИК

**ВЫБРАННЫЕ ЯЗИКИ ПРОГРАММИРОВАНИЯ, ОБОСНОВАНИЕ ВЫБОРА:**

Язык программирования Python был выбран в связи с его превосходной поддержкой нейронных сетей и обширным набором библиотек, что необходимо для реализации нашей идеи.

Мы приняли решение в пользу PostgreSQL, поскольку мы уже знакомы с ним благодаря предыдущим курсам, и работа с ним кажется нам более интуитивно понятной. Это важно для понимания структуры баз данных и способов работы с различными типами файлов, которые могут быть в ней сохранены.

**ДИАГРАММА ПАКЕТОВ:**

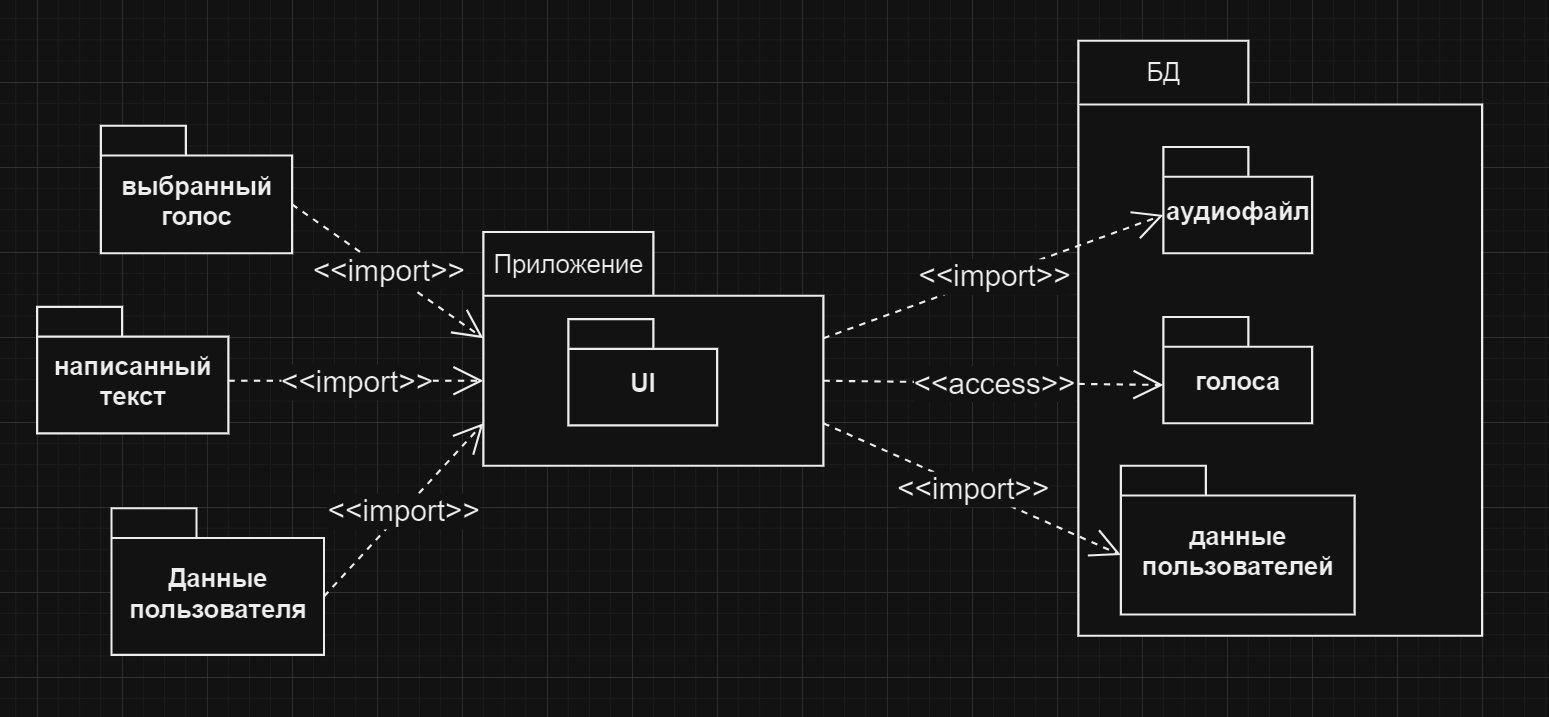
****

Рисунок 6 – диаграмма пакетов

**ДИАГРАММА ФИЗИЧЕСКОЙ МОДЕЛИ:**

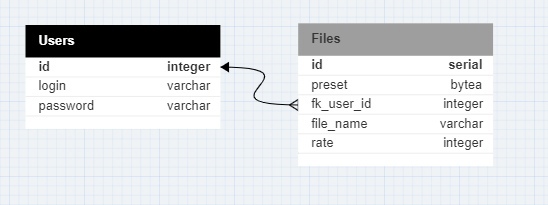


Рисунок 7 – диаграмма физической модели