

Московский государственный технический
университет имени Н.Э. Баумана

***Разработка методов
создания виртуального
помощника с визуализацией
на Unreal Engine 5***

Научный руководитель Витюков Ф.А.
Россия, Москва, Редколис Р.Р. РК6-82Б 2024



Цель и задачи

Цель работы

Создать функционального визуализированного помощника, который может обрабатывать голосовые команды и отвечать голосом.

Задачи:

Анализ существующих методов и инструментов разработки виртуальных ассистентов.

Проектирование и реализация модели виртуального помощника.

Оптимизация производительности приложения.

Анализ результатов и перспектив применения разработанного помощника .

1-й модуль

Реализация виртуального помощника

Настройка интерактивного поведения помощника

2-й модуль

Разработка системы распознавания речи

3-й модуль

Интеграция с внешними сервисами через REST API

4-й модуль

Разработка системы контекстного понимания

Создание модели MetaHumans и привязка lipsync

1

Синхронизация речи и движения губ

Lipsync, или синхронизация речи с движением губ персонажа, является важным аспектом для достижения реалистичности в воспроизведении речи в виртуальной среде.

2

Использование MetaHumanSDK

Мы используем MetaHumanSDK, предоставленный Unreal Engine, для создания главной и точной синхронизации между аудиофайлом и движением губ виртуального персонажа.

3

Повышение реализма взаимодействия

Это позволяет обеспечить естественное и убедительное выражение лица персонажа во время речи, что повышает иммерсию и реализм взаимодействия с виртуальным помощником.

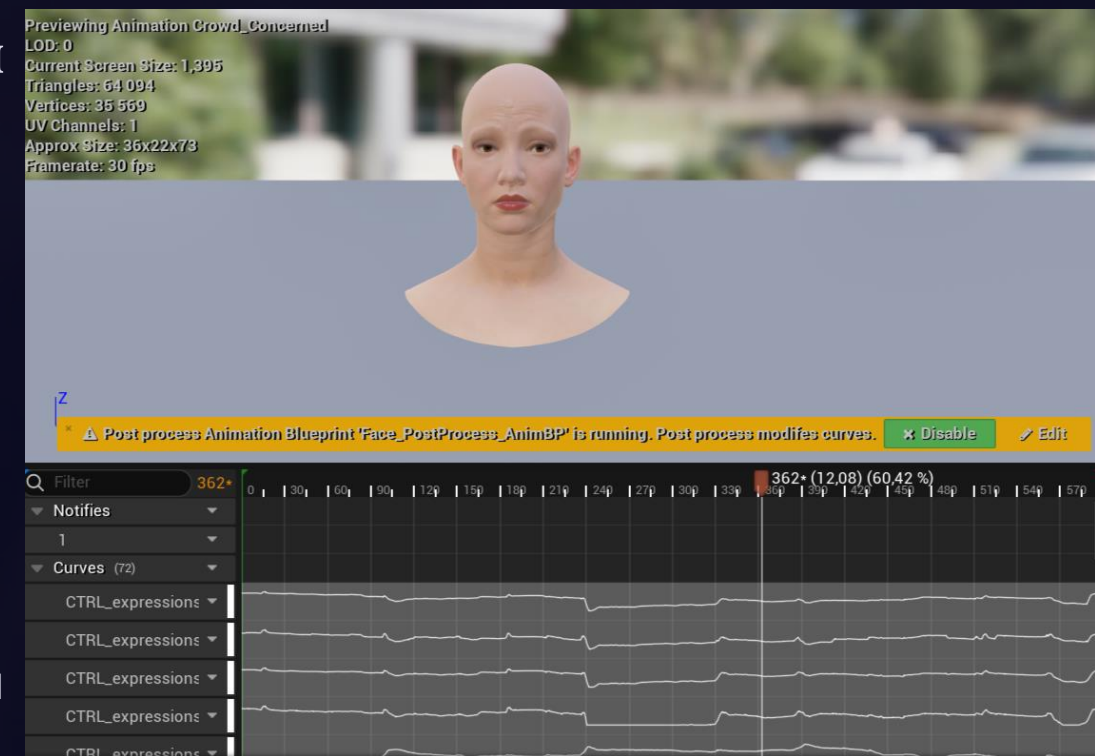


Рис. 1 анимация головы



Разработка системы обработки голоса

Настройка параметров

Плагин предоставляет широкий спектр настраиваемых параметров, параметров, таких как количество потоков, размер шага, максимальное количество токенов и другие, для оптимизации производительности и точности распознавания речи.

1

Распознавание речи

Класс `SpeechRecognition` позволяет
позволяет распознавать и
интерпретировать аудиоданные,
преобразуя их в текстовый формат
с использованием алгоритмов и
моделей машинного обучения.

2

Захват аудиоданных

Плагин `CapturableSoundWave` предназначен для захвата аудиоданных с устройств ввода, таких как микрофон, и последующего воспроизведения.

3

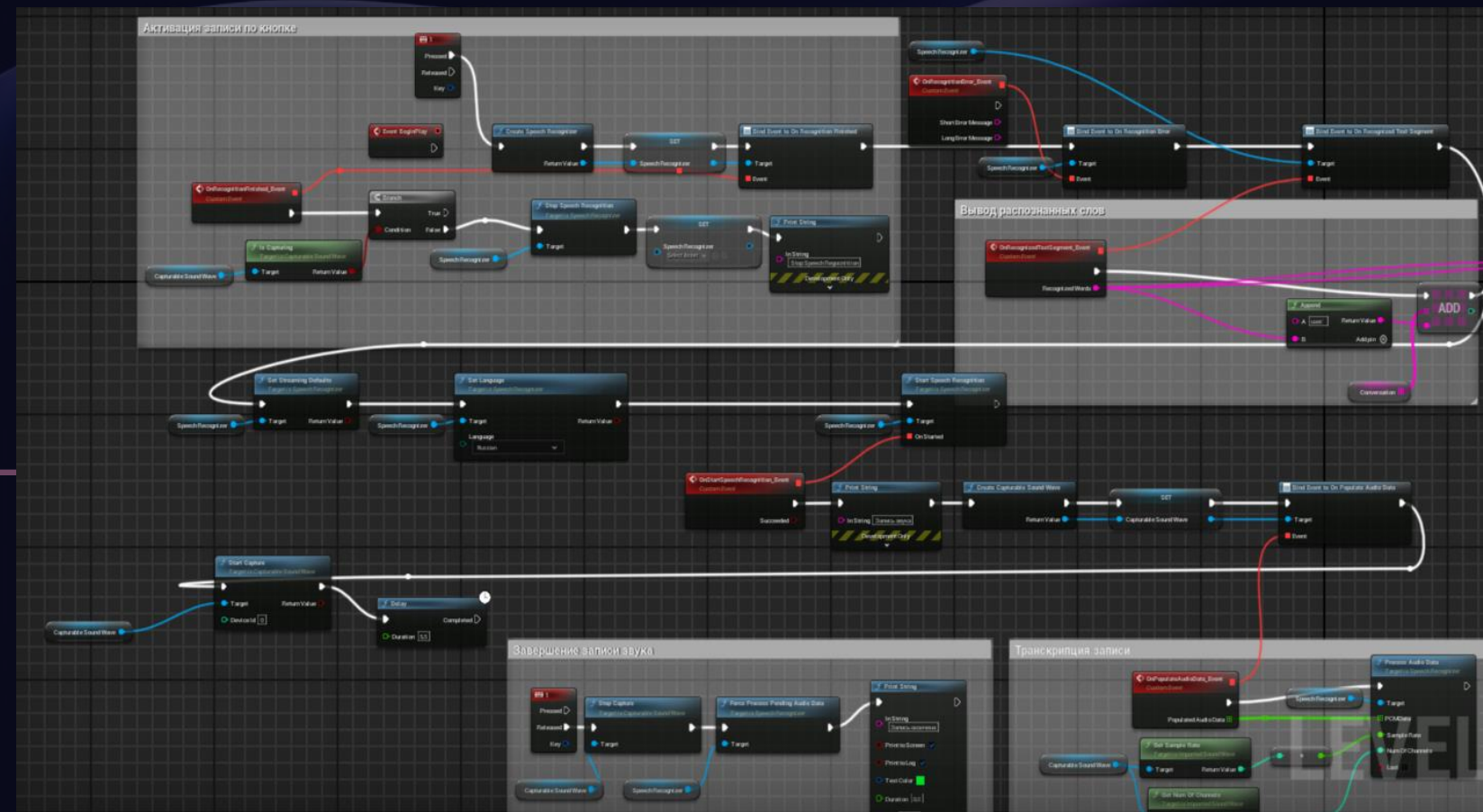
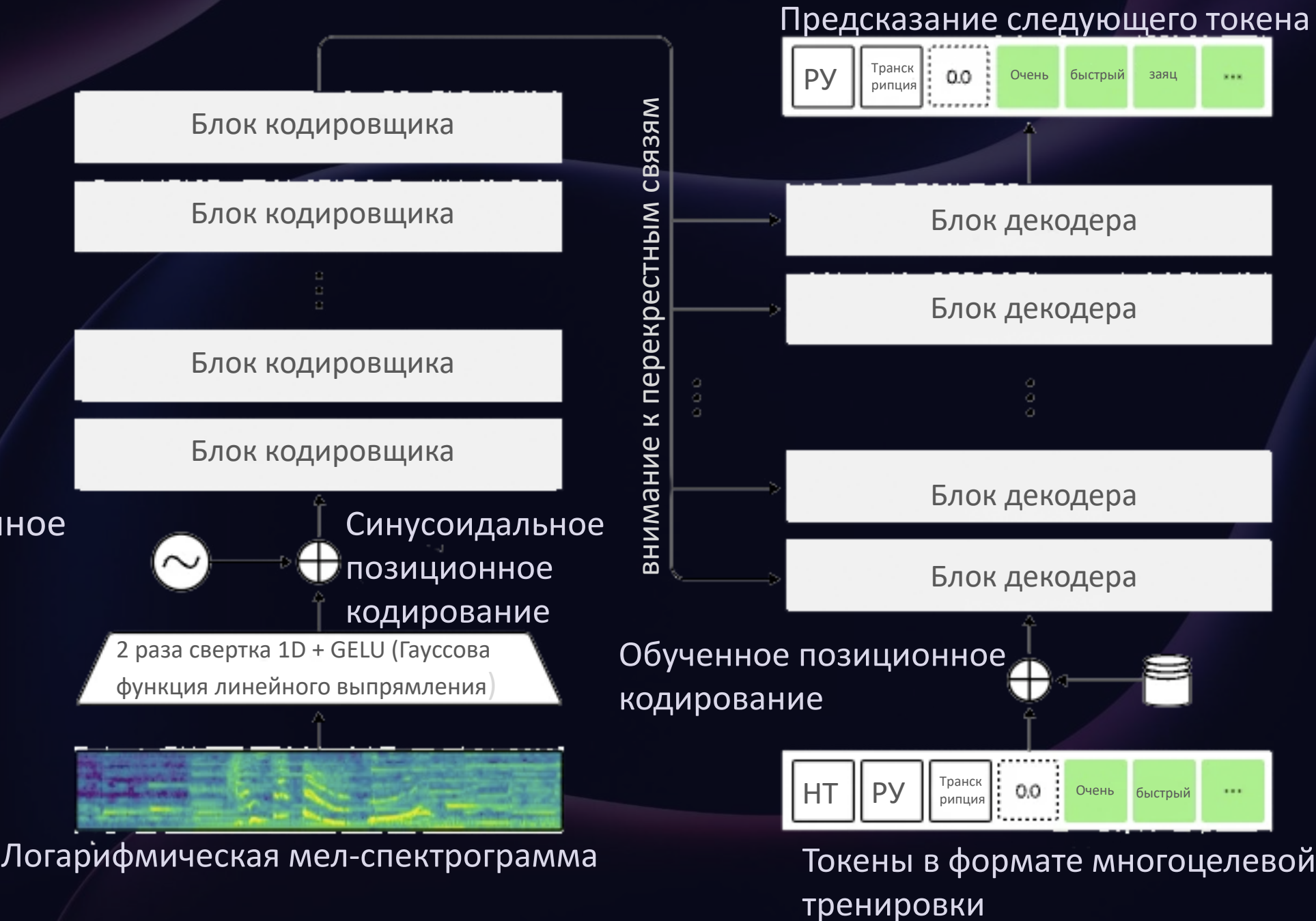


Рис. 2 Логика обработки голоса

Разработка системы обработки голоса

- 1. Загрузка аудиофайла
- 2. Предварительная обработка
- 3. Преобразование аудиосигнала в спектрограммы
- 4. Применение конволюционных слоев и позиционного кодирования
- 5. Обработка спектрограммы в блоках кодировщика
- 6. Перекрестное внимание
- 7. Обработка в блоках декодера и обученное позиционное кодирование
- 8. Генерация текста
- 9*. Дополнительная обработка текста
- 10. Формат многоцелевой тренировки



Разработка API запросов

Взаимодействие с облачными сервисами

API-сервисы предоставляют возможность взаимодействия с облачными ресурсами и сервисами через программные интерфейсы, позволяя разработчикам создавать, развивать и масштабировать свои приложения без необходимости управления физической инфраструктурой.

Интеграция с ProxyApi

Были реализованы API-запросы к ChatGPT, используя методы POST и GET для отправки данных на сервер и получения ответов. Это позволило расширить функционал виртуального помощника и обеспечить более интеллектуальное взаимодействие с пользователем.

Использование Elevenlabs

Также было интегрировано использование API Elevenlabs для синтеза речи, что позволило виртуальному помощнику не только распознавать речь, но и генерировать естественные голосовые ответы, повышая реалистичность взаимодействия.

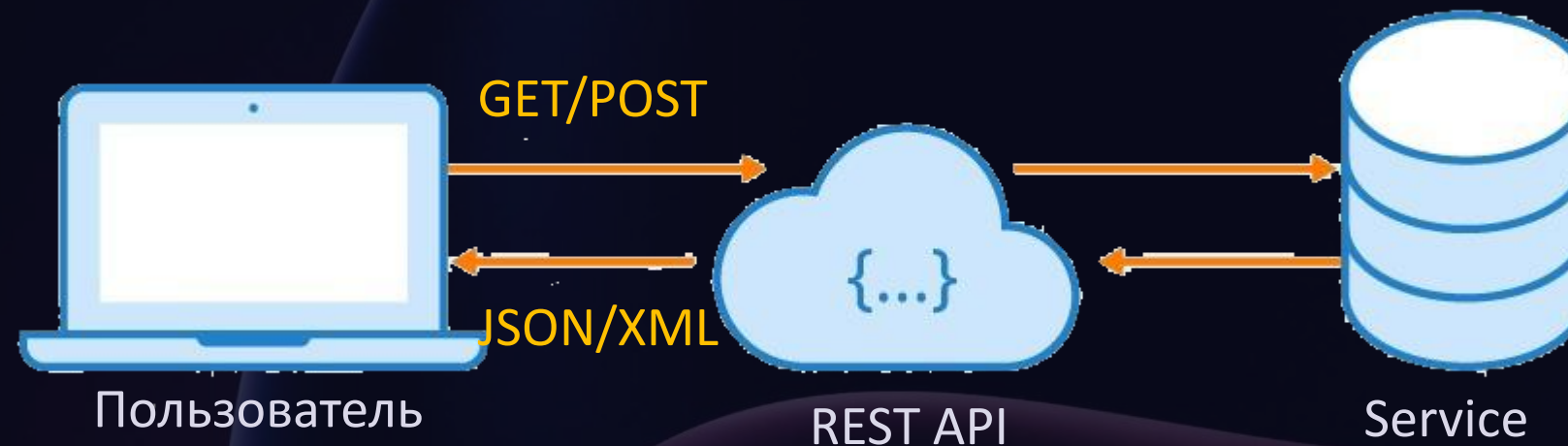


Рис. 3 схема работа API запросов

Разработка API запросов

Использование Elevenlabs

Также было интегрировано использование API Elevenlabs для синтеза речи, что позволило виртуальному помощнику не только распознавать речь, но и генерировать естественные голосовые ответы, повышая реалистичность взаимодействия.

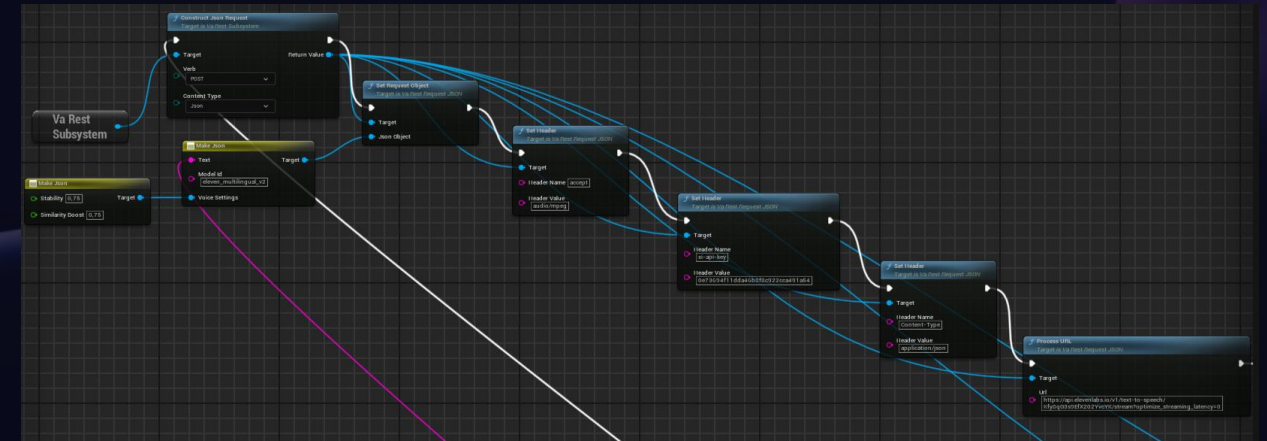


Рис. 4 Логика составления JSON объекта и отправка методом POST

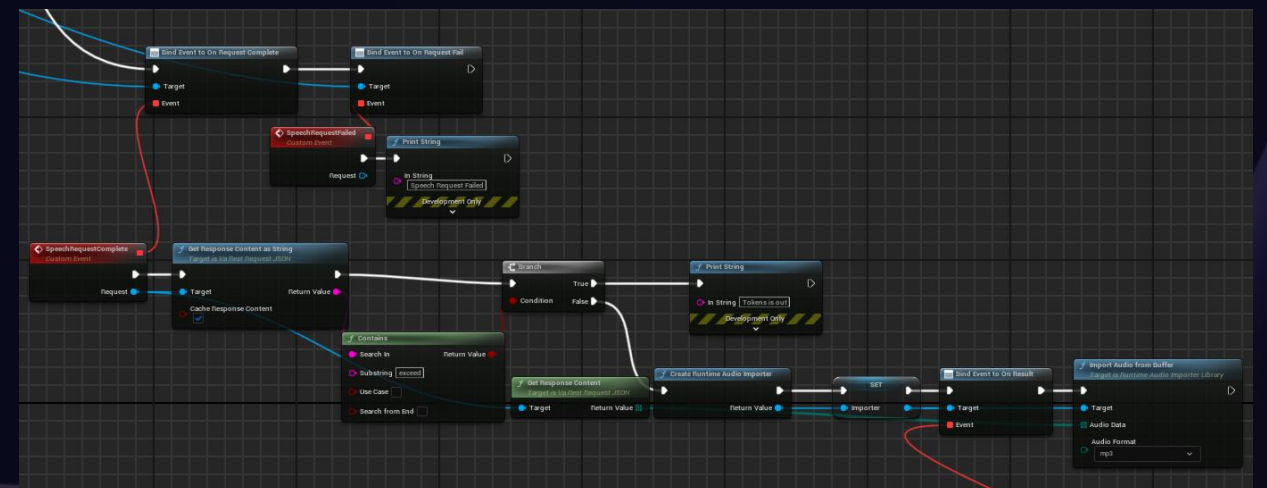


Рис. 5 Отправка GET запроса и обработка ответа

Разработка API запросов

Отправка POST-запроса

POST-запросы отправляют данные в теле запроса, что позволяет передавать большие объемы информации и использовать их для изменения состояния на сервере.

1

```
1 curl https://api.proxyapi.com/v1/chat/completions
2 -H "Content-Type: application/json"
3 -H "Authorization: Bearer $OPENAI_API_KEY"
4 -d '{
5   "model": "gpt-3.5-turbo",
6   "messages": [
7     {
8       "role": "system",
9       "content": "You are a helpful assistant."
10    },
11    {
12      "role": "user",
13      "content": "Hello!"
14    }
15  ]
16 }'
```

Рис. 6 Формат JSON объекта отправленного POST-запросом

```
{
  "id": "chatcmpl-123",
  "object": "chat.completion",
  "created": 1677652288,
  "model": "gpt-3.5-turbo-0613",
  "system_fingerprint": "fp_44709d6fcb",
  "choices": [{
    "index": 0,
    "message": {
      "role": "assistant",
      "content": "Здравствуйте, как я могу помочь вам сегодня ?"
    },
    "logprobs": null,
    "finish_reason": "stop"
  }],
  "usage": {
    "prompt_tokens": 9,
    "completion_tokens": 12,
    "total_tokens": 21
  }
}
```

Рис. 7 Формат JSON объекта полученного GET-запросом

2

Отправка GET-Запроса И получение ответа

GET-запросы отправляют данные на сервер в виде параметров URL и используются для получения информации.

Здравствуйте, Руслан! Рад приветствовать вас. Чем могу помочь сегодня?
Привет, меня зовут Руслан.
Запись окончена
Запись звука

Рис. 8 результат парсера

Обработка ответа

Полученный ответ из GET-запроса обрабатывается с помощью парсера, полученная информация в дальнейшем используется для генерации аудиофайла

3

Функциональность виртуального помощника



Распознавание речи

Виртуальный помощник способен распознавать речь пользователя и преобразовывать ее в текстовый формат для дальнейшей обработки.



Синтез речи

Помощник также может генерировать естественные голосовые ответы на основе текста, используя технологии text-to-speech.



Интеллектуальный диалог

Интеграция с API ChatGPT позволяет виртуальному помощнику вести осмысленные и интеллектуальные диалоги с пользователем.



Реалистичный облик

Использование технологий MetaHuman обеспечивает виртуальному помощнику реалистичный и выразительный внешний вид.

Слайд не готов

Перспективы развития



Интеграция в рабочие процессы

Виртуальные помощники могут быть интегрированы в различные рабочие процессы, повышая эффективность и производительность сотрудников.



Применение в образовании

Использование виртуальных помощников в образовательной среде может способствовать персонализации обучения и повышению вовлеченности студентов.



Использование в медицине

Виртуальные помощники могут быть задействованы в медицинской сфере для облегчения работы медперсонала и улучшения взаимодействия с пациентами.

Слайд не готов

Заключение

Тип работы	Выпускная квалификационная работа
Тема	Разработка виртуального помощника на Unreal Engine 5
Объекты исследования	3D-моделирование, разработка системы на основе искусственного интеллекта
Основные результаты	Создание цифрового аватара, разработка системы обработки голоса, интеграция API-сервисов, привязка lipsync и модели MetaHumans
	Было реализовано 450+ строк кода

Слайд не готов