**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

Изображение выглядит как ночное небо

Автоматически созданное описание**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ**

**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

**«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук



Образовательная программа «Программная инженерия»

|  |  |
| --- | --- |
| ***Подп. и дата*** |  |
|  |
| ***Инв. № дубл.*** |  |
|  |
| ***Взам. инв. №*** |  |
|  |
| ***Подп. и дата*** |  |
|  |
| ***Инв. №***    ***подл*** |  |
|  |

СОГЛАСОВАНО УТВЕРЖДАЮ



Руководитель группы разработки синтеза речи, АО «Тинькофф Банк»

Академический руководитель образовательной программы «Программная инженерия» профессор департамента программной инженерии, канд. техн. наук



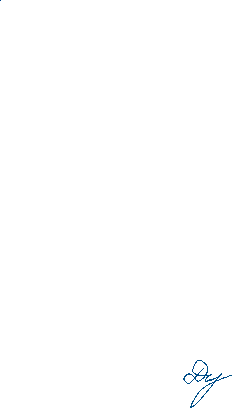
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_С. В. Шишкин \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Шилов



«17» февраля 2022 г. «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

**РАЗРАБОТКА MVP СЕРВИСА СИНТЕЗА РЕЧИ НА АНГЛИЙСКОМ ЯЗЫКЕ**

**Техническое задание**



**ЛИСТ УТВЕРЖДЕНИЯ**

# RU.17701729.07.04-01 ТЗ 01-1-ЛУ

Исполнители студенты групп:

БПАД204 \_\_\_\_\_\_\_\_\_ /Д. И. Калинина/

БПИ202 \_\_\_\_\_\_\_\_\_ /Д. О. Соколова/

БПИ208 \_\_\_\_\_\_\_\_\_ /М. Д. Молокова/

БПМИ203 \_\_\_\_\_\_\_\_\_ /А. Е. Ишутин/

«17» февраля 2022 г

Изображение выглядит как силуэт, ночное небо

Автоматически созданное описание







Изображение выглядит как ночное небо

Автоматически созданное описание

# 

# Москва 2022

# 

УТВЕРЖДЕН

RU.17701729.07.04-01 ТЗ 01-1-ЛУ

|  |  |
| --- | --- |
| ***Подп. и дата*** |  |
|  |
| ***Инв. № дубл.*** |  |
|  |
| ***Взам. инв. №*** |  |
|  |
| ***Подп. и дата*** |  |
|  |
| ***Инв. №***    ***подл*** |  |
|  |

**РАЗРАБОТКА MVP СЕРВИСА СИНТЕЗА РЕЧИ НА АНГЛИЙСКОМ ЯЗЫКЕ**

**Техническое задание**

# RU.17701729.07.04-01 ТЗ 01-1

**Листов 21**

# Москва 2022

ОГЛАВЛЕНИЕ

[АННОТАЦИЯ 4](#_Toc95915124)

[ГОЛОССАРИЙ 5](#_Toc95915125)

[1. ВВЕДЕНИЕ 6](#_Toc95915126)

[1.1 Наименование программы 6](#_Toc95915127)

[1.2 Краткая характеристика области применения программы 6](#_Toc95915128)

[2. ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ 7](#_Toc95915129)

[3. НАЗНАЧЕНИЕ РАЗРАБОТКИ 8](#_Toc95915130)

[3.1 Функциональное назначение 8](#_Toc95915131)

[3.2 Эксплуатационное назначение 8](#_Toc95915132)

[4. ТРЕБОВАНИЯ К ПРОГРАММЕ 9](#_Toc95915133)

[4.1 Требования к функциональным характеристикам 9](#_Toc95915134)

[4.1.1 Требования к серверной части 9](#_Toc95915135)

[4.1.2 Требование к взаимодействию клиентской и серверной частей 10](#_Toc95915136)

[4.1.3 Требования к клиентской части 10](#_Toc95915137)

[4.2 Требования к надежности 10](#_Toc95915138)

[4.2.1 Требования к обеспечению надежного (устойчивого) функционирования программы 10](#_Toc95915139)

[4.2.2 Время восстановления после отказа 11](#_Toc95915140)

[4.2.3 Отказы из-за некорректных действий оператора 11](#_Toc95915141)

[4.3 Условия эксплуатации 11](#_Toc95915142)

[4.3.1 Климатические условия эксплуатации 11](#_Toc95915143)

[4.3.2 Требования к видам обслуживания 11](#_Toc95915144)

[4.3.3 Требования к численности и квалификации персонала 11](#_Toc95915145)

[4.4 Требования к составу и параметрам технических средств 11](#_Toc95915146)

[4.5 Требования к информационной и программной совместимости 11](#_Toc95915147)

[4.5.1 Требования к исходным кодам и языкам программирования 11](#_Toc95915148)

[4.5.2 Требования к программным средствам, используемым программой 11](#_Toc95915149)

[4.6 Требования к маркировке и упаковке 11](#_Toc95915150)

[4.7 Требования к транспортированию и хранению 11](#_Toc95915151)

[5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ 12](#_Toc95915152)

[5.1 Состав программной документации 12](#_Toc95915153)

[5.2 Специальные требования к программной документации 12](#_Toc95915154)

[6. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ 13](#_Toc95915155)

[7. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ 14](#_Toc95915156)

[7.1  Стадии разработки 14](#_Toc95915157)

[7.2 Сроки разработки и исполнители 15](#_Toc95915158)

[8. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ 16](#_Toc95915159)

[8.1 Виды испытаний 16](#_Toc95915160)

[8.2 Общие требования к приемке работы 16](#_Toc95915161)

[9. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 17](#_Toc95915162)

[ПРИЛОЖЕНИЕ 1 19](#_Toc95915163)

[ПРИЛОЖЕНИЕ 2 20](#_Toc95915166)

[ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 21](#_Toc95915169)

# АННОТАЦИЯ

Техническое задание – это основной документ, оговаривающий набор требований и порядок создания программного продукта, в соответствии с которым производится разработка программы, ее тестирование и приемка.

Настоящее Техническое задание на разработку «Разработка MVP сервиса синтеза речи на английском языке» содержит следующие разделы: «Введение», «Основание для разработки», «Назначение разработки», «Требования к программе», «Требования к программным документам», «Технико-экономические показатели», «Стадии и этапы разработки», «Порядок контроля и приемки» и приложения.

В разделе «Введение» указано наименование и краткая характеристика области применения программы.

В разделе «Основания для разработки» указан документ, на основании которого ведется разработка.

В разделе «Назначение разработки» указано функциональное и эксплуатационное назначение программного продукта.

Раздел «Требования к программе» содержит основные требования к функциональным характеристикам, к надежности, к условиям эксплуатации, к составу и параметрам технических средств, к информационной и программной совместимости, к маркировке и упаковке, к транспортировке и хранению, а также специальные требования.

Раздел «Требования к программным документам» содержит предварительный состав программной документации и специальные требования к ней.

Раздел «Технико-экономические показатели» содержит ориентировочную экономическую эффективность, предполагаемую годовую потребность, экономические преимущества разработки программы.

Раздел «Стадии и этапы разработки» содержит стадии разработки, этапы и содержание работ.

В разделе «Порядок контроля и приемки» указаны общие требования к приемке работы.

Настоящий документ разработан в соответствии с требованиями:

1)   ГОСТ 19.101-77 Виды программ и программных документов [10];

2)   ГОСТ 19.102-77 Стадии разработки [11];

3)   ГОСТ 19.103-77 Обозначения программ и программных документов [12];

4)   ГОСТ 19.104-78 Основные надписи [13];

5)   ГОСТ 19.105-78 Общие требования к программным документам [14];

6)   ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом [15];

7)   ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению [16].

Изменения к данному Техническому заданию оформляются согласно ГОСТ 19.603-78 [22], ГОСТ 19.604-78 [23].

ГОЛОССАРИЙ

|  |  |
| --- | --- |
| **Термин** | **Определение** |
| TTS (Text-To-Speech), синтез речи | Формирование речевого сигнала по текстовому представлению |
| MVP (Minimum Viable Product) | Самая ранняя версия продукта, у которой есть минимальный набор функций, достаточный для презентации публике и проверке на первых потребителях. |
| Микросервисная архитектура | Стиль разработки ПО, заключающийся в разбиении монолита системы на отдельные компоненты, которые представляют собой независимые сервисы. |
| Семиотический класс | Множество лексических единиц, обозначающих объекты одного типа и имеющих схожие правила написания и произношения. |
| Нормализация текста | Процесс разделения текста на лексические единицы и соотнесения каждой из них с определенным семиотическим классом. |
| NLP (Natural Language Processing) | Обработка естественного языка. |
| Фонема | Единица звукового строя языка. |
| Мел-спектрограмма | Спектрограмма, где частота выражена не в Гц, а в мелах. Переход к мелам осуществляется с помощью применения мел-фильтров к исходной спектрограмме. Мел-фильтры представляют из себя треугольные функции, равномерно распределенные на мел-шкале. (см. Приложение 2) |
| Вокодер (vocoder) | Устройство (или алгоритм), позволяющее синтезировать речь на основе минимальной информации, некоторого кода. |
| Акустическая модель | Функция, принимающая на вход признаки на небольшом участке акустического сигнала (фрейме) и выдающая распределение вероятностей различных фонем на этом фрейме. |

1. ВВЕДЕНИЕ
   1. Наименование программы

Наименование программы на русском языке: «Разработка MVP сервиса синтеза речи на английском языке».

Наименование программы на английском языке: «Development of an MVP service for speech synthesis in English».

* 1. Краткая характеристика области применения программы

Синтез речи - это процесс создания из текста на каком-то языке аудио, содержащее речь, которую носитель этого языка распознает как этот же самый текст. Синтез речи обычно используется в системах с голосовым управлением таким, как, например, голосовые помощники.

Во многих ситуациях можно было бы записать отдельные куски речи конкретного человека и склеивать их. Такой подход является жизнеспособным в системах с ограниченным количеством реплик. Например, навигаторах. Однако в общем случае могут потребоваться произвольные фразы, которые невозможно синтезировать таким способом. Данная использует NLP-подходы, что помогает решить описанные ограничения и превращать произвольный текст на английском языке в речь.

Данная программа является MVP проектом представляющим Telegram бота, позволяющего синтезировать аудио файл с речью на английском языке по введенному пользователем тексту.

1. ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ

Разработка ведется на основании учебного плана подготовки бакалавров по направлению 09.03.04 «Программная инженерия» и утвержденной академическим руководителем программы «Разработка MVP сервиса синтеза речи на английском языке».

1. **НАЗНАЧЕНИЕ** РАЗРАБОТКИ
   1. Функциональное назначение

Программа предоставляет возможность получать аудиофайлы на основе предоставленного текста на английском языке. Кроме того, программа предоставляет возможность получения аналитики по работе сервисов, отвечающих за нормализацию текста, представление нормализации текста в виде списка фонем, преобразование фонем в мел-спектрограмму, преобразование мел-спектрограммы в аудиофайл.

* 1. Эксплуатационное назначение

Программа является MVP сервиса синтеза речи на английском языке, позволяющим облегчить процесс преобразования текстовых данных в аудио-формат. Каждый пользователь данной программы может создавать аудиофайлы на основе произвольного текста, переданного программе, и просматривать аналитику, связанную с созданием этих файлов.

1. ТРЕБОВАНИЯ К ПРОГРАММЕ
   1. Требования к функциональным характеристикам

Программа состоит из двух основных компонент: клиентской и серверной частей, между которыми должно быть налажено взаимодействие.

* + 1. Требования к серверной части

Серверная часть программы имеет микросервисную архитектуру (см. Приложение 1 – Схема архитектуры сервисной части приложения). Программа содержит следующие связанные между собой сервисы:

1. Telegram Bot service

Управляет Telegram ботом (см. пункт 4.1.2) и обеспечивает взаимодействие бота с TTS Service:

* 1. Отправляет текст, введенный пользователем на обработку TTS Service;
  2. Получает обрабатывает и отправляет пользователю ответ полученный от TTS Service;

1. TTS Service

Является единой точкой входа в систему. Отвечает на з*апрос на вывод синтезированного аудио из переданного текста.* Полученный текст последовательно отправляется на обработку в Normalization Service, Grapheme2phoneme service, Acoustic service, Vocoder service (подробное взаимодействие описано в следующих пунктах). Полученный аудио файл (Формат файла json) отправляется Telegram Bot service

В случае, если запрос невозможно выполнить, передает сообщение об ошибке Telegram Bot service.

1. Normalization Service

Отвечает за нормализацию текста. Отвечает на запрос от TTS Service с нормализованным текстом в качестве параметра и возвращает:

1. *Нормализованный текст;*

Нормализует текст в соответствии с семиотическими классами, представленными в библиотеке NVIDIA NeMo Text Normalization. Возвращает json файл с нормализованным текстом.

1. В случае, если запрос невозможно выполнить, передает сообщение об ошибке TTS Service.
2. Grapheme2phoneme Service

Отвечает за представление нормализацию текста в виде списка фонем. Отвечает на запрос от TTS Service с введенным пользователем текстом в качестве параметра и возвращает:

1. *Текст из фонем;*
2. В случае, если запрос невозможно выполнить, передает сообщение об ошибке TTS Service.
3. Acoustic Service

Отвечает за преобразование фонем в мел-спектрограмму (см. Приложение 2). Отвечает на запрос от TTS Service с текстом из фонем в качестве параметра и возвращает:

1. *Мел-спектрограмму;*
2. В случае, если запрос невозможно выполнить, передает сообщение об ошибке TTS Service.
3. Vocoder Service

Отвечает за преобразование мел-спектрограммы в аудио файл. Отвечает на запрос от TTS Service с мел-спектрограммой в качестве параметра и возвращает:

* 1. *Аудио-файл.*
  2. В случае, если запрос невозможно выполнить, передает сообщение об ошибке TTS Service.

Все взаимодействие сервисов и передача информации осуществляются через по протоколу HTTP. Формат файлов для передачи – .json.

Более подробное описание функционирования каждого микросервиса представлено в индивидуальных ТЗ участников команды в соответствии с распределением работы над проектом.

* + 1. Требование к взаимодействию клиентской и серверной частей

Взаимодействие между клиентской и серверной частями должно осуществляться посредством HTTP-запросов. При получении GET-запроса от клиента, сервер должен ответить сообщением в формате JSON.

* + 1. Требования к клиентской части

Клиентская часть приложения представлена чат ботом на в мессенджере «Telegram».

Бот поддерживает 2 команды:

1. Ввод текстового сообщения на английском языке для перевода его в аудио файл;
   1. Формат выходного аудиофайла mp3
   2. Бот должен отправлять пользователю сообщение об ошибке, если текст не может быть преобразован в аудио.
   3. Требования к надежности
      1. Требования к обеспечению надежного (устойчивого) функционирования программы

Программа не должно аварийно завершаться при любом наборе входных. Программа должна обеспечивать проверку корректности входных данных.

* + 1. Время восстановления после отказа

В случае отказа работы серверной части и последующей недоступности Telegram бота, время восстановления не должно превышать одни рабочие сутки.

* + 1. Отказы из-за некорректных действий оператора

После запуска программы на сервере отказ программы вследствие некорректных действий оператора должен быть исключён. В том числе должна быть исключена возможность непреднамеренного выключения программы, не связанного с техническими неполадками сервера.

* 1. Условия эксплуатации
     1. Климатические условия эксплуатации

Требований к климатическим условиям эксплуатации не предъявляется.

* + 1. Требования к видам обслуживания

Обслуживание не требуется.

* + 1. Требования к численности и квалификации персонала

Не требует специального обслуживания. Требуемая классификация – пользователь, свободно владеющий английским языком, а также Интернетом.

* 1. Требования к составу и параметрам технических средств

Для надёжной работы программы требуется следующий состав технических средств. Для работы клиентского приложения:

Устройство, поддерживающее работу с Telegram.

Устойчивое интернет-соединение.

* 1. Требования к информационной и программной совместимости
     1. Требования к исходным кодам и языкам программирования

Программа реализована на языке программирования Python 3.6 или выше.

* + 1. Требования к программным средствам, используемым программой

Программа должна работать на операционной системе Linux.

Программа должна исползать NVIDIA® Tesla® V100 в Yandex Cloud для синтеза аудио.

* 1. Требования к маркировке и упаковке

Требования к маркировке и упаковке не предъявляются.

* 1. Требования к транспортированию и хранению

Специальные требования к транспортировке не предъявляются.

1. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ
   1. Состав программной документации

Техническое задание (ГОСТ 19.201-78) [16];

Программа и методика испытаний (ГОСТ 19.301-79) [17];

Текст программы (ГОСТ 19.401-78) [18];

Пояснительная записка (ГОСТ 19.404-79) [19];

Руководство оператора (ГОСТ 19.505-79) [20].

* 1. Специальные требования к программной документации

Документы к программе должны быть выполнены в соответствии с ГОСТ 19.106-78 [15] и ГОСТами к каждому виду документа (см. п. 5.1.);

Пояснительная записка должна быть загружена в систему Антиплагиат через LMS «НИУ ВШЭ».

Техническое задание и пояснительная записка, титульные листы других документов должны быть подписаны руководителем разработки и исполнителем

Документация и программа сдается в электронном виде в формате .pdf или .docx. в архиве формата .zip или .rar;

За три дня до защиты комиссии все материалы курсового проекта:

программная документация,

программный проект,

исполняемый файл,

отзыв руководителя

отчет системы Антиплагиат

должны быть загружены одним или несколькими архивами в проект дисциплины «Курсовой проект» в личном кабинете в информационной образовательной среде SmartLMS НИУ ВШЭ.

1. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

Обзор и сравнительный анализ источников и аналогов

|  |  |  |  |
| --- | --- | --- | --- |
| Сервис | Yandex SpeechKit | Google Cloud Text-To-Speech | Amazon’s Polly |
| Потоковая передача аудио в режиме реального времени | + | + | + |
| Выбор голоса из списка | + | + | + |
| Возможность использования кастомного голоса | - | + | - |
| Настройка аудио под конечное устройство для проигрывания | - | + | - |
| Настройка речевого выхода (скорость, произношение, громкость, высота голоса) | выбор из 3х интонаций | выбор высоты и скорости произношения | произношение, громкость, высота голоса, скорость |
| Поддержка SSML | + | + | + |

В рамках данной работы расчёт экономической эффективности не предусмотрен.

1. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

7.1  Стадии разработки

1. Техническое задание:

1.1. Этапы разработки

– обоснование необходимости разработки программы;

– постановка задачи;

– сбор исходных материалов;

– выбор и обоснование критериев эффективности и качества разрабатываемой; программы;

1.2. Разработка и утверждение технического задания

– определение требований к программе;

– определение стадий, этапов и сроков разработки программы и документации на неё;

– согласование и утверждение технического задания.

2. Эскизный проект:

2.1. Разработка эскизного проекта

– предварительная разработка структуры входных и выходных данных;

– уточнение методов решения задачи;

– разработка общего описания алгоритма решения задачи.

2.2. Утверждение эскизного проекта

– разработка пояснительной записки;

– согласование и утверждение эскизного проекта.

3. Технический проект:

3.1. Разработка технического проекта

– уточнение структуры входных и выходных данных;

– разработка алгоритма решения задачи;

– определение формы представления входных и выходных данных;

– разработка структуры программы;

– окончательное определение конфигурации технических средств.

3.2. Утверждение технического проекта

– разработка пояснительной записки;

– согласование и утверждение технического проекта.

4. Рабочий проект:

4.1. Разработка программы

– программирование и отладка программы.

– обучение моделей

4.2. Разработка программной документации

– разработка программных документов в соответствии с требованиями гост 19.101-77. [10]

4.3. Испытания программы

– разработка, согласование и утверждение порядка и методики испытаний;

– корректировка программы и программной документации по результатам испытаний.

7.2 Сроки разработки и исполнители

Разработка должна закончиться к 15 мая 2022 года.

|  |  |
| --- | --- |
| Исполнитель: | Часть проекта, за которую отвечает: |
| Д. И. Калинина | Исследование существующих датасетов на подготовительном этапе, Grapheme2phoneme Service, тестирование программного продукта. |
| Д. О. Соколова | Разработка Telegram бота (Telegram Bot service), Acoustic Service, разработка программной документации |
| А. Е. Ишутин | Разработка архитектуры программного продукта, TTS Service, Vocoder Service |
| М. Д. Молокова | Анализ существующих средств и реализаций на этапе планирования, Normalization Service  , разработка программной документации |

1. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ
   1. Виды испытаний

    Производится проверка корректной работы программы, а также проверка на соответствие требований технического задания. Проверка программы также на соответствие техническому заданию осуществляется заказчиком совместно с исполнителем согласно «Программе и методике испытаний» (ГОСТ 19.301-79) [17].:

- Перечень функций программы, выделенных в программе для испытаний, и перечень требований, которым должны соответствовать эти функции

- Перечень необходимой документации и требования к ней

- Методы испытаний и обработки информации

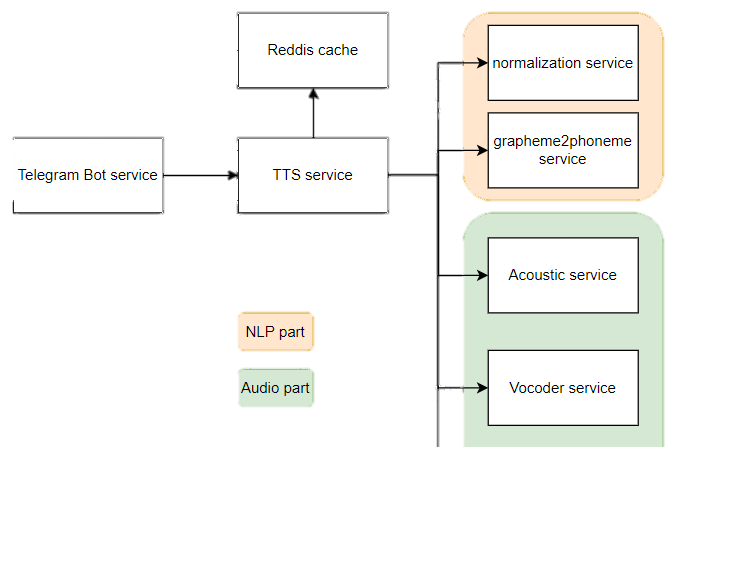
* 1. Общие требования к приемке работы

Прием программного продукта производится только при полной работоспособности Telegram бота, тестировании входных и выходных данных, соответствия технической документации, а также при наличии полной документации к программе.

1. **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**
   1. Eric Engelhart, Mahsa Elyasi, Gaurav Bharaj // Grapheme-to-Phoneme Transformer Model for Transfer Learning Dialects / / AI Foundation, USA [Электронный ресурс]: Technical Report, 2021 – Режим доступа: https://arxiv.org/pdf/2104.04091.pdf, свободный. (дата обращения: 17.02.22).
   2. Hao Zhang // Neural Models of Text Normalization for Speech Applications // Richard Sproat, Axel H. Ng, Felix Stahlberg, Xiaochang Peng, Kyle Gorman, Brian Roark [Электронный ресурс]: Technical Report, 2019 – Режим доступа: https://direct.mit.edu/coli/article/45/2/293/1637/Neural-Models-of-Text-Normalization-for-Speech, свободный. (дата обращения: 17.02.22).
   3. Jungil Kong, Jaehyeon Kim, Jaekyoung Bae // HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis [Электронный ресурс]: Technical Report, 2020 – Режим доступа: https://arxiv.org/pdf/2010.05646.pdf, свободный. (дата обращения: 17.02.22).
   4. Jungil Kong, Jaehyeon Kim, Jaekyoung Bae // HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis [Электронный ресурс] / GitHub. Режим доступа: https://github.com/jik876/hifi-gan, свободный. (дата обращения: 17.02.22)
   5. Kyubyong Park , Jongseok Kim // g2pE: A Simple Python Module for English Grapheme To Phoneme Conversion [Электронный ресурс] / GitHub. Режим доступа: https://github.com/Kyubyong/g2p, свободный. (дата обращения: 17.02.22)
   6. NVIDIA, FastPitch 1.1 for PyTorch [Электронный ресурс] / GitHub. Режим доступа: https://github.com/NVIDIA/DeepLearningExamples/tree/master/PyTorch/SpeechSynthesis/FastPitch, свободный. (дата обращения: 17.02.22)
   7. NVIDIA// NVIDIA NeMo User Guide [Электронный ресурс]: Documentation – Режим доступа: https://docs.nvidia.com/deeplearning/nemo/user-guide/docs/en/main/, свободный. (дата обращения: 17.02.22).
   8. Yi Ren // FastSpeech 2: Fast and High-Quality End-to-End Text to Speech // Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao , Tie-Yan Liu [Электронный ресурс]: Technical Report, 2021 – Режим доступа: https://arxiv.org/abs/2006.04558, свободный. (дата обращения: 17.02.22).
   9. Yuxuan Wang // Tacotron: Towards End-to-End Speech Synthesis// RJ Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J. Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, Quoc Le, Yannis Agiomyrgiannakis, Rob Clark, Rif A. Saurous [Электронный ресурс]: Technical Report, 2017 – Режим доступа: https://arxiv.org/abs/1703.10135, свободный. (дата обращения: 17.02.22).
   10. ГОСТ 19.101-77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
   11. ГОСТ 19.102-77 Стадии разработки. //Единая Система Программной Документации. – М.: ИПК Издательство стандартов, 2001.
   12. ГОСТ 19.103-77 Обозначения программ и программных документов. //Единая Система Программной Документации. – М.: ИПК Издательство стандартов, 2001.
   13. ГОСТ 19.104-78 Основные надписи. //Единая Система Программной Документации. – М.: ИПК Издательство стандартов, 2001.
   14. ГОСТ 19.105-78 Единая система программной документации. Общие требования к программным документам. //Единая Система Программной Документации. – М.: ИПК Издательство стандартов, 2001
   15. ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
   16. ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
   17. ГОСТ 19.301-79 Программа и методика испытаний. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
   18. ГОСТ 19.401-78 Текст программы. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
   19. ГОСТ 19.404-79 Пояснительная записка. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
   20. ГОСТ 19.505-79 Руководство оператора. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
   21. ГОСТ 19.602-78 Правила дублирования, учета и хранения программных документов, выполненных печатным способом. – М.: ИПК Издательство стандартов, 2001.
   22. ГОСТ 19.603-78 Общие правила внесения изменений. //Единая Система Программной Документации. – М.: ИПК Издательство стандартов, 2001
   23. ГОСТ 19.604-78 Единая система программной документации. Правила внесения изменений в программные документы, выполненные печатным способом. //Единая Система Программной Документации. – М.: ИПК Издательство стандартов, 2001

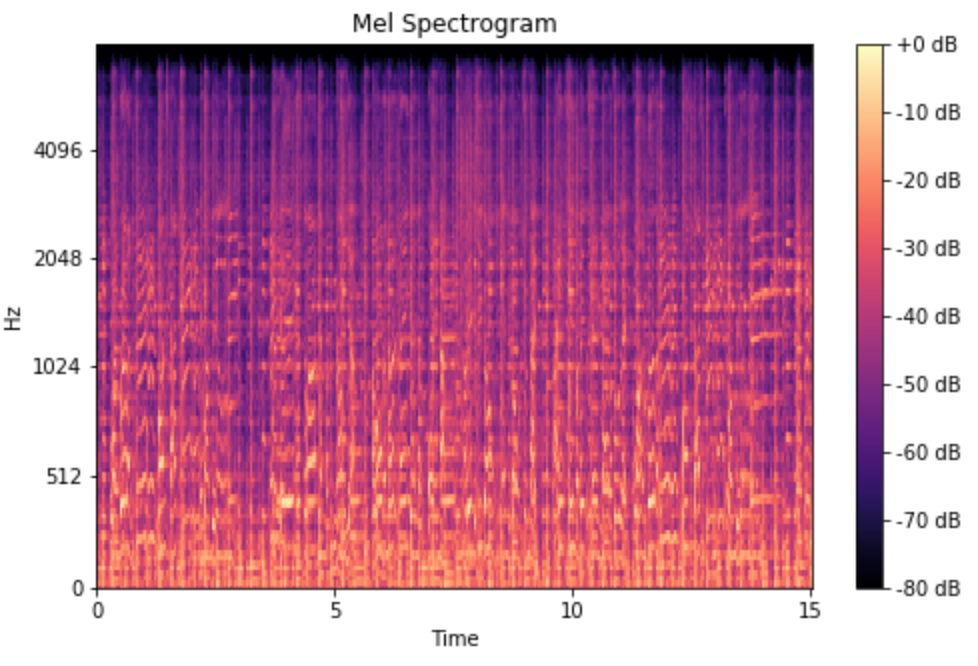
ПРИЛОЖЕНИЕ 1

Схема архитектуры сервисной части приложения



ПРИЛОЖЕНИЕ 2

Пример мел-спектрограммы.



# ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Номера листов (страниц) | | | | | Всего листов (страниц в докум.) | № документа | Входящий № сопроводительного докум. и дата | Подп. | Дата |
| Изм. | Измененных | Замененных | Новых | Аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |