**Оглавление**

[**Введение** 2](#_heading=h.30j0zll)

[**Используемые термины и условные обозначения** 3](#_heading=h.1fob9te)

[**1.**](#_heading=h.3znysh7) **Аналитическая часть** 6

[**1.1.**](#_heading=h.4d34og8) **Интервью с заказчиком** 6

[**1.2.**](#_heading=h.2s8eyo1) **Описание предметной области** 8

[**1.3.**](#_heading=h.17dp8vu) **Определение пользователей и вариантов использования системы** 10

[**1.4.**](#_heading=h.3rdcrjn) **Анализ требований к приложению c использованием CASE средств** 11

[**1.5.**](#_heading=h.26in1rg) **Описание архитектуры приложения** 24

[**1.6.**](#_heading=h.lnxbz9) **Организация данных** 26

[**1.7.**](#_heading=h.35nkun2) **Определение средств разработки** 29

[**2.**](#_heading=h.2et92p0) **Технологическая часть** 32

[**2.1**](#_heading=h.1ksv4uv) **Разработка функций приложения** 32

[**2.2**](#_heading=h.44sinio) **Разработка пользовательского интерфейса** 70

[**2.3**](#_heading=h.2jxsxqh) **Разработка тестовых наборов данных и тестовых сценариев** 73

[**3.**](#_heading=h.tyjcwt) **Экономическая часть** 76

[**Заключение** 80](#_heading=h.3dy6vkm)

[**Используемые источники информации** 81](#_heading=h.1t3h5sf)

**Введение**

В современном мире информационные технологии играют значительную роль в повседневной жизни людей. Они помогают упростить выполнение различных задач, сэкономить время и повысить эффективность работы. Одним из актуальных направлений развития информационных технологий является разработка текстовых редакторов с голосовыми помощниками.

Голосовые помощники становятся всё более популярными среди пользователей, так как они упрощают процесс взаимодействия с компьютером и повышают комфорт использования технологий. В связи с этим возникает необходимость создания текстовых редакторов, которые будут оснащены голосовыми помощниками для облегчения работы с текстом.

Целью данной дипломной работы является разработка текстового редактора с голосовым помощником, который будет обладать широким набором функций и возможностей для удобной работы с текстом. В процессе работы будут решены следующие задачи:

* разработка алгоритма распознавания речи для преобразования голосовых команд в текст;
* создание системы обработки запросов для выполнения действий на основе полученных команд;
* разработка синтезатора речи для ответа пользователю на его вопросы;
* реализация системы управления голосовым помощником для настройки и управления его работой.

Актуальность темы обусловлена растущим спросом на текстовые редакторы с голосовыми помощниками, которые могут существенно облегчить работу с текстом и повысить производительность труда.

Основанием для разработки проекта является приказ №305-ОД директора БПОУ «Омавиат» от «10» апреля 2023 г.

**Используемые термины и условные обозначения**

**Библиотека в программировании** — это собрание предварительно написанных функций, классов, модулей или других компонентов, которые облегчают разработку программного обеспечения. Библиотеки предоставляют готовые решения для распространенных задач и позволяют разработчикам повторно использовать код, что экономит время и усилия при создании новых приложений.

Библиотеки могут быть написаны на одном или нескольких языках программирования и предоставлять функциональность, которая относится к различным областям.

**PyAudio -** это популярная библиотека Python для работы с аудио, которая позволяет записывать и воспроизводить звук на различных платформах, включая Windows, macOS и Linux.

**Speech Recognition -** это популярная библиотека Python для распознавания речи, которая предоставляет простой и удобный интерфейс для работы с различными сервисами распознавания речи.

**Vosk -** это библиотека Python для распознавания речи с открытым исходным кодом, которая предоставляет кроссплатформенный и офлайн-ориентированный подход к распознаванию речи.

**Webbrowser** - это встроенная библиотека Python, которая предоставляет простой интерфейс для открытия веб-браузеров и отображения веб-страниц из вашего Python-приложения.

**Pyttsx3** - это библиотека Python для преобразования текста в речь (Text-to-Speech, TTS), которая предоставляет кроссплатформенный и офлайн-ориентированный подход к синтезу речи.

**Sys** - это встроенная библиотека Python, которая предоставляет доступ к некоторым переменным, используемым или поддерживаемым интерпретатором Python, а также к функциям, взаимодействующим с самим интерпретатором.

**Os** - это встроенная библиотека Python, которая предоставляет функции для взаимодействия с операционной системой. Она позволяет выполнять различные операции, связанные с файловой системой, процессами, окружением и другими системными ресурсами.

# **Аналитическая часть**

***Основные требования заказчика:***

1. Простой и интуитивно понятный интерфейс
2. Возможность голосового ввода и управления редактором
3. Функции форматирования текста, работы с файлами, проверки орфографии и т.д.
4. Кроссплатформенность и доступность на различных устройствах
5. Высокая надежность и безопасность работы с данными

**Описание предметной области:**

**https://vk.com/wall-156817212\_9312**

Развитие технологий распознавания голоса даёт пользователям новые возможности для взаимодействия с различными устройствами. Мы постепенно привыкаем к голосовым помощникам, но с помощью голоса можно не только узнавать погоду или включать музыку, но и набирать большие объёмы текста.  
  
Голосовой ввод текста – это функция, основанная на технологии распознавании речи. Программа голосового ввода воспринимает произнесенные слова и преобразует их в письменный текст.  
  
Применение специальных программ для речевого ввода данных позволяет частично освободить руки и не напрягать зрение, а также выполнять задачи быстрее. Это неоценимо для представителей многих профессий, в числе которых юристы, врачи, писатели, копирайтеры и прочие специалисты, работающие с набором текстов.  
  
Для использования голосового ввода нужен лишь микрофон. Владельцы ноутбуков могут использовать микрофон, встроенный в аппарат, а владельцы стационарных компьютеров –тот, что есть в веб-камерах, которые подключаются к ПК.  
  
Если вы планируете пользоваться голосовым вводом постоянно, то самое удобное решение – установить одну из программ, которая предоставляет такую возможность.

Преобразовать свой голос для облегчения работы с текстами

**Определение пользователей и вариантов использования системы:**

Основные пользователи: офисные работники, студенты, люди с ограниченными возможностями, авторы и редакторы.

***Варианты использования:***

1. **Создание, открытие, сохранение и печать текстовых документов:**

* Пользователь может создавать новый текстовый документ.
* Пользователь может открывать существующие текстовые документы из файловой системы.
* Пользователь может сохранять текстовые документы в различных форматах (например, .txt, .docx, .pdf) на локальном устройстве или в облачном хранилище.
* Пользователь может печатать текстовые документы на подключенном принтере.

1. **Форматирование текста (шрифт, размер, стиль, выравнивание и т.д.):**

* Пользователь может выбирать различные шрифты, размеры и стили текста.
* Пользователь может изменять цвет текста и фона.
* Пользователь может применять различные варианты выравнивания текста (по левому краю, по центру, по правому краю, по ширине).
* Пользователь может создавать маркированные и нумерованные списки.
* Пользователь может вставлять изображения, таблицы и другие объекты в текстовый документ.

1. **Проверка орфографии и грамматики:**

* Текстовый редактор должен иметь встроенную функцию проверки орфографии, которая будет выделять ошибки и предлагать варианты исправления.
* Текстовый редактор должен иметь функцию проверки грамматики, которая будет анализировать структуру предложений и выявлять возможные ошибки.
* Пользователь должен иметь возможность принимать или отклонять предложенные исправления.

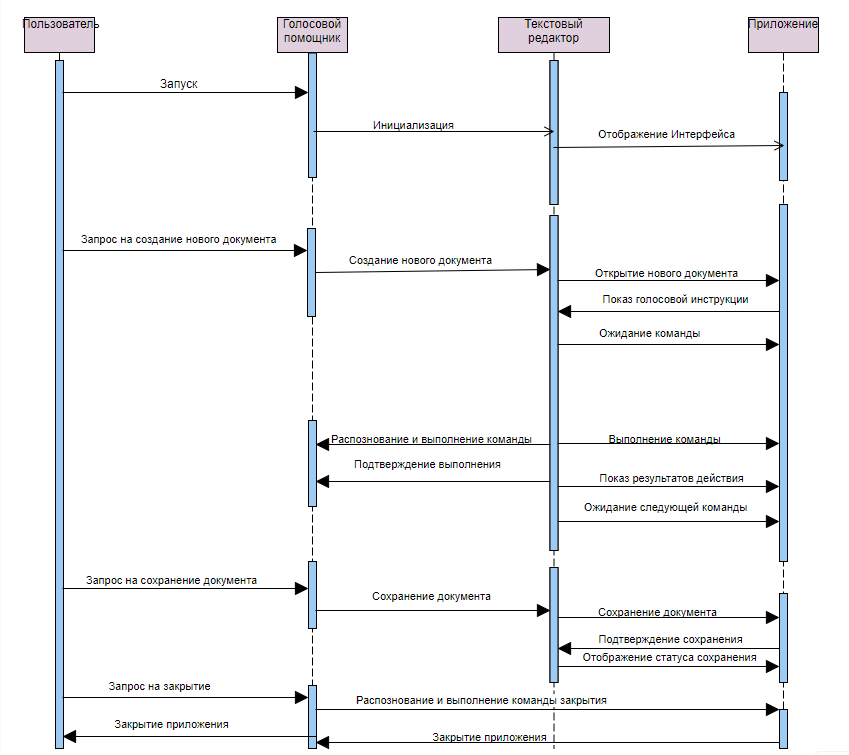
1. **Голосовой ввод и управление редактором:**

* Пользователь должен иметь возможность диктовать текст, используя встроенный голосовой ввод.
* Пользователь должен иметь возможность управлять основными функциями текстового редактора (открытие, сохранение, форматирование и т.д.) с помощью голосовых команд.
* Текстовый редактор должен иметь функцию распознавания речи, которая будет преобразовывать голосовой ввод в текст.

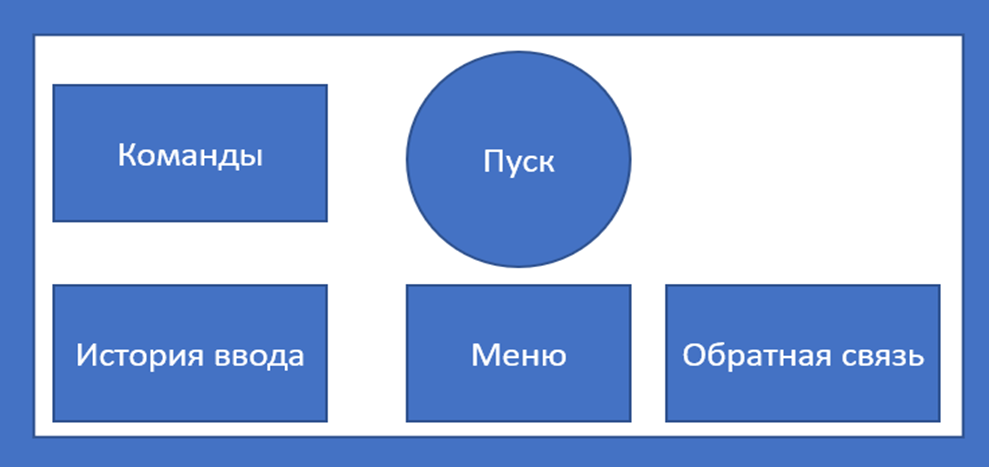
1. **Совместная работа над документами:**

* Пользователи должны иметь возможность совместно работать над одним текстовым документом в режиме реального времени.
* Пользователи должны видеть изменения, внесенные другими участниками, и иметь возможность оставлять комментарии.
* Текстовый редактор должен иметь функцию отслеживания изменений, чтобы пользователи могли просматривать историю правок.
* Пользователи должны иметь возможность предоставлять права доступа (чтение, редактирование, комментирование) другим участникам.

**Анализ требований к приложению с использованием Диаграммы последовательности:**



**Описание архитектуры приложения:**

1. Выбор архитектурного стиля (например, модель-представление-контроллер) 
2. Описание основных компонентов и их взаимодействия

* Пуск-запуск распознавания и обработки голоса
* Команды-Список доступных команд
* Меню - Настройки функционала приложения
* История ввода - Список распознанных команд и истории изменений
* Обратная связь - Передача данных в облако

1. Интеграция голосового помощника и текстового редактора

**Организация данных:**

1. **Структура и формат хранения текстовых документов:**

* Текстовые документы будут храниться в файловой системе в виде отдельных файлов.
* Для хранения текстовых документов можно использовать различные форматы, такие как:
* Простой текстовый формат (.txt) - для хранения чистого текста без форматирования.
* Форматы с разметкой, такие как Markdown (.md) или Rich Text Format (.rtf) - для хранения текста с базовым форматированием.
* Формат Microsoft Word (.docx) - для хранения текста с расширенным форматированием, включая стили, таблицы, изображения и другие объекты.
* Выбор формата будет зависеть от требований пользователей и необходимого уровня поддержки форматирования.

1. **Управление версиями и сохранение истории изменений:**

* Для управления версиями текстовых документов и сохранения истории изменений можно использовать систему контроля версий, такую как Git.
* При сохранении документа, приложение будет создавать новую версию файла в системе контроля версий, сохраняя информацию об авторе, дате и описании изменений.
* Пользователи смогут просматривать историю изменений, сравнивать версии документа и восстанавливать предыдущие состояния при необходимости.
* Система контроля версий также позволит реализовать функцию совместной работы над документами, где несколько пользователей смогут вносить изменения и сливать их в единый документ.

1. **Интеграция с облачными хранилищами данных:**

* Текстовый редактор будет интегрирован с популярными облачными хранилищами данных, такими как Google Drive, Dropbox или Microsoft OneDrive.
* Пользователи смогут сохранять и открывать текстовые документы непосредственно из облачных хранилищ, не требуя предварительной загрузки файлов на локальное устройство.
* Интеграция с облачными хранилищами позволит пользователям получить доступ к своим документам с любого устройства, а также обеспечит дополнительную защиту и резервное копирование данных.
* Для реализации интеграции с облачными хранилищами можно использовать соответствующие API, предоставляемые провайдерами, или библиотеки, упрощающие взаимодействие с ними.

**Определение средств разработки:**

1. **Выбор языка программирования Python:**

* Python является отличным выбором для разработки текстового редактора с голосовым помощником, так как он обладает следующими преимуществами:
* Простой и понятный синтаксис, что упрощает разработку и поддержку кода.
* Широкий спектр библиотек и фреймворков, которые могут быть использованы для различных задач.
* Кроссплатформенность, что позволяет разрабатывать приложение для различных операционных систем.
* Наличие развитого сообщества разработчиков, что обеспечивает доступность документации, примеров кода и решений для распространенных задач.

1. **Использование фреймворков и библиотек:**

* Для разработки пользовательского интерфейса можно использовать фреймворки, такие как PyQt, Tkinter или Kivy. Они предоставляют широкий набор виджетов и инструментов для создания интуитивно понятного и функционального интерфейса.
* Для обработки голоса и реализации голосового ввода можно использовать библиотеки, такие как SpeechRecognition, PyAudio или pyttsx3. Они позволяют распознавать речь, преобразовывать ее в текст и синтезировать речь.
* Для работы с файлами (открытие, сохранение, печать) можно использовать встроенные в Python модули, такие как os, shutil и subprocess.
* Для реализации совместной работы над документами можно использовать библиотеки для сетевого взаимодействия, такие как Flask или Django.

1. **Инструменты для проектирования, разработки, тестирования и развертывания приложения:**

* Для проектирования и моделирования пользовательского интерфейса можно использовать инструменты, такие как Figma, Adobe XD или Sketch.
* Для разработки и написания кода можно использовать интегрированные среды разработки (IDE), например, PyCharm, Visual Studio Code или Spyder.
* Для тестирования приложения можно использовать фреймворки, такие как unittest или pytest, которые позволяют автоматизировать модульные и интеграционные тесты.
* Для развертывания приложения можно использовать инструменты, такие как Docker или Ansible, которые позволяют упаковывать и развертывать приложение в различных средах.
* Для управления версиями кода можно использовать системы контроля версий, такие как Git, и хостинговые сервисы, такие как GitHub или GitLab.

# **Технологическая часть**

Разработка функций приложения

1.1. Функции работы с текстовыми документами:

1. Создание, открытие, сохранение и закрытие текстовых документов
2. Редактирование текста (вставка, удаление, копирование, вставка)
3. Форматирование текста (шрифт, размер, стиль, выравнивание и т.д.)
4. Вставка изображений, таблиц, ссылок и других объектов в документ
5. Проверка орфографии и грамматики

1.2. Функции голосового управления:

1. Распознавание голосовых команд для навигации по приложению
2. Преобразование речи в текст для ввода и редактирования документов
3. Управление функциями приложения с помощью голосовых команд
4. Настройка параметров голосового ввода (язык, чувствительность и т.д.)

1.3. Функции совместной работы:

1. Возможность совместного редактирования документов в реальном времени
2. Отслеживание и управление версиями документов
3. Комментирование и обсуждение документов

1.4. Функции безопасности и управления данными:

1. Шифрование и защита текстовых документов
2. Резервное копирование и восстановление данных
3. Управление правами доступа к документам
4. Разработка пользовательского интерфейса

2.1. Дизайн и макет интерфейса:

1. Создание интуитивно понятного и эргономичного пользовательского интерфейса
2. Использование современных подходов к дизайну (Material Design, Fluent Design и т.д.)
3. Обеспечение визуальной привлекательности и последовательности интерфейса

2.2. Реализация основных элементов интерфейса:

1. Меню, панели инструментов, панели состояния
2. Окна редактирования текста, просмотра документов
3. Диалоговые окна, всплывающие подсказки, уведомления
4. Интеграция голосового ввода и управления

2.3. Адаптивность и кроссплатформенность:

1. Обеспечение отзывчивого дизайна, подходящего для различных устройств и экранов
2. Реализация кроссплатформенного интерфейса, работающего на Windows, macOS и Linux
3. Разработка текстовых наборов данных и тестовых сценариев

3.1. Создание текстовых наборов данных:

1. Сбор и подготовка текстовых документов различных форматов и тематик
2. Формирование репрезентативной выборки текстов для тестирования и оценки

3.2. Разработка тестовых сценариев:

1. Определение ключевых функциональных и нефункциональных требований к приложению
2. Создание тестовых сценариев для проверки основных возможностей текстового редактора
3. Разработка тестовых сценариев для проверки голосового ввода и управления

3.3. Автоматизация тестирования:

1. Интеграция тестовых сценариев в систему непрерывной интеграции и развертывания
2. Использование фреймворков для автоматизации модульного, интеграционного и End-to-End тестирования
3. Обеспечение регулярного запуска тестов и мониторинга качества приложения

**Протокол тестирования**

Тестирование происходило поэтапно, все данные приведены в таблице 19

Таблица 19 – Тестирование функций приложения.

|  |  |  |  |
| --- | --- | --- | --- |
| **Метод тестирования** | **Входные данные** | **Ожидаемый результат** | **Выходные данные** |
| ТП1: start() стартовая команда | Не корректный ввод команды в элементы управления | Отсутствие ответа на команду | Отсутствие ответа на команду |
| ТП2: store\_managment() вывод меню каталога | Корректный ввод команды в элементы управления | Вывод каталога категорий и товаров, если база данных не является пустой | Вывод каталога категорий и товаров, если база данных не является пустой |
| ТП3: show\_product() вывод товара, информации о нем и вспомогательные навигационные кнопки | Корректный ввод команды в элементы управления | Вывод товара, а также дополнительные навигационные inline-кнопки | Вывод товара, а также дополнительные навигационные inline-кнопки |
| ТП4: show\_cart() вывод корзины товаров пользователя | Некорректный ввод команды в элементы управления | Отсутствие ответа на команду | Отсутствие ответа на команду |

# **Экономическая часть**

1. **Оценка проекта**

Оценка данного дипломного проекта произведена методом PERT.

Инженерный метод оценки трудоемкости проекта PERT (Program/Project Evaluation and Review Technique) был разработан в 1958 году в ходе проекта по созданию баллистических ракет морского базирования «Поларис». Входом для данного метода оценки служит список элементарных пакетов работ.

Перечень элементарных пакетов работ данного игрового приложения для проведения оценки:

* инициализация приложения (количество 1 шт.);
* разработка структуры базы данных (количество 11 шт.);
* разработка клавиатур (количество 49 шт.).

Диапазон неопределенности достаточно охарактеризовать тремя оценками:

* Mi – Наиболее вероятная оценка трудозатрат;
* Oi – Минимально возможные трудозатраты на реализацию пакета работ;
* Pi – Максимально возможные трудозатраты.

Выполним оценку для наилучших, наихудших и наиболее вероятных ситуаций. Среднюю трудоемкость для каждого отдельного пакета можно вычислить с использованием следующей формулы:

Ei = (Pi + 4Mi + Oi) / 6, (1)

где

Pi – Максимально возможные трудозатраты,

Mi – Наиболее вероятная оценка трудозатрат

Oi – Минимально возможные трудозатраты на реализацию пакета работ.

1. Инициализация приложения:

* Oi = 0,5
* Pi = 1
* Mi = 0,5

Средняя трудоемкость рассчитана по формуле (1):

Еi = (Pi + 4Mi + Oi)/6 = (1 + 40,5 + 0,5)/6 = 0,5 чел. час

1. Разработка базы данных:

* Oi = 3
* Pi = 10
* Mi = 6

Средняя трудоемкость рассчитана по формуле (1):

Еi = (Pi + 4Mi + Oi)/6 = (10 + 46 + 3)/6 = 6,1 чел. час

1. Разработка клавиатур:

* Oi = 16
* Pi = 52
* Mi = 32

Средняя трудоемкость рассчитана по формуле (1):

Еi = (Pi + 4Mi + Oi)/6 = (52 + 432 + 16) / 6 = 32,6 чел. час

В таблице 20 показаны оценки наиболее вероятного и ожидаемого случаев.

Таблица 20 – Тестирование функций приложения.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Компоненты | Лучший случай | Наиболее вероятный случай | Худший случай | Ожидаемый случай |
| Инициализация приложения | 0,5 | 0,5 | 1 | 0,5 |
| Разработка структуры базы данных | 3 | 6 | 10 | 6,1 |
| Разработка скриптов | 16 | 32 | 52 | 32,6 |
| **Итого:** | **19,5** | **38,5** | **63** | **39,2** |

Для расчета среднеквадратичного отклонения используется формула:

CKOi = (Pi - Oi)/6, (2)

где

Pi – Максимально возможные трудозатраты

Oi - Минимально возможные трудозатраты на реализацию пакета работ.

Для инициализации приложения, среднеквадратичное отклонение рассчитано по формуле (2).

CKOi = (Pi - Oi)/6 = (1-0,5)/6=0,08 чел. час

Для разработки структуры базы данных, среднеквадратичное отклонение рассчитано по формуле (2).

CKOi = (Pi - Oi)/6 = (10-3)/6=1,1 чел. час

1. Для разработки клавиатур, среднеквадратичное отклонение рассчитано по формуле (2).

CKOi = (Pi - Oi)/6 = (52-16)/6=6 чел. Час

Суммарная трудоемкость проекта может быть рассчитана по формуле:

Е = Σ Ei (3)

Е = Σ Ei = 10,5 + 116,1 + 4932,6 = 1665

Среднеквадратичное отклонение для оценки суммарной трудоемкости будет составлять:

(4)

где

CKO – среднеквадратичное отклонение

Среднеквадратичное отклонение для оценки суммарной трудоемкости рассчитана по формуле (4)

= 294 чел. час

Оценка суммарной трудоемкости проекта, которая с вероятностью 95% точности расчета, можно применить формулу:

(5)

где

CKO = Среднеквадратичное отклонение

E = Ei

= (1665+ 2 294) = 2253 чел. час

В месяц сотрудник будет работать по проекту, примерно 176 чел.\*час/мес.

Следовательно, трудоемкость проекта в человеко-месяцах составит, приблизительно 2253/176 ~ 12,8

Тогда, согласно формуле Б. Боэма, оптимальная продолжительность проекта составит:

5,8 месяцев.

Средняя численность команды – 12,8/5,8 = 2 человека.

**Заключение**

В ходе выполнения дипломной работы была разработана система текстового редактора с голосовым помощником, которая позволяет пользователям создавать, редактировать и форматировать текстовые документы с помощью голосовых команд.Проведенный анализ предметной области, определение пользователей и вариантов использования системы, а также анализ требований с использованием Диограммы последовательностей позволили сформировать четкое понимание задач и функциональности разрабатываемого приложения. Описание архитектуры приложения, организация данных и выбор средств разработки обеспечили надежную и эффективную реализацию системы. Интеграция голосового помощника и текстового редактора в единое приложение позволила создать удобный и интуитивно понятный интерфейс для пользователей. Использование библиотек Python, таких как PyAudio, Speech Recognition, Vosk и другие, значительно упростило процесс разработки и обеспечило кроссплатформенность и надежность работы приложения. В результате проделанной работы был создан текстовый редактор с голосовым помощником, который обладает следующими ключевыми характеристиками:

1. Простой и интуитивно понятный интерфейс с возможностью голосового управления.
2. Функции форматирования текста, работы с файлами, проверки орфографии и другие необходимые инструменты для создания и редактирования текстовых документов.
3. Кроссплатформенность и доступность на различных устройствах.
4. Высокая надежность и безопасность работы с данными.

Разработанное приложение может быть использовано в различных сферах деятельности, где требуется эффективная работа с текстовыми документами, особенно для людей с ограниченными возможностями или в ситуациях, когда руки заняты. Дальнейшее развитие системы может включать в себя расширение функциональности, улучшение алгоритмов распознавания речи, интеграцию с облачными сервисами и другие усовершенствования, направленные на повышение удобства и эффективности работы пользователей.

**Используемые источники информации**