Software Requirement Specification (SRS)

Project Name: "TapVision" – Combines "Tap" (interaction method) with "Vision" (for accessibility).

# Table Of Contents:

* 1. Introduction
  2. Purpose
  3. Intended Audience and Reading Suggestion
  4. Scope
  5. Reference
  6. Overall Description
  7. Project Perspective
  8. Project Functionalities
  9. Operating Environment
  10. Requirements
  11. Hardware Requirement
  12. Software Requirement
  13. System Features
  14. Use case diagram
  15. ER diagram
  16. Other Nonfunctional Requirements
  17. Performance Requirements for Smart Banking System (SBS)
  18. Safety Requirements for Smart Banking System (SBS)
  19. Security Requirements for Smart Banking System (SBS)
  20. Software Quality Attributes for Smart Banking System (SBS)
  21. Business Rules for Smart Banking System (SBS)

# Introduction:

* 1. Purpose:

The Text-to-Speech (TTS) system is designed to convert written text into natural-sounding speech. The primary goal of this system is to provide an accessible, user-friendly, and efficient solution for applications such as assistive technologies, automated customer service, and voice-enabled devices. This document outlines the functional and non-functional requirements of the TTS system, ensuring clarity in its purpose, features, and scope.

* 1. Intended Audience and Reading Suggestions:

This document is intended for the following stakeholders involved in the

development and evaluation of the **Project**. Each audience group has a specific role in understanding and utilizing the information provided:

This document is intended for the following stakeholders:

* **Project Guide/Advisor:** To evaluate technical feasibility and alignment with project goals.
* **Evaluation Committee:** To assess the project's scope, objectives, and deliverables.
* **Team Members:** To ensure clarity on individual responsibilities and technical implementation.
* **End Users:** To understand how the system will address their needs and improve workflows.

**Reading Suggestions:**

* **Project Guide/Advisor:** Focus on functional and non-functional requirements, architecture, and methodology.
* **Evaluation Committee:** Review problem statement, project goals, and implementation details.
* **Team Members:** Thoroughly review all sections for alignment with project goals.
* **End Users:** Refer to functional requirements and system features to understand benefits.
  1. Scope

The TTS system will handle functionalities such as:

* **Text Input:** Accept text from users for conversion to speech.
* **Speech Synthesis:** Convert text into natural-sounding speech.
* **Voice Customization:** Allow users to select different languages, and speech parameters.
* **File Support:** Extract text from PDF, DOCX, and TXT files for processing.
* **Advanced Features:** Summarization, translation, and sentiment analysis of text.
  1. Reference
* UI Reference: [Streamlit Documentation](https://docs.streamlit.io/" \t "_blank)
* Inspirational References: [Google TTS](https://cloud.google.com/text-to-speech)
* Designing Reference: [ChatGPT](https://chatgpt.com/" \t "_blank)
* Python Libraries: [pyttsx3](https://pyttsx3.readthedocs.io/), [gTTS](https://gtts.readthedocs.io/" \t "_blank)
* NLP Models: [Hugging Face Transformers](https://huggingface.co/transformers/)

# Overall Description

* 1. Project Perspective

The Text-to-Speech (TTS) system is designed to provide a robust and scalable platform for converting written text into natural-sounding speech. The system is intended to serve as a versatile tool for a wide range of applications, including assistive technologies, automated customer service, voice-enabled devices, and human-computer interaction. Below is a detailed breakdown of the project perspective:

**1. System Context:**

The TTS system acts as a bridge between written text and audible speech, enabling users to interact with digital content in a more accessible and intuitive way.

It provides a unified interface for users to input text, select voice parameters and receive high-quality speech output.

The system is designed to be user-friendly, efficient, and scalable, making it suitable for both individual users and large-scale deployments.

**2. Relation to Existing Systems:**

The TTS system integrates seamlessly with existing technologies and platforms, including:

* **Web Applications:** The system can be embedded into websites or web applications to provide voice-enabled features.
* **Mobile Applications:** It can be integrated into mobile apps for on-the-go text-to-speech conversion.
* **Assistive Technologies:** The system can be used in tools designed for visually impaired individuals, such as screen readers.
* **Customer Service Platforms:** It can be integrated into chatbots or automated customer service systems to provide voice responses.
* **IoT Devices:** The system can be integrated with smart home devices (e.g., Amazon Echo, Google Home) for voice-enabled interactions.

**3. Technical Dependencies:**

The TTS system relies on modern technologies toensure performance, reliability, and scalability:

* **Backend:** Built using Python and frameworks for scalable and secure data handling.
* **Frontend:** A responsive web interface using Streamlit for ease of use and interactivity.
* **Third-party APIs:** Integration with APIs like Google TTS, Hugging Face Transformers, and OpenAI for advanced NLP and speech synthesis capabilities.

**4. Interfaces:**

The system interacts with various interfaces to provide a seamless user experience:

* **User Portal:** For text input, voice selection, and speech output.
* **Admin Dashboard:** For managing system configurations, user roles, and permissions.
* **APIs:** For integration with external systems, such as payment gateways, customer service platforms, or IoT devices.

**5. Constraints:**

The TTS project is subject to the following constraints:

* **Compliance:** The system must comply with data privacy regulations (e.g., GDPR, CCPA) and industry standards for secure data handling.
* **Hardware and Network Requirements:** The system requires stable internet connectivity and adequate hardware resources for optimal performance.
* **Project Timeline and Budget:** The development and deployment of the system are constrained by the project timeline and budget allocated by stakeholders.

**6. Assumptions:**

The following assumptions are made for the TTS system:

* Users have access to stable internet connectivity for cloud-based services or API calls.
* Users are familiar with basic text input and voice selection features.
* Third-party integrations (e.g., Google TTS, Hugging Face) will be available and functional during the project lifecycle.
* The organization will provide training to users for effective system utilization.

**7. Future Enhancements:**

The TTS system is designed with scalability and extensibility in mind, allowing for future enhancements such as:

* **Multilingual Support:** Adding support for more languages and regional accents to cater to a global audience.
* **Advanced Voice Customization:** Providing options for pitch, speed, and tone adjustment to create more personalized voices.
* **Real-Time Speech Generation:** Implementing real-time speech synthesis for dynamic content, such as live news or streaming text.
* **Integration with IoT Devices:** Enabling TTS functionality with smart home devices or assistive technologies for enhanced accessibility.
* **Emotional Intelligence:** Incorporating sentiment analysis to make speech delivery reflect emotions and context for a more natural sound.
* **Improved Accuracy with AI:** Enhancing voice recognition and pronunciation using advanced deep learning models.
* **Cross-Platform Support:** Developing mobile or desktop applications for a more versatile and portable solution.
* **Custom Voice Models:** Allowing users to create personalized voices using advanced deep learning techniques.
  1. Project Functionalities

The TTS system provides the following key functionalities:

* **Text Input and Processing:**
  + Accept text input from users via a user-friendly interface.
  + Extract text from various file formats (e.g., PDF, DOCX, TXT) for processing.
* **Speech Synthesis:**
  + Convert text into natural-sounding speech using advanced NLP techniques.
  + Support for multiple languages and voice customization.
  + **Advanced**. **Features:**
  + Summarization: Summarize long texts using T5 models for quick understanding.
  + Translation: Translate text into multiple languages using MarianMT models.
  + Sentiment Analysis: Analyze the sentiment of text for emotional context.
* **User Interface:**
  + Provide an intuitive interface for text input and speech output.
  + Support voice commands for hands-free operation.
  1. Operating Environment
* **Server:**
  + **OS:** Windows, macOS, or Linux.
  + **Languages:** Python
  + **Tools:** Streamlit, VS Code, PyCharm.
* **Client:**
  + **Platform:** Any modern web browser.
  + **Internet Connection:** Stable broadband connection.

# Requirements

* 1. Hardware Requirements

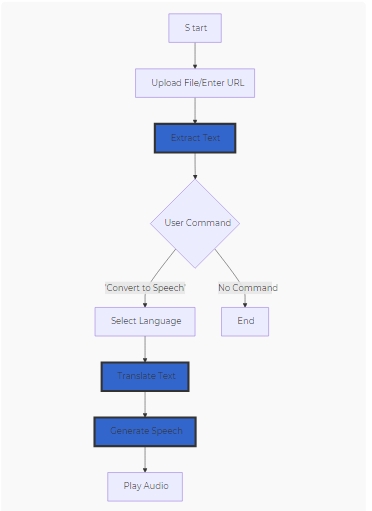
|  |  |  |
| --- | --- | --- |
| **No.** | **Components** | **Minimum Requirements** |
| 1. | Processor | Intel i7/i9 or AMD Ryzen 7/9 |
| 2. | RAM | 16GB (32GB recommended) |
| 3. | Disk Space | 500GB SSD |
| 4. | GPU | NVIDIA RTX 30/40 series (8- 16GB VRAM) |
| 5. | Audio Hardware | Professional sound card |

* 1. Software Requirements

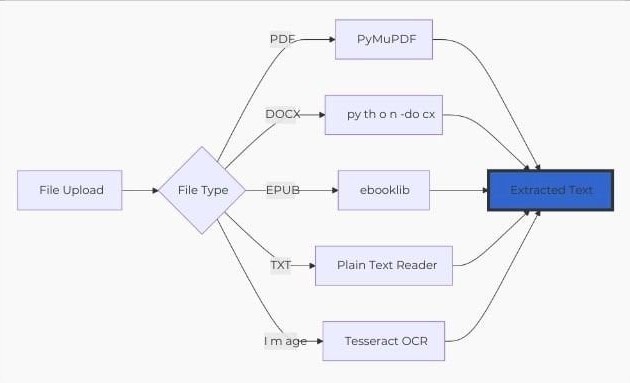
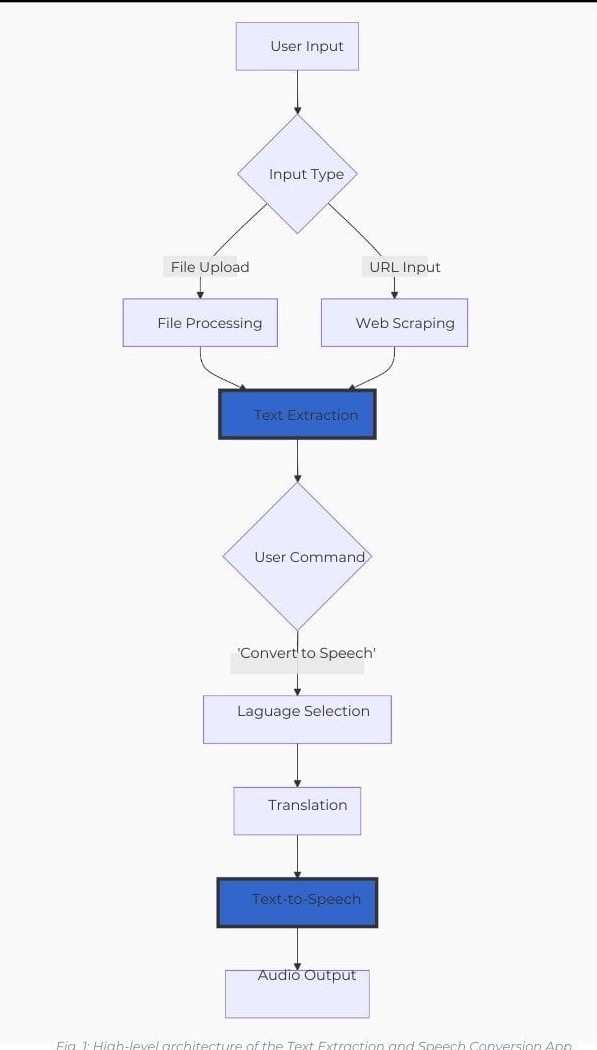
|  |  |  |
| --- | --- | --- |
| **No.** | **Components** | **Requirements** |
| 1. | Operating System | Windows, macOS, or Linux |
| 2. | Browser | Chrome, Firefox, Edge (latest versions recommended) |
| 3. | Internet Connection | Stable broadband connection |

# System Features:

* 1. Use Case Diagram



* 1. ER Diagram:



# Other Nonfunctional Requirement

Nonfunctional requirements define the quality attributes, constraints, and performance standards of the Text-to-Speech (TTS) system. These requirements ensure that the system is reliable, secure, scalable, and user-friendly. Below are the detailed nonfunctional requirements for the TTS system:

**5.1 Performance Requirements:**

* **Response Time:** The system should process text and generate speech within 3 seconds for optimal user experience.
* **Concurrency:** The system should support 50+ concurrent users without performance degradation.
* **Real-Time Processing:** The system should provide real-time speech synthesis for dynamic content (e.g., live text or streaming data).
* **Scalability:** The system should be scalable to handle increasing user loads, with the ability to support 1000+ users in the future.

**5.2 Safety Requirements:**

* **Data Safety:** All user data, including text input and speech output, should be securely stored and transmitted using HTTPS encryption.
* **Backup and Recovery:** Regular backups of user data and system configurations should be performed to prevent data loss in case of system failure.
* **Error Handling:** The system should provide clear error messages and recovery options in case of failures (e.g., invalid text input, API errors).

**5.3 Security Requirements:**

* **Data Encryption:** Sensitive data (e.g., user credentials, transaction logs) should be encrypted using AES-256 encryption.
* **Secure APIs:** All third-party API integrations (e.g., Google TTS, Hugging Face) should use secure communication protocols (e.g., HTTPS, OAuth).
* **Regular Security Audits:** Conduct regular security audits and penetration testing to identify and mitigate vulnerabilities.

**5.4 Software Quality Attributes:**

* **Usability:** The system should have a simple and intuitive interface that is easy to navigate for users of all skill levels.
* **Reliability:** The system should have minimal downtime and provide consistent performance under normal operating conditions.
* **Maintainability:** The system should be designed with modular code to facilitate easy updates, bug fixes, and feature enhancements.
* **Portability:** The system should be compatible with multiple platforms (e.g., Windows, macOS, Linux) and browsers (e.g., Chrome, Firefox, Edge).
* **Accessibility:** The system should be accessible to users with disabilities, including support for screen readers and keyboard navigation.

**5.5 Business Rules:**

* **Transaction Limits:** Transactions or operations above a set limit (e.g., large text inputs) may require admin approval.
* **Notifications:** Users should receive notifications for key activities, such as successful login, speech generation, and system updates.
* **Data Privacy:** The system must comply with data privacy regulations (e.g., GDPR, CCPA) and ensure that user data is not shared with third parties without consent.