
Software Requirements Specification

for

VoiceRad Reporter

Version 2.0 approved

Prepared by: Ertuğrul KÖKSAL

Fatih ATALAY

Muaz ASILKAN

Konya Food and Agriculture University

Computer Engineering Department

01.07.2024

Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction.....	1
1.1 Purpose.....	1
1.2 Document Conventions.....	1
1.3 Intended Audience and Reading Suggestions	2
1.4 Product Scope	3
1.5 References.....	3
2. Overall Description	4
2.1 Product Perspective.....	4
2.2 Product Functions	4
2.3 User Classes and Characteristics	4
2.4 Operating Environment.....	5
2.5 Design and Implementation Constraints	5
2.6 User Documentation	6
2.7 Assumptions and Dependencies	6
3. External Interface Requirements	6
3.1 User Interfaces	6
3.2 Hardware Interfaces	8
3.3 Software Interfaces	8
3.4 Communications Interfaces	8
4. System Features	9
4.1 System Feature 1.....	9
4.2 System Feature 2.....	10
4.3 System Feature 3.....	11
5. Other Nonfunctional Requirements.....	12
5.1 Performance Requirements	12
5.2 Safety Requirements	12
5.3 Security Requirements	13
5.4 Software Quality Attributes	13
5.5 Business Rules	14
6. Other Requirements	14

Revision History

Name	Date	Reason For Changes	Version
VoiceRad Reporter	01.10.2023	First prototype	Version 1.0
VoiceRad Reporter	01.07.2024	Beta Version	Version 2.0

1. Introduction

1.1 Purpose

This document outlines the software requirements for VoiceRad Reporter version 2.0. VoiceRad Reporter is an innovative tool designed to automate the transcription of radiology voice recordings into structured reports. The document describes the system's functional and non-functional requirements, capabilities, interfaces, and performance standards.

VoiceRad Reporter utilizes advanced natural language processing (NLP) and artificial intelligence (AI) technologies to improve radiologists' efficiency. The scope of this Software Requirements Specification (SRS) includes user interface design, security protocols, customization options, and integration with electronic health record (EHR) systems, in addition to core transcription functionalities.

The SRS provides a detailed framework covering all aspects of the system, ensuring comprehensive documentation of the entire system, including subsystems and components. This document aims to guide developers, stakeholders, and users, ensuring all requirements are met to deliver a high-quality product. It highlights the importance of Speech-to-Text technology in generating effective radiology reports and is part of the broader healthcare system. Future versions of this SRS will focus on enhancing the system with additional NLP capabilities.

1.2 Document Conventions

Terminology:

- "User": Person who accesses the platform and provides voice data.
- "Artificial Intelligence (AI)": Artificial intelligence model that processes audio data and produces text output.
- "Reports": Reports created in a specific template and presented to the user.
- "Website": The platform where the user can input audio and view reports.

Documentation Format and Structure:

- Format and Tools: Documentation will be prepared using Microsoft Word.
- Headings and Subheadings: Headings and subheadings will follow a hierarchical format (e.g., H1, H2, H3).
- Text Formatting: Times New Roman, 12 font and bold text where appropriate.
- Images and Graphics: Images and graphics will be added with captions and will be formatted consistently.

References and Links:

- Quotations and references to external sources will be provided in the text under the References heading.

Documentation Sections and Content:

- Sections: The document will include sections such as Introduction, User Guide, API Documentation, and System Requirements.
- Content Organization: Each section will contain relevant information organized logically to facilitate understanding.

Language and Style Guide:

- Language Style: The documentation will use a formal and technical tone.
- Grammar and Spelling: Adhere to standard grammar and spelling rules.

Update and Revision Policy:

- Updating Documentation: Procedures for updating and revising documentation will be established.
- Version Control: Maintain version control to track updates and revisions.

1.3 Intended Audience and Reading Suggestions

This document is intended for developers, project managers, marketing staff, users, testers, and documentation writers involved in the VoiceRad Reporter project. It outlines the software

requirements, capabilities, interfaces, and performance standards. All readers should start with the Introduction for an overview and the Overall Description for context. Developers and testers should focus on Specific Requirements and System Features. Project managers should review the Introduction, Overall Description, and Non-functional Requirements. Marketing staff should read the Introduction and System Features. Users should focus on the System Features, and documentation writers should review all sections for comprehensive information.

1.4 Product Scope

VoiceRad Reporter is an advanced radiology voice transcription software designed to automate the conversion of voice recordings into structured radiology reports. Its goal is to improve the efficiency and accuracy of radiologists' documentation processes by leveraging artificial intelligence (AI) and natural language processing (NLP) technologies. This software aims to reduce time spent on manual transcription, minimize errors and streamline workflow in radiology departments. VoiceRad Reporter is a website that supports business strategies focused on innovation and quality patient care, aligned with corporate goals to improve healthcare delivery and operational efficiency.

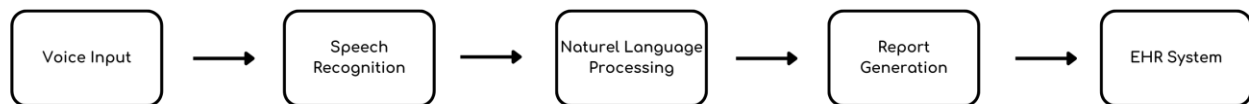
1.5 References

- <https://pypi.org/project/SpeechRecognition/>
- <https://www.abbadox.com/blog/impact-voice-recognition-radiology-report-accuracy>
- T.C. ANADOLU ÜNİVERSİTESİ YAYINI NO: 2525, AÇIKÖĞRETİM FAKÜLTESİ YAYINI NO: 1496 (Radyoloji Terminolojisi)
- Voice Recognition Software: Effect on Radiology Report Turnaround Time at an Academic Medical Center by American Journal of Roentgenology 195(1):194-7
- <https://insightsimaging.springeropen.com/articles/10.1186/s13244-023-01392-y>
- <https://radiopaedia.org/articles/natural-language-processing>

2. Overall Description

2.1 Product Perspective

VoiceRad Reporter is a new, self-contained product designed to automate the transcription of radiology voice recordings into structured reports. It is not a follow-on member of a product family nor a replacement for existing systems, but a novel solution aimed at improving the efficiency and accuracy of radiology reporting. The product integrates with existing electronic health record (EHR) systems to streamline workflows in radiology departments. A simple diagram showing the main components and interfaces of the VoiceRad Reporter system is shown in img2.0.



img2.0

2.2 Product Functions

VoiceRad Reporter performs several major functions, including:

- Voice Recording Upload: Allows users to upload voice recordings.
- Speech Recognition: Converts voice recordings into text using AI.
- Natural Language Processing (NLP): Processes text to extract relevant medical information.
- Report Generation: Automatically generates structured radiology reports.
- Review and Edit: Enables users to review and edit the generated reports.
- Integration with EHR: Integrates the final reports into EHR systems with APIs for easy access and storage.

2.3 User Classes and Characteristics

VoiceRad Reporter is designed for various user classes, each with distinct characteristics:

- Radiologists: Primary users who frequently use the system for dictating and reviewing reports. They require high accuracy and seamless integration with EHR systems.

- Medical Technicians: Occasionally use the system to assist radiologists in managing voice recordings and reports. They need an intuitive interface.
- IT Staff: Manage the technical aspects of the system, ensuring it runs smoothly and integrates with other hospital systems.
- Administrators: Oversee the usage and performance of the system, requiring access to usage reports and analytics.

2.4 Operating Environment

VoiceRad Reporter operates in a hospital or clinical environment and requires the following:

- Hardware: Compatible with standard hospital computer systems.
- Operating System: Windows 10 or later, macOS 10.14 or later.
- Other Software: Must integrate with existing EHR systems, such as Epic, Teleradyoloji or Cerner.
- Network: Requires a secure and stable internet connection for cloud-based processing and data storage.

2.5 Design and Implementation Constraints

The development of VoiceRad Reporter is subject to several constraints:

- Regulatory Compliance: Must comply with healthcare regulations such as HIPAA and GDPR.
- Hardware Limitations: Must operate efficiently on existing hospital hardware without requiring significant upgrades.
- Integration: Must integrate seamlessly with EHR systems and other hospital software.
- Security: Must implement robust security measures to protect sensitive patient data.
- Technology: Uses specific technologies like AI and NLP, which dictate certain design choices

2.6 User Documentation

The following user documentation will be provided:

- User Manual: Comprehensive guide for using VoiceRad Reporter, available in PDF and online formats.
- Online Help: Context-sensitive help within the application.
- Tutorials: Step-by-step guides and video tutorials for key features.

2.7 Assumptions and Dependencies

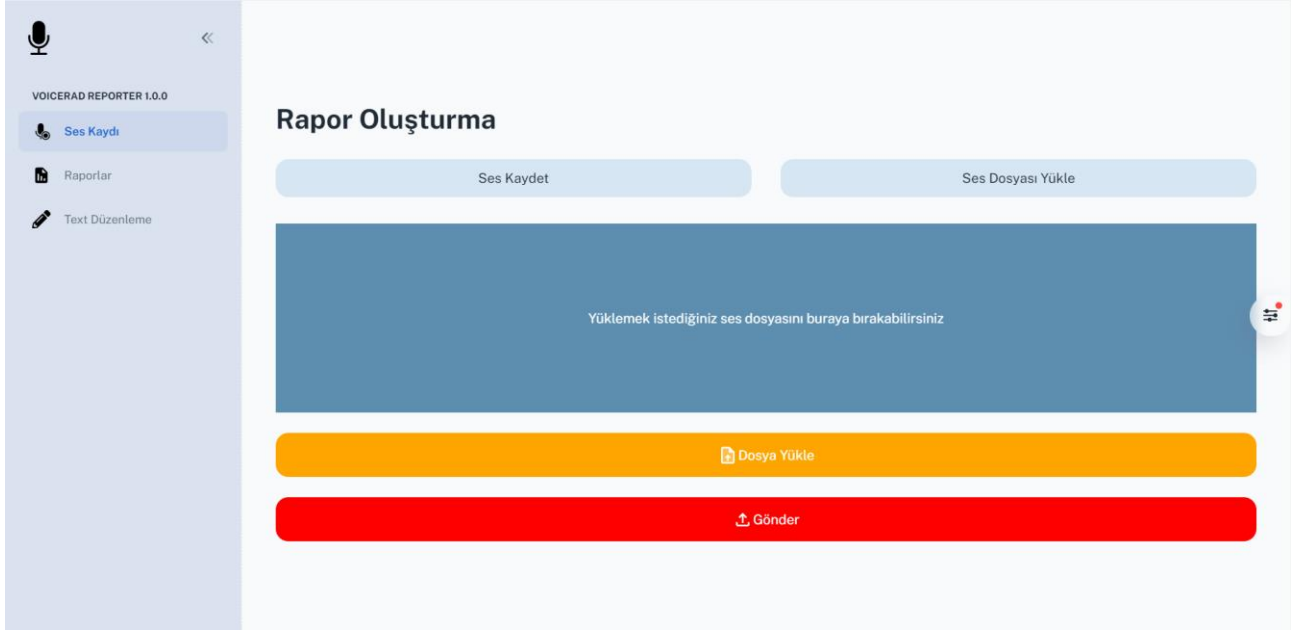
VoiceRad Reporter's development and deployment are based on several assumptions and dependencies:

- Third-Party Services: Relies on third-party AI and NLP services for speech recognition.
- Stable Internet Connection: Assumes the availability of a stable internet connection for cloud-based functionalities.
- EHR Compatibility: Assumes compatibility with major EHR systems like Epic, Teleradyoloji and Cerner.
- Regulatory Changes: Assumes no significant changes in healthcare regulations that could impact system functionality.

3. External Interface Requirements

3.1 User Interfaces

We aimed to achieve a successful result in terms of UI/UX with a simple and plain design in our front end that we created with React. There are three buttons on the left panel, including Voice Recording, Report and Account Information. Voice can be easily converted into a report with the voice recording and file upload sections in the voice recording section, and in the Report section, you can search among the reports with the advanced filtering system in the future as desired. In addition, editing and sharing reports can be easily done. Images of the design are below.



<VoiceRecordPage>img3.0



<VoiceToText>img3.1



<CreatingReportAndReturnPdfURL>img3.2

3.2 Hardware Interfaces

VoiceRad Reporter interfaces with various hardware components essential for its operation. Key characteristics include:

- **Supported Devices:** The software supports desktop computers, laptops, and mobile devices (iOS and Android).
- **Data and Control Interactions:** Interaction between the software and hardware involves audio data input from microphones and output to speakers or headphones.
- **Communication Protocols:** USB, Bluetooth, and standard audio input/output protocols are supported for hardware communication.

3.3 Software Interfaces

VoiceRad Reporter integrates with several software components to provide a seamless user experience. Key aspects include:

- **Operating Systems:** The software is compatible with Windows 10 or later, and macOS 10.14 or later.
- **APIs and Libraries:** Utilizes APIs for speech recognition (e.g., Whisper API) and NLP services. Specific libraries and SDKs are used for processing and data handling.
- **Data Exchange:** Data items include audio files, transcription text, and structured report data. These are exchanged using secure protocols and are stored in databases.
- **Application Protocols:** Detailed API documentation will specify the protocols for application programming interfaces.

3.4 Communications Interfaces

The communications interfaces for VoiceRad Reporter cover several key requirements:

- **Network Communications:** Supports HTTP/HTTPS protocols for secure web communication. Data transfers between the client and server are encrypted.
- **Email:** Configured to send notifications and reports to users via secure email protocols (SMTP/IMAP).

- **Web Browser Compatibility:** The platform is accessible via modern web browsers such as Chrome, Firefox, Safari, and Edge.
- **Security and Encryption:** Implements TLS/SSL for secure data transmission and end-to-end encryption for sensitive patient data.
- **Data Transfer Rates:** Optimized for fast and reliable data transfer, ensuring efficient handling of large audio files and rapid report generation.

4. System Features

4.1 Voice Recording Upload

4.1.1 Description and Priority

Description: This feature allows users to upload voice recordings into the system for transcription. Users can either upload pre-recorded audio files or record audio directly within the application.

Priority: High

4.1.2 Stimulus/Response Sequences

- Stimulus: User clicks on the "Upload" button.
 - Response: System opens a file dialog for the user to select an audio file.
- Stimulus: User selects an audio file and clicks "Open."
 - Response: System uploads the audio file and confirms the upload.
- Stimulus: User clicks on "Record" button to start a new recording.
 - Response: System starts recording audio and displays a recording timer.
- Stimulus: User clicks on "Stop" button.
 - Response: System stops recording, saves the audio file, and confirms the save.

4.1.3 Functional Requirements

- REQ-1: The system shall allow users to upload audio files in formats other than formats such as MP3 or WAV.

- REQ-2: The system shall provide a recording feature that captures audio directly within the application.
- REQ-3: The system shall display a confirmation message after a successful upload or recording.
- REQ-4: The system shall handle invalid file formats with an appropriate error message.

4.2 Speech Recognition

4.2.1 Description and Priority

Description: This feature converts uploaded or recorded voice data into text using AI and NLP technologies.

Priority: High

4.2.2 Stimulus/Response Sequences

- Stimulus: User uploads or records an audio file.
 - Response: System processes the audio file and converts it into text.
- Stimulus: User click “Send” button and then views the transcribed text.
 - Response: System displays the transcribed text for user review.

4.2.3 Functional Requirements

- REQ-1: The system shall process audio files and convert them to text with a transcription accuracy of at least 95%.
- REQ-2: The system shall handle various accents and speech patterns.
- REQ-3: The system shall process audio files in under one minute for files up to 10 minutes in length.
- REQ-4: The system shall provide feedback if the transcription process fails.

4.3 Report Generation

4.3.1 Description and Priority

Description: This feature automatically generates structured radiology reports from the transcribed text.

Priority: High

4.3.2 Stimulus/Response Sequences

- Stimulus: User reviews the transcribed text.
 - Response: System generates a draft report based on the transcribed text.
- Stimulus: User edits the draft report.
 - Response: System saves the changes and updates the report.
- Stimulus: User clicks the "Submit" button and then views the URL of the generated report PDF.
 - Response: System send reported PDF's URL to user.

4.3.3 Functional Requirements

- REQ-1: The system shall generate reports in a standardized format.
- REQ-2: The system shall allow users to edit generated reports.
- REQ-3: The system shall automatically save changes made to reports.
- REQ-4: The system shall integrate generated reports into EHR systems.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Minimum Latency:

The application targets minimal latency for report generation from audio data. This is achieved by using a fast and reliable speech recognition API that quickly converts voice data into text.

Text Processing Speed:

Optimized text processing algorithms are used, including advanced natural language processing models like Gemini and GPT, to process texts quickly and accurately, creating draft report outputs efficiently.

Maximum Response Time:

Caching is implemented to deliver frequently accessed reports quickly, ensuring rapid response times.

5.2 Safety Requirements

Data Security:

Ensure the security, confidentiality, and integrity of user voice data and reports using strong encryption methods (e.g., AES), firewalls, and regular security audits. User data is encrypted and securely stored in the database, with access permissions regularly reviewed.

System Stability:

Minimize errors and interruptions using redundant systems, automatic error detection and correction mechanisms, and regular maintenance.

User Security:

Protect user security and privacy with authentication processes and access controls. Encourage secure password practices and two-factor authentication.

Artificial Intelligence Security:

Ensure the security and reliability of AI models with safe training practices and continuous monitoring to detect and correct errors.

Emergency Response:

Prepare for unexpected situations with emergency plans, crisis communication plans, and rapid response teams.

5.3 Security Requirements

Security and Privacy of Personal Data:

Encrypt user voice data using strong encryption algorithms (e.g., AES). Strictly control database access, limiting it to authorized personnel only.

Security of Reports:

Encrypt report data in the database, regularly review access permissions, and restrict unnecessary access.

Secure Communication:

Use HTTPS protocol for secure communication with users. Employ SSL certificates to encrypt data during transfer.

5.4 Software Quality Attributes

Accuracy and Reliability:

Regularly update and retrain AI models with new datasets to improve accuracy.

Availability and Speed:

Develop user-friendly interfaces and optimize website performance to enhance usability.

Accuracy and Timeliness:

Continuously update text processing algorithms and regularly check data sources to ensure report accuracy.

Modularity and Sustainability:

Maintain a modular code base to facilitate the addition of new features or updates to existing ones.

5.5 Business Rules

Permission Process:

Implement an approval process to obtain necessary permissions from users before receiving audio data, allowing users to clearly indicate and withdraw consent.

Data Use Permission:

Obtain explicit permission from users to use their data for training AI models, ensuring confidentiality and security.

Report Access:

Provide users with optional access to reports, requiring authentication for access and the ability to withdraw access as needed.

6. Other Requirements

Database Requirements:

Data Structure and Retention:

Design the database model to store voice data, text data, user information, reports, and other relevant data. Ensure proper database management for efficient data handling and updates.

Performance and Scalability:

Optimize database performance for heavy usage and ensure scalability to handle increasing data volumes.

Backup and Recovery:

Perform regular data backups and establish a rapid recovery process to prevent data loss.

Internationalization Requirements:

Language Support:

Provide multilingual support for users, ensuring the user interface, text outputs, and reports are available in different languages.

Regional Differences:

Adapt to regional differences such as date formats, time zones, and currencies.

Legal Requirements:

Data Privacy and Security:

Comply with relevant data privacy laws and regulations. Implement necessary measures to protect users' personal data.

User Approval and Permissions:

Obtain explicit consent from users for processing voice data and generating reports. Manage user permissions according to legal requirements.

Reuse Goals for the Project:

Modularity and Flexibility:

Design the project with a modular structure to facilitate the reuse of components in similar projects or different contexts.

Documentation and Guidance:

Provide comprehensive documentation for project components to enable effective reuse and integration by other team members or developers.