Software Requirements Specification (SRS)

Voice Transcription Web Application

Date: November 2025

Authors:

1. Ahmed Mostafa Anwar | ID: 2023/03479
2. Abdelrahman Hesham | ID: 2023/06186
3. Raafat Fayez | ID: 2023/07237
4. Kevin Ayman | ID: 2023/04144
5. Mohamed Walid | ID: 2023/00151

Project: Voice Transcription Web Application

Table of Contents

1. Preface

1.1 Purpose

This document describes the software requirements specification for the Voice Transcription Web Application. It serves as a formal agreement between the development team and stakeholders regarding the system's functionality, constraints, and behavior.

1.2 Scope

The Voice Transcription Web Application is a comprehensive system that allows users to record their voice, transcribe it to text in real-time, categorize transcriptions, and manage their transcription history through a web-based interface.

1.3 Intended Audience

- Project Stakeholders
- Development Team
- Quality Assurance Team
- Project Managers
- End Users

1.4 Document Conventions

- **Bold text** indicates key terms.
- *Italic text* indicates emphasis.
- `Monospace` indicates code or technical terms.

1.5 References

- IEEE Std. 830-1998 - IEEE Recommended Practice for Software Requirements Specifications
- Project Charter Document
- Stakeholder Requirements Document

2. Introduction

2.1 Product Perspective

The Voice Transcription Web Application is a standalone web-based system that integrates modern web technologies (React, Node.js, Python) to provide seamless voice-to-text functionality.

2.2 Product Functions

- User registration and authentication
- Real-time voice recording and transcription
- Transcription categorization and management
- History tracking and retrieval
- Multi-user support with data isolation

2.3 User Characteristics

- **End Users**: Individuals needing voice transcription services.
- **Administrators**: System maintenance and user management
- **Technical Level**: Basic computer literacy required.

2.4 Constraints

- Requires modern web browser with microphone support.
- Internet connection required for transcription services
- Limited by speech recognition accuracy
- Browser compatibility constraints

2.5 Assumptions and Dependencies

- Users have access to functioning microphones.
- Google Speech Recognition API remains available.
- Sufficient server resources for concurrent users

3. Similar Systems

3.1 Existing Solutions Analysis

3.1.1 Otter.ai

- **Strengths**: High accuracy, real-time transcription, good mobile app
- **Weaknesses**: Limited free tier, subscription-based model
- **Differentiator**: Our system offers open-source alternative with self-hosting capability

3.1.2 Google Docs Voice Typing

- **Strengths**: Free, good accuracy, integrates with Google ecosystem
- **Weaknesses**: Limited to Google Docs, no categorization features
- **Differentiator**: Our system provides dedicated transcription management

3.1.3 Dragon NaturallySpeaking

- **Strengths**: High accuracy, extensive features
- **Weaknesses**: Expensive, desktop-only, steep learning curve
- **Differentiator**: Web-based, user-friendly interface

3.2 Competitive Advantage

- Open-source and self-hostable
- Custom categorization system
- User-friendly web interface
- Real-time processing capabilities
- Multi-platform accessibility

4. Glossary

Term	Definition
Transcription	The process of converting spoken language into written text
Audio Chunk	A segment of audio data processed for transcription
Category	User-defined classification for organizing transcriptions
JWT	JSON Web Token - used for secure authentication
MediaRecorder API	Web API for recording audio in browsers
Speech Recognition	Technology that converts speech to text
Base64 Encoding	Method for encoding binary data as ASCII text
Real-time Processing	Immediate processing of audio as it has recorded
Transcription History	Collection of user's previous transcriptions
Audio Visualization	Graphical representation of audio waveforms

5. Functional User Requirements

5.1 Authentication & User Management

1. **UR-1**: The system shall allow new users to register with email, username, and password
2. **UR-2**: The system shall allow registered users to log in with email and password
3. **UR-3**: The system shall maintain user sessions for authenticated users
4. **UR-4**: The system shall allow users to log out securely

5. **UR-5**: The system shall protect user data with proper authentication

5.2 Voice Recording & Transcription

6. **UR-6**: The system shall allow users to start voice recording with a single click
7. **UR-7**: The system shall provide visual feedback during recording
8. **UR-8**: The system shall allow users to stop recording when desired
9. **UR-9**: The system shall automatically transcribe recorded audio to text
10. **UR-10**: The system shall display transcription results immediately after processing

5.3 Transcription Management

1. **UR-11**: The system shall allow users to categorize transcriptions.
2. **UR-12**: The system shall allow users to create new categories.
3. **UR-13**: The system shall display transcription history chronologically.
4. **UR-14**: The system shall allow filtering transcriptions by category.
5. **UR-15**: The system shall display transcription timestamps.

5.4 User Interface

6. **UR-16**: The system shall provide intuitive navigation between features.
7. **UR-17**: The system shall display loading indicators during processing.
8. **UR-18**: The system shall show error messages for failed operations.
9. **UR-19**: The system shall provide visual feedback for user actions.
10. **UR-20**: The system shall work on desktop and mobile browsers.

6. Functional System Requirements

6.1 Authentication Module

6.1.1 User Registration

```
SR-1: The system shall validate registration data:
  - Email format validation
  - Username uniqueness check
  - Password strength requirements (min 8 characters)
```

```
  - Required field completion

SR-2: The system shall hash passwords using bcrypt before storage
SR-3: The system shall generate JWT tokens upon successful registration
```

6.1.2 User Login

```
SR-4: The system shall authenticate users by:
  - Verifying email existence
  - Comparing hashed passwords
  - Generating new JWT token

SR-5: The system shall expire tokens after 24 hours
SR-6: The system shall require token for protected routes.
```

6.2 Voice Processing Module

6.2.1 Audio Recording

```
SR-7: The system shall capture audio using MediaRecorder API
SR-8: The system shall record audio in WAV format, 16kHz sample rate
SR-9: The system shall chunk audio data for processing
SR-10: The system shall encode audio to base64 for transmission.
```

6.2.2 Speech Transcription

```
SR-11: The system shall send audio data to Python service via HTTP POST
SR-12: The system shall use Google Speech Recognition API for
transcription
SR-13: The system shall handle transcription errors gracefully
SR-14: The system shall process audio chunks up to 15 seconds duration
```

6.3 Data Management Module

6.4.1 Transcription Storage

```
SR-15: The system shall store transcriptions with:
  - User ID reference
  - Transcription text
```

```
  - Category assignment
  - Creation timestamp

SR-16: The system shall retrieve transcriptions ordered by creation
date
SR-17: The system shall support filtering by category and date range
```

6.4.2 Category Management

```
SR-18: The system shall allow category creation with unique names per
user
SR-19: The system shall prevent duplicate categories for same user
SR-20: The system shall update category references when categories are
modified.
```

7. Non-functional Requirements

7.1 Performance Requirements

1. **NR-1**: The system shall transcribe audio within 5 seconds of recording completion
2. **NR-2**: The system shall support up to 100 concurrent users
3. **NR-3**: The system shall have page loads under 3 seconds
4. **NR-4**: The system shall handle audio files up to 10MB in size
5. **NR-5**: The system shall maintain 99% uptime during business hours

7.2 Security Requirements

6. **NR-6**: All user passwords shall be hashed using bcrypt algorithm
7. **NR-7**: API endpoints shall be protected with JWT authentication
8. **NR-8**: The system shall implement CORS protection for cross-origin requests
9. **NR-9**: User data shall be isolated and accessible only to account owners
10. **NR-10**: The system shall sanitize all user inputs to prevent injection attacks

7.3 Reliability Requirements

11. **NR-11**: The system shall have mean time between failures (MTBF) of 720 hours
12. **NR-12**: The system shall recover from failures within 5 minutes

13. **NR-13**: Data loss shall not exceed 1 transaction in 10,000
14. **NR-14**: The system shall maintain data consistency across all modules

7.4 Usability Requirements

15. **NR-15**: New users shall be able to record and transcribe audio within 2 minutes of registration
16. **NR-16**: The interface shall be intuitive with learning curve under 15 minutes
17. **NR-17**: The system shall provide clear error messages and recovery suggestions
18. **NR-18**: All functionality shall be accessible with less than 3 clicks from main page

7.5 Compatibility Requirements

19. **NR-19**: The system shall support Chrome 80+, Firefox 75+, Safari 13+, Edge 80+
20. **NR-20**: The system shall work on desktop, tablet, and mobile devices
21. **NR-21**: The system shall support microphone access across all compatible browsers

8. Scenarios

8.1 Scenario 1: First-time User Registration and Transcription

Actor: New User (Sarah)

Precondition: Sarah has not used the system before

Main Flow:

1. Sarah navigates to the Voice Transcription application
2. She clicks "Register" and fills in her details (username, email, password)
3. System validates her information and creates account
4. Sarah receives confirmation and is automatically logged in
5. She sees the main dashboard with voice recorder
6. Sarah selects "Work" category from dropdown
7. She clicks the recording button and speaks for 30 seconds about meeting notes
8. System shows audio visualization while recording

9. Sarah stops recording, system processes audio
10. Transcription appears showing her meeting notes
11. Sarah clicks "Save" and the transcription is stored in her history under "Work" category

Postcondition: Sarah has successfully registered and created her first transcription.

8.2 Scenario 2: Regular User Managing Transcription History

Actor: Existing User (Mohamed)

Precondition: Mohamed has existing account with multiple transcriptions

Main Flow:

1. Mohamed logs into the system
2. He navigates to "History" page
3. System displays all his transcriptions sorted by date
4. Mohamed filters by "Meeting" category using the category filter
5. System shows only meeting-related transcriptions
6. Mohamed finds a specific transcription from last week
7. He reviews the content and confirms it's accurate
8. Mohamed needs to share this transcription, so he copies the text
9. He then creates a new category called "Client Meetings."
10. Mohamed moves several transcriptions to this new category
11. System updates the categorization immediately

Postcondition: Mohamed has organized his transcriptions and found needed information.

9. Use Cases

9.1 Detailed Use Cases

Use Case 1: User Registration

Use Case ID: UC-001

Actor: New User

Description: User creates a new account in the system

Preconditions: User is not logged in, system is accessible

Main Flow:

11. 1. User navigates to registration page
12. 2. System displays registration form
13. 3. User enters username, email, and password
14. 4. User confirms password and submits form
15. 5. System validates input data
16. 6. System checks for duplicate username/email
17. 7. System hashes password and creates user record
18. 8. System generates JWT token
19. 9. System redirects user to dashboard
20. 10. Use case ends successfully

Alternative Flows:

- A1: Validation fails - System shows specific error messages.
- A2: Duplicate credentials - System suggests login instead.

Postconditions: User account created, user logged in.

Use Case 2: Voice Transcription

Use Case ID: UC-002

Actor: Authenticated User

Description: User records voice and gets text transcription

Preconditions: User is logged in, microphone access granted

Main Flow:

1. 1. User selects category for transcription
2. 2. User clicks "Start Recording" button
3. 3. System requests microphone access (if not already granted)
4. 4. System begins recording audio
5. 5. System displays recording duration and audio visualization
6. 6. User speaks desired content
7. 7. User clicks "Stop Recording" button
8. 8. System stops recording and encodes audio data
9. 9. System sends audio to Python transcription service
10. 10. Python service processes audio and returns transcription
11. 11. System displays transcription text to user
12. 12. Use case ends successfully

Alternative Flows:

- A1: Microphone access denied - System shows error and instructions.
- A2: Transcription fails - System allows retry or manual input.
- A3: Network error - System queues request for later processing.

Postconditions: Transcription available for saving or discarding.

10. Form-based Specification & Tabular Specification

10.1 Form-based Specifications

From: User Registration

From ID: F-001

Purpose: Collect user information for account creation

Fields:

- Username: Text input, required, 3-50 characters, alphanumeric
- Email: Email input, required, valid email format
- Password: Password input required, minimum eight characters
- Confirm Password: Password input required must match password field.

Validation Rules:

- Usernames must be unique across system.
- Email must be unique and properly formatted.
- Password must meet complexity requirements.
- All fields must be completed.

Actions:

- Submit: Validates and creates user account
- Cancel: Returns to login page

From: Voice Recording

From ID: F-002

Purpose: Capture and transcribe user speech

Fields:

- Category: Dropdown selection, required, default "General"
- Recording Control: Button toggling between Start/Stop
- Audio Visualization: Real-time audio level display
- Transcription Result: Read-only text area displaying results.

Validation Rules:

- Microphone access must be granted.
- Minimum recording duration: 1 second
- Maximum recording duration: 5 minutes

Actions:

- Start Recording: Audio capture begins.
- Stop Recording: Ends audio capture and initiates transcription.
- Save: Stores transcription with selected category
- Discard: Clears current recording without saving

10.2 Tabular Specifications

Table: API Endpoints Specification

Endpoint	Method	Parameters	Response	Status Codes
`/api/register`	POST	` {username, email, password} `	` {success, token, user} `	200, 400, 500
`/api/login`	POST	` {email, password} `	` {success, token, user} `	200, 401, 500
`/api/transcribe`	POST	` {audioData, category} `	` {success, transcription} `	200, 400, 500
`/api/transcriptions`	GET	`category (optional)`	` {success, transcriptions []} `	200, 401, 500
`/api/categories`	GET	-	` {success, categories []} `	200, 401, 500

Table: Error Codes and Handling

Error Code	Description	User Message	Recovery Action
AUTH-001	Invalid credentials	"Email or password is incorrect"	Check credentials and try again

AUTH-002	Token expired	"Session expired. Please log in again"	Redirect to login page
VOICE-001	Microphone access denied	"Microphone access is required. Please allow microphone permissions"	Guide user to browser settings
VOICE-002	Transcription failed	"We couldn't process your audio. Please try again"	Show retry button
VOICE-003	Audio too long	"Recording too long. Please keep under 5 minutes"	Auto-stop recording at limit

Table: Database Schema Specification

Table	Column	Type	Constraints	Description
users	id	INT	PRIMARY KEY, AUTO_INCREMENT	User identifier
users	username	VARCHAR (50)	UNIQUE, NOT NULL	User's display name
users	email	VARCHAR (100)	UNIQUE, NOT NULL	User's email address
users	password_hash	VARCHAR (255)	NOT NULL	Hashed password
users	created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Account creation time
categories	id	INT	PRIMARY KEY, AUTO_INCREMENT	Category identifier

categories	user_id	INT	FOREIGN KEY (users.id)	Category owner
categories	name	VARCHAR (50)	NOT NULL	Category name
transcriptions	id	INT	PRIMARY KEY, AUTO_INCREMENT	Transcription identifier
transcriptions	user_id	INT	FOREIGN KEY (users.id)	Transcription owner
transcriptions	category_id	INT	FOREIGN KEY (categories.id)	Transcription category
transcriptions	text	TEXT	NOT NULL	Transcribed text content
transcriptions	created_at	TIMESTAMP	DEFAULT CURRENT_TIMESTAMP	Transcription time