Requirements Report -

Speech Capture, Transcription, and Analysis App

Sponsor – Dr. Zoran Djordjevic

Reed Cummings – Group Leader

Mario Jelev

Eric Rogers

James Lussier

ABSTRACT

In this document we will outline the requirements we must meet upon the completion of this project. This includes functional and non-functional requirements as well as a use case that they should be used in. We will give a graphical and a textual description of for each use case we present.

Each use case will contain the following:

– Participating Actors

– Entry Condition(s)

– Normal Flow of Events

– Exit Condition(s)

– Exceptions (Alternate Flow of Events)

– Special Requirements

We will then go over the rationale for our use case model we used.

TABLE OF CONTENTS

List of Figures 4

List of Tables 5

Introduction 6

Use Case Model for Functional Requirements 7

Graphical Model 7

Textual Model 12

Functional Requirements 13

Rationale for Use Case 14

Non-functional Requirements 15

Evidence the Requirements have been Places Under Config Mgmt. 16

References 17

LIST OF FIGURES

None.

LIST OF TABLES

UC-1 pg 7

UC-2 pg 8

UC-3 pg 9

UC-4 pg 10

UC-5 pg 11

INTRODUCTION

The purpose of our Requirements plan is to make sure the customer and the software design team are on the same page. We will present the plan, so the customer knows how we foresee the software being used, and so we can have written proof that we came to an agreement on what the basis of the software should be. It covers the basic functions of the proposed software as well as non-functional requirements the software must meet. It will show how we expect the software to work and in what manner it is supposed to be used (use cases). We start with the functional requirements of the software and demonstrate how these will be met in specific use cases. We will then discuss the rationale we used for the use cases and then cover the non-functional requirements. At the end, the design team and the customers should be on the same page considering the functionalities of the proposed software.

USE CASE MODEL FOR FUNCTIONAL REQUIREMENTS

GRAPHIC USE CASE MODEL

|  |  |
| --- | --- |
| **Use Case ID** | UC-1 |
| **Use Case Name** | Speech Capture |
| **Primary Actor** | User |
| **Entry Condition** | Clicked on Record File from home screen |
| **Exit Condition** | Failure/System Begins Transcription/User goes back to home screen |
| **Actor’s Goal** | To capture the speech into an audio file |
| **Priority** | Critical |

***Main Success Scenario***

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | Actor clicks record button | System starts recording audio |
| **2** | Actor clicks stop recording button | System stops recording audio |
| **3** | Actor clicks next button to begin transcription | System begins transcription and produces a new screen. |

***Alternative Success Scenario 1: Download File/Record Again***

**Trigger Condition**: Actor wants to download file to local system and record again

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | Actor clicks stop recording button | System stops recording audio |
| **2** | Actor clicks button to download audio file | System produces audio file in downloads folder |
| **3** | Actor clicks back button to go back to home screen | System takes you back to home screen |

***Alternative Success Scenario 2: Download File/Transcribe***

**Trigger Condition**: Actor wants to download file to local system and begin transcription

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | Actor clicks stop recording button | System stops recording audio |
| **2** | Actor clicks button to download audio file | System produces audio file in downloads folder |
| **3** | Actor clicks next button to begin transcription | System begins transcription and produces a new screen. |

|  |  |
| --- | --- |
| **Use Case ID** | UC-2 |
| **Use Case Name** | Upload Audio File |
| **Primary Actor** | User |
| **Entry Condition** | Clicked on “Select File” from the upload screen |
| **Exit Condition** | Failure/System begins transcription |
| **Actor’s Goal** | User uploads an audio file for analysis |
| **Priority** | Critical |

***Main Success Scenario***

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | Actor clicks the “Select file” button | System prompts the actor to choose a file from their file system explorer with an audio file format |
| **2** | Actor selects the audio file they want to use in their file system explorer | The user is taken to a “Transcription in Progress” page to await the transcription to be complete. |

***Alternative Success Scenario 1: Invalid or corrupt file***

**Trigger Condition**: Step 2 user input an invalid or corrupt file to the system

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | Actor submits an invalid or corrupt file for transcription | System begins file transcription, recognizes the file as invalid or corrupt, and returns an error to the actor prompting them to try another file. |
| **2** | Actor goes back to step 1 of the Main Success Scenario |  |

|  |  |
| --- | --- |
| **Use Case ID** | UC-3 |
| **Use Case Name** | Upload Transcription |
| **Primary Actor** | User |
| **Entry Condition** | Clicked on Upload Transcription from home screen |
| **Exit Condition** | An analysis of the uploaded transcription is completed |
| **Actor’s Goal** | Upload a pre-existing text file for analysis |
| **Priority** | High |

***Main Success Scenario***

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | Actor clicks the “Select file” button | System prompts the actor to choose a file from their file system explorer with a text file format |
| **2** | Actor selects the text file they want to use in their file system explorer | The user is taken to an “Analysis in Progress” page to await the analysis to be complete. |

***Alternative Success Scenario 1: Invalid or corrupt file***

**Trigger Condition**: Step 2 user input an invalid or corrupt file to the system

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | Actor submits an invalid or corrupt file for analysis | System begins file analysis, recognizes the file as invalid or corrupt, and returns an error to the actor prompting them to try another file. |
| **2** | Actor goes back to step 1 of the Main Success Scenario |  |

|  |  |
| --- | --- |
| **Use Case ID** | UC-4 |
| **Entry Condition** | Clicking the “Past Analysis” button on the main page |
| **Exit Condition** | User clicks open or back button |
| **Use Case Name** | Search past jobs |
| **Primary Actor** | User |
| **Actor’s Goal** | View any jobs they may have done in the past |
| **Business Value** | High |

***Main Success Scenario***

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | User types in a job name into the text field provided | System enables a search button for the user to click |
| **2** | User clicks the search button | User provides a list of jobs that matches the query searched |
| **3** | User clicks on one of the queries provided and clicks “Open” | System takes the user to the analysis page that matches the job selected |

***Alternative Success Scenario 1: No jobs match query***

**Trigger Condition**: Step 3 User tries to search for a job, but no jobs match their query

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | User enters a search query that matches no past jobs for them | System responds by putting text on the screen that says “No jobs match your query” |
| **2** | User can either go back to step 1 of the main success scenario, or click the “Back” button to return to the main page of the site |  |

|  |  |
| --- | --- |
| **Use Case ID** | UC-5 |
| **Use Case Name** | Analyze Transcription |
| **Primary Actor** | User |
| **Entry Condition** | User has an already transcribed file |
| **Exit Condition** | User closes browser/clicks back |
| **Actor’s Goal** | Produce the analysis on the speech transcription |
| **Priority** | Critical |

***Main Success Scenario***

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | The user chooses to Analyze file by clicking the analyze button | System will take the .txt file and run in through the analysis functions which will produce a .json file which we will use to create a web page |
| **2** | The user clicks download file | The system will download the file to the local system as a .json file |

***Alternative Success Scenario 1: User Does not want to analyze file***

**Trigger Condition**: User wants to transcribe more files

|  |  |  |
| --- | --- | --- |
| **Step** | **Actor Action** | **System Response** |
| **1** | User clicks the download button. | System downloads .txt file to local machine |
| **2** | User clicks back button | System takes user back to the home screen. |

TEXTUAL DESCRIPTION

UC-1:

Description:

When the user wants to record a meeting/conversation they will choose the record option on the home screen so they can create an audio recording of the speech.

UC-2:

Description:

When the user reaches the main page of the application, they are given the option to upload an MP3 file for transcription and analysis.

UC-3:

Description

The Upload Transcription use case, which can be initialized from the home screen, lets users upload already transcribed .mp3/audio files and proceed with the text analysis. This bypasses uploading/creating audio files and making the application transcribe them.

UC-4:

Description:

If the user wants to view past transcription jobs, they are given a button on the main page to take them to this webpage where they can input search queries of past job names

UC-5:

Description:

When the user has finished transcribing the file, they will click on the analyze button which will begin the analysis of the transcription.Functional Requirements:

The product shall capture speech from speaker

The product shall create an mp3 file from the audio captured

The product shall have an upload button to upload mp3 file

The product shall transcribe the mp3 file into a txt file

The product shall analyze the txt file

The product shall delineate topics - 2

The product shall extract key phrases - 3

The product shall extract dates - 4

The product shall extract sentiment - 6

The product shall create a brief summary of the data. - 1

The product shall determine what words were used most. - 5

The product shall be able to search through past transcriptions for key phrases

Stretch Goals:

The product shall transcribe straight from the speaker into the application

The product shall transcribe a few languages that are not English

RATIONALE FOR YOUR USE CASE MODEL:

We had used this use case model in the past and we understood the format and how to approach each problem. We believe this expedited the process a bit and helped us more fully understand each requirement and how to accomplish them.

NON-FUNCTIONAL REQUIREMENTS:

The product shall be able to be used with less than 10 minutes of training.

The product shall give analysis within 2 minutes of choosing to analyze file.

The product shall be available 90% of the time, with down time for maintenance.

The product shall transcribe correctly 90% of the time.

We will add as we see fit and understand more about the software needs.

EVIDENCE THE REQUIREMENTS HAVE BEEN PLACED UNDER

CONFIGURATION MANAGEMENT

<https://github.com/reedpcummings/SeniorDesign-SpeechCapture>

<https://drive.google.com/open?id=1OyBwhqh_zyKAeEYYu49XRbNLiF9aBvdU>

REFERENCES

None