Software Requirements Specification for Harkify

Version 1.0.4

Prepared by

Bravo Team

University of Maryland Global Campus

07 June 2021

Revision History

Author	**Date**	**Reason For Changes**	**Version**
Team Bravo	06/07/2021	Initial version	1.0
Roy Gordon	06/11/2021	Recommended mention of saving PII as risk	1.0.1
Team Bravo	06/11/2021	General revisions. Added PII to security requirements.	1.0.2
Team Bravo	06/19/2021	General revisions. Added encrypt documents, notifications, and favorites page to use cases.	1.0.3
Team Bravo	07/10/2021	Added requirement changes from Milestone 2 meeting.	1.0.4

Table of Contents

1.0 Introduction

1.1 Purpose

This SRS describes the functional and non-functional software requirements for release 1.0 of the Harkify application. This document is intended to be used by the members of the Bravo team that will implement and verify the correct functioning of the system. Unless otherwise noted, all requirements specified here are high priority and committed for release 1.0. The document can therefore be used to illustrate various use scenarios for Harkify application. It can also be used to illustrate the underlying features and possible improvements that can be incorporated in the future. Some of the use cases illustrated in this document include the following:

1. Starting the application
2. Training the application for voice recognition
3. Recording save and end trigger
4. Saving and editing note
5. Searching note by voice and on screen
6. Viewing and editing settings
7. Stopping the application

1.2 Scope

The software to be created is a speech recognition application for memory-impaired users as a short-term memory assistant. It will perform basic functions that will allow the user to record and save based on speech-to note functionality, where the user will then utilize the functions to search the notes by either voice or through the developed Graphical User Interface (GUI). Login access and saving of user information are out of scope for this application. An exploration of various features of the application will be made below.

1.3 Definitions, Acronyms, and Abbreviations

Some of the terms used in this document in relation to SRS have been defined below:

- UI: User Interface
- GUI: Graphical User Interface
- SRS: Software Requirements Specification
- STM: Short-term Memory
- Trigger: This is a feature where the application allows starting, stopping, and searching when a specified identifier is heard.
- Speech Synthesis: Allows the artificial production or simulation of human speech by a computer or other device.

1.4 References

Capstone Project Guide. (2020). Retrieved from
https://learn.umgc.edu/d2l/le/content/583665/viewContent/21626188/View

Speech synthesis (n.d.). Wikipedia. Retrieved from
https://en.wikipedia.org/wiki/Speech_synthesis

Software Requirement Specification (SRS) format. (2020). GeeksforGeeks. Retrieved from
https://www.geeksforgeeks.org/software-requirement-specification-srs-format/

Your guide to writing a software requirements specification (SRS) document. (n.d.). Relevant. Retrieved from https://relevant.software/blog/software-requirements-specification-srs-document/

1.5 Overview

This document has been organized into four major sections: Overall Description, Specific Requirements, Supporting Information, External Interface Requirements and Other Non-Functional Requirements. The overall description provides system requirements on a high level. Specific requirements provide illustrations of specific features of the Harkify application. The supporting information section provides additional information that can help users quickly familiarize themselves with the application. The External Interface Requirements section covers major external systems with which the application will interact. These external requirements include user interface, hardware and communication guidelines to be enforced. The Other Non-Functional Requirements section details performance and security requirements of the final application.

2.0 Overall Description

The application is intended as a memory aid for users by recording their speech and playing it back for them. When the user first starts the application, they shall take several steps to configure the application for use. First, the user shall train the application to recognize their voice. Also, the user shall record trigger phrases in the application to indicate that it should begin recording a new note or to indicate that it should begin searching through existing notes. Once voice recognition and trigger phrases are established, the user can fully operate the application.

When the user speaks one of their trigger phrases for recording a new note, then the application shall translate the user’s speech after the trigger phrase into text which shall be saved. Afterwards, when the user speaks one of their trigger phrases for searching through existing notes, the application shall find matching notes by comparing them to the user’s speech after the trigger phrase. These matching notes are then repeated back to the user as speech.

In addition to speech-based note recording and searching, the application shall allow the user to search for notes and edit notes through the application GUI as well. When the user turns off the application, existing notes shall persist but the application shall no longer be listening for trigger phrases to initiate operations.

2.1 Use-Case Model Survey

The UML use case diagram below outlines all uses cases for this project.

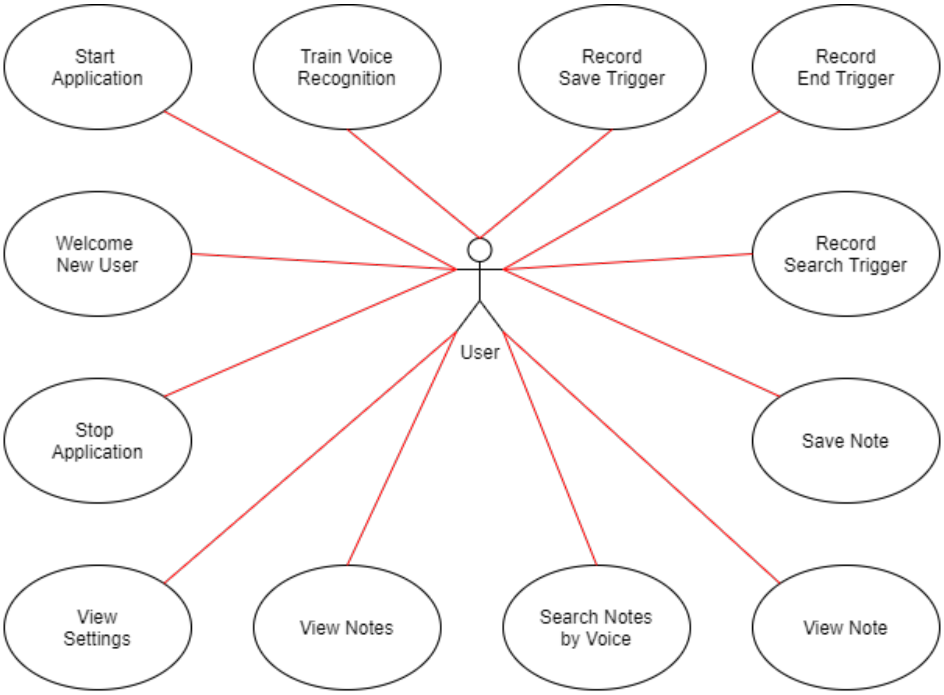

Figure 1. UML diagram of all system use cases

The level 0 data flow diagram (DFD) below specifies all input and output data between the application and external systems or actors.

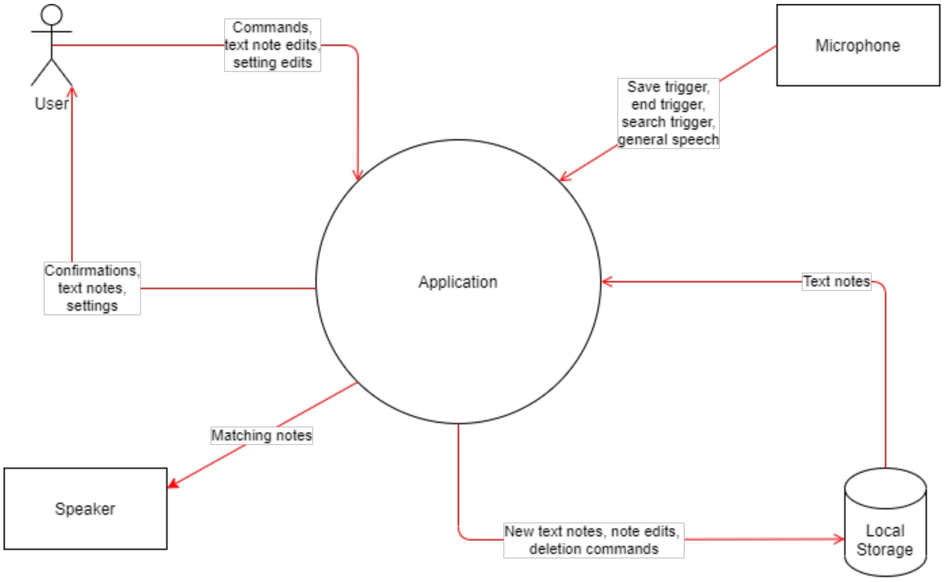

Figure 2. Level 0 data flow diagram of application

The following use cases explain the main requirements for this application. Each use case is described in further detail in Section 3.0.

Use Cases	Description
1. Start Application	This feature allows a user to start the application so that it begins to listen for trigger phrases and is ready to accept commands.
2. Train Voice Recognition	This feature allows a user to train the application to recognize the user's voice for commands.
3. Record Save Trigger	This feature allows a user to record a trigger phrase so that the application initiates the “Save Note” use case when the trigger is heard.
4. Record End Trigger	This feature allows a user to record a trigger phrase so that the application stops recording a note during “Save Note” use case when the trigger is heard.
5. Record Search Trigger	This feature allows a user to record a trigger phrase so that the application initiates the “Search Notes by Voice” use case when the trigger is heard.
6. Save Note	This feature allows a user to record speech as a text note by speaking the save trigger phrase or selecting the “Save Note” option.
7. View Note	This feature allows a user to employ the GUI to view, edit or delete an existing note on the screen.
8. View Settings	This feature allows a user to view current application settings, edit those settings, record speech triggers or train voice recognition.

9. Search Notes by Voice	This feature allows the user of the software to search a specific event of the recorded conversations using key phrases provided.
10. View Notes	This feature allows the user of the software to view existing text notes and search for them using key words displayed on the GUI.
11. Stop Application	This feature allows the user of the software to end the session /close out of the application.
12. Welcome New User	This feature allows a user to configure application settings after starting the application for the first time.
13. Encrypt saved document	This feature allows a user to provide personal information to the application with knowledge of security.
14. Favorites Page	This feature allows a user to store notes that cannot be deleted directly from this page or from the application deleting settings.
15. Notifications	This feature reminds a user of the application and to remember to record notes.
16. Training videos	This feature demonstrates training videos on how to use the application.

2.2 Assumptions and Dependencies

The application is expected to operate under the following assumptions. The application shall be developed in Flutter and Dart.js as a mobile application for both Android and IOS smartphones. Only the user who have trained the application to recognize their voice shall be recorded. All other voices heard shall be ignored. Furthermore, the application will have a wake command to that will initiate a recording and this command must be customizable. In addition, the application shall be designed specifically for its target demographic of elderly users suffering from memory loss. This means that screens shall be easy to read and easy to navigate with high usability. Finally, recorded text notes shall be retained for only seven days (see the “View Settings” use case), after which those notes shall be purged.

In terms of dependencies, the application depends upon the smartphone where it is installed for access to a microphone, speaker and text note storage. If the smartphone does not provide access to these features, the application shall not function. Furthermore, given the project’s minimal budget, application development is completely dependent upon free and open-source software.

3.0 Specific Requirements

This section provides specific details on all use cases covered in this project. Each use case is described in a use-case report which includes a summary, pre-conditions, post-conditions, triggers and the basic course of events. If the use case may traverse alternate paths from the basic course of events, then those are detailed as well.

3.1 Use-Case Reports

3.1.1 Use Case Name: Start Application

Summary: This feature allows a user to start the application so that it begins to listen for trigger phrases and is ready to accept commands.

Pre-conditions: The user must have turned on the smartphone where the application is hosted.

Triggers: The user selects the application on their phone.

Basic Course of Events:

Internal Pre-condition: The user has opened the application before.

Actor	System
1. The user selects the application.	
	2. The system purges any text notes that are older in days than the application setting (see the “View Settings” use case).
	3. The system shows the user the following options to select from: View Settings, Save Note, View Notes, Stop Application.
	4. The system begins listening for commands.

Internal Post-condition: None

Alternative Paths:

A. The user selects the application for the first time.

Internal Pre-condition: None

Actor	System

1. The user selects the application.	
	2. The system initiates the “Welcome New User” use case.

Internal Post-condition: None

Post-conditions: None

3.1.2 Use Case Name: Train Voice Recognition

Summary: This feature allows a user to train the application to recognize the user's voice for commands.

Pre-conditions: The user must have started the application and selected the “View Settings” option.

Triggers: The user initiates the Train Voice Recognition event.

Basic Course of Events:

Internal Pre-condition: None

Actor	**System**
1. The user selects the “Train Voice Recognition” option.	
	2. The system requests that the user repeat a series of phrases.
3. The user repeats the requested phrases.	
	4. The system listens to those phrases through the microphone and saves data to identify the user when speaking.
	5. The system shows confirmation to the user that the voice recognition was trained successfully.

Internal Post-condition: Voice recognition data is saved.

Alternative Paths:

A. The user repeats the requested phrases but the system cannot interpret them correctly.

Internal Pre-condition: The user has already selected to Train Voice Recognition and spoken the requested phrases.

Actor	System
3.The user repeats the requested phrases.	
	4. The system provides an error message regarding the failed voice recognition training. Then the system asks the user to repeat a series of phrases again.
5. The user repeats the requested phrases again.	
	6. The system listens to those phrases through the microphone and saves data to identify the user when speaking.
	7. The system shows confirmation to the user that the voice recognition was trained successfully.

Internal Post-condition: Voice recognition data is saved.

Post-conditions: The system has been trained to recognize the user's voice.

3.1.3 Use Case Name: Record Save Trigger

Summary: This feature allows a user to record a trigger phrase so that the application initiates the "Save Note" use case when the trigger is heard.

Pre-conditions: The user must have started the application, successfully trained the application to recognize the user's voice and selected the "View Settings" option.

Triggers: The user initiates the Record Save Trigger event.

Basic Course of Events:

Internal Pre-condition: Voice recognition data must be saved.

Actor	System
1. The user selects the "Record Save Trigger" option.	
	2. The system requests that the user repeat the trigger phrase.

3. The user repeats the phrase they want to trigger starting the “Save Note” use case.	
	4. The system listens to the phrase through the microphone and saves data to identify it when the user is speaking.
	5. The system shows confirmation to the user that the phrase was recorded successfully.

Internal Post-condition: The save trigger phrase has been saved.

Alternative Paths:

A. The user repeats the trigger phrase but the system cannot interpret it correctly.

Internal Pre-condition: The user has already selected to Record Save Trigger and spoken the trigger phrase.

Actor	System
3. The user repeats the phrase they want to trigger starting the “Save Note” use case.	
	4. The system provides an error message regarding the failed trigger phrase saving. Then the system asks the user to repeat the trigger again.
5. The user repeats the trigger phrase again.	
	6. The system listens to the phrase through the microphone and saves data to identify it when the user is speaking.
	7. The system shows confirmation to the user that the phrase was recorded successfully.

Internal Post-condition: The save trigger phrase has been saved.

Post-conditions: The user may now initiate the “Save Note” use case with the save trigger phrase.

3.1.4 Use Case Name: Record End Trigger

Summary: This feature allows a user to record a trigger phrase so that the application stops recording a note during “Save Note” use case when the trigger is heard.

Pre-conditions: The user must have started the application, successfully trained the application to recognize the user’s voice and selected the “View Settings” option.

Triggers: The user initiates the Record End Trigger event.

Basic Course of Events:

Internal Pre-condition: Voice recognition data must be saved.

Actor	System
1. The user selects the “Record End Trigger” option.	
	2. The system requests that the user repeat the trigger phrase.
3. The user repeats the phrase they want to trigger ending the “Save Note” use case.	
	4. The system listens to the phrase through the microphone and saves data to identify it when the user is speaking.
	5. The system shows confirmation to the user that the phrase was recorded successfully.

Internal Post-condition: The end trigger phrase has been saved.

Alternative Paths:

A. The user repeats the trigger phrase but the system cannot interpret it correctly.

Internal Pre-condition: The user has already selected to Record End Trigger and spoken the trigger phrase.

Actor	System
3. The user repeats the phrase they want to trigger ending the “Save Note” use case.	

	4. The system provides an error message regarding the failed trigger phrase saving. Then the system asks the user to repeat the trigger again.
5. The user repeats the trigger phrase again.	
	6. The system listens to the phrase through the microphone and saves data to identify it when the user is speaking.
	7. The system shows confirmation to the user that the phrase was recorded successfully.

Internal Post-condition: The end trigger phrase has been saved.

Post-conditions: The user may now end the "Save Note" use case with the end trigger phrase.

3.1.5 Use Case Name: Record Search Trigger

Summary: This feature allows a user to record a trigger phrase so that the application initiates the "Search Notes by Voice" use case when the trigger is heard.

Pre-conditions: The user must have started the application, successfully trained the application to recognize the user's voice and selected the "View Settings" option.

Triggers: The user initiates the Record Search Trigger event.

Basic Course of Events:

Internal Pre-condition: Voice recognition data must be saved.

Actor	System
1. The user selects the "Record Search Trigger" option.	
	2. The system requests that the user repeat the trigger phrase.
3. The user repeats the phrase they want to trigger starting the "Search Notes by Voice" use case.	
	4. The system listens to the phrase through the microphone and saves data to identify it when the user is speaking.

	5. The system shows confirmation to the user that the phrase was recorded successfully.

Internal Post-condition: The search trigger phrase has been saved.

Alternative Paths:

A. The user repeats the trigger phrase but the system cannot interpret it correctly.

Internal Pre-condition: The user has already selected to Record Search Trigger and spoken the trigger phrase.

Actor	System
3. The user repeats the phrase they want to trigger starting the "Search Notes by Voice" use case.	
	4. The system provides an error message regarding the failed trigger phrase saving. Then the system asks the user to repeat the trigger again.
5. The user repeats the trigger phrase again.	
	6. The system listens to the phrase through the microphone and saves data to identify it when the user is speaking.
	7. The system shows confirmation to the user that the phrase was recorded successfully.

Internal Post-condition: The search trigger phrase has been saved.

Post-conditions: The user may now start the "Search Notes by Voice" use case with the search trigger phrase.

3.1.6 Use Case Name: Save Note

Summary: This feature allows a user to record speech as a text note by speaking the save trigger phrase or selecting the "Save Note" option.

Pre-conditions: The user must have started the application and successfully trained the application to recognize the user's voice. If a save trigger phrase is used to start this use case, then the save trigger phrase must have been successfully saved.

Triggers: The user speaks the save trigger phrase or selects the "Save Note" option.

Basic Course of Events:

Internal Pre-condition: Voice recognition data must be saved.

Actor	System
1. The user selects the “Save Note” option.	
	2. The system requests that the user begin speaking.
3. The user speaks.	
	4. The system listens to the speech through the microphone and translates it into text.
5. The user speaks the end trigger phrase or does not speak for a number of seconds greater than the application setting (see the “View Settings” use case).	
	6. The system saves all text translated to that point in a single text note on the application’s host smartphone. The system also infers a conversation name from the spoken words to also be saved with the text note.
	7. The system shows confirmation to the user that the text note was saved while initiating the “View Note” use case.

Internal Post-condition: The text note has been saved.

Alternative Paths:

A. The user speaks the save trigger phrase.

Internal Pre-condition: Voice recognition data must be saved.

Actor	System
1. The user speaks the save trigger phrase.	

Internal Post-condition: Control returns to step #3 of the Basic Course of Events of the “Save Note” use case.

Post-conditions: The spoken words are translated to text and saved as a text note.

3.1.7 Use Case Name: View Note

Summary: This feature allows a user to employ the GUI to view, edit or delete an existing note on the screen.

Pre-conditions: The user must have started the application and must have executed the “Save Note”, “Search Notes by Voice” or “View Notes” use cases so that an existing text note shows on the screen.

Triggers: The user initiates the View Note event.

Basic Course of Events:

Internal Pre-condition: An existing text note must be showing on the screen.

Actor	System
1. The user selects the “View Note” option.	
	2. The system shows the text note details in an editable format with the following options: “Save”, “Delete” and “Cancel”. The system also shows the conversation name associated with the text note (in an editable format) and the date/time when the text note was saved (not in an editable format).
3. The user makes edits to the text note.	
4. The user selects the “Save” option.	
	5. The system saves all changes to the text note.

Internal Post-condition: The text note has been saved.

Alternative Paths:

A. The user cancels changes made to an existing text note.

Internal Pre-condition: The user has made edits to the text note.

Actor	System

3. The user makes edits to the text note.	
4. The user selects the “Cancel” option.	
	5. The system abandons changes made to the text note.

Internal Post-condition: None

B. The user selects the “Delete” option.

Internal Pre-condition: The user has selected the “View Note” option.

Actor	System
3. The user selects the “Delete” option.	
	4. The system requests confirmation that the note should be deleted.
5. The user selects to confirm deletion.	
	6. The system deletes the text note, shows confirmation that the note was deleted, and initiates the “View Notes” use case.

Internal Post-condition: The text note has been deleted.

Post-conditions: None

3.1.8 Use Case Name: View Settings

Summary: This feature allows a user to view current application settings, edit those settings, record speech triggers or train voice recognition.

Pre-conditions: The user must have started the application.

Triggers: The user initiates the View Settings event.

Basic Course of Events:

Internal Pre-condition: None

Actor	System

1. The user selects the “View Settings” option.	
	2. The system shows all application settings that the user can edit. These include how many days before text notes are purged (defaulting to 7, see the “Start Application” use case) and how many seconds without speech before a text note is saved (defaulting to 5, see the “Save Note” use case). The system also shows options for “Record Save Trigger”, “Record End Trigger”, “Record Search Trigger”, “Train Voice Recognition”, “Save Changes” and “Cancel Changes”.
3. The user makes edits to the application settings.	
4. The user selects the “Save Changes” option.	
	5. The system saves all changes to the application settings.

Internal Post-condition: Changes to the application setting have been saved.

Alternative Paths:

A. The user cancels unsaved changes to an existing application setting.

Internal Pre-condition: The user has made edits to the application setting.

Actor	System
3. The user makes edits to one of the application settings.	
4. The user selects the “Cancel” option.	
	5. The system abandons changes made to the application setting.

Internal Post-condition: None

B. The user selects the “Record Save Trigger” option.

Internal Pre-condition: The user has selected the “View Settings” option.

Actor	System
3. The user selects the “Record Save Trigger” option.	
	4. The system initiates the “Record Save Trigger” use case.

Internal Post-condition: None

C. The user selects the “Record End Trigger” option.

Internal Pre-condition: The user has selected the “View Settings” option.

Actor	System
3. The user selects the “Record End Trigger” option.	
	4. The system initiates the “Record End Trigger” use case.

Internal Post-condition: None

D. The user selects the “Record Search Trigger” option.

Internal Pre-condition: The user has selected the “View Settings” option.

Actor	System
3. The user selects the “Record Search Trigger” option.	
	4. The system initiates the “Record Search Trigger” use case.

Internal Post-condition: None

E. The user selects the “Train Voice Recognition” option.

Internal Pre-condition: The user has selected the “View Settings” option.

Actor	System
3. The user selects the “Train Voice Recognition” option.	

	4. The system initiates the “Train Voice Recognition” use case.

Internal Post-condition: None

Post-conditions: None

3.1.9 Use Case Name: Search Notes by Voice

Summary: This feature allows the user of the software to search a specific event of the recorded conversations using key phrases provided.

Pre-conditions: The application must be started on the smartphone.

Triggers: User triggers the Search Trigger event.

Basic course of events (Scenario):

Internal Pre-condition: None

Actor	**System**
1. The user gives a voice command using key phrases to the opened application.	
	2. The system uses the key phrases to play the audio that are customized for the user to search through the saved speech to text notes via the voice command.

Internal Post-condition: None

Post-conditions: None

3.1.10 Use Case Name: View Notes

Summary: This feature allows the user of the software to view existing text notes and search for them using key words displayed on the GUI.

Pre-conditions: The application must be started on the smartphone.

Triggers: User selects the View Notes option

Basic course of events (Scenario):

Internal Pre-condition: None

Actor	**System**
1. The user selects the View Notes option from the main menu.	
	2. The system shows a summary of existing text notes with “View Note” option for each one. The system also shows a search bar with an editable keyword field. The summary for each text note should include the date/time when the note was saved, the conversation name associated with the note, and the first 100 characters of the note itself.
3. The user enters a keyword in the search bar.	
	4. The system displays all the available notes matching the selected keyword. Each note includes a “View Note” option. Matching notes should be determined by ensuring that all keywords submitted are found either in the text of the note or in the conversation name for that note in order for the note to appear in results.
5. The user selects the “View Note” option on a matching text note from the list.	
	6. The system initiates the “View Note” use case.

Internal Post-condition: None

Alternative Paths:

A. The system finds no matches to the user’s keyword.

Internal Pre-condition: The user has entered a keyword in the search bar

Actor	**System**
3. The user enters a keyword in the search bar.	
	4. The system shows a message indicating that there were no matching text notes and that the user should try again.

Internal Post-condition: Control returns to step #3 of the Basic Course of Events of the “View Notes” use case.

Post-conditions: None

3.1.11 Use Case Name: Stop Application

Summary: This feature allows the user of the software to end the session /close out of the application.

Pre-conditions: The application must be already started and in use on the smartphone.

Triggers: User triggers the Stop event

Basic course of events (Scenario):

Internal Pre-condition: None

Actor	System
1. The user clicks on the Stop icon from the main menu.	
	2. The system checks and retains unsaved changes.
	3. The system automatically closes the application.
	4. The system no longer listens for trigger phrases to initiate operations.

Internal Post-condition: None

Post-conditions: None

3.1.12 Use Case Name: Welcome New User

Summary: This feature allows a user to configure application settings after starting the application for the first time.

Pre-conditions: The user must have turned on the smartphone where the application is hosted.

Triggers: The user selects the application on their phone.

Basic Course of Events:

Internal Pre-condition: The user has never opened the application before.

Actor	System
1. The user selects the application.	

	2. The system initiates the “Train Voice Recognition” use case.
	3. The system initiates the “Record Save Trigger” use case.
	4. The system initiates the “Record End Trigger” use case.
	5. The system initiates the “Record Search Trigger” use case.
	6. The system initiates the “Save Note” use case.

Internal Post-condition: None

Post-conditions: None

3.1.13 Use Case Name: Encrypt Documents

Summary: This feature allows a user to provide personal information to the application with knowledge of security.

Pre-conditions: The user must have begun recording a note.

Triggers: The user selects to save the note to the application.

Basic Course of Events:

Internal Pre-condition: None

Actor	System
1. The user saves a note to the application	
	2. The system initiates the “Record Save Trigger” use case
	3. The system initiates the “Save Note” use case.
	4. The system encrypts Note to add a level of security.
	5. The system saves note in the internal memory of phone.

Internal Post-condition: None

Post-conditions: None

3.1.14 Use Case Name: Favorite Pages

Summary: This feature allows a user to store notes that cannot be deleted directly from this page or from the application deleting settings.

Pre-conditions: The user must have saved notes.

Triggers: The user selects a note and selects as favorite.

Basic Course of Events:

Internal Pre-condition: None

Actor	System
1. The user opens note.	
1. User marks note as a favorite note.	
	System changes log system to favorite page.
2. User opens Favorites page.	
	Retrieve notes with favorites page.
	Present favorites to page

Internal Post-condition: None

Post-conditions: None

3.1.15 Use Case Name: Notifications

Summary: This feature reminds a user of the application and to remember to record notes.

Pre-conditions: The user has not updated or opened the system in 12 hours.

Triggers: Time trigger of 12 hours.

Basic Course of Events:

Internal Pre-condition: Last update recorded 12 hours ago.

Actor	System
	1. System checking time from last update or application open.
	2. System sends out notification to user to remind of recording abilities.

Internal Post-condition: None

Post-conditions: None

Alternative Paths:

A. The system finds that the last notification was within 12 hours.

Internal Pre-condition: None

Actor	System
	3. Notification is not sent.

Internal Post-condition: Condition returns to whether system must send out a notification.

Post-conditions: None

3.1.16 Use Case Name: Training Videos

Summary: This feature demonstrates training videos on how to use the application.

Pre-conditions: The User goes to settings page.

Triggers: User selects training videos section

Basic Course of Events:

Internal Pre-condition: None

Actor	System
1. User selects training videos page.	
	2. System retrieves training videos for application.
	3. The system organizes videos in order of going through application.
4. User selects a video.	
	5. The system plays video that was requested.
	6. Video is over, presents opportunity to continue to other videos or exit training.
7. Continues with training.	
	8. Proceeds to next video in order and plays the video.

Internal Post-condition: None

Post-conditions: None

Alternative Paths:

A. The user wants to exit training sessions.

Internal Pre-condition: None

Actor	System
7.Wants to exit the training	
	8.System exits the video and returns to page of listing videos.

Internal Post-condition: Condition returns to point where a video is stopped.

Post-conditions: None

4.0 Supplementary Requirements

Harkify will work with users who installed the application on the specified device. The users of the application are identified by their corresponding trained voice of the user. It can also be noted that the application is purely dependent on the internet. Although, it has limited capability in the offline environment.

5.0 Supporting Information

N/A

6.0 External Interface Requirements

6.1 User Interfaces

UI-1: The home screen of the system conforms to the application User Interface Standard, Version 1.1.

6.2 Hardware Interfaces

HI-1: The system shall store all data locally.

HI-2: The system shall be deployed in both Android and IOS operating systems.

6.3 Communication Interfaces

CI-1: The system shall recognize, train based on the user’s voice, record, speech synthesize, save, and search phrases by the user using the speech triggers.

7.0 Other Non-Functional Requirements

7.1 Performance Requirements

PE-1: The system shall accommodate a single user during the peak usage time windows of their specific local time, with an estimated average session duration of 10 seconds.

PE-2: All usable pages generated by the system shall be fully downloadable in no more than 10 seconds with a proper internet connection for generalized saved text notes.

7.2 Security Requirements

SE-1: The system shall only give access to the recognized and trained voice of the user every time they want to access the application.

SE-2: The system shall store saved notes of a 1-week duration, or the duration decided by the user in the “View Settings” use case.

SE-3: The system shall use AES 128-bit encryption for all text notes to ensure that any Personally Identifiable Information (PII) contained within them is protected from accidental access.