TEXT -TO-SPEECH AND SPEECH-TO-TEXT CONVERTOR

(Software Requirements Specification)

**Team guide**-

Prof. Lydia Jane

**Team Members**-

ISHAN KRISHNA (12BCE0501)

SHOBHIT GUPTA (12BCE0503)

**Table of Contents**

**Description................................................................................................... Page No.**

**1. Introduction**

1.1 Purpose.................................................................................................... 3

1.2 Scope....................................................................................................... 3

1.3 Background Information.......................... .............................................. 4

1.3 Definition, Acronyms, and Abbreviations.............................................. 4

1.4 Technologies Used.................................................................................. 5

1.5 References............................................................................................... 5

**2. Overall Description**

2.1 Product Perspective................................................................................. 5

2.2 Requirements........................................................................................... 6

2.3 Software Interface................................................................................... 7

2.4 Hardware Interface.................................................................................. 7

2.5 User Characteristics................................................................................. 8

2.6 Constraints............................................................................................... 8

2.7 Use Case Model Description................................................................... 9

**3. Project Planning**

3.1. Project Planning...................................................................................... 12

3.1.1 Gantt Chart…………………………………………………………… 12

3.1.2 Pert Chart…………………………………………………………….. 12

**1. Introduction**

**1.1. Purpose**

* To develop a Text-to-speech Android application which will convert the input text into voice. Further, Also incorporate Speech-To-Text in the same application. Also add an option to save the text that is converted from speech.
* Although the task of building very high quality, unlimited vocabulary text-to-speech (TTS) system is still a difficult one, with many open research questions, we will build an application with two to three language support.
* Our main objective for converting text to speech is to reduce the delay time. Delay time is the time difference between input text and the output speech. Lesser the delay time better will be the program and will not create confusion between the texts.
* We are looking forward to use as many as languages as we can which can serve people with different languages.
* Also we will be providing the user an option to save the converted speech in the form of a .wav file.

**1.2. Scope**

* Speech can serve as an excellent interface for sightless people, or people with motor neuron disorders. For some people who have some physical disability like blind people can use this program to listen to any text.
* Text to Speech is most helpful when it highlights the words as they are spoken. Dyslexic people say this focuses their attention and helps their understanding of the content. Some studies have shown that a combination of text-to-speech and highlighting improve reading skills.
* Text to Speech is most helpful when it highlights the words as they are spoken. Dyslexic people say this focuses their attention and helps their understanding of the content. Some studies have shown that a combination of TTS and highlighting improve reading skills.
* Text to Speech is also finding new applications outside the disability market. For example, speech synthesis, combined with speech recognition, allows for interaction with mobile devices via natural language processing interfaces.

**1.3. Background Information**

Text-to-speech (TTS) is the ability to play back written text as spoken words. As the Human-Computer Interfaces (HCI) come of age, the need for a more ergonomic and natural interface than the current one (keyboard, mouse, etc.) is being constantly felt. Talking of natural interfaces, what comes to mind, is sound (speech) and sight (vision). These form the basis of many intelligent systems research like robotics.

For people with **reading and writing difficulties**, having text reinforced by hearing it read aloud can be very useful. Specialised programs have existed to do this for a long time, and in many cases are extremely helpful and highly appropriate and should be seriously considered, perhaps in consultation with professional advice where necessary.

Our application is different from others in the market in the sense that it will give an option to save the converted text into a word document. This can enable the users to share the document with their friends who can make use of this converted document as per their requirement.

**1.4. Definitions, abbreviations and acronyms**

* **ADK**: Android Development Kit, What people use to develop anything for the Android such as APPs and ROM's.
* **adb**: Android Debug Bridge, a command-line debugging application included with the SDK. It provides tools to browse the device, copy tools on the device, & forward ports for debugging. If you are developing in Eclipse using the ADT Plugin, adb is integrated into your development environment.
* **Android** Unveiled on 5 November 2007, Android is a mobile operating system running on the Linux kernel developed by Google.
* **Dalvik**: An open source, register-based virtual machine (VM) that’s part of the Android OS. The Dalvik VM executes files in the Dalvik Executable (**.dex**) format & relies on the Linux kernel for additional functionality like threading & low-level memory management. The virtual machine is register-based, and it can run classes compiled by a Java language compiler that have been transformed into its native format using the included "dx" tool.
* **DDMS**: Dalvik Debug Monitor Service, a GUI debugging application included with the SDK. It provides screen capture, log dump, and process examination capabilities. If you are developing in Eclipse using the ADT Plugin, DDMS is integrated into your development environment.
* **JDK** - "Java Development Kit" an SDK for the java platform. It is needed to run the Android SDK.
* **JRE** - "Java Runtime Environment" a collection of binarys and files to allow java software to execute.
* **.apk or APK's**: An .apk file extension denotes an Android Package (APK) file, an .apk file can be opened & inspected using common archive tools. Each Android application is compiled and packaged in a single file that includes all of the application's code (.dex files), resources, assets, and manifest file. The application package file can have any name but must use the .apk extension. For example: myExampleAppname.apk. For convenience, an application package file is often referred to as an ".apk".

**1.5. Technologies Used**

The various technologies used in this software will be android , text-to-speech engine. All the definitions of these technologies are given above.

**1.6. References**

1. [www.google.com](http://www.google.com)

1. www.stackoverflow.com
2. www.developer.android.com
3. [www.wikipedia.org](http://www.wikipedia.org)

**2. Overall Description**

**2.1. Product Perspective**

The product is supposed to be an open source, under the GNU general Public License. It is an android based application. The application provides simple mechanism for users to convert written text into speech and speech into text.

The following are the main features that are included in the application:

* **Platform support**: The application will be able to run on all android devices.
* **Number of users** being supported by the system: Though the number is precisely not mentioned but the system is able to support a large number of online users at a time.
* **Text-to-speech:** The application will convert the written text into speech and play it to the user.
* **Speech-to-Text**: The system will also convert speech into written text and display it to the user.
* **File Saving**: The system will be able to save the converted speech as a ".wav" file on the user's device.

The various modules that we will implement are:

* **Text-to-speech:** In this module, the application will convert the written text into speech and play it to the user.
* **Speech-to-Text**: In this module, the system will convert speech into written text and display it to the user.
* **File Saving**: In this module, the system will be able to save the converted speech as a ".wav" file on the user's device.

**2.2. Requirements**

1. Functional Requirements:

* Our application will convert text into speech. The text will be entered in a text box and when the button is pressed, the written text will be spoken loudly by the system.
* Our application will also convert speech into written text. This text can be saved in the form of a word document and can be shared with other people if required.

b) Non-Functional Requirements:

* **Availability:** This application will be available 24\*7 to the users so that the users can convert speech to text or text to speech whenever they need to.
* **Reliable:** The application should be reliable enough. It should give correct translation of speech to text.
* **Responsive:** The application should be fast and reduce the delay time in conversion.

**2.3. Software Interface**

API, an abbreviation of **a**pplication ***p***rogram ***i***nterface, is a set of [routines](http://www.webopedia.com/TERM/R/routine.html), [protocols](http://www.webopedia.com/TERM/P/protocol.html), and tools for building [software applications](http://www.webopedia.com/TERM/A/application.html). The API specifies how software components should interact and are used when programming graphical user interface (GUI) components.  A good API makes it easier to develop a [program](http://www.webopedia.com/TERM/P/program.html) by providing all the building blocks. A [programmer](http://www.webopedia.com/TERM/P/programmer.html) then puts the blocks together.

## Types of APIs

There are many different types of APIs for operating systems, applications or for websites. Windows, for example, has many [API sets](http://msdn.microsoft.com/en-us/library/windows/desktop/hh802935%28v=vs.85%29.aspx) that are used by system hardware and applications — when you copy and paste text from one application to another, it is the API that allows that to work.

Most [operating environments](http://www.webopedia.com/TERM/O/operating_environment.html), such as [MS-Windows](http://www.webopedia.com/TERM/M/MS_Windows.html), provide an API so that programmers can write applications consistent with the operating environment.  Today, APIs are also specified by websites. For example, Amazon or eBay APIs allow developers to use the existing retail infrastructure to create specialized web stores. Third-party software developers also use Web APIs to create software solutions for end-users.

## Popular API Examples

Programmable Web, a site that tracks more than 9,000 APIs, lists Google Maps, Twitter, YouTube, Flickr and Amazon Product Advertising as the most popular APIs.

1. [Google Maps API](https://developers.google.com/maps/): Google Maps APIs lets developers embed Google Maps on webpages using a JavaScript or Flash interface. The Google Maps API is designed to work on mobile devices and desktop browsers.

2. [YouTube APIs](https://developers.google.com/youtube/): YouTube API: Google's APIs lets developers integrate YouTube videos and functionality into websites or applications. YouTube APIs include the YouTube Analytics API, YouTube Data API, YouTube Live Streaming API, YouTube Player APIs and others.

3. [Flickr API](http://www.flickr.com/services/api/): The Flickr API is used by developers to access the Flick photo sharing community data. The Flickr API consists of a set of callable methods, and some API endpoints.

4. [Twitter APIs](https://dev.twitter.com/): Twitter offers two APIs. The REST API allows developers to access core Twitter data and the Search API provides methods for developers to interact with Twitter Search and trends data.

**2.4. Hardware Interface**

There are various hardware devices that are involved with this project .

For example our application can be installed on devices with android support such as mobile phones and tablets so that the application which is aimed at helping customer can come in handy .

The customers can press the button on any of these devices and can use the functionalities of text to speech as well as speech to text whenever he/she likes.

Our application also provides an option to save the converted speech as a ".wav" file. This can come in handy in later situations where a user wants to share the speech with some other person.

**2.5. User Characteristics**

The user is expected to be phone literate and should have an ability to read and understand English. He should be able to use a personal device such as a mobile phone with internet connectivity well. The main screen of the application will have a link to both the text-to-speech as well as speech-to-text functionalities. The user should have an intellect such that he is able to process simple instructions like mentioned above as well as fill up some simple fields such as text and file name.

**2.6. Constraints**

* Resource availability : Network Connectivity Required.

: Android Device Needed to run the Application.

* Free space needed on user's device to save the speech.
* Basic Skill for using mobile phone needed.
* Funding needed in case of putting the application on Google Play Store.

**2.7. Use Case Model**

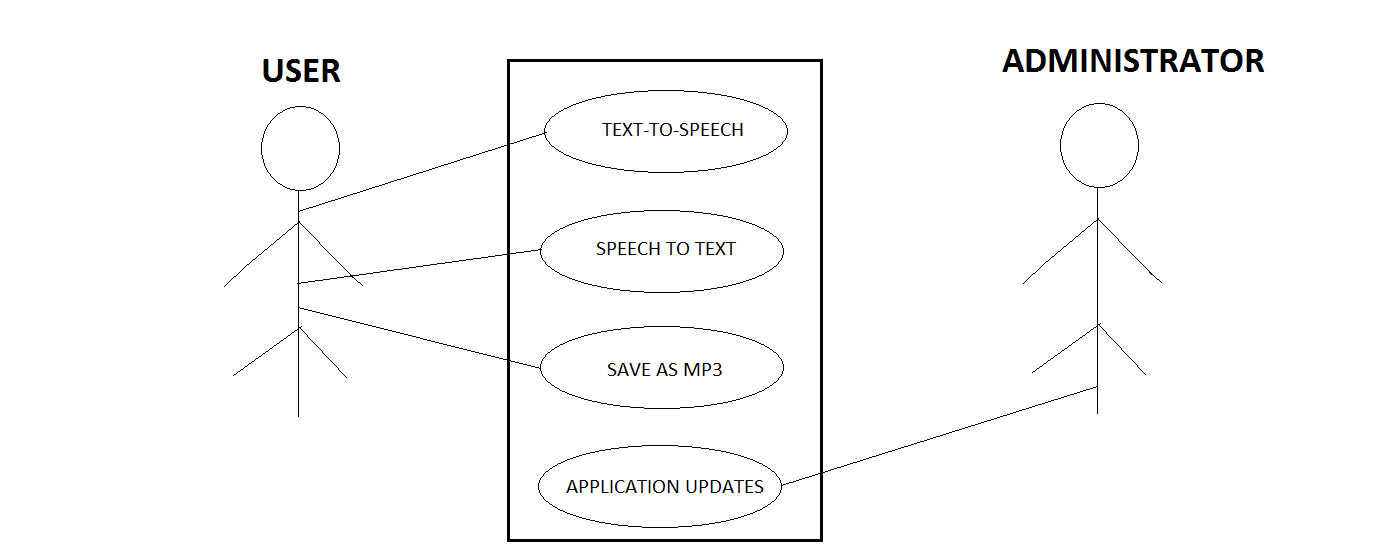


Fig. 1: Use case model

Table 1: Use case Description

|  |  |  |
| --- | --- | --- |
| Use Case ID | 1 | |
| [Use Case name](http://requirmentengineeringsrs.blogspot.in/) | Text -to-Speech Module | |
| Summary | It is responsible for converting input text into speech. | |
| Preconditions | Text should be entered into the text box. | |
| Success End Condition | Correct text recognition and delivery. | |
| Failed End Condition | Try Again. | |
| Primary, Secondary Actors | User. | |
| Trigger | This use case is initiated based on the request from the User. | |
| DESCRIPTION | Step | Action |
|  | 1 | The End user selects the Convert Button. |
|  | 2 | The text is automatically converted into speech and played out to the user. |

|  |  |  |
| --- | --- | --- |
| Use Case ID | 2 | |
| [Use Case name](http://requirmentengineeringsrs.blogspot.in/) | Speech to Text | |
| Summary | It is responsible for converting speech into written text. | |
| Preconditions | Verify that your mouthpiece is working correctly and you are not in a noisy environment and you are connected to the internet. | |
| Success End Condition | Speech is converted into text. | |
| Failed End Condition | Check network connection. Try again. | |
| Primary, Secondary Actors | User. | |
| Trigger | This use case is initiated based on the request from the User. | |
| DESCRIPTION | Step | Action |
|  | 1 | The End user selects the Convert Button. |
|  | 2 | The speech is automatically converted into text and displayed to the user. |

|  |  |  |
| --- | --- | --- |
| Use Case ID | 3 | |
| [Use Case name](http://requirmentengineeringsrs.blogspot.in/) | Save as mp3/.wav | |
| Summary | It is responsible for saving the converted speech in the form of an mp3/.wav file on user's device. | |
| Preconditions | File name should be provided. | |
| Success End Condition | File saved successfully. | |
| Failed End Condition | Check free space on the device. | |
| Primary, Secondary Actors | User. | |
| Trigger | This use case is initiated based on the request from the User. | |
| DESCRIPTION | Step | Action |
|  | 1 | The End user selects the convert Button. |
|  | 2 | The file is saved as "[filename].wav". |

|  |  |  |
| --- | --- | --- |
| Use Case ID | 4 | |
| [Use Case name](http://requirmentengineeringsrs.blogspot.in/) | Application Updates | |
| Summary | It is responsible for updating the application. | |
| Preconditions | The update should be made by the administrator. | |
| Success End Condition | Successfully Updated. | |
| Failed End Condition | Check for errors. | |
| Primary, Secondary Actors | Administrator. | |
| Trigger | This use case is initiated based on the request from the Administrator. | |
| DESCRIPTION | Step | Action |
|  | 1 | The administrator selects the updates and applies the updates on the application. |
|  | 2 | Put the updated application back in the market. |

**3.1. Project Planning**

**3.1.1. Gantt Chart**

Fig.: Gantt Chart

**3.1.2. Pert Chart**

Fig.: Pert Chart