**Software Requirements Specification**

**for**

**Text-to-Speech Synthesis**

**Version 1.0 approved**

**Prepared by:**

**Asim Aryal [NCE074BCT004]**

**Bimal Adhikari [NCE074BCT007]**

**Bishnu Khojwar [NCE074BCT011]**

**Bishworam Ghimire [NCE074BCT012]**

**Prepared for: Sharmila Bista**

**NATIONAL COLLEGE OF ENGINEERING**

**20****20**

# **Table of Contents**

[**Table of Contents** ii](#_Toc32529196)

[**1. Introduction** 1](#_Toc32529197)

[**1.1 Purposes** 1](#_Toc32529198)

[**1.2 Document Conventions** 1](#_Toc32529199)

[**1.3 Intended Audience and Reading Suggestions** 1](#_Toc32529200)

[**1.4 Product Scope** 1](#_Toc32529201)

[**1.5 References** 2](#_Toc32529202)

[**2. Overall Description** 2](#_Toc32529203)

[**2.1 Product Perspective** 2](#_Toc32529204)

[**2.2 Product Functions** 3](#_Toc32529205)

[**2.3 User class and characteristics** 4](#_Toc32529206)

[**2.4 Operating Environment** 5](#_Toc32529207)

[**2.5 Design and Implementation Constraints** 5](#_Toc32529208)

[**2.6 User Documentation** 5](#_Toc32529209)

[**2.7 Assumption and Dependencies** 5](#_Toc32529210)

[**3. External Interface Requirements** 6](#_Toc32529211)

[**3.1 User Interface** 6](#_Toc32529212)

[**3.2 Hardware Interfaces** 6](#_Toc32529213)

[**3.3 Software Interfaces** 7](#_Toc32529214)

[**3.4 Communication Interfaces** 7](#_Toc32529215)

[**4. System Features** 7](#_Toc32529216)

[**4.1 Signup** 7](#_Toc32529217)

[**4.2 Login** 8](#_Toc32529218)

[**4.3 Language Selection** 8](#_Toc32529219)

[**4.4 Generate speech** 9](#_Toc32529220)

[**5. Other Non-functional Requirements** 9](#_Toc32529221)

[**5.1 Performance Requirements** 9](#_Toc32529222)

[**5.2 Safety Requirements** 9](#_Toc32529223)

[**5.3 Security Requirements** 9](#_Toc32529224)

[**5.4 Software Quality Attributes** 9](#_Toc32529225)

[**5.5 Business Rules** 10](#_Toc32529226)

[**Appendix A: Glossary** 11](#_Toc32529227)

[**Appendix B: Analysis of Models** 12](#_Toc32529228)

# **1. Introduction**

This software requirements document specification provides complete information about the Text-to-Speech synthesis system which will be developed by our project team. The system is planned as software with good user interface for the user. In this section, we are going to give the introduction of the purpose, document conventions, intended audience and reading suggestion and scope of this document, definitions, acronyms and abbreviations, references and overview of the project.

## **1.1 Purposes**

The purpose of this document is to describe the requirements for the Test-to-Speech Synthesis system. Developers should consult this document and its revisions as the only source of requirements for the project. They should not consider any requirements statements, written or verbal as valid until they appear in this document or its revisions.

## **1.2 Document Conventions**

This document is prepared using Microsoft Word 2016 and has used the font type ‘Times New Roman’. The fixed font size that has been used is 12pt with 1.5 line spacing. It has used the bold property (18pt) to set headings of the document and (14pt) bold for sub-headings. All pages except the cover page are numbered; the numbers appear on the upper right-hand corner of the page. Every image and data table are numbered and referred to main text. Use case diagram is written with IEEE convention.

## **1.3 Intended Audience and Reading Suggestions**

The intended audience of this document would be client, developers, testers, project managers, project teams, supervisor. The SRS document can be used in any case regarding the requirements of the project and the solutions that have been taken. The document would provide clear idea about the system that is building.

## **1.4 Product Scope**

The proposed software product is the Text to Speech Synthesizer. The system will be used to get the audio form of the text entered. Also, the documents in .txt,.pdf,.docx format can be converted into speech with the help of this system. Till now, there are many application and software related to Text-to-speech conversion. People have done this using different programming language and platform, and for different-different purposes, and for different languages.

Text to speech software can be enormously helpful for anyone who's a condition like dyslexia that makes reading on screens tricky. It can also help overcome language barriers for people who are in the process of learning. Text to speech software is also ideal if we want to listen to a document while doing something else, or if we want to sense-check something we've written. It is also a valuable computational aid for those with speech disorders.

The scope of the SRS is basically for everyone involved to understand and have an idea about how and what is going to happen in the system. Using ER, use case diagrams which are in a form where everyone can understand how the interfaces finally appear.

## **1.5 References**

[1].Ian Sommerville, software Engineering

[2].Roger S.Pressman, Software Engineering-A practitioner's Approach

[3].https://www.academia.edu

# **2. Overall Description**

## **2.1 Product Perspective**

The project is about the development of the Speech Synthesis System that could be used input text to analyse, synthesize and generate the output in the form of the audible sound. Speech synthesis is the process of translating text into human understandable speech. It is known as Text-to-Speech (TTS) synthesis. The system model only uses the speech synthesis engine to convert text into speech. Hence, speech recognition engine is not an issue. During speech synthesis, first analysis of the text takes place that comprises of several steps like prosody analysis, structure analysis and text-to-phoneme conversion. In prosody analysis, speech attributes like the pitch, pausing, timing, peaking rate and others are concerned. After completion of these analysis the text’s prosody and structural information is gathered for conversion of it into phonetic or any linguistic description. Further, the prosody and phonetic information are used for the generation of speech waveforms using any of the two ways. One is to use words of pre-recorded human speech and concatenate them for full speech generation and other is to use signal processing techniques that are based on the mechanisms the phonemes sound and how they are affected by the prosody (Preston, 2005). Like that of BOSS (Bonn Open Synthesis System), a cross-platform developer framework for researchers in the field of speech synthesis, most of the traditional speech-based applications were written in C++ or other native languages (Breuer and Hess, 2010). However, the project is developed using Java programming language because of its integration capability with other language platforms.

## **2.2 Product Functions**

The Text to Speech Synthesis (TTS) functions can be described as follows:

* Text Processing:
  + To analyze, normalize and transcribe text into a phonetic or some other linguistic representation.
  + To deal with low level processing issues such as sentence and word segmentation.
  + To handle abbreviation and acronyms and match text by normalization.
  + To perform morphological analysis for proper word pronunciation and syntactic analysis to facilitate accenting and phrasing to handle ambiguities in text by linguistic analysis.
* Speech generation:
  + To generate the speech using different parameters as mentioned below.
  + Phonetic analysis to focus on phone level within each word about what sound to produce and how to produce.
  + To determine extra pronunciation of each word by grapheme to phoneme conversion.
  + To identify a word tense system.
  + Phonological prosodic processing i.e analysis of prosody for marking prosodic effect.
  + To create an abstract descriptive system which characterizes the behavior of the prosody within the acoustic signal.
  + To create a phonological system which can be used to input to process which eventually result in acoustic signal jugged by listeners to have proper prosody.
* Speech synthesis:
  + To generate speech by synthesizer using specific synthesis techniques such as articulator, formant and concatenative synthesis.

## **2.3 User class and characteristics**

The Text to Speech Synthesis (TTS) will be used in android and in pc’s as well. The TTS makes it easier in general for all people by ensuring that information is available on both written and audio format. The system is also designed to be user friendly.

* Client

They are the ones who take service from the TTS and are the general users of the system. They can know about the features and functions of the system.

Functions

1. Can convert direct text into speech.
2. Can choose the male or female voice
3. Can upload the desired document for processing of speech.
4. Can get information about using the system.

## **2.4 Operating Environment**

Smartphone with following requirements:

Software Requirements

* Any web browser that supports HTML5

Hardware Requirements

* 512MB RAM
* 128MB free memory space

## **2.5 Design and Implementation Constraints**

* The system development should be completed before December 28th 2020.
* Access to storage of smartphone is required.
* Users should download the application on their phone in order to use it.
* The system should be user friendly.
* Smartphone with newer operating system is required.
* Speech is only available in English language.

## **2.6 User Documentation**

As soon as the app is downloaded step by step tutorial (interactive video) is played showing user where to type the text on the screen and how to convert it to speech. It will also introduce user with all the options available to the user like speed of the audio played, voice options of the audio and formatting of the written text. After the initial tutorial is completed user can access video of each section individually if needed from Menu >>Tutorials.

## **2.7 Assumption and Dependencies**

* Users know basics of how to use smartphone.
* Users understands English language.
* Users will allow app the access to shared storage.

# **3. External Interface Requirements**

## **3.1 User Interface**

User Interface of our application will be user friendly. The user is expected to know basics of how to use smartphone and write on the smartphone in the English language. The interface of the application can be divided into five parts. They are: -

**Text box interface**

It is the big white space on the middle of the screen where input text is written as soon as the interface is touched keyboard appear for typing.

**File Interface**

Users can use this button to open any text file which will be converted into speech.

**Speed interface**

This button on bottom left of screen will be able to change the speed of the audio output. The options to either increase or decrease the speed of audio will be available.

**Formatting Interface**

In this interface located at middle top there will be options for formatting input text color, size, and font.

**Convert Interface**

In this interface there will be a button register which when clicked will convert text written into speech.

## **3.2 Hardware Interfaces**

Smartphone: Software is installed in the smartphone by downloading it from Google Play Store or Apple App Store. Smartphone should be able to take text as input and give speech (audio) as output.

## **3.3 Software Interfaces**

Developing end

* Python v3.9.1
* Pycharm2020.2.2: IDE for Java developing
* Adobe Photoshop CC 2020 –Designing such as User Interface

## **3.4 Communication Interfaces**

WIFI: For connection of the client with the cloud server.

# **4. System Features**

## **4.1 Signup**

4.1.1: Description and Priority:

This feature registers the user to the system.

Priority: Medium

4.1.2: Response Sequence:

* User selects the signup option
* Login form is displayed by system
* User fills the required details
* New account is created by the system

4.1.3 Functional Requirements:

The user mustn’t have created the account with the same details before. If already created, the system should generate the error saying that the details are already used. The user should be connected to internet to use this feature. If not, generate an error saying connect to the internet. User must provide the valid email addresses and other details. If not valid, throw an error saying the details are invalid.

## **4.2 Login**

4.2.1 Description and Priority:

This feature provides login feature.

Priority: High

4.2.2 Response Sequence:

* User selects login option
* System displays login box
* User enters valid id and password
* System logins the user to the application

4.2.3 Functional Requirements:

The user must have created an account. If not, the system should generate a msg saying to sign up. The user should be connected to internet to use this feature. If not, generate an error saying connect to the internet. The user must enter valid id and password otherwise generate an error saying invalid id/password.

## **4.3 Language Selection**

4.3.1 Description and priority:

This feature allows user to select the language.

Priority: Medium

4.3.2 Response Sequence

* User selects the language selection option
* System shows the list of available language
* User selects a language from the list

4.3.3 Functional Requirements

The user must have logged in to the system to access this feature. If not suggest user to log in.

## **4.4 Generate speech**

4.4.1 Description and Priority:

This feature allows user to generate the speech of the text entered.

Priority: High

4.4.2 Response sequence:

* User press the speak button
* System generate the speech

4.4.3 Functional Requirements:

The text must have been entered in the text box. If there is no text generate the error saying enter the text to generate speech.

# **5. Other Non-functional Requirements**

## **5.1 Performance Requirements**

* Response time- The system will generate speech within 1 sec of clicking the button.
* Capacity- The system must support 10,000 words at a time.

## **5.2 Safety Requirements**

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage.

## **5.3 Security Requirements**

Each user has their own login id and password. User must login to the system to choose a language and to generate the speech.

## **5.4 Software Quality Attributes**

AVAILABILITY: The system shall be available all the time.

CORRECTNESS: Bug free software which fulfill the correct need/requirements of the client.

MAINTAINABILITY: The ability to maintain, modify information and update fix problems of the system.

USABILITY: software can be used again and again without distortion.

ACCESSIBILITY: Administrator and many other users can access the system but the access level is controlled for each user according to their work scope.

STABILITY: The system outcome/output won’t change time to time. Same output will be given always for a given input.

## **5.5 Business Rules**

* Won’t take the responsibility of failures due to hardware malfunctioning.
* Additional payments will be charged for various languages.
* Advertisement will be displayed at any time.
* No money back returns for the software.

# **Appendix A: Glossary**

SRS - Software Requirements Specification

TTS- Text to speech

IEEE- Institute of Electrical and Electronics Engineers

GUI- Graphical User Interface

Sign up- Creating the new account.

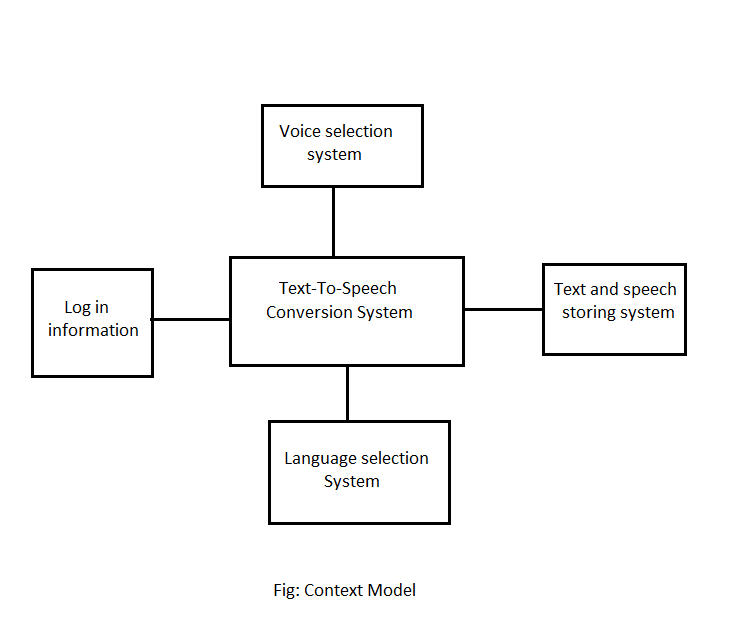
Login ID - A user identification number to enter the system

Password - A word that enables one to gain admission into the system.

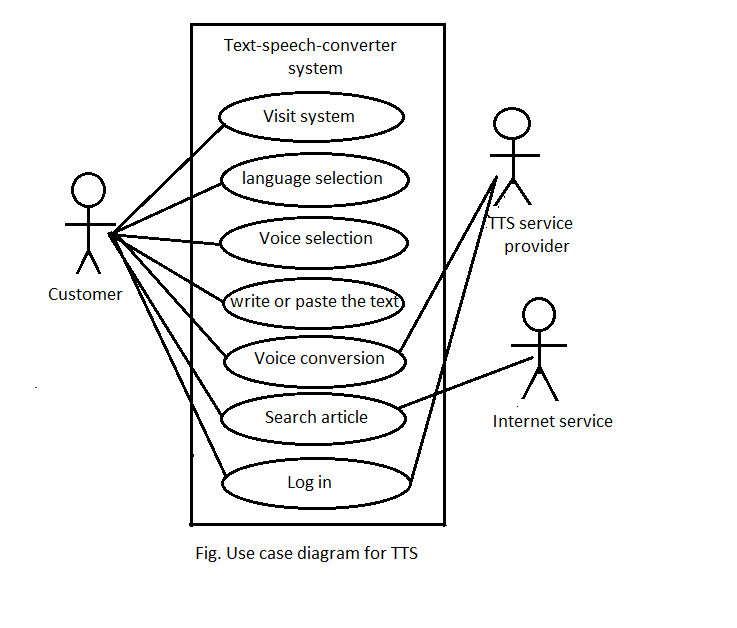
RAM-Random Access Memory

WIFI- Wireless Fidelity

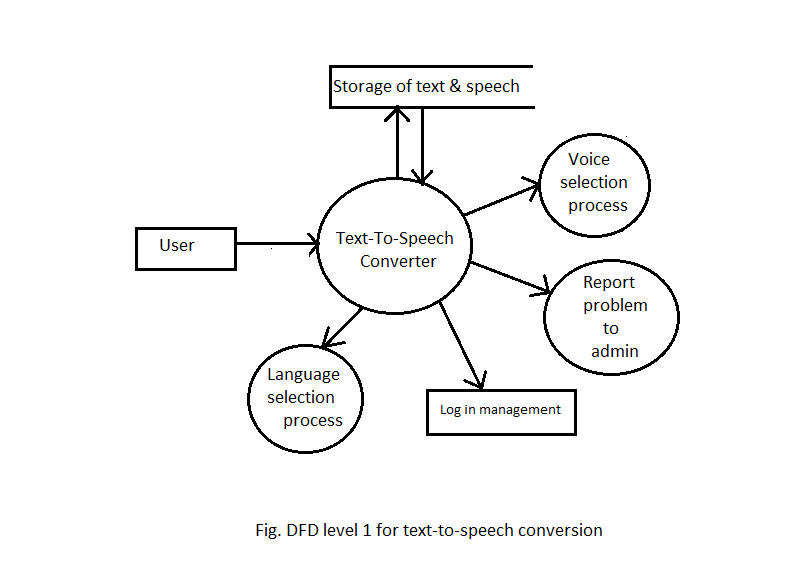
# **Appendix B: Analysis of Models**

1. **Context Model**

Above figure shows the structure of Text-To-Speech conversion system. User entry get details of a user and add record to the log in information file. Language selection selects a language in which the user is going to write or paste the text. Voice selection selects the type of voice in which the text is going to be read. Finally, the TTS converter converts the given text to speech and the user can listen to the given text.

1. **Use Case Diagram**

Above diagram shows the use case diagram for text to speech conversion system. It consists of three actor costumer, Text-to-speech service provider and internet service provider. It also consists of seven use cases namely visit system, language selection, voice selection, write or paste the test, voice conversion, search article and log in. User perform all the seven use cases while TTS service provider perform voice conversion and store and retrieve log in information. Costumer can also search different article form the internet service provider and finally can listen to that article as well.

1. **DFD**

Above diagram shows the DFD level 1 diagram. It consists of only 1 entities user. It consists of four process enclosed in the oval, and also it consists one data base for data store enlisted in open rectangles. Any user can write text of different language and listen to it. User can easily enter to the system by simple log in. Then he/she can select the language and write or paste the text to the system. User can also select the type of voice, for example male voice or female voice and then the system convert the text to voice. User can also store the data, either text or speech to the system. System also provide facility to user to report any problem they are facing while using the system.