Software Requirements Documentation for

<TalkBox>

Version Alpha 1.0 approved

Walker, Matt – 214779300  
Kaplan, Allen – 215494925  
Sakib, Saadman – 215916232

Lassonde School of Engineering

2019-02-01

Table of Contents44

Table of Contents ii

Revision History (Master branch) iii

1. Introduction 1

1.1 Purpose 1

1.2 Intended Audience and Reading Suggestions 1

1.3 Product Scope 1

1.4 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 3

2.7 Assumptions and Dependencies 3

3. Testing 5

3.1 Test Cases 5

3.2 Test

# Introduction

## Purpose

To produce software for the embedded TalkBox system. Achieved through the development of two separate applications: The TalkBox simulator – a piece of software that simulates the behavior of any TalkBox device and the TalkBox configurator – a user-friendly GUI based application which allows for the configuration of a TalkBox device with appropriate audio.

Typographical conventions used in this SRS are as follows; **bolded** words represent significant ideas and technical terms while underlined text represents titles

## Intended Audience and Reading Suggestions

This document is intended for project managers and clients. This SRS contains an overall description of the project, external interface requirements, *use cases,* and *test cases.* It is suggested to read this document sequentially, beginning with the overview *section* and proceeding through the technical sections as one sees fit.

## Product Scope

Goals:

The development of the following:

* TalkBox Simulator App
* TalkBox Configurator App

## References

[1] "proj [EECS2311]", Wiki.eecs.yorku.ca, 2019. [Online]. Available: <https://wiki.eecs.yorku.ca/course_archive/2018-19/W/2311/proj>. [Accessed: 04- Feb- 2019].

[2] Ehud Reiter. "SimpleNLG”, GitHub, 2019. [Online]. Available: <https://github.com/simplenlg/simplenlg>. [Accessed: 04- Feb- 2019].

[3] DFKI’s Language Technology Lab and the Institute of Phonetics at Saarland University."marytts/marytts", GitHub, 2019. [Online]. Available: <https://github.com/marytts/marytts>. [Accessed: 04- Feb- 2019].

[4] "The TalkBox Community Research Project", Talkbox.apps01.yorku.ca, 2019. [Online]. Available: http://talkbox.apps01.yorku.ca/. [Accessed: 04- Feb- 2019].

# Overall Description

## Product Perspective

The TalkBox simulator and configuration apps are derived from the TalkBox Community Research Project, which has been undertaken by faculty and students of the Lassonde School of Engineering at York University to provide a low-cost open-source, open-hardware Speech Generating Device for those who need such devices.

## Product Functions

TalkBox Simulator:

* Accurately simulate the behavior of any TalkBox device
* Is fully tested to behave as the hardware device
* Provide a clean, user-friendly interface which allows for the generation of sentences via sequential button inputs
* Provide Text-to-Speech output

TalkBox Configurator:

* Enable the configuration of TalkBox device qualities such as;
  + Record and select audio sets
  + Dictionary selection
  + Text-to-Speech voice selection
* Provide a way to launch the TalkBox Simulator app
* Create a directory called TalkBoxData which contains a serialized object (of type .tbc) and relevant audio files
* Provide the TalkBox Simulator app with the file path to the TalkBoxData directory

Both the TalkBox Simulator and the TalkBox Configurator apps must be able to communicate through the use of a TalkBoxConfiguration object that will be serialized.

## Operating Environment

Both the TalkBox Simulator and the TalkBox Configurator will operate in the JVM environment. This environment will communicate with a RaspberryPi through a USB which will act as the input/output device.

## Design and Implementation Constraints

The TalkBox Configurator must implement the TalkBoxConfiguation interface. This interface defines the following functions:

* Get number of physical buttons that when pressed will play an audio file
* Get the number of sets of audio files that this configuration supports
* Get the total number of buttons in this TalkBox
* Get a Path relative to this configuration object where all audio files can be found
* Get a 2-dimensional array of Strings that contains the names of all audio files.

## User Documentation

* Software Requirements Specification (this document)
* Testing Document
* User Manual

## Assumptions and Dependencies

SimpleNLG: The TalkBox Simulator app uses maven to import the SimpleNLG library as a dependency. SimpleNLG is used to realize sentences given keyword input from the user.

MaryTTS: The TalkBox Configurator app uses maven to import the MaryTTS library as a dependency. MaryTTS is used to generate spoken word output from the sentences realized by SimpleNLG.

# Testing

## Test Cases

* Capitalizing the first letter of the sentence
* Making sure the verbs use the correct tenses and make it agree with the subject
* Adding prefixes ‘-ing’ or ‘ed’ to the infinitives needed as such. E.g (I asked her a question) / (I was asking her a question)
* Making sure all the words are in correct grammatical form and make sense when used in a sentence.
* Appropriate whitespace between the words of the sentence and putting a period at the end of the sentence.
* Inserting line breaks between words (rather than in the middle of a word) in order to fit text into rows of, for example, 80 characters (or whatever length you choose).

## Test Case Derivation

Our objective is to make sure the people who are using our software can express themselves the way they really want. These test cases help shape up the sentences being produce and grammatically correct them out as much as possible, making sure the other person is clearly able understands them.

By selecting the subject or noun, various verbs and words will pop up helping the user generate the appropriate sentences.

# 