REQUIREMENTS DOCUMENT

By Sarah Feroz, Sara Attalla, and Sean Doyle

Software Development Project
EECS 2311

Table of Contents

Software Development Project: Requirements Document (version 1.0)

Project: EECS 2311 Software Development Project – The Authoring App
Date: January 30th, 2018
Prepared by: Sarah Feroz
Reviewed By: Sara Attalla, Sean Doyle

Document status: __ Draft _X_ Proposed __ Validated __ Approved

1. Introduction

This requirement document has been prepared by Team 7 to describe the user needs, required features, acceptance test cases and their sufficiency for this project. In this phase of the project we allow the user to construct scenarios for their students through the use of GUI's.

1.1 Purpose of This Document

This document is intended to guide development of ***The Authoring App***. This document will help describe and demonstrate the use of The Authoring App. Its aim is to inform the reader of the capabilities and features of the application to aid in a seamless execution. It will provide sufficient information for the user to understand the scenario building phase of the program.

1.2 Overview

The Authoring App is an app that is intended for the use of educators to aid in accessibility for the hearing and visually impaired. The objective of this app is to imitate the function of a Treasure Box Braille. A Treasure Box Braille is known as a device that helps kids learn how to read Braille. This is done by presenting Braille characters/words to the user who can then respond by pressing buttons. This device helps to ensure that children are able to learn Braille in an engaging manner, and also provides the educators with an option to load and create different scenarios.

This software provides a user-friendly interface that helps the educators to create, edit and play different scenarios for education purposes. The simulator also includes screen reading capabilities for individuals who are visually impaired to aid them in working the program and computers.

1.3 Scope and Overview of the Requirements

The goal of this project is to create our software in a user-friendly way, this includes allowing individuals who may have no back ground in software, and/or technology to be able to easily use the program and navigate through swiftly. The program is catered towards visually impaired users by implementing a text-to-speech feature into the existing screen readers. Our aim is to have the program be very simple and instantly convert any story line/ scenario that the educator may have written directly into braille cell and onto the screen reader.

- Software must be usable for visually-impaired users
- Software must be able to work with screen reader
- Software must be able to support 2/3 of the three platforms: Linux, Windows, Mac
- Must have an easy-to-use lay out
- All labels must be clear and flow from each window to another
- Scenario files can be created, edited, opened, and saved.
- User should only be able to see the simplified versions of the scenarios, more complex versions for the program to read are hidden.
- Must be able to rearrange, add and delete events in a scenario.
- Handle user errors

2. General Description

This section will give the reader an overview of the project, including why it was conceived, what it will do when complete, and the types of people we expect will use it. We also list constraints that were faced during development and assumptions we made about how we would proceed.

2.1 Product Perspective

This application was inspired by the Treasure Braille Box, abbreviated as TBB. The TBB presents Braille characters/ words to users and awaits the respond through buttons. The main downside of the TBB is that the device is very expensive to obtain, making it very hard for an educator to afford. To solve this problem, the Authoring App was developed as a downloadable application for multiple operating systems. The authoring app serves the need for the visually impaired to be able learn efficiently and effectively through the application. The primary stakeholders are the teachers who would like to use this application for educational purposes and the visually impaired who would benefit from the execution of this application. Both parties will benefit from the finished product of the Authoring App.

2.2 Use Cases

A use case is known as a methodology used to identify, clarify, and organize possible sequences of interactions between systems and users. The Authoring App opens to a main menu that gives the user options to play an existing scenario, edit an existing scenario, learn about the creators, and exit the application. If the user chooses the option to play an exiting scenario, a file chooser will be opened, leading to the current file directory on the operating system. Once the file has been chosen, the authoring app activates the screen reader, and allows the scenario to be played in an engaging manner.

If the user presses the button to edit a scenario, a window is opened asking the user whether it would like to create a new scenario or edit an existing scenario. If the user decides to create a new scenario, a window is opened which consists of simple text boxes for the requirements of the scenario. These requirements include the question for the main section of the scenario, an option to choose which button on the Braille cell represents the correct answer, and a response option for when the student gets the answer correct or incorrect.

Alternatively, if the user chose the option to edit an existing scenario, a new window would be opened where a list would show the current timeline of all the existing scenario events, where the user could choose which event they would like to edit. Once they have chosen the event, they would then be redirected to the previous window asking for the question, correct button answer, and response statements. Once the event was saved, it would be overwritten in the timeline and ready to be used with the screen reader.

On the main menu, if the user presses the button to learn about the authors, the window is redirected to display the names of the creators in a list and state the purpose of development. Lastly, if the user presses the “Exit” button, the user is exited out of the application, allowing it to end.

2.3 User Characteristics

The user of our finished product is expected to be both an educator, as the implementor, and a visually impaired individual gaining knowledge through the app. This is because the educator will be able to effectively provide exercises for individuals to gain an understanding of Braille. The technical background for both the educator and the student may be close to null, as the application has been created in order to provide step by step instructions for easy use. The application has been made to be exceptionally interactive to allow the users to be at ease knowing that no complications will come their way

2.4 General Constraints

While developing this application, there were some general constraints that our team had to work around. The main constraint was the time constraints. The customer is typically counting on the company to have a finished product at a certain time. Working to deliver exceptional results while simultaneously pleasing the client was the biggest constraint in completing this product. Another constraint was making our product compatible with the software given to the team from the professor. We had to ensure that any implementation we did conformed with the code given prior to.

3. Acceptance Test Cases

Acceptance Test Cases are known as tests conducted to determine whether specific requirements of the project are met.

Execution Procedure	Expected result	Pass or Fail
Main menu opens during the execution of the Simulator	Main menu displays the various options for the user to select	Pass
User pressing buttons on menus	The program will proceed depending on the button pressed and its requirements	Pass
Hovering over a button with a mouse to use accessibility features	System will use screen reader to read the titles of the specified buttons	Pass
Simulator allows a file to be chosen when "Play a Scenario" option is chosen	File chooser allows user to pick a file in their documents	Pass
File chooser plays Scenario File chosen	Interactive learning experience will begin	Fail
Pins in cells reset after every event is completed	All cells are cleared	Pass
Simulator should allow the selected event on timeline to be edited for removed	Edit and Remove button should become enabled to allow for action	Pass
Save function allows for files to be saved	Scenarios are able to be saved in a file to be accessed afterwards	Pass

4. Glossary

GUI – Graphical User Interface
TBB – Treasure Box Braille

5. References

- Multiple API's that are provided by Java Oracle (ex. clear, remove etc.)
- TA in 2311 Lab Section
- Professor Bil Tzerpos