

# Requirements Document

## Treasure Box Braille: Authoring App

Group 19: Connor Ahearn, Micah Arndt, and Kevin Chan

# Table of Contents

<b>1.0 Document Overview</b>	<b>3</b>
<b>2.0 Requirements</b>	
<b>2.1</b> Main Requirements	<b>3</b>
<b>2.2</b> Additional Requirements	<b>4</b>
<b>2.3</b> Accessibility Requirements	<b>5</b>
<b>3.0 Use Cases</b>	
<b>3.1</b> Create scenario	<b>5</b>
<b>3.2</b> Save Scenario	<b>6</b>
<b>3.3</b> Load Scenario	<b>6</b>
<b>3.4</b> Test Scenario	<b>7</b>
<b>3.5</b> Save Section	<b>7</b>
<b>3.6</b> Set Pins on Braille Cell	<b>8</b>
<b>3.7</b> Record Audio	<b>8</b>
<b>3.8</b> Import Audio	<b>9</b>
<b>3.9</b> GUI Audio Accessibility	<b>10</b>
<b>3.10</b> Hot Keys	<b>10</b>
<b>3.11</b> Logger	<b>10</b>
<b>4.0 Acceptance Test Cases</b>	<b>11</b>

## 1.0 Document Overview

This document aims to provide a detailed description of the system requirements as developed with the client for the Treasure Braille Box (TBB) program. It includes a list of the project requirements, use cases for the authoring app, as well as acceptance test cases when using the Treasure Braille Box program.

## 2.0 Requirements

This section provides a closer look at the functional and accessibility requirements of the program, as well as an explanation of the design choices made to accommodate the requirements.

### 2.1 Main Functional Requirements

This section details the key requirements of the Scenario Creator, which is the authoring app for the Treasure Braille Box program, as set forth by the client at the onset of the project.

Requirement Description	Design
The program will allow users to create and determine the flow of the scenario (ask questions, receive answers).	<p>The Scenario Creator provides a Story text field that allows users to write questions and an Answer text field that determines the button associated with the correct response. Additionally, the Correct and Incorrect text fields allow the user to determine the response that is triggered after a button is pressed.</p> <p>Each question and answer is broken up into different sections. Users are able to create as many sections as they want for their scenario, and they can load and edit different sections as they work on their scenario.</p>
The program allows users to save the scenario in an appropriate format.	After users have completed and saved at least one section, the Scenario Creator allows users to save their scenario. The program will then automatically create, format and save a .txt file on the user's system.
The program allows users to test the scenario using provided software.	The Scenario Creator allows users to test saved scenarios using software included in the Treasure Braille Box program. Once the test scenario option is selected, users will see a graphic user interface (GUI) that simulates a braille cell.

The program will allow users to record sound.	The Scenario Creator allows users to record sounds and add it into the Story, Correct or Incorrect text fields. After they have finished recording, a .wav file will be saved onto the user's system.
The program needs to be compatible with 2 out of the 3 main operating systems (Mac, Windows, and Linux).	Our implementation of the Treasure Braille Box program is fully functional on all systems.

## 2.2 Additional Functional Requirements

This section highlights additional functional requirements for the Treasure Braille Box software. These are either smaller requirements needed to help achieve the main requirements, or, optional features that aim to provide a more comprehensive experience for the user.

Requirement Description	Design
The program allows users to set the pins displayed on the braille cell.	The Scenario Creator provides a Braille field that lets the user type the letter they want displayed on the Braille Cell. The software will automatically set the correct pins to be displayed.
The program allows users to load and edit a previously saved Scenario.	The Scenario Creator provides users with the ability to load and edit scenarios in the Scenario Creator program, and save any changes they might have made.
The program allows users to import .wav sound files.	To compliment the ability to record sound files, the Scenario Creator also provides users with the ability to import .wav sound files into their scenarios.
The program contains a logger, that keeps track of accessed features and errors that occur.	The Scenario Creator keeps a log of features that were used and errors that occur in the software. A .txt containing the log is saved onto the user's system.

## 2.3 Accessibility Requirements

This section highlights the accessibility features of the Scenario Creator.

Requirement Description	Design
The program needs to have audio accessibility for visual-impaired users.	The Scenario Creator is compatible with the audio accessibility features of the 3 main operating systems. Audio descriptions are provided for all text as well as changes that occur in the graphical user interface.
The main features of the program should include keyboard shortcuts in order to improve accessibility.	The Scenario Creator provides users with keyboard short cuts to the main functions of the program. Similarly, hot keys are used in order to allow users to quickly navigate through the various sections of the graphical user interface with their keyboard.

## 3.0 Use Cases

This section details the main ways in which users can interact with the Treasure Braille Box, specifically, the Scenario Creator software.

3.1_Create Scenario
<b>Description:</b> User starts a new scenario
<b>Actor:</b> User
<b>Preconditions:</b> <ul style="list-style-type: none"><li>1) User selects a name for the scenario they want to create.</li><li>2) User indicates the number of braille cells they have available.</li><li>3) User selects the number of answer buttons available.</li></ul>
<b>Basic flow of events:</b> <ul style="list-style-type: none"><li>1) User starts Scenario Creator.</li><li>2) User chooses Create New Scenario in starting window.</li><li>3) User names the scenario.</li><li>4) User indicates the number of braille cells and answer buttons available.</li></ul>
<b>Exceptions:</b> <ul style="list-style-type: none"><li>1) Scenario Name field cannot be empty. The system will warn the user to enter a scenario name.</li><li>2) Braille Cells Available field must contain a number. The system will warn the user to enter a number if the field is left empty or a non-numerical character is entered in this field.</li></ul>

<p><b>3)</b> Answer Buttons Available field must contain a number. The system will warn the user to enter a number if the field is left empty or a non-numerical character is entered in this field.</p>
<p><b>Post-conditions:</b> The user will be brought to the main Scenario Creator window, where they can create different sections for their scenario.</p>

3.2 _Save Scenario
<p><b>Description:</b> User saves a scenario as a text file</p>
<p><b>Actor:</b> User</p>
<p><b>Preconditions:</b></p> <ol style="list-style-type: none"> <li>1) At least one section has been completed and saved</li> </ol>
<p><b>Basic flow of events:</b></p> <ol style="list-style-type: none"> <li>1) User fills out the Section Name, Answer Buttons Used, Story, Braille and Answer fields in the main Scenario Creator window. The correct and incorrect fields may be left empty.</li> <li>2) User saves the section.</li> <li>3) User can choose to write and save more sections.</li> <li>4) User selects the Save Scenario option in the Scenario menu.</li> </ol>
<p><b>Exceptions:</b></p> <ol style="list-style-type: none"> <li>1) At least one section has to be saved before the scenario itself can be saved. The system will warn the user to save a section before trying to save the scenario.</li> </ol>
<p><b>Post-conditions:</b> The scenario will be saved and a text file will be saved on the user's system.</p>

3.3 _Load/Edit Scenario
<p><b>Description:</b> User edits a previously created scenario</p>
<p><b>Actor:</b> User</p>
<p><b>Preconditions:</b></p> <ol style="list-style-type: none"> <li>1) User has to have already created and saved one scenario. This implies that there is also at least one section that has been created and saved.</li> </ol>
<p><b>Basic flow of events:</b></p> <ol style="list-style-type: none"> <li>1) User selects the Edit Scenario button in the first window that appears after opening the Scenario Creator program. Users may also select the Load Scenario option in the main window.</li> <li>2) In both cases, a file chooser window will appear, prompting users to select the file they would like to edit.</li> <li>3) After selecting an appropriate file, users can use the combo box to select which section they would like to review and edit.</li> </ol>
<p><b>Exceptions:</b></p> <ol style="list-style-type: none"> <li>1) If the user selects a file that is not compatible with the Scenario Creator program, the system will warn the user to select another file.</li> </ol>

**Post-conditions:** Users can make changes to individual sections. They will then be able to save changes to these sections as well as the scenario itself.

### 3.4\_Test Scenario

**Description:** User is able to test out a created scenario using a simulated braille cell

**Actor:** User

**Preconditions:**

- 1) User must have created and saved one scenario

**Basic flow of events:**

- 1) In the opening window, user selects the Test Scenario option or in the main Scenario Creator Window, user selects the Test Scenario option from the Scenario menu or shortcut.
- 2) User can then choose between using a Visual Player which will visually simulate the braille cell or Audio Player, which will accompany the visual display with audio accessibility options.
- 3) A file chooser will appear and prompt the user to choose a scenario to test.

**Exceptions:**

- 1) If the user chooses a file that is not compatible with the Scenario Creator, the system will prompt the user to choose another file.

**Post-conditions:** The scenario will be loaded and the user can test out the scenario.

### 3.5\_Save Section

**Description:** User saves a section

**Actor:** User

**Preconditions:**

- 1) User has selected Create a New Scenario in the opening window.
- 2) User has named the scenario and indicated the number of braille cells and answer buttons available.
- 3) User completes the Section Name, Answer Buttons Used, Story, Braille and Answer text fields.

**Basic flow of events:**

- 1) User starts Scenario Creator and chooses Create New Scenario.
- 2) User provides a name for the scenario, and indicates the number of braille cells and answer buttons available.
- 3) User fills out the Section Name, Answer Buttons Used, Story, Braille and Answer text fields.
- 4) Correct and Incorrect text areas are completed (optional).
- 5) User presses the Save Section button or uses the Save Section shortcut from the menu.

**Exceptions:**

<ol style="list-style-type: none"> <li>1) Section Name field cannot be empty. The system will warn the user to enter a section name.</li> <li>2) Answer Buttons Used field cannot be empty, contain a non-numerical character, or a number greater than the number of answer buttons available indicated in the previous window. The system will warn the user of these conditions and prompt them to enter another number.</li> <li>3) The Story text area cannot be empty. The system will warn the user to write something in the Story Text Area.</li> <li>4) The Braille text field cannot be empty, contain a non-alphabetical character, or a word that has more numbers than the number of braille cells available as indicated in the previous window. The system will prompt the user to enter another word.</li> <li>5) The Answer field cannot be empty, contain a non-numerical character, or a number that is greater than the number of answer buttons used. The system will prompt the user to enter another number.</li> </ol>
<b>Post-conditions:</b> The section that the user was working on is saved. The section name appears in the drop down combo box.

3.6_Set Pins on Braille Cell(s)
<b>Description:</b> User sets the letter / word they want displayed on the braille cells
<b>Actor:</b> User
<b>Preconditions:</b> <ol style="list-style-type: none"> <li>1) User has selected Create New Scenario or Edit Scenario in the opening window.</li> <li>2) User has named the scenario and indicated the number of braille cells and answer buttons available.</li> </ol>
<b>Basic flow of events:</b> <ol style="list-style-type: none"> <li>1) In the main Scenario Creator window, the user indicates the letter or word they would like displayed on the braille cell in the Braille text field.</li> </ol>
<b>Exceptions:</b> <ol style="list-style-type: none"> <li>1) The word entered can only contain alphabetical characters and cannot contain more letters than the number of braille cells available as indicated in the previous window. The system will warn the user to enter only alphabetical characters, and that the number of letters has to be less than or equal to the number of braille cells available.</li> </ol>
<b>Post-conditions:</b> The pins on the braille cells will display the word that the user has indicated in the braille text field when the scenario is run.

3.7_Record Audio
<b>Description:</b> User record audio
<b>Actor:</b> User
<b>Preconditions:</b> <ol style="list-style-type: none"> <li>1) User has selected Create New Scenario or Edit Scenario in the opening window.</li> </ol>



<ol style="list-style-type: none"> <li>2) User has named the scenario and indicated the number of braille cells and answer buttons available.</li> <li>3) User clicks the add sound button or uses the menu shortcut in the main Scenario Creator window.</li> </ol>
<b>Basic flow of events:</b> <ol style="list-style-type: none"> <li>1) In the main Scenario Creator window, the user clicks the Add Sound button or uses the menu short cut.</li> <li>2) The user selects the record sound option.</li> <li>3) A new window will pop up, prompting the user to enter a name for the sound file and to enter the length of the recording in seconds.</li> </ol>
<b>Exceptions:</b> <ol style="list-style-type: none"> <li>1) The user has to name the sound file. If the sound file name field is empty, the system will prompt the user to enter a name.</li> <li>2) The user has to indicate the length of time they want to record for in seconds. Otherwise, the system will default to only record for 60 seconds.</li> </ol>
<b>Post-conditions:</b> A .wav sound file will be saved onto the user's system. The user can then add the sound file they created anywhere into the story, correct, or incorrect text field.

3.8_Import Audio
<b>Description:</b> User imports an audio file into their story
<b>Actor:</b> User
<b>Preconditions:</b> <ol style="list-style-type: none"> <li>1) User has selected Create New Scenario or Edit Scenario in the opening window.</li> <li>2) User has named the scenario and indicated the number of braille cells and answer buttons available.</li> <li>3) User clicks the add sound button or uses the menu shortcut in the main Scenario Creator window.</li> </ol>
<b>Basic flow of events:</b> <ol style="list-style-type: none"> <li>1) In the main Scenario Creator window, the user clicks the Add Sound button or uses the menu short cut.</li> <li>2) The user selects the import sound option.</li> <li>3) A file chooser window will pop up allowing users to select the sound file they want to import.</li> </ol>
<b>Exceptions:</b> <ol style="list-style-type: none"> <li>1) If the user tries to import a file that does not have a .wav extension, the system will warn the user that only .wav files are excepted, and they are prompted to select another file from the file chooser.</li> </ol>
<b>Post-conditions:</b> A .wav sound will be imported for use in the Scenario Creator. The user can then add the sound file they imported anywhere into the story, correct or incorrect text field.

3.9_GUI Audio Accessibility
<b>Description:</b> User can have audio descriptions of the GUI and actions they perform on the system.
<b>Actor:</b> User
<b>Preconditions:</b> The user needs to have a computer that has screen reader software enabled.
<b>Basic flow of events:</b> <ol style="list-style-type: none"> <li>1) User needs to enable the screen reader on their computer</li> <li>2) For creating and editing scenarios, the audio accessibility function will already be enabled when the screen reader is on.</li> <li>3) When the user chooses Test Scenario, they can select the Audio Player option to turn on the Audio Accessibility function.</li> </ol>
<b>Exceptions:</b> <ol style="list-style-type: none"> <li>1) Audio accessibility will not be functional if the computer that is being used does not contain a screen reader program.</li> </ol>
<b>Post-conditions:</b> The computer will use audio to describe what element of the GUI the user is focused on, along with the purpose of that component. The system will also tell users about actions that are taken.

3.10_Hot Keys
<b>Description:</b> User can use hot keys to quickly transverse through the different elements of the GUI and access functions of the system.
<b>Actor:</b> User
<b>Preconditions:</b> <ol style="list-style-type: none"> <li>1) User has selected Create New Scenario or Edit Scenario in the opening window.</li> <li>2) User has named the scenario and indicated the number of braille cells and answer buttons available.</li> </ol>
<b>Basic flow of events:</b> <ol style="list-style-type: none"> <li>1) Once in the main Scenario Creator window, users can quickly navigate between different elements in the GUI. They can also use hot keys (labeled in the menu) to quickly access commonly used functions.</li> </ol>
<b>Exceptions:</b> N/A
<b>Post-conditions:</b> Users will be able to quickly navigate back and forth between different components of the GUI. They can also use hot keys to jump to a specific element in the GUI. Users can also use hot keys to access functions related to sound, sections and scenarios.

3.11_Logger
<b>Description:</b> The program creates a log of functions used and exceptions thrown
<b>Actor:</b> Program
<b>Preconditions:</b> <ol style="list-style-type: none"> <li>1) User has selected Create New Scenario or Edit Scenario in the opening window.</li> <li>2) User has named the scenario and indicated the number of braille cells and answer</li> </ol>

buttons available.
<b>Basic flow of events:</b> 1) Once in the main Scenario Creator window, the system will automatically log key events and errors into a text file that is saved on the user's system.
<b>Exceptions:</b> N/A
<b>Post-conditions:</b> A text file containing a log for the key functions accessed and error messages is created

#### 4.0 Acceptance Test Cases

This section details the software testing conducted to determine the acceptability to deliver the system to the client. The test results are aimed to illustrate the software's compliance with the requirements as developed together with the client.

Acceptance Test	Results / Comments
Users can determine the flow of their scenario. They can demonstrate what a letter looks like or ask questions and receive answers wherever they want in the scenario.	Pass  // Breaking the scenario up into sections allows users to easily manage their scenario.
User is able to save their scenario in an appropriate format.	Pass  // A .txt file is created and saved in the system. The software formats the text so that it can be tested using the provided software.
User is able to test a saved scenario.	Pass  // User can test a selected file from the starting menu and within the Scenario Creator as well.  Note: Due to an incompatible voice directory, Mac OS users should instead test their scenarios through the ToyAuthoring App included within the software package.

User can record their own audio track and add it into their scenario.	Pass  // A .wav file is created and saved in the system.
The program is compatible with 2 out of the 3 main operating systems.	Pass
User can set the pins they want to display on the braille cell.	Pass  // This version will only use one cell
User can load and edit their scenario.	Pass  // User can do so directly from the starting window or in the Scenario Creator.
User can import sound files.	Pass  // imported sound file must be of the .wav format.
A log of the main features is created.	Pass  // The log is saved in the system as a .txt file.
The software needs to have audio accessibility features.	Pass  // The software is designed to be user friendly for the visually impaired. Audio descriptions are provided for each button and text field. Tool tips are used to provide new users with audio prompts on what to enter in each text field.  Audio accessibility is also used to provide feedback and describe events that occur on the screen.
A log of the main features accessed in the software will be kept on the user's system.	Pass  // A .txt file is created and saved.