Braille Box Authoring App

Requirements Document
EECS 2311: Software Engineering Project
Team 9
May 19, 2018

Authors

Jeremy Winkler

Nisha Sharma

Tyler Thomson

Revision Level	Date	Pages affected	Comments
1.0.0.0	May 19, 2018	ALL	Final Submission
1.0.0	February 23, 2018	1-6	Midterm Submission
1.0	February 05, 2018	1	Initial draft

Table of Contents

List of Figures

1.0 Introduction

This document specifies the required features of the scenario creator Authoring App software. The Authoring App software allows educators to create, save, edit and test educational scenarios for visually impaired users. The program is capable of accepting text inputs and audio files of ".wav" format for creating interactive scenarios. The program also allows educators to record and save new audio files. Current version just allows to play audio in record mode after saving the file, however, the final version will have a play feature allowing to play selected audio files. By following the simple input syntax, the educators can create educational scenarios and save them as ".txt" files. This document lists the system overview, system requirements, standard use cases and acceptance tests.

1.1 System Overview:

The system under development (SUD) is the Authoring App, a computer based user interface that allows educators to create flow of educational scenarios to be used with a braille device called Treasure Box Braille (TBB). TBB is a device that helps kids learn how to read Braille. TBB presents Braille characters/words to the user who then responds by pressing buttons. Educators can load different scenarios to teach different letters and keep the kids entertained. Authoring App is a piece of software to help educators create these scenarios. The Authoring app will provide facilities to

- Create the flow of the scenario (ask questions, receive answers)
- Record audio
- Save the scenario in the appropriate format
- Test the scenario using the provided software

The Authoring App must also be usable by visually-impaired users.

To accomodate for limited number of TBBs (and also because they are expensive and hard to replace), a software that simulates the TBB is used. This way, as an educator is preparing a scenario with using the Authoring App, they can try it out to see if it works. The system must log all actions the user takes and keep a count of how many times each feature is accessed. The system should have keyboard shortcuts to the most frequently used features. Such features should be easy to access with a mouse as well.

This requirements document follows above mentioned specifications.

2.0 Requirements

This section describes what the Authoring App software system does for the client. These requirements have been elicited from the specifications given by the client and further interviewing the client and potential users.

2.1 Key Requirements:

Here are the key requirements from the software application as requested by the client to suite the need of potential customers.

Requirement Number	Brief Description	Comments and Rationale
REQ1	Software shall allow the users to create the flow of scenarios (ask questions, receive answers).	The users should be able to add a prompt/question and should also be able to receive a response/answer in a flow.
REQ2	Software shall allow the users to edit the scenarios.	The users should be able to edit a previously created and saved scenario anytime. The users should also be able to save those changes.
REQ3	Software shall allow the users to record audio in *.wav* format.	The users should be able to record an audio file and save it in .wav format to be inserted in the scenario file as needed.
REQ4	Software shall allow the users to test/play scenarios.	The users should be able to test/play their saved scenarios before potentially handing them over to their students to interact with.

2.2 User Requirements:

Following is an important user requirement for the Authoring App.

Requirement Number	Brief Description	Comments and Rationale
REQ5	The Authoring App shall be usable by visually-impaired users.	The authoring app *must* be accessible and user-friendly for visually impaired users. The assumption is that those users have a screen reader software installed on their devices and they know how to interact with it.

2.3 System Requirements:

Following is the system requirement for successfully running the Authoring App.

Requirement Number	Brief Description	Comments and Rationale

Requirement Number	Brief Description	Comments and Rationale
REQ6	The Authoring App shall be able to run successfully on Windows, Linux and Mac operating systems.	The authoring app should be fully functional on at-least two of the operating systems.

2.4 Additional Requirements:

Some other requirements are:

Requirement Number	Brief Description	Comments and Rationale
REQ7	The Authoring App shall allow users to set number of cells and buttons at the start of scenario creation.	After interviewing the client, we reached at mutual conclusion that only finite number of cells and buttons are needed. Client indicated that having an option to have upto *10 cells* and upto *6 buttons* would be sufficient. The users should be able to raise pins on all those cells and should be able to add functionality to all the buttons.
REQ8	The Authoring App shall allow users to add audio responses in scenario.	For example, on a button click an audio file could be played to indicate right or wrong selection.
REQ9	The Authoring App shall allow users to save files in .txt format.	This is the desired format to test the scenario file using TBB simulator.
REQ10	The Authoring App shall allow users to edit a file previous saved as .txt file with correct format.	A correctly formatted text file should have first two lines indicating the number of cells and number of buttons, as shown below: Cell 1 Button 4 Also every other specification mentioned in the "Scenario File Format Documentation" [1] should be met in order for scenario file to be properly formatted.

2.5 New Requirements:

Following requirements have been requested by the client after first release:

Requirement Number	Brief Description	Comments and Rationale

REQ11	The Authoring App shall keep count of number of times an action is used.	Your system must log all actions the user takes and keep count of how many times each feature is accessed.
REQ12	The Authoring App shall have keyboard shortcuts for most commonly used functions.	Based on the log data obtained during testing, the app should have keyboard shortcuts for the most frequently used features. Such features should be easy to access with a mouse as well.

3.0 Use Case

The use cases give information about how different users interact with an application or system (Authoring App here). The target users of the Authoring App are educators teaching visually impaired students. There are many ways users can interact with the app. This section includes most common use cases of the Authoring App. Each use case is illustrated briefly with a textual use case followed by a use case diagram is provided for main success scenario.

3.1 UC1: Standard Success Scenario

This use case describes the normal operation of the Authoring App by the educators. Here is one flow of actions for creating, saving and testing a scenario.
Precondition: The App has been installed on the operating system (Linux, Windows or Mac) under consideration.
Primary Actor: Educator
Main Success Scenario:
1. **Launch** the Authoring App by double clicking the app icon.
2. **Create** a **New** Scenario by clicking the "New" button on the user interface being displayed or using shortcut Ctrl+n.*
3. **Save** the created scenario once done by clicking the Save option in the menu or by using keyboard shortcut Ctrl+s in a text file.
4. **Edit** the saved scenario selecting the edit option on the Authoring App user interface.
5. **Save** the changes made to the scenario similar to step 3.
6. **Test/Play** the saved scenario on TBB simulator.
7. **Exit** the Authoring App once done.

* Some actions can have multiple actions associated with it. For example, "Create a New Scenario" would have actions associated with setting number of cells and buttons, setting the title, adding an audio file, recording audio files, adding questions, raising pins, adding response actions, reordering cards/scenario sections and so on.
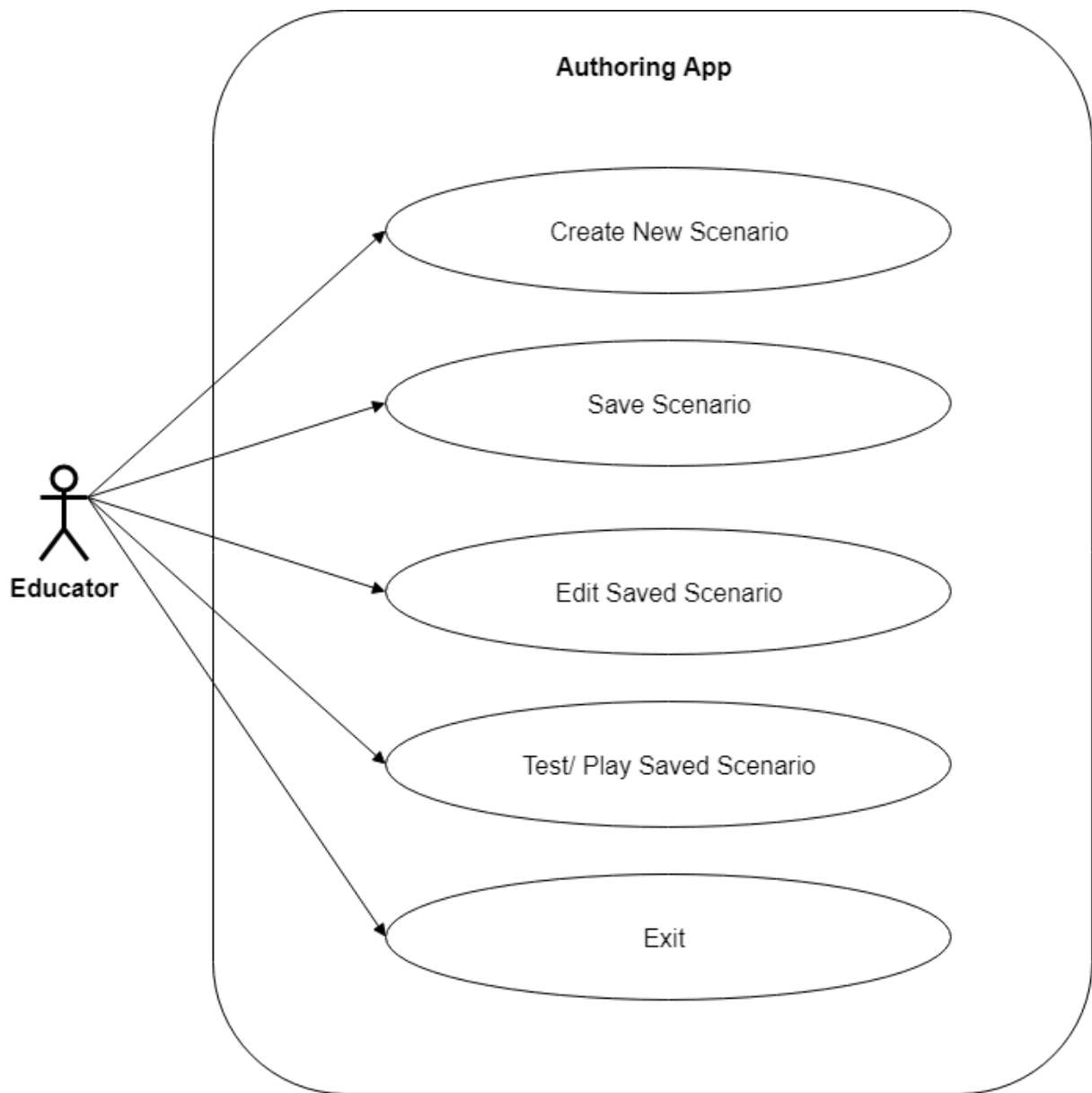
Figure 1. Use case diagram for UC1: Standard Success Scenario

3.2 UC2: Create a Scenario involving user responses

Precondition: The App running on the operating system.
Primary Actor: Educator
Main Success Scenario:
1. Select the option to create a New Scenario.
2. Set up the number of cells and buttons for the scenario.
3. Set up the title of the scenario
4. Enter some text for text-to-speech feature.
5. Select the option to enable user response.
6. Enter details for relative path the scenario takes for a particular user response. (This response may involve raising pins, playing an audio etc).
7. Save the Scenario by selecting "Save" option.

8. Exit the App.

3.3 UC3: Create a Scenario without user responses

Precondition: The App running on the operating system.
Primary Actor: Educator
Main Success Scenario:
1. Select the option to create a New Scenario.
2. Set up the number of cells and buttons for the scenario.
3. Set up the title of the scenario
4. Enter some text for text-to-speech feature.
5. Add some other features to the scenario, for example raising pins on braille cell, display a character and playing an audio file etc.
6. Save the Scenario by selecting "Save" option.
7. Exit the App.

3.4 UC4: Record Audio

Precondition: The App running on the operating system.
Primary Actor: Educator
Main Success Scenario:
1. Select the option to create a New Scenario.
2. Set up the number of cells and buttons for the scenario.
3. Set up the title of the scenario
4. Select the option to Record an audio file.
5. From the Recorder user interface start recording audio.
6. Once done, save the audio file in .wav format.
7. Play the audio.
8. Exit the recorder user interface.
9. Exit the App.

3.5 UC5: Edit a Scenario

Precondition: The App running on the operating system and a previously created scenario file.
Primary Actor: Educator
Main Success Scenario:
1. Select the option to Edit a Scenario.
2. From the Editor user interface, select the option to edit the number of buttons and cells.
3. Increase number of cells and buttons by 2.
4. Change is successfully reflected.
5. Edit other details of the scenario file.
6. Save the made changes a separate scenario file.
7. Exit the App.
Alternate Course:

1. The new number of buttons and cells is less than the ones in previously saved scenario.
2. A warning is given before finalizing the changes.
3. User agrees and proceeds with steps 4 - 7 as normal.

3.6 UC6: Rearrange the components of a scenario

Rearranging ordering of questions.
Precondition: The App running on the operating system.
Primary Actor: Educator
Main Success Scenario:
1. Select the option to create a New Scenario.
2. Set up the number of cells and buttons for the scenario.
3. Set up the title of the scenario
4. Enter some text for text-to-speech feature asking a question.
5. Select the option to enable user response.
6. Enter details for relative path the scenario takes for a particular user response. (This response may involve raising pins, playing an audio etc).
7. Create another section within the scenario following steps 4 - 6.
8. Move the section created in step 7 before the section created in steps 4 - 6.
9. Change is successfully reflected.
10. Save the made changes to a scenario file.
11. Exit the App.

A similar use case can be constructed for rearranging the components for a pre-existing scenario file.

3.7 UC7: Test a Scenario

Precondition: The App running on the operating system and a previously created and saved scenario file.
Primary Actor: Educator
Main Success Scenario:
1. Select the option to Test a saved Scenario.
2. From the options menu select the file to be tested.
3. The app launches the software that simulates the TBB.
4. The simulator software runs the scenario file on the simulator.
5. The scenario finishes.
6. Close the simulator software.
7. Exit the App.
Alternate Scenario:
1. While the scenario file is running on the simulator software, the user closes the software.
2. Simulator successfully closes stopping the audio playback from the file.
3. Exit the app.

4.0 Acceptance Test Case

This section list the acceptance tests (performed by users to test the real life use of the app).

Acceptance Test	Description	Success Criteria
AT0	The Authoring App launches successfully on all three specified operating systems (Windows, Linux and Mac)	The app launches successfully on all platforms displaying the user interface with four options to create a New scenario, Edit or Test a scenario and Exit the app..
AT1	The Authoring App allows users to edit scenario file. Rationale: User should be able to load a properly formatted scenario '.txt' file in the authoring app to edit different components associated with it.	The scenario file to be edited loads successfully and all the components of the scenario file can be edited.
AT2	Inserting audio files to a scenario. Rationale: User should be able to insert audio files of '.wav' format in the scenario. This audio would be played while running test simulator.	Audio file successfully inserted to the scenario file in proper format and the audio gets played correctly during the simulation testing.
AT3	Recording a new audio file using the Authoring App. Rationale: User should be able to record new audio files and save them in '.wav' format.	The app successfully launches user interface to record the audio. The audio files gets successfully saved in specified folder with specified name in .wav format on choosing to save.
AT4	Raising pins on the TBB simulator after specifying inside a scenario file created using Authoring App. Rationale: User should be able to raise pins particular pin on braille cell in TBB simulator.	The intended pins get raised during simulation testing.
AT5	Specify finite number of Cells and Buttons for a scenario during creation of scenario using the Authoring App.	Successfully set the number of Cells and Buttons to a

	Rationale: User should be able to specify finite number of cells and buttons for a particular scenario on creation.	finite number (as mentioned in the specifications).
AT6	Enable user response for a scenario. Rationale: User should be able to ask a question and receive a response in terms of button clicks.	User input successfully enabled.
AT7	Respond to a button click with an audio and text-to-speech feedback. Rationale: User should be able to associate a sound file and text feedback with each button click.	Audio file and text-to-speech component related to a button response gets successfully simulated on TBB simulator.
AT8	Rearrange different sections of a scenario with relative ease. Rationale: User should be able to rearrange the order of different question and response segments of scenario.	Different components can be successfully moved up or down during the creation or editing of a scenario file.
AT9	Test a saved scenario file created using the Authoring App. Rationale: User should be able to successfully test a saved scenario file with TBB simulator.	The TBB simulator successfully launches from within the authoring app and successfully runs the scenario file created using the Authoring App.
AT10	Insert Pause to the scenario.	The app allows to successfully insert pauses at different times within the scenario.
AT11	Display a character and or word on the braille cell.	The app allows user to successfully set a character or word to be displayed.
AT12	Reset pins or lower pins after some time.	The app allows users to successfully specify action to lower the pins on braille cell.
AT13	Use a keyboard shortcut to perform a common task.	The app allows user to easily use most commonly used features using a

	Rationale: For example use Ctrl+N to create a new file.	keyboard shortcut.
AT14	Perform an action involving button click or choosing an option in the software and see how many times a feature was used. Rationale: Based on most frequently used features, design easy access keyboard shortcuts.	The app records a log of actions keeping a count of how many times it was accessed. The log is displayed either on terminal/console or saved as a logfile.

Therefore, major success criteria for a potential Authoring App is:

1. User should be able to load a properly formatted scenario '.txt' file in the authoring app to edit different components associated with it.
2. User should be able to insert audio files of '.wav' format in the scenario. This audio would be played while running test simulator.
3. User should be able to record new audio files and save them in '.wav' format.
4. User should be able to raise pins particular pin on braille cell in TBB simulator.
5. User should be able to specify finite number of cells and buttons for a particular scenario on creation.
6. User should be able to ask a question and receive a response in terms of button clicks.
7. User should be able to associate a sound file and text feedback with each button click.
8. User should be able to rearrange the order of different question and response segments of scenario.
9. User should be able to successfully test a saved scenario file with TBB simulator.

5.0 References

[1] B. Tzerpos,"Scenario File Format Documentation" *EECS 2311 – Software Development Project*, https://wiki.eecs.yorku.ca/course_archive/2017-18/W/2311/_media/scenarioformat.pdf. [Accessed: 21- Jan- 2018].