User Manual

TreBBA: Treasure Braille Box Authoring

Version 2.0

EECS2311 Submission: Group 10

Arsh Punia

Table of contents

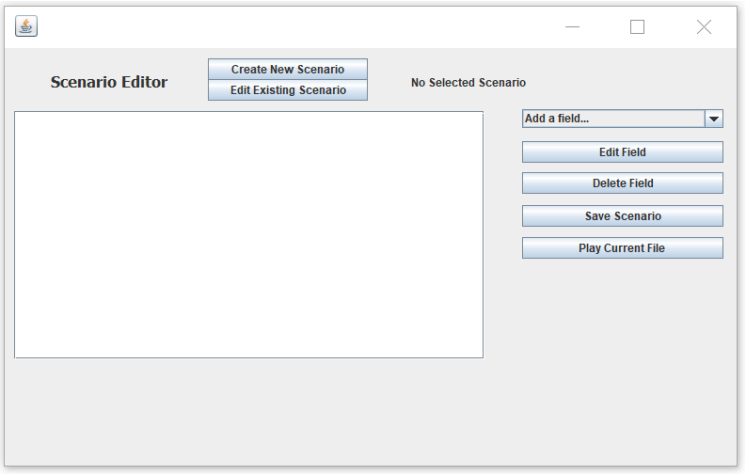
Here are the detailed instructions to now download the project:

1. Download the Source Code (zip) file from https://github.com/Shogri/EECS2311-TBB/releases/tag/1.0
2. Open the zip folder and navigate as: \EECS2311-TBB-1.0.zip\EECS2311-TBB-1.0\Enamel
3. Click on the treeBBA-app
4. You should now see the main interface for TreBBA

Documentation:

The documentation for version 1.0 of TreBBA can be found at: https://github.com/Shogri/EECS2311-TBB/tree/master/Enamel/Documentation

User interface:



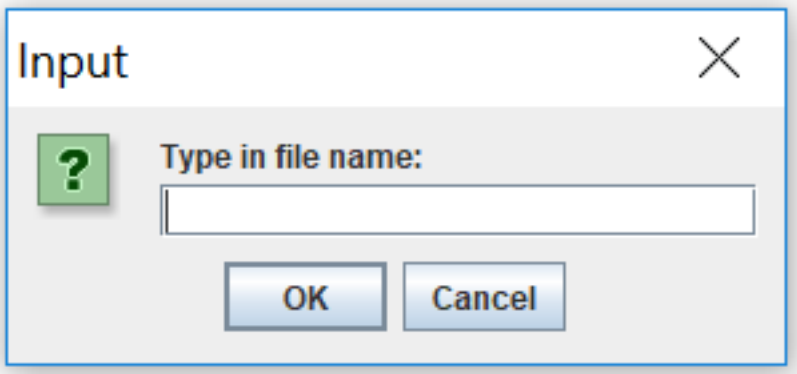
Upon launching the .jar file, the TreBBA user interface pops up. In the top left corner, one can see the option to either create a new file, or edit an existing file. To its right is the name of the chosen file. In this figure, that field is blank as no file has been chosen.

Below the Create/Edit buttons is a plain area which is where the user sees what they have input, and what their scenario file is structured like.

To the right of the plain area, one can see various buttons. On the top is a drop-down menu which enables you to select all the fields that you would like to add. These functions are explained in later sections.
To the bottom of the drop-down menu are buttons to edit a field, delete a field, save the scenario, and to play a given scenario. These functionalities too, are explained in later sections.

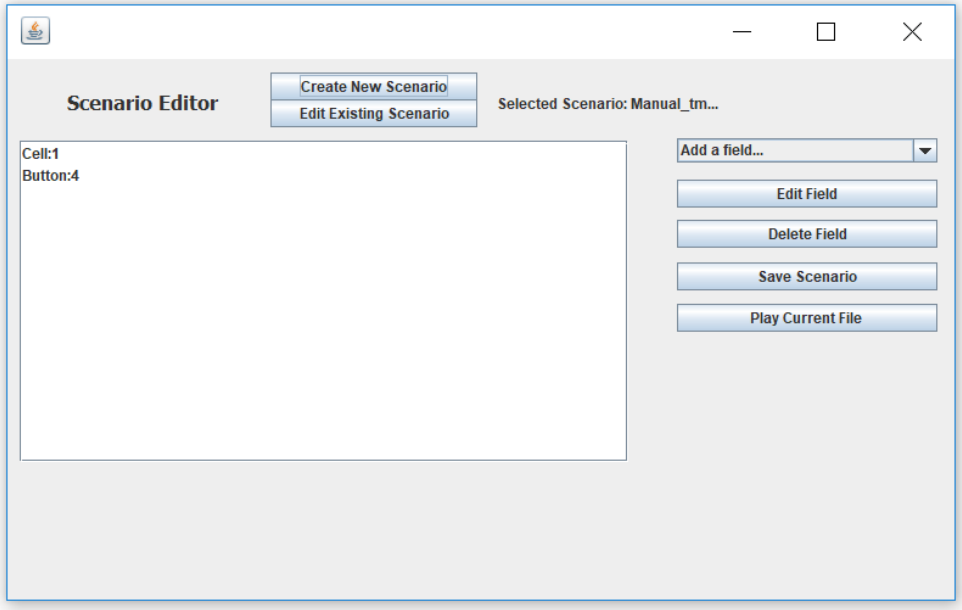
Getting Started

To get started, launch the.jar file. Once the TreBBA window pops up, click on "Create New Scenario" to get started on making a new scenario. This will lead to a popup asking for the file name. By default, files are saved in the same folder as the .jar file.



After you've entered the file name, there is another prompt for you to enter the number of cells and buttons.
'Cells' means the number of Braille cells that you would like to emulate.
'Buttons' Means the number of interactive buttons you would like to have in your scenario.

The number of cells and buttons that you would like has to be separated by a space.



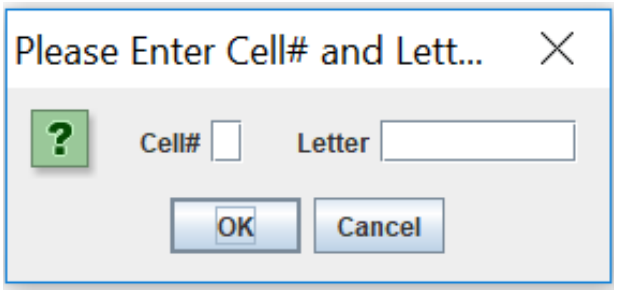
In the example above, the user has chosen to add 1 cell and 4 interactive buttons in a file called Manual.

DROP DOWN

The drop-down menu on the right of the plain area supports multiple operations which are as follows:

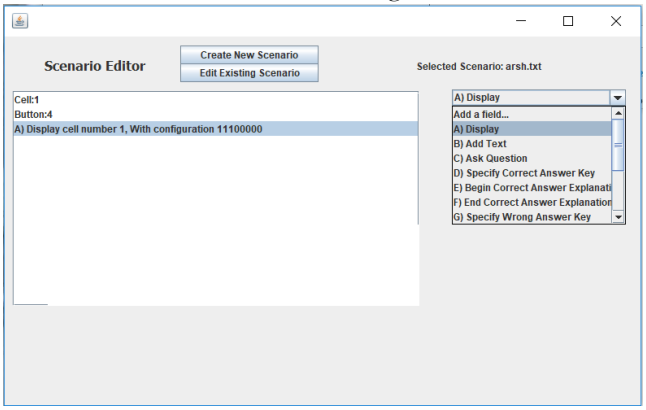
1. Display
2. Add Text
3. Ask Question
4. Specify Correct Answer Key
5. Begin Correct Answer Explanation
6. End Correct Answer Explanation
7. Specify Wrong Answer Key
8. Begin Wrong Answer Explanation
9. End Wrong Answer Explanation
10. Import Sound File
11. Display String
12. Add Pause

Display: When the user clicks "Display", they are prompted for a cell Number and a Letter.
In the String field, the user has to input an 8-character string of 1's and 0's to display a certain string.

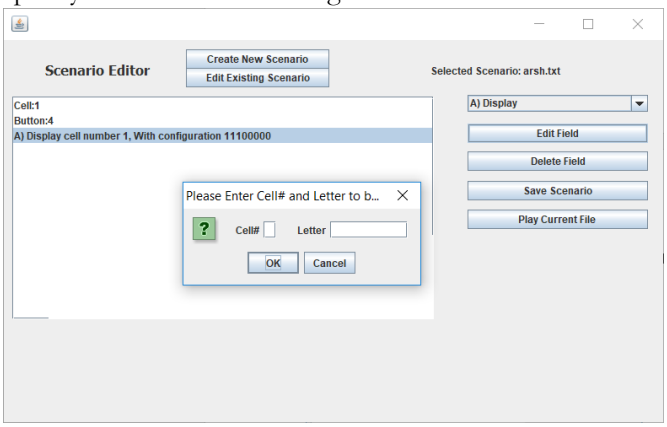


The prompt for Display

Once the cell number and text have been entered, the white text area is appended with the following line:
(In this example I am imputing character 11100000 into cell 1)
"A) Display cell number 1 for the letter 11100000"
The user can always edit this cell or even a different cell to show a different configuration. All the user has to do is click on the line in the editing area and then click on "Edit Field" button.



This will lead to a dialog box, just like the one that came up when you first added the field, asking you to specify the cell and the configuration.



Add Text: This is a simple feature if the user wants the text to be read out as a part of the scenario. If clicked, the user is prompted for the string that they would like to enter, and that line is added to the white text area with "Say:" prefixed to the text that has to be read out.
This field can be edited, by clicking on the line and hitting "Edit field" to reflect whatever else you want to be read.

Ask Question: This Button leads to two prompts:
1. The Question
2. The buttons that would record the answers (These buttons are then activated)

When entering buttons, the user can input button numbers separated by a comma.
The editor then reads he questions, the activated keys and a line to signify that a user input is now required.
For example, if the user would like to activate buttons 1 and 2, the can simply type in "1,2".
The question is then added to the interface, along with the activated keys

In the current version, this field can be edited, but only the question. The activated keys cannot be changed.

Specify Correct Answer Key: This button helps the user keep track of the right answer. This key has to be one from the keys entered when the question was asked. When this option is chosen, and a valid key is entered (for example button 1), then the line "Correct Answer: 1" is added to the interface.

Specify Wrong Answer Key: Just like "Specify Correct Answer Key" except, this option is to specify the wrong answer key.

Begin Correct Answer Explanation: Choosing this option starts a user defined series of events if the right answer is chosen. For example, if the user would like the learner to say "Good Job!" after they have chosen the right answer, the user can click on "Add Text" after they have chosen "Begin correct answer explanation" and type in "Good Job".
In addition to that, the user interface then has the line, "Correct Answer Explanation Starts here"
DISCLAIMER: **It is important that the user chooses "End Correct Answer Explanation" after the**

explanation has ended. If "End Correct Answer Explanation" is not chosen, this may lead to errors and the software might not be able to read the scenario file.

End Correct Answer Explanation: This button is for the user's reference to indicate that the correct answer explanation has ended. Note that of this option is not chosen the scenario file might not be able to be read by the software.

Begin Wrong Answer Explanation: Just like "Begin Correct Answer Explanation" except for the wrong answer. In this case, "End Wrong Answer Explanation" has to be chosen to signal the end of the explanation.

End Wrong Answer Explanation: Used to signal the end of the wrong answer explanation.
NOTE: If this question is the last question in the scenario, it is important that the user add some text after this, else there might be issues in the software not being able to read the scenario file.

Import Sound: Using this option, the user can play a given sound as a part of the scenario. When this option is selected, the user is prompted to select a file from a file chooser. TreBBA takes care of the rest.
The user can choose to play a different sound file by clicking on the line in the editor and clicking on "Edit Field". This will help you chose a different sound file.
NOTE: The file has to be a .wav file

Display a String: Displays a certain string on the braille cell. When this option is selected, the user can add a user-defined string to be displayed on the cell.
The user can edit the string that they want to be displayed by clicking on the line in the editor and clicking on "Edit Field". This will enable them to change the string that was originally set to be displayed.

Clear Cell: The user can use this to clear any given cell. The user is prompted with what cell they want to clear. The user can come back at a later time to edit whatever cell they had originally chosen to be cleared by clicking on the line and then clicking on "Edit Field".

Add Pause: The user has the ability to add a pause of any given time duration in seconds. When you click this, you are prompted with the duration of the time in seconds you'd like the scenario to be paused for. You can change this duration by clicking on the field and then clicking on "Edit Field".

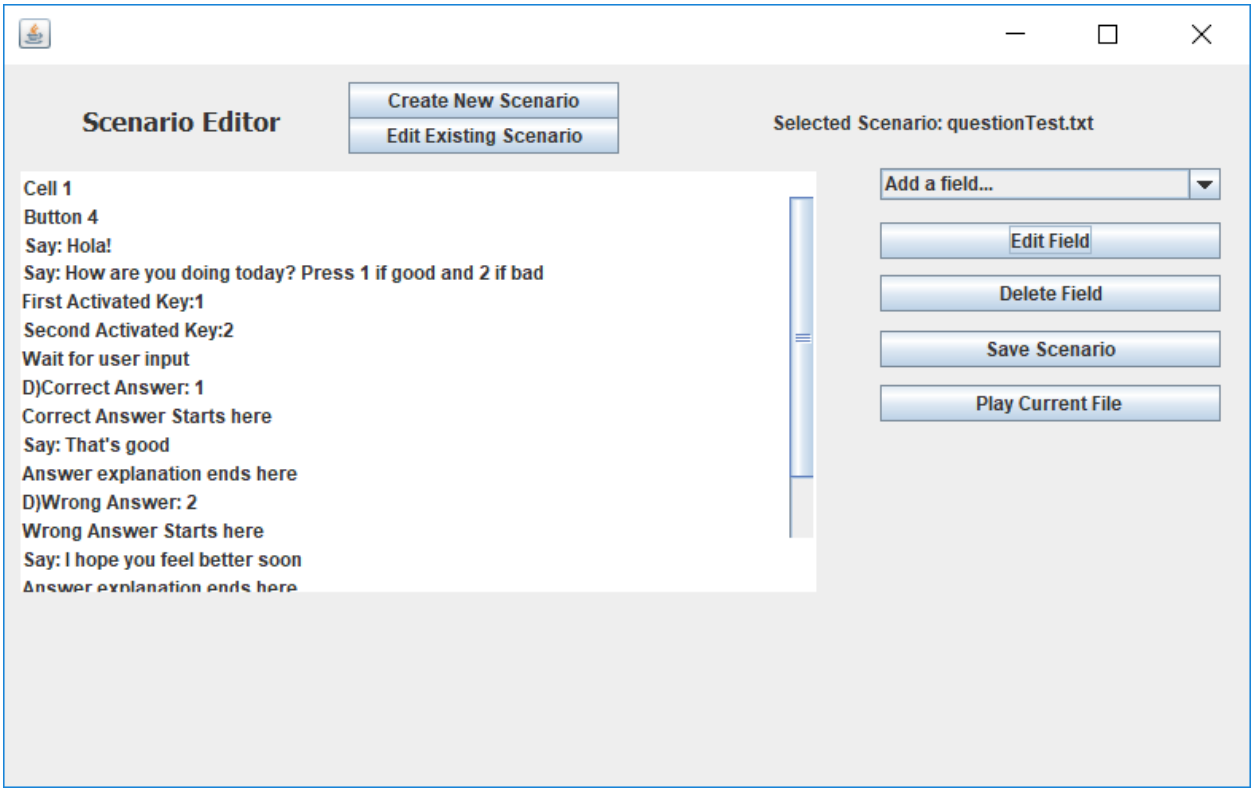
Reset Buttons: Using this functionality, the user can reset all the buttons on the cells in a given scenario.

Clear Cells: Users can use this to clear all cells in the scenario

Clear a Cell: Users can clear a cell of their choice by using this functionality. When you add this field, you are prompted with a dialogue box asking for the cell number you want to clear. You can also change the cell you want to be cleared by clicking on "Edit Field".

Editing an Existing Scenario

In version 2.0, users now have the ability to edit an existing scenario. All a user needs to do is launch the .jar file and click on "Edit Existing Scenario". This opens a file chooser, and the user can select an already existing scenario.
This loads the file in the editor, and the user can now add, edit, or remove fields as desired.



Saving Files:

To save a file, click on "save File" Button and your scenario would be saved.

Playing a Scenario:

To play a scenario, click on "Play Current Scenario Button"

Known Issues

Display Issue: The text at the edge of the editor window is hard to read. Something like in the figure below:

Workaround: Scroll down in the editor window and the text becomes clearer