

# User Manual



By: Zaeem Israr, Kevin Banh, Amir Khademi

## Table of Contents

Introduction .....	2
Definitions .....	2
Starting the Application .....	2
Recording Sound .....	3
Creating a Node .....	3
Specifying Cell Number/Button Number .....	5
Anatomy of a Node .....	6
Tag: No Tag .....	6
Tag: Pause .....	6
Tag: Voice .....	7
Tag: Display String .....	7
Tag: Sound .....	7
Tag: Repeat .....	7
Branching .....	8
Condition of Branching: .....	8

## Introduction

This document is the user manual of the Authoring App. This app is the third part of a three-part project in creating software for a hardware device used to help kids read Braille. In the second part of the project, a Player App was created to play stories for the users. The “stories” are text files with special keywords that allows display of Braille characters onto the hardware, input from the hardware (and based on the input, go to a different branch of the story) and more. The purpose of the Authoring App is to allow educators to create scenarios as they see fit.

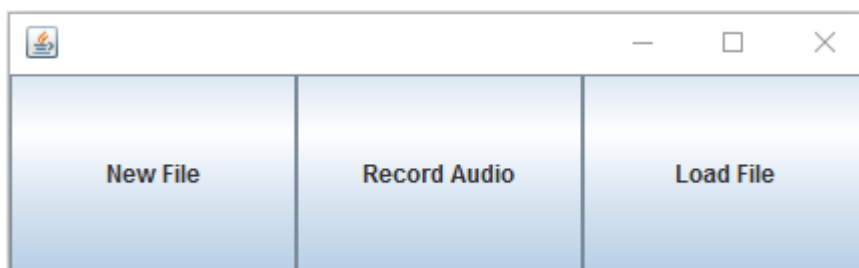
## Definitions

For the purpose of this document, scenarios are text documents (.txt extensions) with keywords in them. Scenarios are played by the Player App and based on certain keywords, do certain actions. A list of keywords can be found at

[https://wiki.eecs.yorku.ca/course\\_archive/2016-17/W/2311/\\_media/playerfileformat.pdf](https://wiki.eecs.yorku.ca/course_archive/2016-17/W/2311/_media/playerfileformat.pdf).

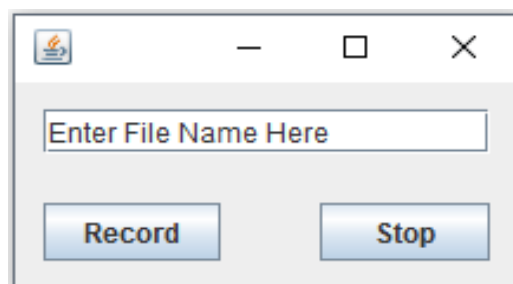
## Starting the Application

To start the application, simply double click on the provided executable file. The following window should appear upon successfully opening the authoring application. This window will be referred to as the “main window” in the document.



## Recording Sound

On the main window, click the button labeled “Record Audio”. The following window will appear:



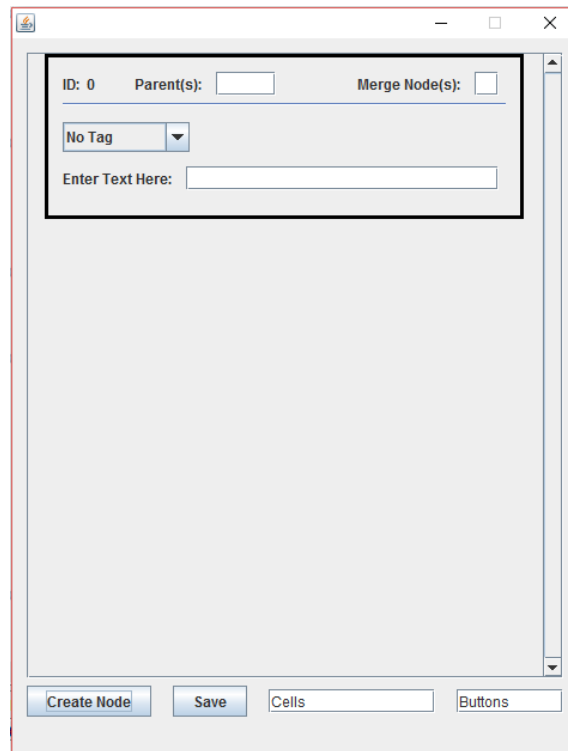
Simply enter the name of the audio file you want to create and press the record button. Speak into the microphone and press stop when you want to stop recording. The audio file will be created in the folder, the app resides in and will have the “.wav” extension. This file can be played using the default audio player on the computer.

## Creating a Node

The node is a tag (see definitions), text to be spoken or a combination of the two. A collection of nodes constitutes a story to be read by the braille. This story can be created by pressing the “New File” button on the main window. The following window appears when the button is pressed:



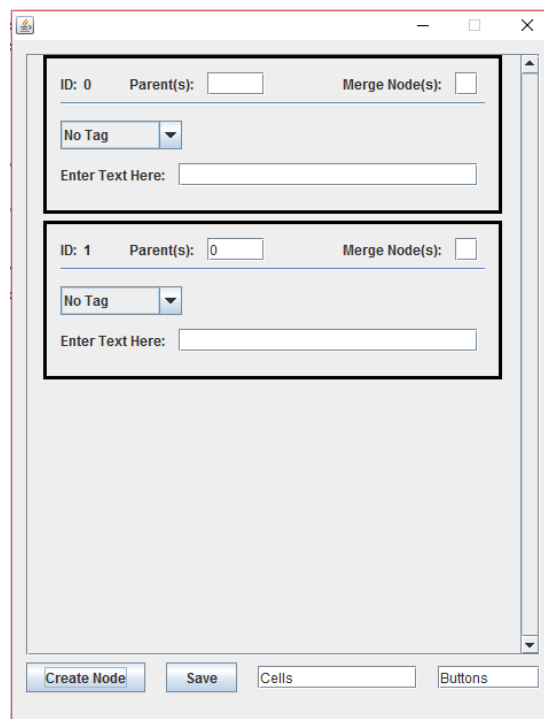
Simply press the “Create Node” button. This will add the node to the GUI and looks as follows:



A screenshot of a GUI window with a title bar containing a small icon, a minus sign, a maximize button, and a close button. The main content area is a light gray rectangle. At the top, there is a form for a node with the following fields: "ID: 0", "Parent(s):" followed by an empty text box, and "Merge Node(s):" followed by an empty checkbox. Below these is a dropdown menu currently showing "No Tag". Underneath the dropdown is a text input field with the placeholder text "Enter Text Here:". At the bottom of the window, there is a horizontal bar containing four buttons: "Create Node", "Save", "Cells", and "Buttons".

A story can be create by adding nodes to the GUI, filling the nodes with valid information, connecting the nodes together and then generating the file.

Subsequent nodes can be added to the file which would cause the GUI to look as follows:



A screenshot of the same GUI window as before, but now it contains two node forms stacked vertically. The top form is identical to the one in the previous screenshot, with ID: 0. The bottom form has "ID: 1", "Parent(s):" followed by a text box containing the number "0", and "Merge Node(s):" followed by an empty checkbox. It also has a "No Tag" dropdown and an "Enter Text Here:" text input field. The bottom bar with buttons remains the same.

The above window shows two nodes which can be filled by the user according to their requirements (See “Node Details” Section for more information).

## Specifying Cell Number/Button Number

The number of cells and buttons the story requires can be entered into the text fields labeled in the following windows.



The number of cells and buttons can be changed at any time before pressing the save button. Creating nodes by pressing the “Create Node” button does not affect the number of cells and buttons, hence the user can change the number as many times as they desire.

## Creating the File

In order to create the file, once all the nodes have been created, simply press the save button and exit the application. The output file will be generated in the folder, the app resides in. Once the file has been created, it can be read using the player app.

## Anatomy of a Node

A node consists of several textboxes. The ID at the top left specifies the identity of the node. The identity is generated automatically as new nodes are added to the GUI, starting at number 0 and incrementing by 1 each time a node is created. The textbox labeled “Parent(s)” is used to specify the parent node. This is also generated automatically unless otherwise specified. It is possible to have multiple parents and in that case, put the ID numbers representing the parents separated by a space (i.e. 10 2 3 4). Numbers specified in the parent section must be smaller than the current ID of the node or the program will not function properly. All nodes beside the first one must have a parent. Nodes with multiple parents are called merge nodes. The textbox labeled “Merge Node(s)” is for users to specify the place where branches from the current node are merged. A node x has branches if there are more than one node with x as its parents. Specify the merge node ID (where the branches will finally merge) into in the box. See branches for more detail.

The combo box (called “Tag”) with “No Tag” on by default determines what tag should be applied at that point. The text field below that (called “Info”) is for extra information needed for each tag. Please see below for more details.

The large textbox is for the text or what the user of the authoring app wants to be spoken in the text file that is to be played by the Player App.

### Tag: No Tag

This tag does not do anything. This option is for a node where the user does not want to use a tag.

### Tag: Pause

This tag will add a pause before the text in the current node is spoken. A text field labeled “Info:” appears when this tag is selected. In the “Info” field, user can add a number specifying the length of the pause in seconds. The “Info” box text should be an integer greater than zero. Anything else (i.e. number followed by a space, letters, negative number and etc.) will result in file not being generated properly.

### Tag: Voice

This tag will change the voice used to read the scenario. The tag carries on to the next node that is linked to the current node until another voice tag is used. In the “Info” field that appears when this tag is selected, user should add a number specifying which voice number to use. The “Info” box text should be either a 1, 2, 3 or 4. Anything else (will result in file not being generated properly).

### Tag: Display String

This tag causes the Player when playing the scenario, to change the output of the Braille Cells before the text of the current node to be played. The text in the “Info” field that appears when this tag is selected gets translated to Braille and is displayed on the Braille Cells. The “Info” field text should contain only lower case letters along with space and length of the text should not exceed the number of cells specified. Anything else will result in file not being generated properly.

### Tag: Sound

This tag causes the Player when playing the scenario, to play a sound file before the text of the current node to be played. The text in the “Info” field that appears when this tag is selected specifies the sound file that is played. The “Info” field text should contain a string of characters ending in .wav. Anything else (i.e. not a .wav file, empty string and etc.) will result in file not being generated properly. The .wav file must exist for the file to be generated properly and can be recorded using the app.

### Tag: Repeat

This tag causes the Player when playing the scenario, to prompt users to press a button after the text of the current node to be played. The “Info” field disappears when this tag is selected. The Repeat keyword can only be used if the node has more than one children. The button number available should be greater or equal to the number of child + 1 if repeat is used.

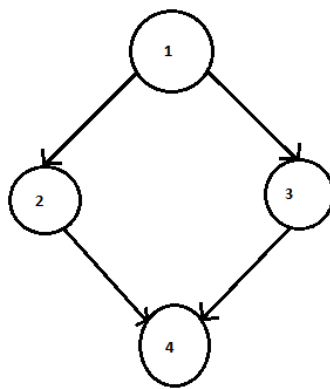


## Branching

A story may contain multiple scenarios with each scenario leading to a different branch. These branches are specified by the “Parent(s)” and “Merge Node(s)” fields in each created node in the GUI. A linear story would consist of a parent field with the ID of its preceding node.

Branching requires the user to manually enter the parents and merge node.

Assume each circle in the following diagram acts as a node in the app.



Here, node 1 acts as the parent node for the. Nodes 2 and 3 are child nodes and node 4 is the merge node. If the user wants to create a scenario with two branches which later merge into the same point in the story, nodes 2 and 3 in the GUI would require the parent field to be specified as “1” and merge node field to be specified as “4”.

### Condition of Branching:

In order to create the branches in a story, the following rules must be kept in mind.

1. All nodes that have multiple children must specify a unique merge node ID. Here unique means the node specified is not used as a merge node twice.
2. Merge node cannot have a branch. A branch cannot be created in a merge node or a merge node cannot be part of a branching scenario.
3. A node with a branch cannot have more than one merge node.