**BunnyWorld Description**

Team Member: Zibo Gong, Ji Yu, Yilun Xu, Yuwen Zhang, Yifan He

Here is a demo for our project: <https://youtu.be/vL3b3Se0kP0>

**Editor Features:**

**Basic Features required by the handout:**

* The ability to create a new game, open an existing game, edit the game, and save the game persistently on an Android device (either using a SQLite DB or a file is fine).
* There should be a way to play the game. As previously mentioned, you can have a single app that both allows editing and playing of games, or you may have the editor pass the game on to a separate game player application.
* There should be a way to create, name, delete, and see pages. Newly created pages should automatically be assigned names. The first page created should always get the special name "page1". Subsequent pages should get the names "page2", "page3", etc. There should be visual feedback of the name of the currently displayed page. There should be a way to edit the name of a page (except page1). There should be a way to delete the current page (except page1), which deletes the shapes on that page (though not the resources they refer to).
* There should be a way to add, name, see, edit, and delete shapes in the current page. Newly created shapes should automatically be assigned globally unique names "shape1", "shape2", and so on. There should be an "inspector" which displays and edits the state of the currently selected shape. The state of the selected shape includes: its geometry (4 ints), name, movable, visible, image-name, text, and associated script.
* You should preload your application with images and sounds that a game designer can use to develop their own adventure game.
* For script editing, because typing on an Android platform is somewhat awkward, you may want to implement these using pulldown menus. For example, creating a pulldown menu allowing the user to choose a script action, which then enables a second pulldown menu allowing you to select a page-name, sound-name, or shape-name as appropriate.
* You’ll also have to decide what to do if the user changes the name of a page or shape that’s referred to in another script. My implementation updates all references when a name is changed, but it’s permissible to simply leave the script unchanged and hope the user knows what they are doing (see error checking in the Extensions section below).
* For error handling, your editor should never crash due to any reason. The editor does not have to do any error checking. However, if your editor does not verify that a game is in fact legal (for example, do all Shapes in scripts in the game refer to actual Shapes created by the user), the game player must gracefully indicate the error, rather than crashing.

**Extensions:**

* Error Check feature: if the script of the shape is illegally formatted, the line of code will not be executed, and the error check was conducted before the object was created from the database.
* Shapes (images and texts) can be dragged from the inventory to add to the current page. (we build a two-page inventory with plenty resources to choose from).
* Copy/Paste shapes features: we can copy and pages shapes in one page, or copy a shape from one page to another page.
* When we create the page, we can name it ourselves or use the default name(Page1, 2, 3) that is already generated in the editText.
* We can go to different pages by pressing the button “Change Page”, and the selected radioGroup will implicate the current page we are in. Same to “Delete Page” button.
* We can change the script of the selected shape directly by pressing the item in the menu “Show Script”
* We can clear the current page.
* Undo support for basic changes to shapes: moving, renaming, changing the script, etc.
* Undo support for adding/deleting shapes(harder)
* When we create the script of the selected shape regarding playing sounds, we can play the demo first and then decide on which sound to select.
* Users can resize the shape by pressing the item “Set Property” in the menu. And they can also change the properties like movable/hidden. Hidden image shape will be shown in a transparent state(the text shape will hide the text when it is set “hidden”) in the editor(movable property will function when playing the game).
* Adorable UI design (☺).

**Game Features:**

All basic features function perfectly required by the handout.

**Extension:**

* We give a demo when entering the game before loading the designed game.
* We can create several games and save them all to the database, and then select one of them to play in the game.
* Also adorable UI design(☺).
* Handle all the edge cases; Almost bug free (Please tell us if you find a bug).