

## Milestone 4 – Polished Game Menu

This is an **GROUP** assignment. Only one member of your team should turn in your files.

### Due

March 31 at 11:55 PM

### Late Policy

See M1 for late policy

### Description

The goal of this assignment is to make your class project feel more like a finished product. One of the best ways to achieve this is to support a starting action that is well integrated into your game via an engaging menu system.

The game menu will provide the option to start a new game, make configuration changes, and provide access to a credits sequence.

The second task will be to add some polish to the menu interaction by implementing a background scene with Image Effects (e.g. blur, motion blur, tilt shift, gray scale, etc.).

You will leverage the “new” UI system introduced in Unity 4.6 / Unity 5. You will **not** be using the legacy GUI Skin (aka IMGUI) for this milestone.

### Grading Criteria

Note that you don’t need to have a game attached to your menu for this assignment, but feel free to submit a project with the full menu and your game at whatever progress you have achieved at the point of submission.

Your submission should satisfy the following requirements:

#### UI Components

- 1) The game should start with an introduction menu scene (implemented with new GUI, **NOT** IMGUI). This menu scene should have a polished feel with intuitive interaction. At minimum, the menu scene should have the title of the game, your team name, a menu button to start the game and a menu button to show the credits.

**(10 points)**

- 2) The background environment (scene behind the menu) should be visually appealing and animated in some fashion. This doesn’t have to be overly complex just something more entertaining than a solid color or static image. Ideally, it will also communicate to the game player something about your game.

**(10 points)**

Suggestions:

- a. A camera floating or moving around the game world in a looping fashion
- b. Use the unity 2D tools to animate a custom background
- c. Use captured gameplay footage on a video loop (video edited with trimmed sequences of interest and transitions)

3) The menu items should be aesthetically positioned and styled

- a. The position of the elements should be placed appropriately such that components are not overlapping, text should be readable, and overall the whole menu should be visually appealing.

**(10 points)**

Suggestions:

- i. Group like components, and utilize padding between groups.
- ii. Font colors should contrast the background so that it is readable
- iii. Apply partial transparency to increase the visual appeal of components
- iv. The use of perspective is easy in Unity 4.6\5 UI system and can add a lot to the style

- b. The elements should be customized for your game.

- i. Do not simply use the default UI selections for font, etc. Develop a style appropriate for your game and document your decisions in your submission write-up.

**(10 points)**

- ii. Create a custom button image for the clickable elements of your interface. Again, document your reasoning for the design.

**(10 points)**

4) Your menu system should be navigable with a controller or keyboard in addition to normal the mouse clicking.

**(10 points)**

Hints:

- a. This will involve highlighting or identifying the currently selected element and be able to go up or down / left or right elements and changing the identified element.
- b. You should be able to loop from left back to right or top back to bottom for convenience.
- c. If you have only two options in a menu, your “highlight” should be easy to determine versus the un-highlighted option. For instance, show a frame around the selected button. Otherwise, the user might not be able to tell the difference between selected/unselected.

5) The credits for the game should be fully informative on all contributions. (Subject to future revisions, of course.)

**(10 points)**

- a. The credits should document all developers in your team and their major contributions.

- b. The credits should document all sources of 3<sup>rd</sup> party content that you should have been documenting in the readmes in all milestones including this one. (Again, subject to future changes as you progress on the project)  
Including but not limited to:
  - i. Plugins & Scripts (Including RAIN AI)
  - ii. Audio (Music & Sound Effects)
  - iii. Artwork (Textures & 3d Assets)

6) The credits for the game should be visually compelling in some way.

**(10 points)**

Hints:

- a. This can include the standard vertical scroll on a dynamic background, like the menu but doesn't have to be as complicated since the focus is on the text.
- b. This can also include some custom effects like individually animated text elements that slide in, fade in and out, etc. Also having other images like screenshots from the game or photos of the developer or corporate logos transition around.

#### Polish via Visual/Image Effects

You will implement a visual or image effect to add polish to your menu backing scene. These include lens flares, halos, bloom, blur, motion blur, depth of field, tilt shift, color changes, edge detection, etc.

You are responsible for researching the technique you deem appropriate for your game menu. Implement the effects and clearly document how it's used. You are welcome to use 3<sup>rd</sup> party effects provided that you list the source in your readme.

**(20 points)**

#### **Resources:**

<http://blogs.unity3d.com/2014/05/28/overview-of-the-new-ui-system/> - overview of the "new" UI system

<http://docs.unity3d.com/Manual/HOWTO-UIScreenTransition.html> - how to integrate UI with animation controller to control screen states and transitions.

<http://openfontlibrary.org/> - has a collection of fonts many of them on very permissive license agreements, please note where you got them from in your writeup.

#### **Submission:**

You should submit a 7ZIP/ZIP file of your Unity project directory via t-square. Please clean the project directory to remove unused assets, intermediate build files, etc., to minimize the file size and make it easier for the TA to understand.

The submissions should follow these guidelines:

- . a) Your team name should appear on the HUD of your game when it is running.
- . b) ZIP file name: <teamname>\_m<num>.zip
- . c) Readme file should be in the top-level directory: < teamname >\_m<num>\_readme.txt and should contain the following
  - i. Full name, email, and prism account name for each member of the team
  - ii. Detail which requirements you have completed, which are incomplete, and which are buggy (be specific)

- iii. Detail any and all resources that were acquired outside of class and what it is being used for (e.g. Asset Bundles downloaded from the Asset Store for double sided cutout shaders, or this file was found on the internet has link <http://example.com/test> and does the orbit camera tracking).
  - iv. Detail any special install instructions the grader will need to be aware of for building and running your code, including specifying whether your developed and tested on Windows or OSX
  - v. Detail exact steps grader should take to demonstrate that your game meets assignment requirements (e.g. "First, walk towards the pile of blocks using WASD and mouse and bump into them to knock them down. This should demonstrate actor movement via physically simulated forces and interactivity with environment...") Please also include game feel description.
  - vi. Which scene file is the main file that should be opened first in Unity
- d) Complete Unity project (each script file you created should include team name and members' names in comments at top of file, and any file you acquired outside should also be attributed with the appropriate source information)

Submission total: (up to 20 points deducted by grader if submission doesn't meet submission format requirements)

**Be sure to save a copy of the Unity project in the state that you submitted, in case we have any problems with grading (such as forgetting to submit a file we need).**