# Wearable’s Game Documentation

1. Structures 1

2. Collectable Objects 2

3. Time 2

Drone battery 2

4. Dangerous terrain 2

5. Player 3

6. TopPanel & EventSystem 3

7. Needs future work 4

*Dangerous terrain* 4

*Clues* 4

1. Structures

Each structure has a virtual representation of a door with a four-digit address.

The campus map has been designed as a complete model **nmsu\_osm.dae** and is a prefab on the game.

This prefab contains all the buildings and structures that make the virtual campus map.

Each building has a door representation and its ID (thought this ID not displayed for all) that is made of an **UI image**, to modify these doors they need to be physically moved (**rect Transform**) to the desired location.

For all buildings on **nmsu\_osm** make sure that:

* have Tag as Building
* have a mesh collider
* the convex option is checked

Because the mesh collider cannot be edited, some buildings that have a variant side shape make take a rectangle collider, thus not having access even when it looks like a hallway. Some of these buildings are:

* Building\_Jett\_Hall
* Buiding\_Frenger\_Food\_Court
* Building\_Harold\_Foreman\_Engineering\_Complex
* Building\_Music\_Center (this building is out from the walls)
* Building\_NMSU\_ICT
* Building\_Rentfrow\_Gymnasium (this building is out from the walls)
* Building\_Regents\_Row\_Residence\_Center (this is one of the buildings that is more affected)
* Building\_Science\_Hall (this is one of the buildings that is more affected)

There is something important to note on mesh colliders:

Faces in collision meshes are one-sided. This means objects can pass through them from one direction, but collide with them from the other. This may affect some character movements sometimes.

1. Collectable Objects

*Classes related to its functionality:* **objectsToCollect.cs, countObjects.cs**

I imported 2 sets of assets from the Unity assets store for the collectible objects the folders are contain inside the Assets folder.

**ClassRoom Stuff**

**HighDefinitionOfficeProps**

Once I placed some objects in the game I made them prefabs, for now the only two objects I have been working on is the **chair** and **FolderOpen**

If you need to place a new object took the objects from the following folders and modify it as your desired size and position:

**ClassRoom Stuff--Meshes**

**HighDefinitionOfficeProps--Models**

If these new objects are going to be used as collectable objects make sure to add them the **objectsToCollect.cs** scripts and check the **IsTrigger** option.

In the beginning of the level the top panel will display inside the magnifying icon named **Clues** the number of objects that need to be collected. The player collects an object by **approaching and passing** through it. If the object is correctly collected it will disappear from the game environment and the counter inside the clues icon will decrease by one.

1. Time

*Classes related to its functionality:* **levelSpawner.cs, batteryControl.cs, droneBatteryControl.cs**

For now every 60 seconds the same scene will be loaded and the character player will be placed in the same initial position with all the collectible objects from the beginning of the game.

As time passes the **top panel** displays the remaining time and the battery icons show and simulates the percentage of one drone battery.

When the time has finished, a new level (reload the same scene) will begin. The number of objects to collect will be reset to the initial number and all the icons in the **top panel** will be reset to their initial state.

## Drone battery

The drones also have a limited battery power simulated with a **DroneHealthBar** object made out if two UI images **HealthBG** & **HealthBar**. The last one is a **Filled image** type with a WhiteSquare.jpg as a source image.

1. Dangerous terrain

*Classes related to its functionality:* **dangerousZoneWarning.cs**

Only 2 of the buildings are dangerous zones for now. Whenever the player gets close to them the **Dangerous** icon will change in the top panel.

To make the collision detectable and trigger the warning sign, I added a sphere collider named **DangerTigger** to each of the **nmsu\_osm buildings** that are dangerous terrains:

Building\_Science\_Hall

Building\_Walden\_Hall

If you want to add another building or structure make sure to add the following features for each of them:

* **MeshCollider** checked as **Convex** and **unchecked IsTrigger**
* **SphereCollider** checked as **IsTrigger** and add to this the **dangerousZoneWarning.cs** script adding to **sprite2change** element in the inspector the **Dangerous** image that is under

Assets—StandardAssets—Icons folder.

1. Player

*Classes related to its functionality:* **characterControl.cs, camMouseControl.cs**

At the beginning of the game, the mouse cursor is invisible and disable. To enable it and get out from the game scene use the **ESC** key.

The first person character (FPC) is a **capsule** control that needs to have the **Capsule Collider** and the **RigidBody** to detect movement and collisions.

Make sure that for the Capsule Collider the following features are setup like this, if not the collider detection will not work as expected:

* radius = 1.0
* height = 2.5
* Direction = Y-axis

The class **characterControl.cs** handles the forward/backward left/right movement of the character.

The **Main Camera** is a child of the FPC so it can detect the player movement.

The class **camMouseControl.cs** handles the camera movement guided by the mouse. The following variables have been tested with different values and it seems that these are the best that fir the game:

* sensitivity=2.0
* smoothing=1.0

To handle the Dangerous Zones detection use a sphere collider that covers the dangerous building or area and check the option **Is Trigger**. For every sphere collider attach the **dangerousZoneWarning.cs** script that handles the changing icon of the top menu.

1. TopPanel & EventSystem

The top panel is an UI Canvas that is Render Mode as Screen Space – Overlay

If you need to add one more icon such as an UI Image or UI Text make sure:

* the **Anchor presets** are set to bottom-stretch
* the **source image** has been formatted as following

For **all and any png** images that are going to be used as icons make sure to change the features for:

Texture type: Sprite (2D and UI)

Uncheck: Generate Mip Maps

Max Size: Preferable to be 512

The EventSystem object is an empty Game Object that handles some of the events of the game.

* **countObjects** (updating the number of objects that are collected)
* **levelSpawner** (time management and reloading the scene as time finishes)
* **batteryControl** (changing colors as battery gets depleted)

1. Needs future work

## *Dangerous terrain*

Certain parts of terrain are dangerous and can detected remotely by the drones.

*dangerous to the player*

If a player spends time in a dangerous area, time is deducted;

*dangerous to the drone*

if a drone spends time in a dangerous area, the drone loses battery faster.

## *Clues*

Clues are a part of the address for the structure with the hidden object;

clues may be hidden inside a structure (accessible only to the player)

clues may be hidden on top (accessible only to the drones)

Once 4 unique clues are assembled, the player can use that information to identify the correct structure.