Computer Interface Design Principles

Higher National Diploma in Interactive Media Y2

2HND1

Leeanne Borg

TASK 1 (P1.1)

Research the current state of the art with respect to game engines and outline what the most common game engines in use today are. Write a short paragraph explaining at least three interactive game engines with one example of a game from each.

1. RAGE Engine – RAGE is a very strong engine it has the ability to handle large streaming worlds, complex A.I. arrangements, weather effects, fast network code and a huge number of game play styles. Game examples: GTA III, Vice City, San Andreas and Bully.

2. CryENGINE - "CryENGINE 3 is the first Xbox 360, PlayStation 3, MMO, DX9 and DX10 all-in-one game development solution that is next-gen ready – with scalable computation and graphics technologies." Crytek. Game examples: Far Cry, Crysis, Crysis Warhead, Crysis 2, Aion: Tower of Eternity.

3. Naughty Dog Game Engine – It has crisp environments with dynamic objects, smoother and various environment-animation interface, amazing improvements for lighting and A.I. and also full support for co-op and competitive multiplayer. Game examples: Uncharted: Drake's Fortune, Uncharted 2: Among Thieves

4. Unreal Engine - Unreal Engine became the multi-format engine of choice for many of the major and minor publishing houses. It turns up in the strangest of places (Surf's Up) and even in games from developers with their own impressive tech (BioWare, EA, Ubisoft). Game examples: Gears of War, Mass Effect, BioShock, Unreal Tournament, Deus Ex, GRAW, Red Steel.

TASK 2 (P1.2)

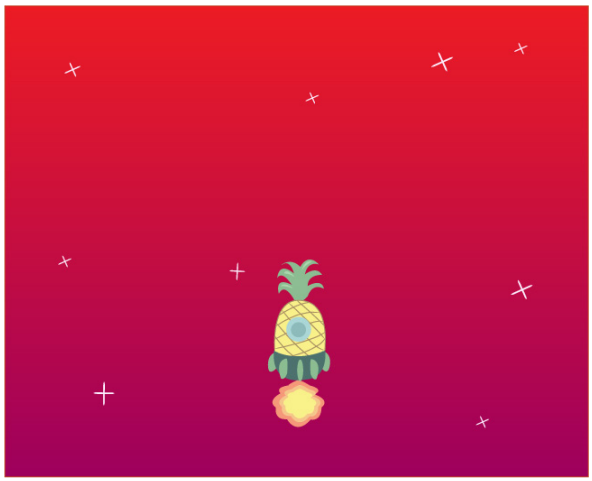
Explain how different screen sizes are catered for using the Unity game engine, with reference to the ScreenToWorldPoint function.

You can change the GUI.matrix to follow the screen resolution - this changes everything in the GUI world: fonts, textures, screen coordinates. If the GUI is not set good it will be pixilated so you have to be responsible to check the proper size.

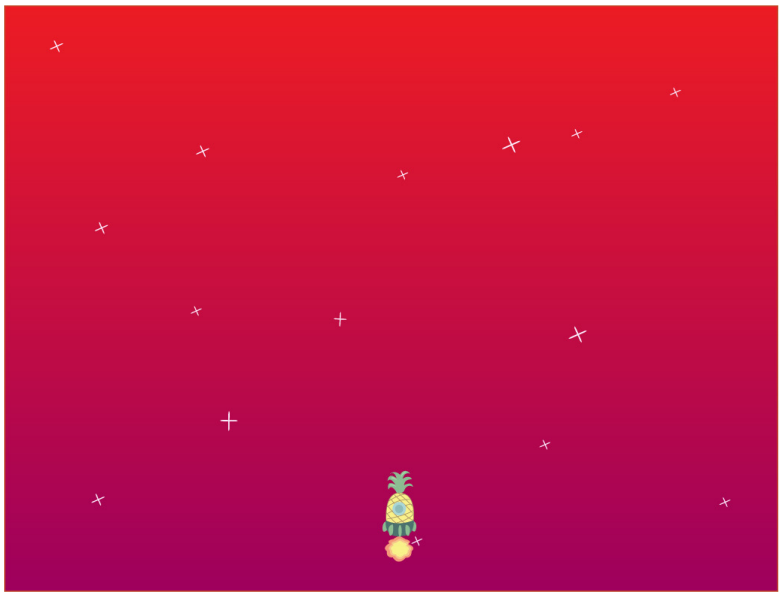
The ScreenToWorldPoint function will transform the position from screen space into world space. Screenspace is defined in pixels. The bottom-left of the screen is (0,0) and the right-top is pixelWidth and pixelHeight. The z position is in world units from the camera.

Example of screen resolutions:

640x480



1024X768

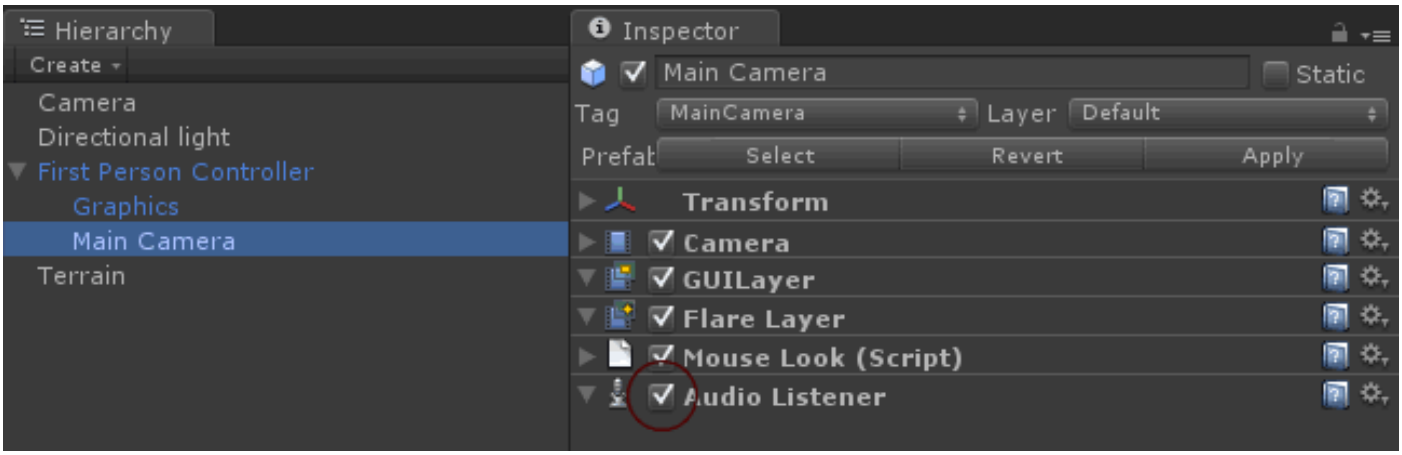


TASK 3 (P2.1)

Explain how a simple sound effect is implemented in Unity, with reference to audio listeners.

Audio Listener is the ears of your player and is found on each camera. The Audio Listener is like a microphone-like tool, it receives input from an Audio Source in the scene and plays the sound through the speakers. It must be added to work and it is always added to the Main Camera by default.

The Audio Listener works with Audio Sources, let you create the hearing for your games. When the Audio Listener is attached to a GameObject in your scene, any Sources that are close enough to the Listener will be picked up and output to the speakers. Each scene can only have 1 Audio Listener to work properly. You should attach the Audio Listener to either the Main Camera or to the GameObject that represents the player. Try both to find what suits your game best.



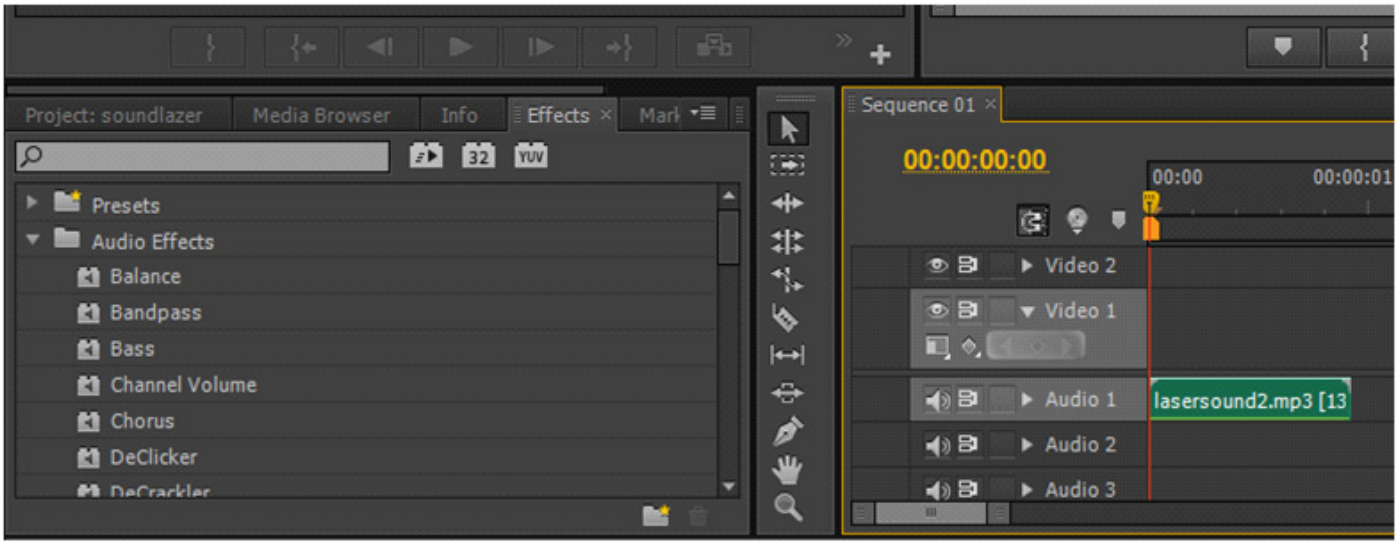
TASK 4 (P2.2)

Record and generate an effective laser sound for your game. To do this you need to present the following three .mp3 format sounds:

The original sound recorded.

- lasersound2.mp3

A description of the effects applied to the sound.



- Bass
- Notch
- Treble
- Volume
- Speed Duration

The final modified sound.

- lasersoundedit.mp3

TASK 5 (P3.1)

Explain how to change materials of different objects based on in-game events, with reference to the OnTriggerEnter and OnTriggerExit functions.

Triggers are good for detecting when things enter and exit, but if something is created or teleported inside the trigger, nothing happens.
When the Collider other enters the trigger OnTriggerEnter is used. This is sent to the trigger collider and the rigidbody that the trigger collider belongs to, and to the rigidbody that thouches the trigger. The trigger is only sent if one of the colliders also has a rigidbody attached.

```
// Destroy everything that enters the trigger
function OnTriggerEnter (other : Collider) {
Destroy(other.gameObject);
}
```

As seen in this example, the OnTriggerEnter function accepts a collider as parameter. It tells you which object entered the trigger. You can check "other" and see whatever which entered your trigger. Check for the tag of your entering collider and only trigger something when the tag of your entering collider eg. "player".

OnTriggerExit is the opposite of OnTriggerEnter. It is used when the Collider other has stopped touching the trigger.
Example:

```
function OnTriggerExit(other : Collider)
{
 if(other.gameObject.transform == myTarget)
{
 myTarget = null;
        }
}
```

The 'myTarget' is for an object moving in and out of the collider. This is so that the turret knows not to aim at 'myTarget' upon exiting the collider.

TASK 6 (P3.2)

Draw a sketch of three screens of your game:

Main Menu page sketch:

Game Over page sketch:

You Win page sketch:

TASK 7 (P4.1)

Conceptualize an interactive system by explaining how to interrupt or control a sequence of events using Unity, with particular reference to the start(), update() and WaitForSeconds() methods.

Start()

Start is called just before any of the Update methods is called the first time.
Start is only called once in the lifetime of the behaviour. Start is only called if the script instance is enabled. This allows you to delay any initialization code, until it is really needed. The Start function is called after all Awake functions on all script instances have been called.
Example:

```
function Start () {
//when the game will start again it will restart the following
        score = 0;
        health = 100;
        fired = 0;
        hit = 0;
        this.renderer.material = colours[0];
}
```

Update()

Update is called every frame, if the MonoBehaviour is enabled.
Update is the most commonly used function to implement any kind of game behaviour.
It is the main workhorse function for frame updates.

```
function Update () {
        //calculate borders
        //leftmost border
        leftmost = Camera.main.ScreenToWorldPoint(Vector3(0,0,0)).x;
}
```

WaitForSeconds()

This function suspends the finishing amount of seconds.

```
// Prints 0
print (Time.time);
// Waits 5 seconds
yield WaitForSeconds (5);
// Prints 5.0
print (Time.time);
```

TASK 9 (P4.3)

Write a short paragraph

explaining what improvements you would have implemented in the game if you had more time.

I would like to point out my improvements during this assignment. Although it was my first time using unity for this year I think I did my best. Moreover I would like to say that although I'm not that confident with coding, I still managed. I also managed to make my game as I imagined and it was a success.

On the other hand I could have implemented more time to make a boss at the end of the game and some power ups, but I think that I couldn't manage the time as I am not that confident with unity program. I tried at least to make some things which I mentioned but I did'nt have time.

On the whole it was a good assignment and I feel that I did my best to make a good result.

Refereces

http://www.ign.com/articles/2009/07/15/the-10-best-game-engines-of-this-generation

http://docs.unity3d.com/Documentation/ScriptReference/Camera.ScreenToWorldPoint.html

http://answers.unity3d.com/questions/156619/scaling-the-gui-with-different-screen-resolutions.html

http://docs.unity3d.ru/Components/class-AudioClip.html

http://docs.unity3d.com/Documentation/Components/class-AudioListener.html

http://forum.unity3d.com/threads/117130-Multiple-Cameras-and-Audio-Listeners

http://docs.unity3d.com/Documentation/ScriptReference/Collider.OnTriggerEnter.html

http://answers.unity3d.com/questions/11512/ontriggerenter-question.html

http://docs.unity3d.com/Documentation/ScriptReference/MonoBehaviour.OnTriggerExit.html

http://docs.unity3d.com/Documentation/ScriptReference/MonoBehaviour.Start.html

http://docs.unity3d.com/Documentation/Manual/ExecutionOrder.html

http://docs.unity3d.com/Documentation/ScriptReference/MonoBehaviour.Update.html

http://docs.unity3d.com/Documentation/ScriptReference/WaitForSeconds.html