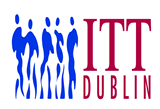
# Institute of Technology Tallaght Dublin

**Department of Computing**



**CA Title:**  Create A 3D Pool Game In Unity Engine

**Module:** Interactive Media Development

**Lecturer:** Enda Lee

**Student Name:** David Kiernan

**Student Number:** X00093830

**Due Date:**  4h May 2017

**The Approach**

The aim was to recreate a pool billiards pool game and implement the rules.

The game should have contained ball for each player to pot (spots & stripes), along with a black ball, cue ball and a cue.

The first thing was to create a pool table. As this was complicated, it was decided that a download of the pool table would be better. This is what the table would look like

[pool table, no date]



Also the textures for the pool ball were also downloaded as a resource as well, this was to reduce the time spent on creating, and finding a suitable textures and colours

[pool ball, No Date]



**Features**

The game allows for two players each player will take a turn to take a shot. If the player doesn’t pot a ball then the other player then takes their shot, this continues till all the balls are potted.

The user can use the “A”,”D”, “left arrow”, “right arrow” to move the cue which will also move the camera.

The left mouse button will allow the user to adjust the power and the user can visually see the cue moving.

There is also a scoreboard that shows the players’ names (Hard-coded) to “P1” and “P2”, it also displays their score.

It will also display a message stating who the winner is.

**Challenges**

The biggest challenge was getting the balls to detect that they were hit by either the cue ball, another ball, side wall or were pocketed.

This was done with physics, and colliders set on each off the ball. This was then applied to the cue ball as well as it too needed to detect when it had being hit.

Another problem was when the balls were pocketed to ensure that they disappeared or in the case of the cue ball reset it.

This was solved by splitting up the cue ball into its own game object

Then a simple for loop and if statement were used, if the cue was potted then reset it position if another ball was potted it would destroy the game object.

Lighting was another problem at first it was too dark then a light was add shining straight down. This made it too bright, so two light sources were used to give the lightning that was used in the game.

These were then main problems in the game, although they are quite small at the time during the course of it they took up too much time to first make, then solve the errors.

**Comments**

This led to features being left out due to the time constraints such

* Having a black ball that would be destroyed only if all of the player balls were destroyed otherwise reset.
* Better lightning conditions
* Having both spots and striped balls
* Have it as a foul if the player hits the wrong ball

**References**

pool ball available from

<https://opengameart.org/content/billiard-balls>

Accessed on 15th April 2017

Pool table available from:

<https://archive3d.net/?a=download&id=52cc9877>

Accessed on 15th April 2017