The core design elements for Quest: To Be Defined

Quest: To Be Defined Design Document

By James Moran

Contents

[Introduction to/Description of the Game 1](#_Toc464216876)

[Running the game 1](#_Toc464216877)

[Starting up the game 1](#_Toc464216878)

[Start-Up sequences 1](#_Toc464216879)

[Main menu 1](#_Toc464216880)

[New Game 1](#_Toc464216881)

[Credits 2](#_Toc464216882)

[Main Menu UMG Widget Sample Layout 2](#_Toc464216883)

[First level layout 2](#_Toc464216884)

[Sample View upon Starting First Level 4](#_Toc464216885)

[Player Movement System 5](#_Toc464216886)

[What the Player can see in the first crevice of the canyon 6](#_Toc464216887)

[How PIBot behaves in the first level 6](#_Toc464216888)

[The CombatAdmin 7](#_Toc464216889)

[The Brick Weapon 7](#_Toc464216890)

[Weapon Pick up Confirmation 7](#_Toc464216891)

[Weapon Ammo HUD Element 8](#_Toc464216892)

[Armour HUD Element 8](#_Toc464216893)

[Carrying Mass HUD Element 8](#_Toc464216894)

[Weapon Statistics System 9](#_Toc464216895)

[Sacramento’s Introduction 9](#_Toc464216896)

[The Tutorial Enemy and the Combat Tutorial/Dialog Tutorial 9](#_Toc464216897)

[Sacramento’s AI 10](#_Toc464216898)

[Sacramento’s Movement Animations 10](#_Toc464216899)

[Second Section of the Introduction Level 11](#_Toc464216900)

[Private Matterson 12](#_Toc464216901)

# Elevator Pitch

A Third-Person shooter, in which the player has statistics related to their health and armour, is able to obtain items such as weapons and armour, follows linear routes through a level to complete an objective and they will also meet certain characters along their route that can help or hinder them. The aforementioned features are tied together by a story, which grants context to the actions the Player is taking, as well as the respective consequences of these actions.

# Story

At the mid-point (absolute), of the twenty-second century,

# Characters

# User Interface and Game Controls

## Main menu

The main menu is a UMG (Unreal Material Graphic) Widget, the mouse cursor is shown to the player, so that they can use this GUI (Graphical User Interface), which shows five buttons upon being created and added to the viewport, these are:

* New Game
* Load Game
* Options
* Credits
* Exit

### New Game

At the moment, this contains one button, the start button, this will start the game by opening the first level.

#### Start

Pressing on this button will stream load, open the first level and spawn the player at the first level’s starting spawn point, this level is called QTBDIntro, mostly a linear canyon, which the player must follow.

### Credits

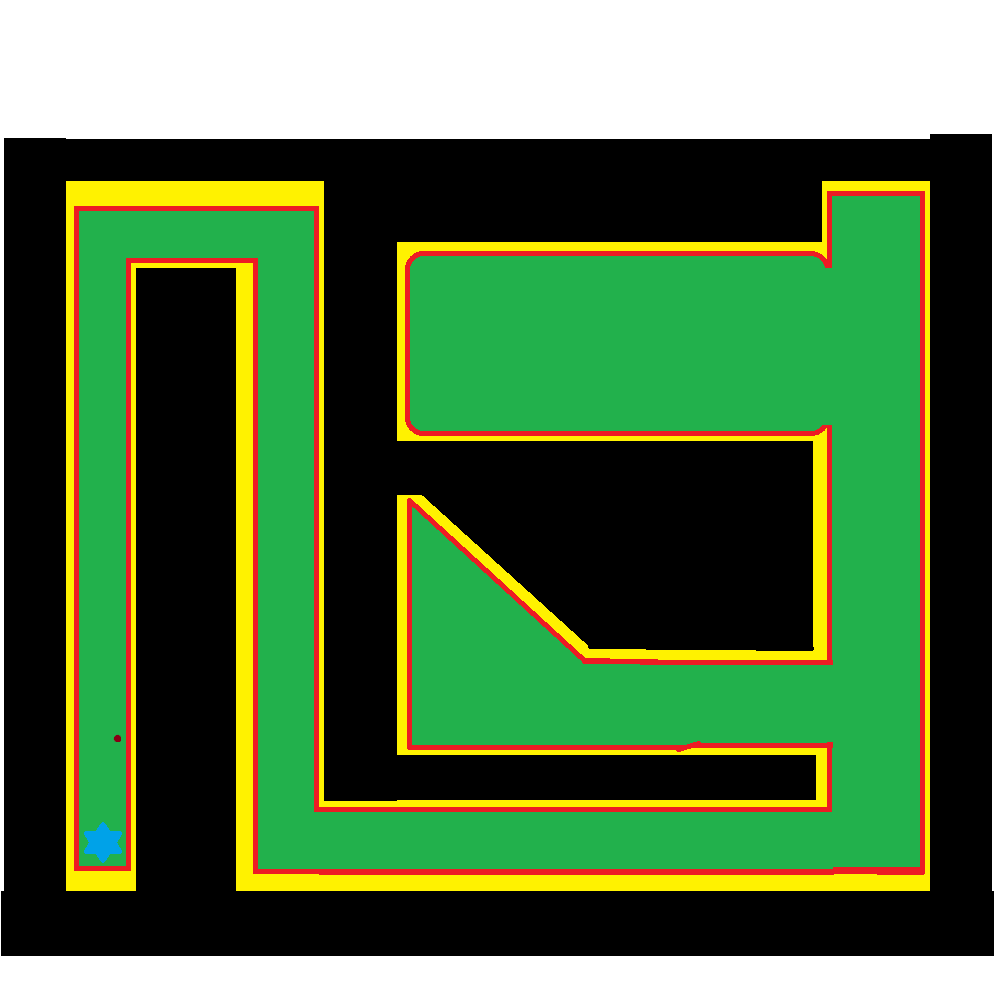
Pressing this button, will display the credits for the game.

### Main Menu UMG Widget Version 1.0.0

### Top-Level

# Level/Environment Design

The layout of the first level will be as follows in this draft layout design image.

Where black indicates areas that the player cannot get too and are at least 2 times higher than the player model.

Red indicates the start of the incline up to the top of the peaks, where it starts getting harder for the player to move up them.

Yellow indicates where the incline is starting to increase steadily, to the point where the player can get no higher.

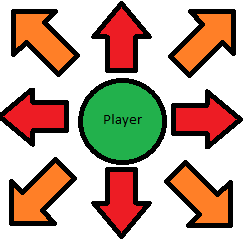
Green indicates where the player can move in the level, even over certain obstacles and through other elements of the level.

The blue star is where the player spawns into the level, to end the level, they have to complete an objective.

Finally, the maroon dot; indicates where there is a target dummy, which the player can damage and even destroy if they so wish.

## Sample View upon Starting First Level

### Player Movement System



The player can move along the X and Y axis, either positively or negatively, to go forwards, backwards, to the left and to the right, with W, S, A and D respectively, in addition, they can combine movement along both of these axes, simultaneously, to more in another 4 directions, forwards and to the left, forwards and to the right, backwards and to the left, backwards and to the right.

This can be represented by the following diagram, where the player is at the centre of the arrows, showing the 8 possible directions they can move in:

Where the red arrows, show directions that can be moved in by pressing one of the 4 movement keys (W, A, S or D), while the orange arrows, showing the other 4 directions that can be moved in with a combination of 2 movement keys (WA, WD, SA or SD).

They can change their movement speed in any and all directions with the left and right Alt keys, left to increase the % of base they go at, right to decrease that % of base speed. Currently +/- 0.01 (1%) of their base speed (1.00, 100%). The animation for moving in any of the 8 directions changes according to their move speed, from running at full pace, to walk, to shuffling, they can also sprint for a 1.75 times multiplier of their current speed, regardless of what % of base speed it is at. The bindings are Left-Alt for plus movement speed, Right-Alt for minus movement speed, if Ctrl is held down before pressing left alt, it will increase the player’s speed by 10%, not just 1%, while if Right-Ctrl plus Alt is pressed, it will decrease the player’s speed by 10%, not just 1%.

Note that the player’s speed, will be a factor in accuracy while shooting.

# Gameplay

### How PIBot behaves in the first level

As described in the section above, PIBot will start off in the first level, on the ground, deactivated. When the player gets close enough (as determined by PIBot’s Proximity Detection Sphere), PIBot will play the death animation that made PIBot appear deactivated, in reverse, so PIBot will appear to jump up, then after an intro sequence, will follow the player around, always attempting to be within 50 units of the player.

PIBot uses a Nav. Mesh to follow the player around, this Nav. Mesh is as big as the whole level, allowing PIBot to find the player, even if PIBot does not have line of sight to the player.

PIBot uses a duplicate of the player’s Skeletal Mesh, with a grey metal coloured material painted onto it, as well as using all of the player’s animations, with a copy of their Animation Blueprint, as their Skeletal Mesh is a duplicate of the one the player uses (replace with a mesh, that is similar to a standard Galactic Ranger, from Ratchet and Clank 3, GET A REFERENCE IMAGE FOR SUCH).

The player can just leave PIBot to jump up and begin their intro, without being held back, but PIBot will cut themselves short when the player gets to a certain point, in the path ahead, then shout; “Hey, wait up!” (or another similar phrase), and run after them (at an increased speed, so that they can catch up to the player?). The player is stopped after causing the trigger to fire, but PIBot does not move to them, they are also stopped if they enter the volume again, fix this, so PIBot does shout something at them, they are stopped only once and are also told to enter their name, before moving on.

### The CombatAdmin

When the player enters the second crevice of the canyon, at a certain point, the CombatAdmin will fly in and give its introduction, similar to PIBot, the player can keep going, then the CombatAdmin will complain about being interrupted by the player, PIBot will also respond to the Com Ad complaining.

The CombatAdmin will not use a Skeletal Mesh (for now), just a few primitive shapes, for a construction; with a similar build to that of certain current age drones (such as a Predator Drone), even though the CombatAdmin is not a drone though, but it is a robot, like PIBot, so is completely automatous.

### The Brick Weapon

The first weapon the player can get, is a brick, which they can throw at enemies and pick it up again if they so wish. I have yet to determine whether to let them keep onto the brick, or for it to be removed from their inventory, upon completing the first level.

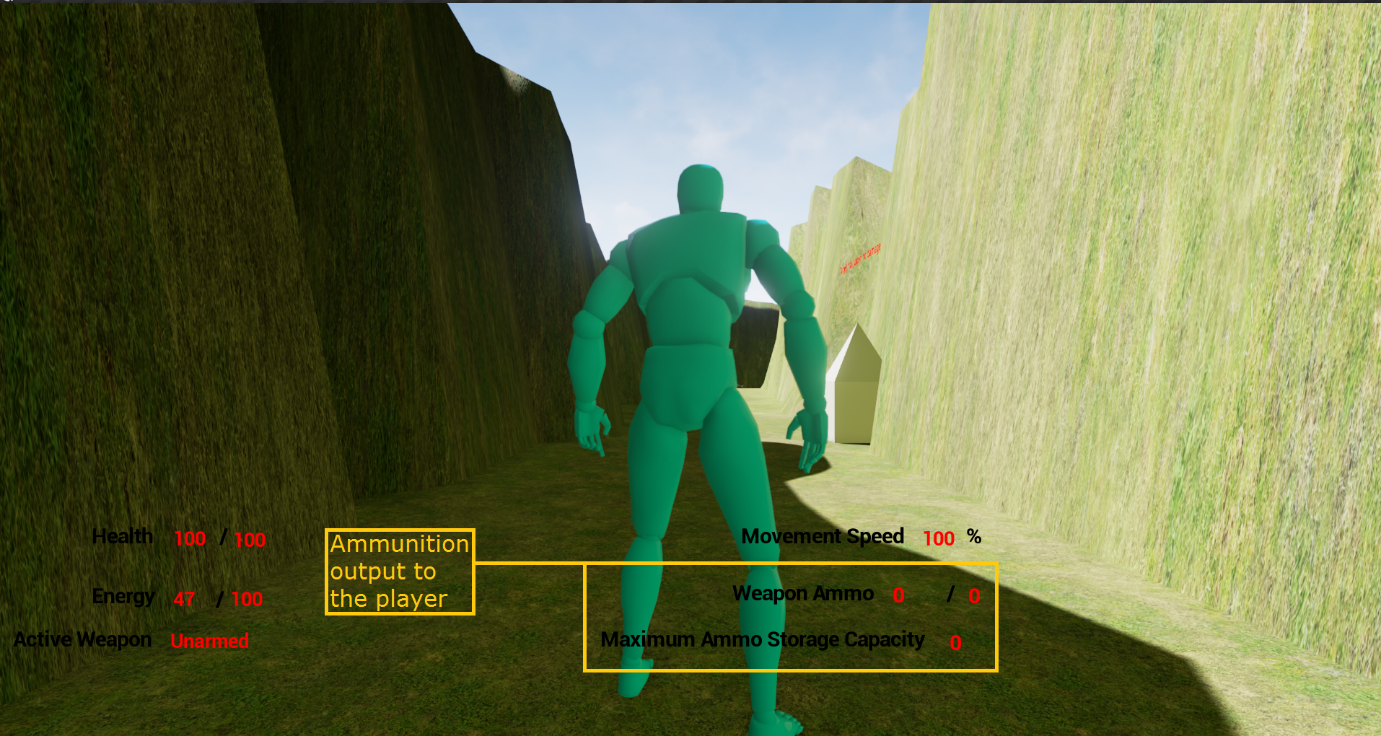
The brick will cause ~10 points of damage to be done to enemies, twice that of unarmed combat (resolve issues with unarmed combat) and so; not too much more, but the brick is a ranged projectile weapon, so they do not have to get as close to enemies, in order to engage combat with them. A radial impulse component provides the force and impulse to the brick, when it is thrown.

The BrickWeapon is a sub-class of QTBDWeapon, which in of itself, is a sub-class of QTBDItem.

## Weapon Pick up Confirmation

When a weapon is picked up by the player, a message telling them what weapon was picked up will be displayed to them for five seconds (display a screenshot of this, change this for any item that has been picked up too, whether it is a weapon or not).

## Weapon Ammo HUD Element



(The above screenshot shows this element of the player’s HUD, highlighted in a yellow box, with a caption to the left)

This text, shows how much ammo is left in that weapon’s total storage capacity, out of its maximum ammo, for weapons that have ammunition, there will always be an initial starting quantity of ammo, for melee weapons, this will be ‘0’, in addition, the loading mechanism’s (whether it is a clip, magazine etc.) storage capacity will also be shown, once again, being ‘0’ for melee weapons.

## Armour HUD Element

The player’s current active armour, plus the damage resistance of said armour, is shown on the HUD, here: (Add a screenshot to show such soon)

## Carrying Mass HUD Element

The player’s current carrying mass, as well as their maximum carrying mass, is shown on the HUD, here: (Add a screenshot to show such)

## Weapon Statistics System

There is a class, which all weapons have as a member, which stores the various stats of that weapon, such as its damage, how much ammo it has, uses per loading mech. Etc. This is used instead of spawning objects, say, above the player’s head, for getting the statistics of a given weapon, such as the BrickWeapon, while the BrickWeapon actor is spawned in a hidden location, this is the case for BrickWeapon, for unarmed, it is an actor without a mesh of any type, so it is hidden away near the spawn point for QTBDIntro.

## Sacramento’s Introduction

Before the combat tutorial begins (if it ever does), Sacramento will give their introduction, the control flow and final result for this is shown in this document, as the player can give many combinations of answers to Sacramento’s questions:[..\SacramentoDialogQAASessionControlFlow.pub](file:///\\CLUSTER1-D27\HOMEA14\M\2MORAJ05\Gameplay%20and%20Game%20Design\SacramentoDialogQAASessionControlFlow.pub)

## The Tutorial Enemy and the Combat Tutorial/Dialog Tutorial

This enemy also uses a copy of the same skeletal mesh and animation blueprint, as the player character and PIBot.

The combat tutorial/dialog tutorial will occur, after PIBot is trying to give the player advice, in either one of two possible routes, then Sacramento will appear and give their introduction.

The player will be put into Cinematic mode at this point in the level, so that Sacramento can give their intro, PIBot will say some dialog towards the player after Sacramento gives their first section of introduction dialog, then will teleport hide out of sight.

Sacramento will be spawned in a section of the canyon, outside the player’s line of sight, but then, will teleport near the player and give their introduction, putting the player in cinematic mode.

In either case, the player can then make various choices, which can affect the combat tutorial in some manor (even causing Sacramento to join their side, so there will not be one), depending on the choices the player makes, Sacramento will either attack the player or side with the player. The player can also begin combat early if they so wish, Sacramento will be frustrated at such though. Sacramento’s introduction can also be interrupted as well, which will cause them to become hostile, only saying part of their second introduction dialog section lines before attacking the player if they get interrupted.

Sacramento has a Prototype Shotgun for their primary weapon, they also have a grenade hidden near where they spawn, before teleporting, which if the player finds, can be used against Sacramento.

If the player has BrickWeapon, it will deal bonus damage against Sacramento, if the player finds and uses Sacramento’s hidden grenade and misses Sacramento, Sacramento will throw an insult their way, if the player hits Sacramento, if Sacramento is hit by the full force of the explosion (for maximum damage, thus, if the grenade explodes within 1,1,1 or so, of Sacramento), Sacramento will teleport away (possibly fighting the player again later on in the level?) and the tutorial will be over.

The Grenade has an explosion particle effect sequence and a sound, for when it explodes, it will kill whichever character is in the blast radius of the time of the grenade exploding, the grenade will be primed to explode upon being thrown, then will explode after approximately 5 seconds have elapsed in game.

### Sacramento’s AI

The logic of Sacramento’s behaviour tree is detailed in this document: [..\SacramentosAIBehaviorTreeBlueprint.pub](file:///\\CLUSTER1-D27\HOMEA14\M\2MORAJ05\Gameplay%20and%20Game%20Design\SacramentosAIBehaviorTreeBlueprint.pub)

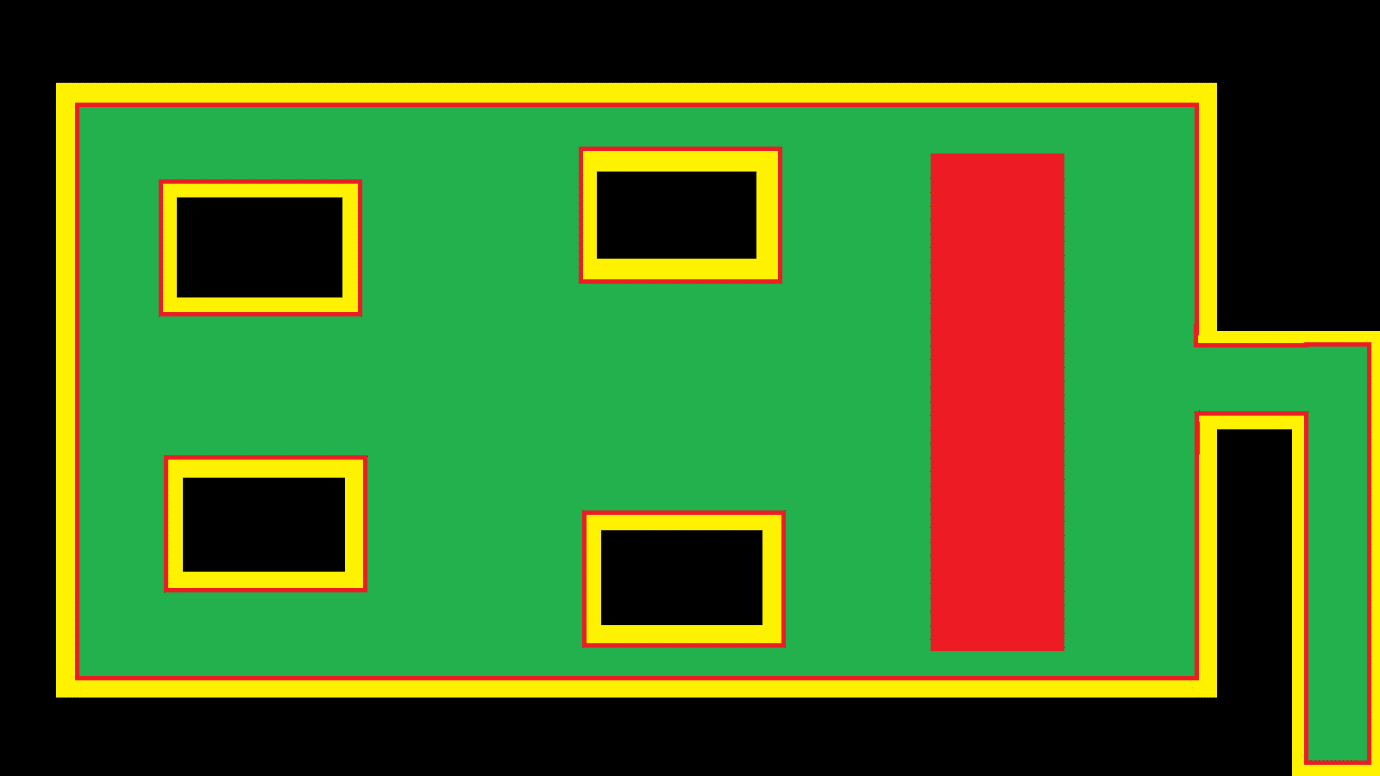
## Sacramento’s Movement Animations

Sacramento has the same animations for moving, jumping and dying as the player, but has additional animations for drawing his shotgun from his back, firing that shotgun, reloading it, as well as moving without firing it and moving while firing or reloading, as can be seen in this video presentation:

[..\..\Videos\Games Dev. Videos\Quest; To Be Defined; Sacramento Animation Sample.mp4](file:///\\CLUSTER1-D27\HOMEA14\M\2MORAJ05\Videos\Games%20Dev.%20Videos\Quest;%20To%20Be%20Defined;%20Sacramento%20Animation%20Sample.mp4)

### 

### Second Section of the Introduction Level

After having their encounter with Sacramento, the player can now freely move into the second section of the canyon, with a planned layout of such:

Where Black indicates where the player cannot reach, yellow where the gradient gets steeper and steeper, to the point the player will be able to go no higher, red, where there is a traversable, but notably gradient and green where it is flat and the player can move around freely.

In this section, there will be a six of instances of the first NPC, which will always be hostile towards the player, Private Matterson.

### Private Matterson

When the player gets near to where the squad of multiple instances of Private Matterson is, the player will not be attacked by them straight away and can hear them having a light chat while the player and/or Sacramento do not reveal themselves.

Private Matterson has a cheap rifle (represented by the assault rifle from the prototype weapons pack), poor body armour and 100 hit points. Whenever any instance of Pvt. Matterson has the player in their line of sight, as well as them being close enough to that particular instance, they will look at the player, as well as notifying the other instances that they have seen a target and they should look in that direction as well.

Upon turning to face their target, they will draw their weapon and either start shooting, or heading to the location they were most recently looking at, then checking to see if they have line of sight once again, they will keep repeating this cycle until they can see their target, or their target has been eliminated.

They will also inform other instances that are nearby, that they have spotted a target nearby, which will cause the other instances to look in that area, also, if they have a target but line of sight is blocked by another instance of Pvt. Matterson, they will move to the left or right, to regain a line of sight to their current target.