

Treball de Fi de Grau

Recycle VR

Arnau Pérez Castrillo

Grau en Multimèdia

Tutors: Enric Vergara Carreras i

Jose Díaz Iriberry

Vic, Maig de 2019

RESUM

Títol: Recycle VR

Paraules clau: Unity, Realitat Virtual i Reciclatge

Autor: Arnau Pérez Castrillo

Tutors: Enric Vergara Carreras i Jose Díaz Iriberry

Data: Juny de 2019

En aquest treball s'explica bàsicament l'aplicació o videojoc enfocat a la conscienciació del reciclatge, on mitjançant unes ulleres o des del mateix ordinador es pot simular un entorn on s'ha de recollir totes les deixalles possibles i reciclar-les al contenidor corresponent. Aquest videojoc no vol complir la funció d'entretenir a un públic que busca jugar a videojocs per divertir-se una estona, sinó que, aquest treball, entraria més en el camp dels *serious games*, per tant la seva funció és més aviat educativa que no pas lúdica.

L'objectiu principal del projecte és demostrar que, amb tots els coneixements adquirits durant aquests últims anys gràcies a la carrera que he cursat, puc ser capaç de crear per mi sol un videojoc de realitat virtual des de zero. El desenvolupament d'un videojoc tridimensional necessita coneixements amplis de moltes disciplines diferents, com ara la programació; coneixements de disseny 3D; edició d'imatge, vídeo i so; coneixements de funcionament de motors de videojocs; experiència en realitat augmentada i realitat virtual; i molts altres aspectes.

Per poder realitzar aquests objectius pel joc, s'han creat un seguit d'escenes o mapes on es simulen uns entorns tridimensionals, com un bosc, que mitjançant els controladors de les ulleres de realitat virtual et pots desplaçar en aquests i recollir les deixalles i dipositar-les en el contenidor corresponent.

Finalment, després de veure el resultat del treball, considero que el joc ha quedat millor del que al principi del projecte imaginava ser capaç de fer, tenint en compte

que he estat treballant tot sol i sense tots els coneixements d'algunes parts del desenvolupament de videojocs.

Title: Recycle VR

Key words: Unity, Virtual Reality and Recycling

Author: Arnau Pérez Castrillo

Tutors: Enric Vergara Carreras and Jose Díaz Iriberry

Date: June 2019

This work explains, basically, the videogame focused on the awareness of recycling, where thanks to a virtual reality headset or from the same computer it can simulate an environment where all possible trash must be collected and recycled in the corresponding container. This video game does not want to play the role of entertaining an audience that seeks to play video games for a while, but this work would be more in the field of serious games, because its role is rather educational than playful.

The main objective of the project is to demonstrate that with all the knowledge acquired during these last years and thanks to the career I have studied, I can be able to create a virtual reality videogame by myself. Since the development of a 3D video game needs lots of knowledge in many different disciplines such as programming, knowledge of 3D design, image and video editing, video game engineer knowledge, augmented reality experience and virtual reality and many other aspects.

To achieve these goals for the game, a series of scenes or maps have been created where three-dimensional environments, such as a forest, are simulated, which thanks to the VR glasses and it's controllers, the user can be moved to them and collect waste and deposit them in the corresponding container.

Finally, after seeing the result of the work, I consider that the game has been better than at the beginning of the project it imagined being able to do, considering that I have been working alone and without all the knowledge of some parts of the development of video games.

ÍNDEX

RESUM.....	2
GLOSSARI DE TERMES	7
INTRODUCCIÓ	9
Contingut del treball	9
Motivació	10
Formulació del problema.....	10
Objectius del projecte	11
Abast del projecte.....	11
CONTEXTUALITZACIÓ	12
Què és un aparell VR i com funciona?.....	12
Productes similars	13
<i>The Elder Scrolls V: Skyrim VR</i>	13
<i>WildLife VR</i>	14
<i>PI VR Earth</i>	14
<i>The Kremer Collection VR Museum</i>	15
PLANIFICACIÓ I METODOLOGIA.....	16
Distribució del temps	16
SOFTWARE UTILITZAT	19
<i>Unity</i>	19
<i>Blender</i>	19
<i>SpeedTree Modeling</i>	19
<i>Photoshop i Illustrator</i>	20
<i>Audacity</i>	20
<i>Unity Cloud</i>	20
DESCRIPCIÓ DEL JOC:.....	21
DISSENY:.....	24
Estètica	24
<i>UI/HUD</i>	25
Controls	26
DISSENY DE MAPA I NIVELLS.....	28
PROGRAMACIÓ	33
Agafar i guardar objectes	33
Crear els objectes	35
Interfície d'usuari	36
Implementació VR	38
PUNTS FORTS I FEBLES	41
CONCLUSIONS I FUTURES MILLORES.....	42
AGRAÏMENTS.....	45
WEBGRAFIA.....	46

ÍNDIX DE FIGURES

FIGURA 1 - Equip complet de realitat virtual d'Oculus Rift.....	12
FIGURA 2 - Exemple de gameplay Skyrim VR.....	13
FIGURA 3 Escena del joc WildLife de realitat virtual.....	14
FIGURA 4 - Escena del simulador.....	15
FIGURA 5 - Sala del museu en VR.....	15
FIGURA 6 Esquema de Gantt que he utilitzat durant el desenvolupament del projecte.....	18
FIGURA 7 - Interfície del programa Blender.....	19
FIGURA 8 Sala principal on es mostren els cartells amb imatges i la informació necessària per jugar.....	21
FIGURA 9 Captura de la escena del mapa de bosc des de vista aèria i en vista isomètrica.....	22
FIGURA 10 Escena final des de l'editor de Unity amb alguns dels objectes per reciclar i els contenidors.....	23
FIGURA 11 Exemple visual d'estètica low-poly per videojocs.....	25
FIGURA 12 - Exemple de la diferència entre les versions de PC o consola i VR.....	26
FIGURA 14 - Exemple de desplaçament mitjançant tele-transportacions amb unes ulleres de realitat virtual i controladors VR.....	27
FIGURA 15 Editor del meu projecte de Unity i a la dreta el menú del Terrain.....	28
FIGURA 16 Exemple de model 3D sense textura creat a blender.....	30
FIGURA 17 Exemple d'arbre e interfície del SpeedTree Modeling 7.1.5.....	31
FIGURA 18 Captura del joc intentant recollir un objecte seleccionat.....	34
FIGURA 19 Captura de l'editor de Unity amb un SpawnPoint.....	35
FIGURA 20 UI del joc que inclou el minimapa, inventari i marcador de temps/objectes.....	37
FIGURA 21 Implementació dels controladors de moviment del Oculus dins l'editor de Unity ...	39

GLOSSARI DE TERMES

UI (User Interface) - És l'espai de la pantalla on l'usuari es pot posar en contacte amb l'ordinador, mòbil o qualsevol altre dispositiu.

HUD (Head Up Display) - És la informació que normalment en els videojocs se'ns mostra per pantalla però que no representa cap forma d'interacció directa amb l'usuari.

C# - Llenguatge de programació orientat a objectes desenvolupat per Microsoft l'any 2002 i que la seva sintaxi deriva de C / C++.

Spawn Point – En els videojocs un punt de spawn sol ser una localització concreta del mapa o l'escenari on normalment apareixen els jugadors, enemics, recompenses o quelcom important.

Asset – Un asset és qualsevol, peça de media o d'informació que pot ser usada dins els nostres projectes de Unity.

Prefab – A Unity, un *prefab* és un sistema que ens permet emmagatzemar objectes del joc ja muntats amb tots els seus components com un *asset* reusable.

RawImage – Peça que mostra una textura bidimensional per la UI.

Frame – És una de les moltes imatges estàtiques que es mostren durant un interval de temps. En el cinema, les pel·lícules solen anar a 30 frames cada segon.

Instanciar – És l'acció de clonar un objecte o *prefab* ja existent a l'escena del joc.

Collider – És un component primitiu dels objectes de Unity que ens permet la col·lisió entre altres objectes tridimensionals.

SDK (Software Development Kit) – Conjunt d'eines de software que ens permeten crear jocs o aplicacions per un cert paquet de programari.

Shooter – Gènere o subgènere de videojocs on mitjançant armes es sol posar a prova els reflexos i les habilitats del jugador per superar els nivells.

BitMap – També conegut com a mapa de píxels on cada píxel pot guardar més de dos colors.

Acceleròmetre – Sensor present en les ulleres de realitat virtual que calcula els moviments en els eixos de coordenades X, Y i Z.

Low-poly – Estil de disseny 3D que consisteix a modelar tots els objectes amb un número relativament petit de polígons. Molt utilitzat en videojocs i pel·lícules d'animació.

Canvas – Zona dins de l'editor de Unity on s'agrupen tots els objectes referents a la UI.

INTRODUCCIÓ

Contingut del treball

El contingut d'aquesta memòria explica en detall tots els procediments seguits per tal d'elaborar aquest treball i quines han estat les eines i programaris utilitzats per aconseguir-ho.

Aquest document es basa en el seguiment de la creació d'una aplicació o videojoc per realitat virtual i ordinador creat en el motor de videojocs *Unity*. L'usuari mitjançant un equip de controladors i ulleres de realitat virtual es veurà immers en un món tridimensional en el qual, mitjançant els controladors, haurà d'anar recollint un seguit d'objectes o residus col·locats de forma aleatòria en un escenari que ha estat creat per simular un bosc el més real possible. Durant la primera fase del joc, el jugador haurà de seguir unes pautes que se li mostraran per pantalla, per tal d'aprendre els controls del joc. Quan tingui tots els controls i els objectius clars se'l transportarà al bosc on, com he dit, haurà de recollir els residus abans d'accedir a l'escenari final, una sala on apareixeran els objectes que prèviament ha recollit, i que haurà de dipositar en el contenidor corresponent per tal de reciclar aquestes deixalles correctament.

La història d'aquest joc consisteix en el fet que el jugador es veu immers en un mapa que reflecteix la realitat dels nostres boscos i altres paratges naturals en l'actualitat, on la contaminació i les deixalles s'hi acumulen d'una forma més preocupant cada dia que passa, per culpa de l'ésser humà. En part aquest projecte l'he creat per conscienciar a alumnes, nens i tothom qui el vulgui jugar sobre el canvi climàtic i la importància del reciclatge per tal d'evitar futures conseqüències. Així que, aquest joc és, en sí, una eina per ajudar a entendre a les persones del mal que suposa, i en intentar ensenyar d'una forma més lúdica com reciclar els diferents tipus de residus que ens podem trobar al nostre dia a dia.

Motivació

Durant els mesos previs a començar aquest treball de fi de grau em van sorgir diverses idees sobre què hauria de treballar o quin seria un projecte adequat. La majoria d'aquestes idees eren al voltant del desenvolupament i creació d'algun tipus de videojoc, ja que durant la carrera han estat les assignatures que més he gaudit al cursar-les. Al final, i gràcies a un projecte de la meva estada d'Erasmus a Finlàndia, vaig decidir que el projecte final seria fer un joc per ensenyar la importància del reciclatge, i que a poder ser, es pogués implementar en realitat virtual.

Com he dit abans, sempre havia volgut posar a prova tots els coneixements que he anat assolint durant els anys del grau i aquest era el moment perfecte, ja que per crear un videojoc, sense tot el que he après durant aquests últims quatre anys hauria estat impossible.

Formulació del problema

Primerament, em vaig haver de plantejar si aquest projecte era viable fer-lo com a treball de final de grau, ja que pensava que en tot un semestre em donaria temps de sobres a poder acabar el projecte que, en un principi, tenia en ment, però no va ser així. En la primera reunió amb els tutors em van aconsellar que no volgués crear un projecte tan gran, ja que no em donaria temps i que seria molt més adient reduir el treball i intentar aconseguir un resultat molt més polit.

Les primeres setmanes van anar destinades a veure com podria resoldre tots els reptes que se'm plantejaven a l'hora de voler crear un projecte que seguia sent tan gran. Per fer-ho vaig intentar buscar informació sobre crear entorns virtuals semblants al que volia crear, si ja existien eines que em permetessin programar les mecàniques que volia aplicar al joc, etc.

Objectius del projecte

El primer objectiu que tenia en ment era ser capaç de crear un joc des de zero, ja que moltes vegades pel meu propi compte havia treballat en projectes que mai acabava per falta de motivació o de temps. En volia experimentar tot el procés de creació des de zero, i que aquest em servís com a experiència per si mai he de treballar en cap altre projecte semblant.

El segon objectiu que tenia present a l'hora de crear un projecte tan gran era que, un cop acabat, pogués utilitzar-lo com a un altre dels meus continguts al meu *port foli*. Per això, un bon resultat era essencial, ja que aquest joc podria ser la meva carta de presentació davant del món laboral per trobar una feina en la indústria dels videojocs, que és on he volgut treballar sempre.

Un altre dels objectius que tenia era el de millorar en alguns camps necessaris per a la creació de videojocs, en els que crec que no sóc tan bo com altres companys de classe o estudiants de graus similars. El principal d'aquests camps és el de la programació, i crec que en aquest és en el que aquests últims mesos, i gràcies a aquest projecte, trobo que he millorat més notablement. Ha sigut, en part, gràcies a totes les hores que he hagut de treballar en la programació de les mecàniques del treball. També he millorat altres disciplines relacionades amb multimèdia o la creació de videojocs, com l'edició d'àudio i imatges, el modelatge 3D i moltes altres.

Abast del projecte

El joc està dirigit a una audiència molt àmplia, ja que tothom el podrà jugar, però el públic al qual el vull enfocar és sobretot a nens i adolescents d'entre 8 a 16 anys, que són als qui més agraden aquests tipus de tecnologies i qui seran més receptius a voler provar un joc educatiu. La intenció del joc és que es pugui utilitzar a les aules o en qualsevol entorn educatiu per tal d'ensenyar a reciclar d'una manera més divertida que, per exemple, veient un documental o llegint un llibre.

CONTEXTUALITZACIÓ

Què és un aparell VR i com funciona?

Durant els últims anys l'ús d'aparells relacionats amb la realitat virtual s'ha disparat. Gràcies, en gran part, a l'abaratiment i la gran competència en el camp dels models d'ulleres VR, quan es va presentar, l'any 2010, el primer prototip d'Oculus rift. També han afectat altres factors com les grans campanyes de màrqueting on es mostren aquests aparells com el futur de l'entreteniment i la gran llibreria en expansió de continguts, pel·lícules i videojocs que es poden consumir amb aquestes ulleres actualment.

Com aquest treball tracta de la creació d'un videojoc, i la possibilitat d'implementar-lo a les ulleres de realitat virtual, m'agradaria contextualitzar i explicar breument com funciona un d'aquests aparells.

Primer de tot, la gran majoria de dispositius de realitat virtual funcionen d'una forma similar. Dins del cos de l'aparell trobem una pantalla d'alta resolució per cada ull, unes lents correctores per adaptar-se a qualsevol classe de vista, i un seguit d'acceleròmetres i altres sensors per detectar la posició del jugador. A part d'això, alguns aparells de realitat virtual necessiten un seguit de sensors extres que es col·loquen a l'habitació on es vol jugar, per tal de detectar millor la



FIGURA 1 - Equip complet de realitat virtual d'Oculus Rift

localització, la rotació, les mans i els moviments de qui porta les ulleres, creant un entorn on es pugui submergir.

Un cop tot l'equip està a punt mitjançant les ulleres i els controladors que normalment s'utilitzen, podem simular un entorn virtual dins el qual se'ns pot projectar un videojoc, una pel·lícula o un paisatge.

Productes similars

Per aquest projecte volia buscar alguns jocs o aplicacions semblants i utilitzar-les com a referències, però no he sabut trobar en cap moment cap projecte o treball semblant per tal d'ajudar-me a l'hora de buscar inspiració pel meu joc. Així que, en comptes de parlar extensament d'un projecte molt semblant al meu, faré un breu *benchmark* de quatre audiovisuals semblants o que integren coses que en el seu moment em van semblar interessants per afegir al meu projecte.

The Elder Scrolls V: Skyrim VR

Skyrim és un dels jocs més populars de la història, ha estat publicat en la majoria de consoles modernes disponibles, i ha venut centenars de milers de còpies arreu del món. Més endavant en parlaré, però d'aquest joc en vaig voler agafar l'ús que en fan de la interfície d'usuari en la realitat virtual.



FIGURA 2 - Exemple de gameplay Skyrim VR

Es basa estèticament a l'edat mitjana, on grans i espessos boscos poblaven la Terra i *The Elder Scrolls* és una de les franquícies millor reconegudes pel disseny de nivells i dels seus enormes mapes.

WildLife VR

WildLife VR és un senzill joc tipus *shooter* que funciona mitjançant un parell d'ulleres virtuals i els controladors de les mans per apuntar amb les pistoles. El que m'agrada d'aquest joc és la estètica *low-poly* que volia implementar en un principi, però que al final no he decidit utilitzar. A més, aquest joc està creat en *Unreal Engine 4*, un motor de creació de videojocs bastant semblant a *Unity*.

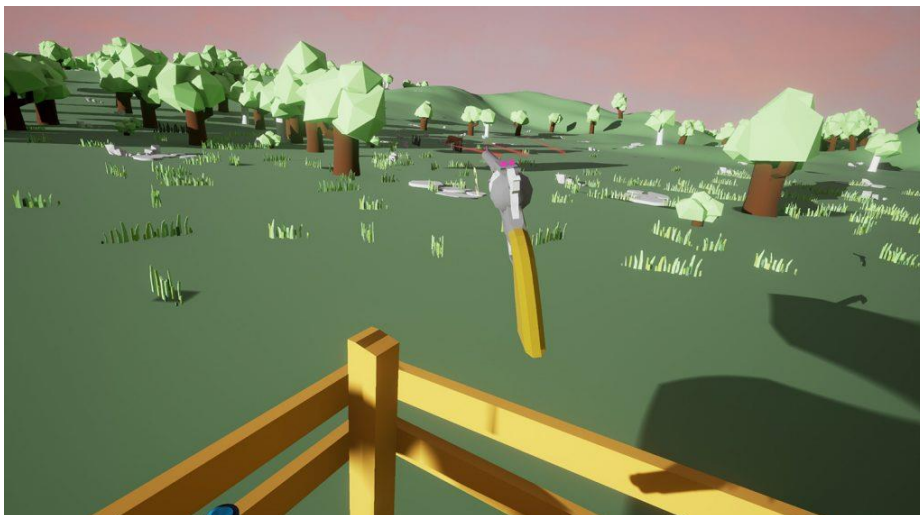


FIGURA 3 Escena del joc *WildLife* de realitat virtual

PI VR Earth

PI VR Earth és un simulador per dispositius mòbils, en el que, un cop l'usuari ha posat el seu dispositiu mòbil en algun dels dispositius que permeten convertir-lo en un equip de realitat virtual, se'ns col·loca en la vista d'un mirador on, depenent del nivell seleccionat, podem veure diferents entorns naturals i aprendre sobre ells mitjançant una sèrie de vídeos o textos explicatius que el mateix simulador incorpora. D'aquest joc m'ha agradat el disseny dels entorns i que es pugui fer servir per a dispositius mòbils.



FIGURA 4 - Escena del simulador

The Kremer Collection VR Museum

Tot i no ser una aplicació d'*Oculus Rift*, aquest *software* ens permet entrar dins un museu privat que, mitjançant tècniques de modelatge digital, s'ha creat un entorn idèntic al museu però en format tridimensional. Se'ns permet entrar-hi amb les ulleres de realitat virtual per tal d'observar les obres d'art que normalment no són obertes al públic. D'aquest programari m'ha agradat la mecànica que utilitza per desplaçar-se pel mapa, ja que, mitjançant tele transportacions, es pot moure d'un lloc a un altre en les diferents sales que componen aquest museu.



FIGURA 5 - Sala del museu en VR

PLANIFICACIÓ I METODOLOGIA

En aquesta secció explico el procés de planificació del temps de tota la feina que comporta un projecte tan gran com ha estat aquest treball de final de grau, i quina metodologia de treball he seguit per tal d'acabar el projecte amb uns resultats satisfactoris. A més, també es detalla el software utilitzat durant el desenvolupament del treball.

Com en la majoria de projectes en els quals he treballat semblants a aquest, he seguit la mateixa metodologia. Aquesta metodologia seguida és la metodologia de desenvolupament basat en funcionalitats o, en anglès *Feature-Driven development*. Aquesta consisteix a desenvolupar un projecte seguint fites. En el meu cas, durant tot el projecte, m'he anat posant petites fites que, durant el temps he anat resolent. Per exemple, si volia acabar la part de programació del projecte, no començava tot el codi de cop, sinó que primer feia una petita part i m'assegurava que funcionava correctament. I així ha estat com he treballat en tots els camps d'aquest projecte.

Distribució del temps

Durant els primers dies de treball vaig voler anar treballant lliurement sense seguir cap pauta ni esquema, però ràpidament em vaig adonar que no era una bona estratègia si volia fer un bon treball i acabar amb un resultat del qual en pogués estar orgullós.

És per això que, per recomanació del meu tutor, vaig crear un document *d'Excel* amb un esquema de planificació de *Gantt*. En aquest document vaig anotar totes les tasques que havia de fer durant el desenvolupament del joc per tal d'acabar-lo tal com jo volia i poder fer-ho a temps.

Aquestes tasques les vaig dividir en tasques més petites i els hi vaig donar un pes en sessions de treball per saber més o menys tot el temps que necessitaria.

Gràcies a aquest esquema m'ha estat molt més fàcil organitzar-me i planificar tot el desenvolupament del videojoc.

En la següent figura, que representa una taula de *Gantt* per fer un esquema de la repartició del temps durant el desenvolupament, es pot veure que, durant les primeres setmanes em vaig centrar a crear les escenes principals i les mecàniques per tal que el joc fos funcional i es pogués jugar sense cap problema. Un cop acabades aquestes tasques em vaig centrar en la creació de l'escenari i dels objectes a recollir. Algunes de les tasques que surten fora de posició, per exemple, la creació de les sales inicials i finals o la creació dels objectes tridimensionals, les vaig voler fer més aviat en el desenvolupament, ja que algunes tasques que pensava que em comportarien més hores de treball van resultar que en poc temps ho vaig tenir llest. En canvi, altres tasques com la implementació de les ulleres, vaig trigar més a posar-m'hi perquè pensava que hi podria treballar simultàniament amb la versió sense realitat virtual.

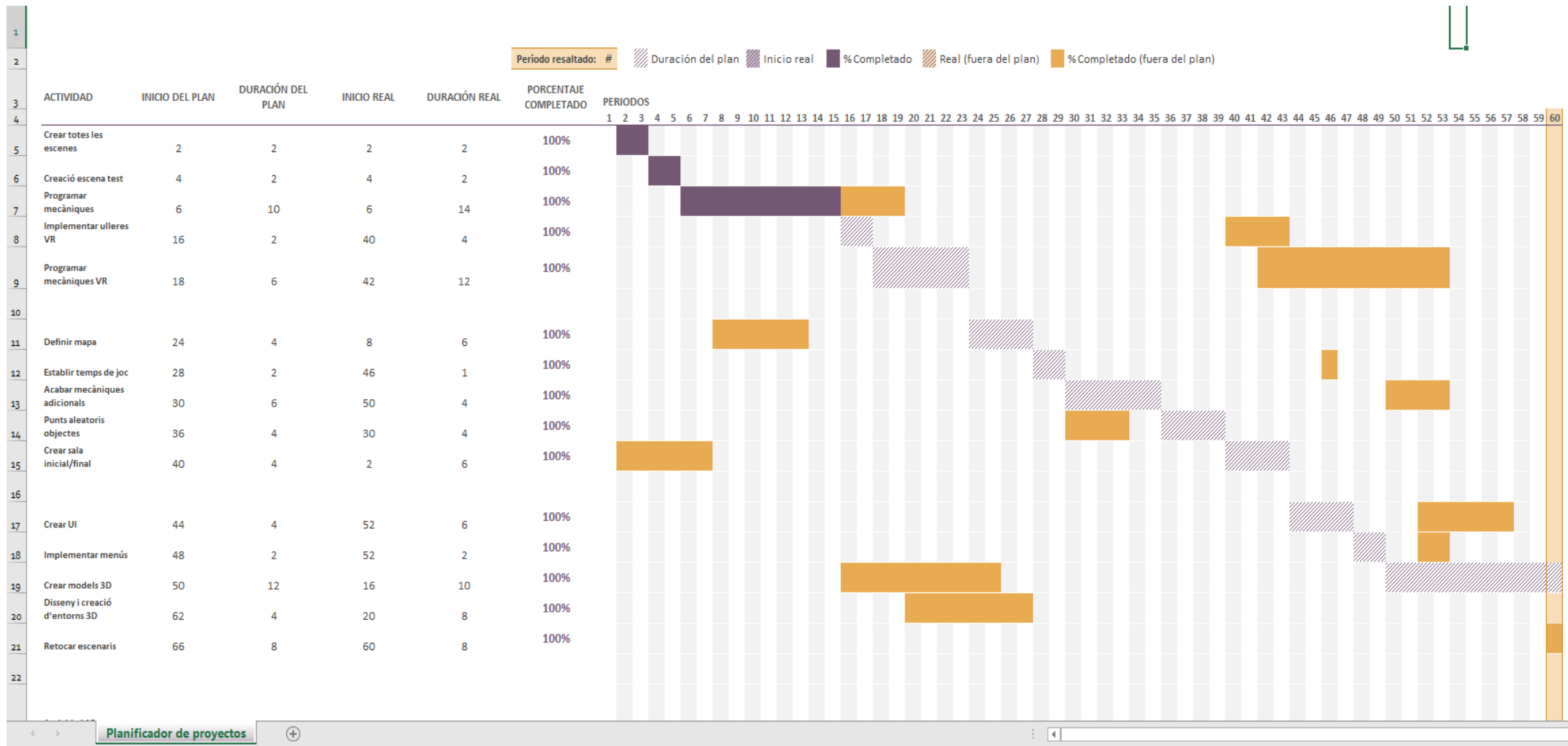


FIGURA 6 Esquema de Gantt que he utilitzat durant el desenvolupament del projecte

SOFTWARE UTILITZAT

Unity

Unity és un dels motors de creació de videojocs més popular de l'actualitat, ja que és gratuït, complet i pràcticament tothom pot aprendre a utilitzar-lo sense necessitat d'una gran formació en videojocs o programació. També és usat, en menys mesura, en indústries alienes als videojocs, com el cinema, l'automoció, l'arquitectura i la construcció.

Blender

Blender és un programari d'edició tridimensional gratuït que permet modelatge d'objectes, edició de materials i textures, animació, renderització d'escenes i també com a motor de jocs, però aquest últim punt no s'ha arribat a popularitzar gaire.

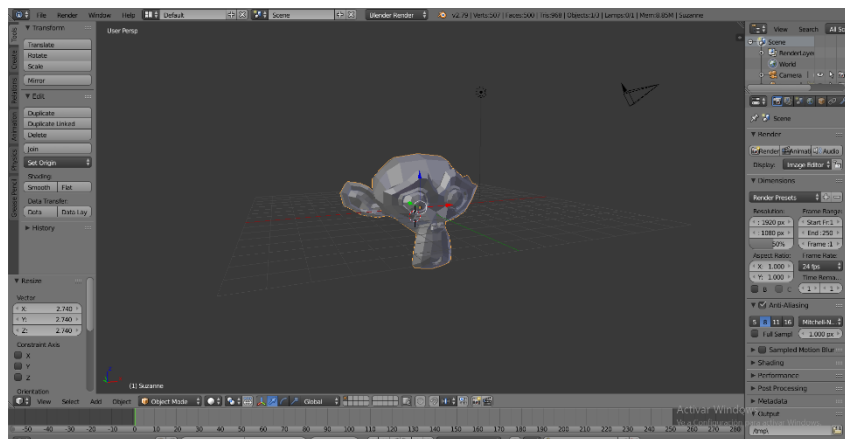


FIGURA 7 - Interfície del programa Blender

SpeedTree Modeling

Per la creació d'arbres per l'escenari del bosc que he creat, he utilitzat aquest *software* de pagament. És un *software* que permet crear models tridimensionals de tota mena i espècies diferents d'arbres d'una forma molt més senzilla que

amb un programari com *Blender*. Aquest és un *software* utilitzat principalment per grans empreses i estudis de videojocs i de cinema d'animació.

Photoshop i Illustrator

Photoshop i *Illustrator* són els programes d'edició d'imatges tant en format *BitMap* (*Photoshop*) i en format Vectorial (*Illustrator*) per excel·lència. Els he usat per petites edicions de textures i d'elements de la *UI*.

Audacity

Audacity és programa d'edició i gravació d'àudio gratuït i molt senzill d'utilitzar. El vaig usar per editar i retallar alguns dels sons que hi ha integrats al projecte.

Unity Cloud

A diferència dels programaris anteriors, aquest punt, és una funcionalitat dins el mateix *Unity* que ens permet emmagatzemar els nostres projectes al núvol i accedir-hi des de qualsevol ordinador. Aquest *software* em va ser molt pràctic a l'hora de treballar en paral·lel des de dos ordinadors, ja que mentre era la universitat o fora de casa i des del meu portàtil podia treballar en tasques que no necessitessin molta potència gràfica com treballar en la *UI* o en la programació, i un cop havia arribat a casa podia seguir treballant amb la versió que estava editant al portàtil prèviament i des de l'ordinador de sobretaula que em permet crear tasques que requereixen un PC amb més potencia.

DESCRIPCIÓ DEL JOC:

Aquest joc està dividit en tres escenaris importants o tres escenes, en la primera ens trobarem en una petita sala on se'ns explicarà els controls per jugar, tots els objectius del joc i com funciona tot. En la segona escena, la del joc com a tal, és on el jugador haurà de fer servir el que prèviament se li ha explicat per tal de superar aquest nivell i passar a l'última escena, que és una altra sala o escenari on hi haurà uns contenidors i serà on acabarà el joc.

Un cop iniciat el joc el jugador serà enviat a una sala on, mitjançant cartells i notes escrites a l'entorn on es troba, se l'explicarà el funcionament del joc. Els cartells explicaran com utilitzar els controls de les ulleres de realitat virtual i els seus controladors o, en cas contrari, del ratolí i el teclat si és que es juga des de PC. En aquest moment es mostraran totes les mecàniques que haurà de tenir en compte durant el joc, els objectius inicials i finals del joc, les normes del joc i més informació que li pot ser d'utilitat mentre jugui.

Sense fer servir cap *UI* a la sala principal s'aconsegueix una sensació més immersiva i menys intrusiva a la vista de l'usuari. En aquesta sala també hi trobem diferents cartells que ens mostren imatges de la importància del reciclatge, i un altre panell on, prement un botó dels controls, se'ns mostraran diferents missatges importants pel joc. Un cop s'han vist tots els missatges es tele-transporta al jugador a l'escena principal del projecte, on podrà començar el joc.



FIGURA 8 Sala principal on es mostren els cartells amb imatges i la informació necessària per jugar

Per tal de situar al jugador dins el mapa un cop s'ha iniciat el joc, es disposarà d'un mini-mapa que mostra la posició en temps real del jugador, i amb uns petits punts vermells al mapa, on es troben els objectes a recollir.

El jugador apareix en un punt prèviament definit al centre del mapa, on haurà de trobar i recollir tots els residus que li sigui possible. El jugador estarà condicionat a intentar trobar totes les escombraries que pugui en menys d'un cert temps que se li mostrarà per *UI* mitjançant un compte enrere.

Els objectes sempre apareixeran en certs punts del mapa definits prèviament mitjançant codi, però un cop es crea l'escena es triarà aleatòriament on estaran localitzats exactament.



FIGURA 9 Captura de la escena del mapa de bosc des de vista aèria i en vista isomètrica

Per crear aquest mapa he necessitat fer models de diferents tipus d'arbres, objectes naturals com pedres, arbustos, flors, gespa, etc. Un cop el temps ha arribat a 0 a l'escena del bosc, encara que el jugador no tingui tots els objectes, se'l transportarà a una sala on estaran tots els tipus de contenidors diferents que necessitarà i els objectes que abans ha recollit. El jugador haurà d'agafar els residus d'un en un i llençar-los al contenidor pertinent per tal d'aconseguir punts.

Si encerta que, per exemple, una llauna de refresc va al contenidor groc, sumarà punts i en cas contrari en perdrà. Agafant la puntuació que ha obtingut durant la

recollida d'objectes i afegint els punts que ha aconseguit dipositant-los en els contenidors pertinents, es farà un total del resultat final que ha aconseguit i es mostrarà un cop ha acabat el temps en la sala final.

A la sala de contenidors del final de la partida i per fer el joc més difícil, hi hauria d'haver molts tipus de contenidors, no només els més clàssics com el de vidre o envasos, sinó també alguns menys freqüents com el de control sanitari, el de metall, etc. Tots ells seran iguals, però es diferenciaran pel color.

La imatge és una representació del que seria la sala del final del joc, on apareixerien tots els objectes que hem recollit i que haurem de llençar al contenidor adient.

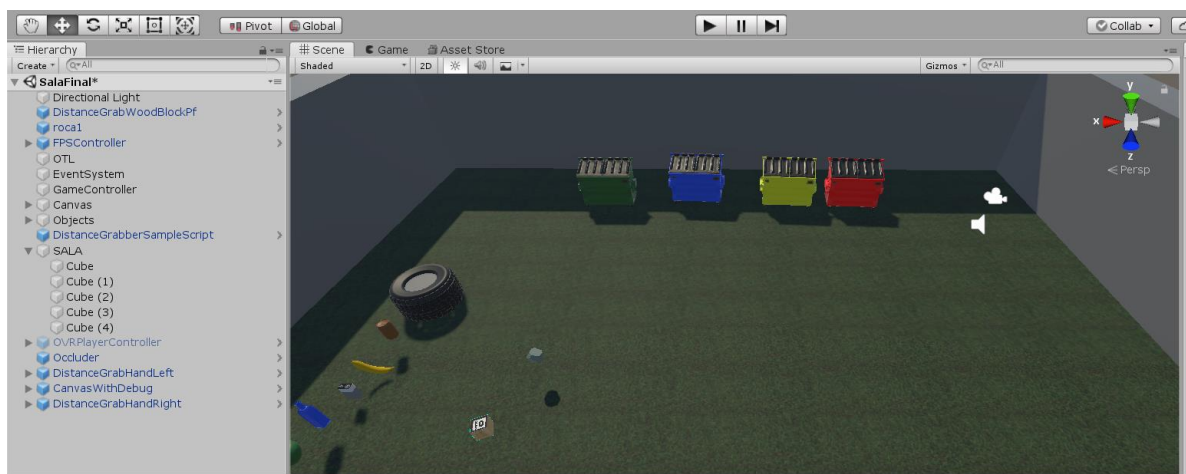


FIGURA 10 Escena final des de l'editor de Unity amb alguns dels objectes per reciclar i els contenidors

En aquesta figura mostro quins contenidors hi ha a la sala final i alguns dels objectes que hi podem llençar a cadascun d'ells. Pot ser que durant el desenvolupament seguïn afegits més objectes o més contenidors.

DISSENY:

En aquest apartat veurem què ha suposat el desenvolupament del joc per la part de disseny, que inclou tant la part estètica, la interfície d'usuari, i els controls per tal de moure's per l'entorn. La plataforma per la qual s'ha fet el desenvolupament principal és per a *PC*, ja que pel desenvolupament i per la implementació de les ulleres de realitat virtual, l'ordinador és la millor opció per poder aconseguir-ho.

El joc està creat amb el motor de videojocs de *Unity* i programat en *C#*. Els he utilitzat, ja que, durant els últims dos anys és el motor i el llenguatge que més he fet servir. Per tant, és amb el que hi tinc més pràctica i amb el que sé que podré fer-ne un millor ús per acabar amb un bon resultat final.

Pel disseny dels elements 3D, durant la versió de proves vaig fer servir objectes simples com cubs o esferes de diferents colors, però un cop que el joc i les seves mecàniques ja han estat ben definides i completament funcionals, els objectes 3D els he creat amb *Blender*, ja que és gratuït, dóna bons resultats i és amb el que hi tinc més practica.

Estètica

L'estètica dels espais que volia crear al principi del desenvolupament seria ambientada en una combinació d'estils d'entre *low-poly* i *cartoon*, ja que el joc el vull dirigir principalment a nens petits i adolescents als quals aquest format els serà més atractiu visualment. Un cop vaig començar a desenvolupar el joc, i a mesura que anava acabant de programar totes les mecàniques, vaig voler seguir el perfil estètic que havia definit en un principi.

Amb *Blender* havia de modelar cada element des de zero, mentre que amb aquest nou *software* em permetia fer uns elements naturals diferents dels que tenia pensat en un principi, per tant vaig decidir canviar i donar un aire més realista al meu joc.

Malgrat això, mentre buscava formes de crear aquests escenaris, vaig trobar un *software* que em permetia crear arbres, arbusts i altres tipus d'elements naturals d'una forma més senzilla.



FIGURA 11 Exemple visual d'estètica low-poly per videojocs

El disseny del mapa ha estat un dels blocs més grans de treball per aquest projecte, ja que crear un entorn des de zero, requereix molts elements 3D i temps per distribuir-los de forma que no es vegi ni massa carregat ni molt buit.

UI/HUD

Ja que la idea principal del joc és que es pugui jugar amb les ulleres de realitat virtual, és important que no utilitzem gaires UIs o HUDs que puguin molestar a la vista del jugador. És per això que durant el joc principal, la visió del jugador intentarà no ser obstruïda per cap menú o interfície massa gran, només veurà un petit compte enrere que li mostrarà el temps que li queda per tal de seguir jugant, i un comptador que mostri el nombre d'objectes que li falten per recollir a un dels costats de la seva vista.

En la versió de PC sí que és necessari implementar un seguit d'elements de *UI* per tal que, qui estigui perdut jugant, se li mostrin per pantalla missatges que puguin ajudar-lo a saber que ha de fer en cada moment.

Un exemple més visual del que he intentat aconseguir per el HUD és com es representa en aquesta imatge del joc *Skyrim*, que va sortir per VR, on es veu la diferència que en la versió per PC i consola (esquerra), té un petit *HUD* on mostra la direcció en què el jugador es mou, la vida i l'energia. En la versió de realitat virtual (dreta) no hi ha res que molesti al jugador.



FIGURA 12 - Exemple de la diferència entre les versions de PC o consola i VR

Controls

En cas que el jugador estigui fent servir unes ulleres de realitat virtual o estigui jugant des d'un PC amb teclat i ratolí els controls per desplaçar-te pel mapa seran molt diferents.

Si s'està jugant des d'un ordinador, el jugador es mourà com en qualsevol altre joc, amb les lletres *W,A,S,D* per caminar i amb el ratolí per girar la vista. Per tal de recollir objectes a la versió d'ordinador, només haurà de fer clic sobre ells amb el ratolí i immediatament seran recollits pel jugador.

Per aconseguir el controlador de moviment per a ordinador, em vaig instal·lar el paquet d'*Assets* inclosos a *Unity* per defecte al meu propi projecte. Aquest paquet conté molts recursos interessants pel desenvolupament de jocs en fases inicials. A més de tots els objectes de mostra i les escenes de prova, vaig agafar el controlador de primera persona que inclou, ja que crear un controlador és relativament senzill, però crear-ne un de tan complet com el que inclou *Unity* per defecte seria un treball de fi de grau en si mateix. Per aquest controlador de moviment, des del mateix editor es pot ajustar tota una sèrie de paràmetres perquè els controls responguin de la forma desitjada.

Un cop el jugador estigui a la sala de reciclatge del final, per moure els residus fins al contenidor adient, fent clic sobre l'objecte l'agafaríem i quedaria "enganxat" a la nostra mà, i per desfer-nos d'ell només hauríem de moure'ns fins on el vulguem dipositar i deixar anar el botó del Mouse.

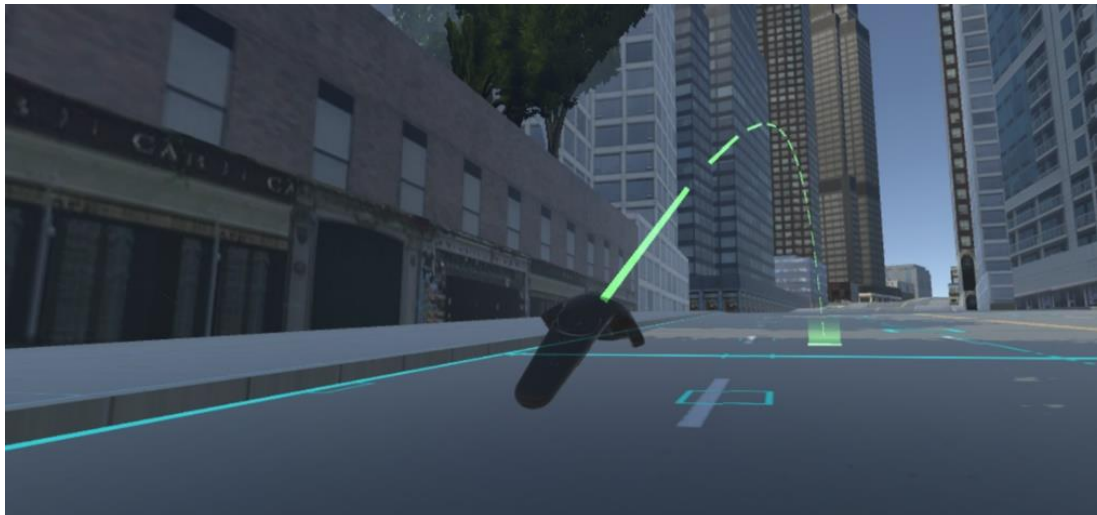


FIGURA 13 - Exemple de desplaçament mitjançant tele-transportacions amb unes ulleres de realitat virtual i controladors VR

En el cas que s'estigui jugant des de les *Oculus*, el moviment es realitzarà de manera que si el jugador vol anar fins a un altre punt, per desplaçar-se haurà d'apuntar amb el controlador i fent servir el *joystick* podrà tele-transportar-se al punt desitjat. No es pot tele-transportar grans distàncies, només uns metres per tal que no vagi d'un extrem a l'altre del mapa en un instant.

Per recollir objectes seria semblant que en la versió de PC, si vol agafar, per exemple, una ampolla que té a prop seu, només l'haurà d'assenyalar amb el controlador i prémer un altre botó per tal de fer l'acció.

Pel controlador de moviment de les ulleres de realitat virtual vaig seguir un procediment similar al d'ordinador i pels mateixos motius. Un cop vaig instal·lar el paquet de desenvolupador d'*Oculus* per *Unity*, vaig afegir el controlador de moviment de la mateixa forma, però en aquest cas sí que calia fer uns ajustos de codi no massa importants. Per recollir objectes en realitat virtual, assenyalem a un objecte que no sigui massa lluny amb un dels controladors de les mans, i prement un botó es transportarà fins a la nostra mà. Per tal de deixar-lo, haurem de deixar anar el botó i aquest caurà a terra. Per tal de moure els objectes a la sala final, on deixem els residus en els contenidors, procediríem de la mateixa manera.

DISSENY DE MAPA I NIVELLS

En les grans empreses de videojocs, on en els seus estudis hi poden treballar centenars de persones, sempre hi ha una gran part de l'equip que es dedica íntegrament en el disseny dels mapes per fer-los tan grans i alhora tan vius com els sigui possible. Fa uns anys aquesta feina era molt més difícil, a causa de les limitacions tècniques requerien que els estudis planifiquessin molt millor el disseny d'aquests mapes per no sobrecarregar de recursos les màquines i no perdre la sensació d'estar tancat en un món petit o sobrecarregat.

Actualment, i gràcies a al fet que cada cop les consoles o, en el meu cas, els ordinadors tenen molta més potencia que fa uns anys, els escenaris solen ser molt més grans i millors gràficament.

La creació d'un mapa des de zero no és feina fàcil, ja que es necessiten molts elements per emplenar un espai i que no ens sembli ni repetitiu ni que està buit. Per crear el meu joc, he creat tres escenaris diferents, dos d'ells són unes sales. En la primera, és on et situes en començar el joc, i és una petita sala on se't donen unes indicacions. En la segona sala és on trobes els contenidors en finalitzar el joc, per abocar tots els residus obtinguts durant aquest. El tercer escenari, és un bosc que representa el mapa principal del joc. En aquest ens trobem tota mena d'elements gràfics a part dels objectes a recollir, plantes, arbusts, arbres, roques, etc.

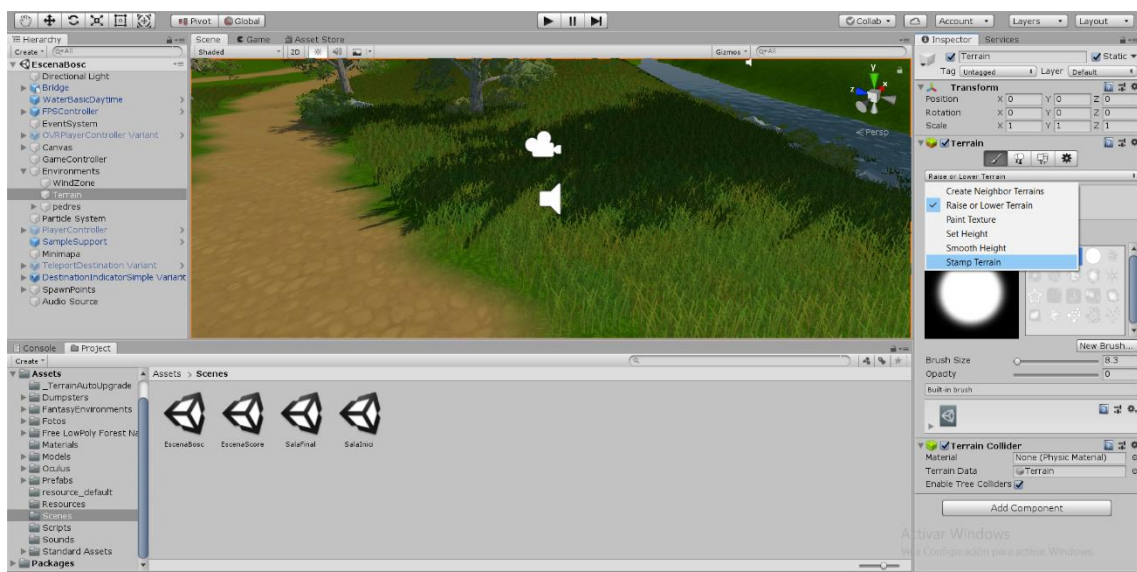


FIGURA 14 Editor del meu projecte de Unity i a la dreta el menú del Terrain

Aquest bosc ha estat creat i dissenyat fent servir diverses eines i tècniques diferents de creació de nivells que explicaré breument a continuació. En el mateix editor de *Unity* s'inclou una eina anomenada *Terrain* que és la que m'ha ajudat a crear el terreny del mapa. En comptes de tenir un terreny pla, aquesta eina et permet crear d'una forma més senzilla el relleu d'un mapa, que per exemple, amb un programa extern de disseny 3D com Blender. Això és possible utilitzant un seguit de pinzells que, ajustant els diferents paràmetres d'aquests, permet ajustar l'elevació del terreny, fer muntanyes, rius i parets.

Aquesta eina també ens permet, amb els pinzells, aplicar textures al terreny d'una forma que s'assimilaria a pintar sobre paper, aquesta tècnica és molt útil si volem donar textures diferents, a punts específics del terreny, com dibuixar camins o herba. I finalment, l'última utilitat d'aquesta eina que he utilitzat és la possibilitat que també mitjançant els pinzells o de forma automàtica, et col·loca elements naturals com arbres, arbusts o gespa per l'escenari.

Per utilitzar aquesta funcionalitat prèviament s'han d'haver importat els models tridimensionals i texturitzats d'aquests elements naturals que volem col·locar. Un cop està tot configurat correctament, des de l'editor de *Unity* podem ajustar certs paràmetres que ens permeten decidir si volem que aquests elements tinguin una mida o una rotació aleatòria en col·locar-los. Pot semblar poc important, però aquestes opcions són molt útils, ja que si tots aquests elements es posessin de la mateixa mida i en la mateixa direcció es veuria un nivell molt repetitiu, sense variació i poc natural.

Gràcies a l'eina *Terrain* es va poder crear una primera versió de l'entorn molt complet, però que, al final, només estava compost d'arbres i gespa. És aquí en el moment que cal, manualment des de l'editor de *Unity*, començar a completar l'escenari manualment. Tan bon punt vaig acabar de fer servir l'eina esmentada, vaig començar a, manualment, emplenar part de l'escenari amb altres elements, com aigua per crear un riu, un pont, pedres, etc.

Els elements creats per emplenar el mapa, han estat construïts mitjançant diferents programes, els més importants serien *Blender*, *SpeedTree Modeling* i el mateix *Unity*. Amb *Blender* vaig crear diversos elements com les roques que trobem pel camí, un sistema d'aigua que va resultar que per incompatibilitats no podia ser utilitzat dins de *Unity* i la primera versió dels arbres que volia utilitzar per col·locar al bosc.

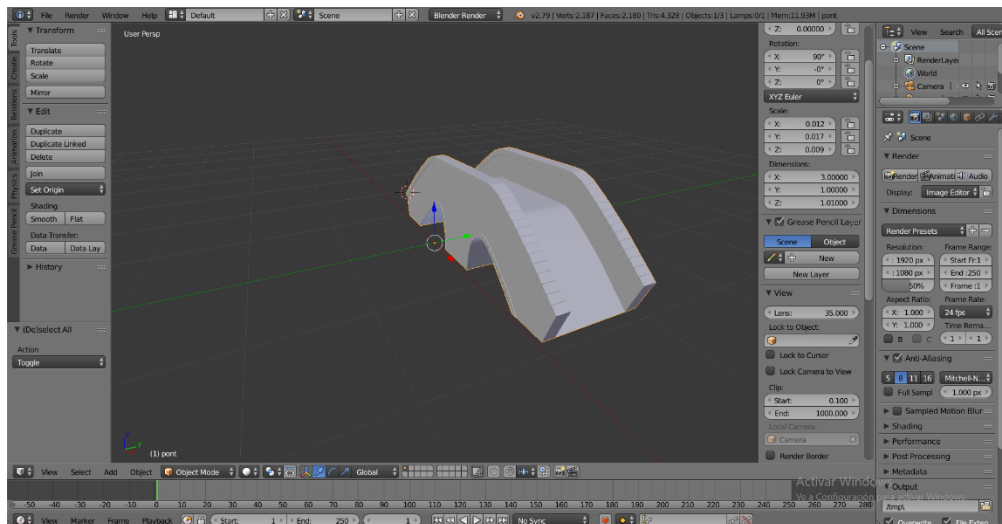


FIGURA 15 Exemple de model 3D sense textura creat a blender

Els arbres són l'element més important d'un bosc i com he dit, a l'inici del desenvolupament del joc els volia crear amb una estètica *low-poly* i amb *Blender*, però un cop vaig començar a crear-los i vaig veure que a l'importar-los a *Unity* no em convenien, vaig començar a buscar altres opcions per crear arbres més realistes. Primer de tot vaig pensar de crear-los directament com ho havia fet amb la primera versió, però era pràcticament impossible, ja que per fer-ho hauria necessitat setmanes per crear dos o tres models d'arbre diferent, així que vaig començar per buscar si existia algun *plugin* de *Blender* que m'ajudés a crear-los.

Vaig trobar un *plugin* que els creava modularment i només podies ajustar alguns paràmetres, però tot i que els resultats eren bons no em va convèncer. Mentre buscava altres opcions i buscava per internet, al manual en línia de *Unity* vaig veure que esmentaven un *software* anomenat *SpeedTree Modeler*, que com també el *plugin* de *Blender*, et crea els arbres de forma modular, però amb moltes més opcions tant de disseny com a l'hora d'editar els arbres.

Tot i ser un programa només de creació d'arbres és bastant complicat d'utilitzar, ja que no és semblant a res que hagués fet servir anteriorment. També ho fa complicat tots els paràmetres que s'han de tenir en compte, com la mida del tronc, el número i la disposició de les fulles, la forma de les branques i fins i tot la resistència d'aquestes al possible vent que hi haurà en el nostre videojoc.

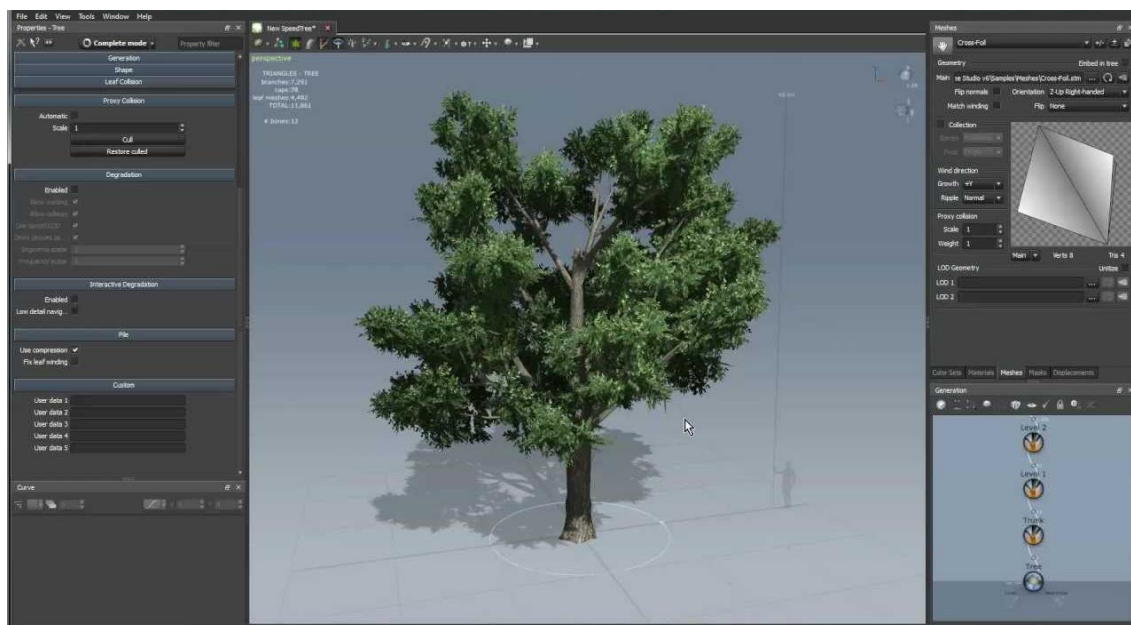


FIGURA 16 Exemple d'arbre e interfície del SpeedTree Modeling 7.1.5

La millor part d'aquest *software* és, que de forma nativa, porta la compatibilitat amb *Unity*, així que un cop havia creat els arbres com jo volia només havia d'exportar-los en un format propi d'aquest programa que el mateix *Unity* ja podia reconèixer. Aquest *software* ha estat usat per alguns dels estudis més importants de videojocs com *EpicGames*, *Activision*, *Ubisoft*, *CD Project...*

L'últim punt per crear el mapa i les sales que vaig tenir en compte va ser el model d'il·luminació i l'ús de sons per fer l'escenari més viu. En l'escena del bosc he afegit un seguit d'efectes sonors que s'activen en el moment que es recull correctament un objecte. Si t'acostes al riu, es van escoltant sons d'aigua, o si el jugador es comença a quedar sense temps per recollir deixalles, se l'avisava amb un altre so.

Per la part d'il·luminació de les diferents sales i el bosc, he utilitzat diferents models d'il·luminació. Com volia que la sala principal fos una habitació tancada, vaig utilitzar uns panells amb un material emissor de llum, aquests panells junts en desactivar l'opció d'il·luminació en temps real, ens permet que la il·luminació

de la sala inicial sigui molt millor. Per la il·luminació de l'escena de bosc he editat alguns paràmetres per tal de millorar la qualitat de l'entorn, com el color de la llum per donar-li un color més viu i la posició i la intensitat dels emissors per donar unes ombres més realistes.

PROGRAMACIÓ

Les tasques de programació del joc han estat una de les parts del desenvolupament que m'ha portat amb diferència més temps. Aquest treball, en alguns punts del desenvolupament de la programació ha suposat un repte personal. Un dels objectius que tenia era millorar la meua capacitat com a programador i amb aquest projecte considero que he millorat molt i he après molts recursos importants que em podran ser útils en un futur.

El joc en tasques de programació es pot dividir en quatre blocs. El d'agafar i guardar els objectes que el jugador va trobant durant l'escena del bosc; haver de crear els objectes que s'han de recollir i els que ja s'han recollit en el nivell anterior; la part de configurar la interfície d'usuari; i la part d'implementar les ulleres de realitat virtual. També hi ha molta feina de programació de blocs més petits de codi com el gestor d'escenes o el codi que et calcula la puntuació que has aconseguit al final de la partida, entre d'altres.

M'agradaria explicar d'una manera menys tècnica que anar explicant el codi línia per línia, així que de cada un dels quatre blocs que abans he mencionat n'explicaré les funcionalitats principals i alguns recursos utilitzats.

Agafar i guardar objectes

Aquest és el bloc principal del joc i el que ha tingut més importància durant el desenvolupament, ja que com he explicat abans, la mecànica principal un cop ha començat el joc és haver d'anar caminant per l'escena triada i recollir totes les deixalles que veiem.

Per aconseguir-ho he creat un *script* que dins el joc he anomenat *saveObject*. En aquest bàsicament el que faig és que, un cop s'està dins el joc, si la persona que porta els controls està a prop d'un dels objectes que ha de recollir, i mentre l'està mirant per tal de seleccionar-lo, fa clic al Mouse el podrà recollir i se li desarà dins l'inventari d'objectes recollits.

Per programar aquesta simple acció, he hagut de, prèviament, donar a cada model 3D una clau ID única. Per exemple, en el cas de les llaunes, la seva clau serà el número 1. Un cop iniciat el joc es disposaran de forma completament aleatòria en certs llocs amagats del mapa.

Un cop tots els objectes han estat instanciats i assignats amb una ID ja es pot passar a recollir els objectes. Per fer-ho, he fet servir una funcionalitat que incorpora *Unity* anomenada *Raycast*, que simplement el seu funcionament és llançar un raig invisible des del punt central de la vista del jugador en línia recta. Si el raig llançat col·lisiona amb el component *BoxCollider* d'algun dels objectes, per codi es comprova si aquest objecte conté alguna de les ID que prèviament s'han assignat a les deixalles.

Aquest raig es pot ajustar per tenir una certa trajectòria i distancia, pel meu joc no ens interessa que estigui constantment llançant aquest raig, així que només es llençarà el raig si fem clic al botó del ratolí.



FIGURA 17 Captura del joc intentant recollir un objecte seleccionat.

En el cas que el raig col·lisioni i es trobi una de les claus prèviament assignades, se sumarà un punt a la variable de quantitat d'aquell objecte i aquest valor quedarà guardat a un sistema propi de *Unity* per emmagatzemar dades denominat *PlayerPrefs*.

PlayerPrefs és un mètode que ens permet guardar i accedir a dades entre diferents sessions de joc, gràcies a aquesta funció la quantitat dels objectes que hem pogut recollir durant el temps de joc d'un determinat residu se'ns quedarà

guardat de forma que més endavant hi podrem accedir i treballar amb aquestes dades.

Un cop hem recollit la informació necessària de l'objecte i s'ha guardat al *PlayerPrefs*, mitjançant codi es desactiven els components *BoxCollider* i el *MeshRenderer* per tal de fer desaparèixer l'objecte de l'escena. Aquests dos components són, en el cas del *BoxCollider*, el que li dóna als objectes a recollir la possibilitat de xocar amb altres objectes tridimensionals i l'altre component dóna la possibilitat de renderitzar el cos de l'objecte.

Crear els objectes

Aquest bloc el dividim en dos punts, el primer inclou l'apartat de crear els objectes en punts aleatoris del mapa del bosc on es troba el jugador, i la part de crear els objectes a la sala de reciclatge on apareixeran les deixalles que s'han recollit en la primera escena.

El primer punt funciona de la següent manera. Manualment he creat uns *prefabs* anomenats *spawnPoints* que porten un *script* de codi en el qual s'inclouen tots els models 3D que es poden instanciar. Un cop s'ha iniciat el joc, en aquests *prefabs*, el codi que contenen fa que només puguin fer aparèixer un dels models d'objecte que se'ls ha assignat des de l'editor. Com que cada *spawnpoint* només crearà com a màxim un objecte com a màxim, s'han de posar varis d'aquests *prefabs* distribuïts pel mapa per assegurar-nos que hi ha molts objectes a recollir.

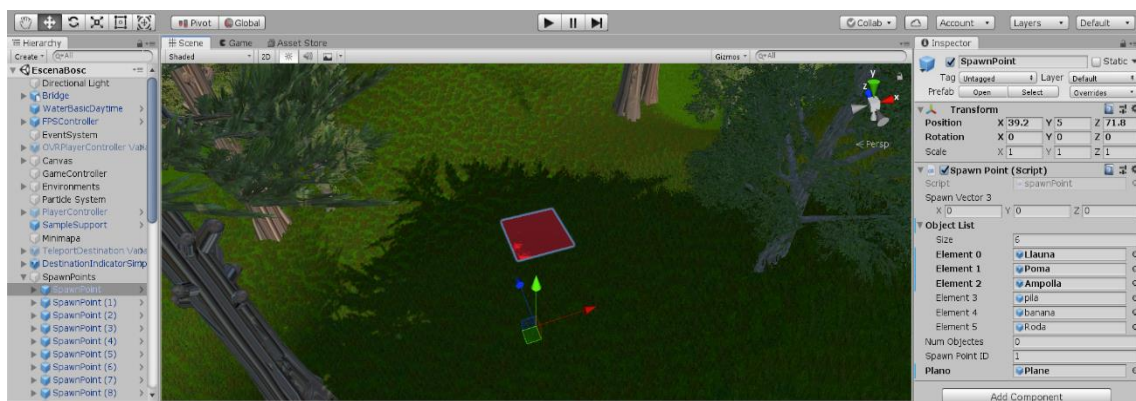


FIGURA 18 Captura de l'editor de Unity amb un *SpawnPoint*

Per crear els objectes que hem guardat a l'inventari un cop ja som a la sala de reciclatge, funciona d'una forma similar a com els guardem dins l'inventari. Mitjançant el *PlayerPrefs* accedim a la informació que havíem guardat en el punt anterior i el codi busca per cada valor de les ID si hi ha més d'un objecte amb aquella clau i en cas que així sigui, s'activarà un model 3D a l'escena dels contenidors equivalent al model que hem recollit en l'escena anterior per tal que pugui ser recollit i dipositat al contenidor adient pel jugador.

Aquests objectes de l'escena final són una mica diferents dels recollits en l'escena anterior, el model tridimensional i el seu material o textura són el mateix però inclouen un seguit de scripts de codi diferents, per tal que puguin ser agafats i transportats fins als contenidors d'una forma diferent que en l'escenari anterior, tant amb les ulleres de realitat virtual com amb l'ordinador.

Interfície d'usuari

Com la majoria dels jocs, és necessari llençar certa informació per pantalla per tal d'ajudar al jugador, controlar que està jugant correctament o que està fent el que els desenvolupadors volen que es faci en el seu joc.

Pel meu projecte he implementat un sistema d'interfície d'usuari (UI) bastant simple per tal de no saturar d'informació als usuaris. Durant el joc principal a l'escena del bosc només es mostren tres tipus de UI. A la cantonada superior esquerra se'ns mostra el temps restant per trobar deixalles abans que se'ns transporti a l'escena dels contenidors i el nombre total d'objectes que s'han recollit. En la part central inferior de la pantalla se'ns mostra un senzill inventari que s'actualitza cada cop que recollim un objecte i ens mostra la quantitat que disposem en cada moment d'aquells objectes. També hi ha un panell que apareix en certs moments importants que ens ensenya missatges en text, per exemple en començar la partida ens dóna alguna instrucció o si, recollim un objecte, se'ns mostra un missatge del tipus: "Molt bé has recollit X objecte!".

A la cantonada superior dreta hi ha un petit mini mapa que ens ensenya on estem actualment, com és el terreny al nostre voltant més immediat des d'una vista

aèria i la posició dels objectes, per ajudar al jugador a trobar-los d'una forma més senzilla, ja que el mapa és molt gran i pot arribar a ser complicat trobar-los.



FIGURA 19 UI del joc que inclou el minimapa, inventari i marcador de temps/objectes

Tots aquests elements esmentats formen part dins de l'apartat de *canvas* de *Unity*, que és la zona on es col·loquen tots els elements destinat a la UI pels jocs. Per crear el comptador de temps, el comptador total d'objectes i el panell de missatges han estat configurats i programats d'una forma bastant similar. Tots ells són uns components *Text* del *canvas* que, un cop referenciats a l'editor, mitjançant el codi els podem editar automàticament.

Per exemple, el temporitzador és un valor fix que en funció del temps va disminuint, i un cop aquest valor ha arribat a 0 es canvia d'escena. El comptador d'objectes, cada cop que amb el *RayCast* es recull un objecte se li suma un a aquest valor. El panell d'informació funciona de forma similar, però és una llista de missatges que en certs moments es fa aparèixer el missatge a la posició indicada de la llista.

L'inventari de la part central de la pantalla és una mica diferent, funciona igual que el comptador total d'objectes, però si es recull un tipus d'objecte concret, i en el moment que s'ha trobat quina clau conté, per codi es suma una unitat al valor del *Text* que correspon a l'objecte amb aquella ID. També està configurat per si en un futur es volen afegir més tipus diferents d'objectes a l'inventari, no caldrà haver de tornar a crear-lo de zero.

La part més costosa de la interfície d'usuari ha estat, crear el mini mapa. Per simplificar-ho, el funcionament d'aquest és una *RawImage* a la que a cada *frame* del joc se li aplica una nova textura, aquesta textura vindria des d'una càmera aèria secundària, que mitjançant codi segueix al jugador mentre es mou. Aquesta càmera està en projecció ortogràfica, fent que la vista no tingui perspectiva i doni una sensació de mapa més real.

Per senyalar els punts on es troben els objectes en el mini mapa, i com he esmentat abans, he creat uns *prefabs* que són els encarregats de crear els objectes a recollir un cop s'ha iniciat el joc. Aquests porten associats un pla 2D de color vermell que només es veu des d'una de les dues cares, aquests plans estan bastant més alts que el jugador. Per tant, el jugador no serà capaç de veure'ls però la càmera aèria, com podrà veure la cara visible d'aquests punts, apareixeran només en el mini mapa.

Implementació VR

No va ser fins que el desenvolupament del joc per a la plataforma d'ordinador va estar bastant avançada que vaig voler començar a implementar la possibilitat de poder utilitzar les ulleres de realitat virtual en el meu projecte. Prèviament al desenvolupament del joc, no havia utilitzat mai aquesta tecnologia per una altra cosa que no fos el mateix entreteniment o de passada en alguna assignatura de la meva estada Erasmus.

Per començar amb el desenvolupament primer de tot cal activar el *SDK* de, en el meu cas, *Oculus Rift* dins l'editor del projecte de *Unity*, que és el model del qual disposa la universitat i amb el que he pogut treballar tots aquests mesos. Un cop activat i vaig instal·lar el *software* de *Oculus* a l'ordinador on he treballat, ja vaig poder passar a importar tots els suplementes d'integració d'*Oculus* per a *Unity* des de *l'Asset Store*. Aquest paquet d'*Oculus* és un recull d'escenes *prefabs* i altres utilitats d'exemple que poden servir d'ajuda a l'hora de desenvolupar qualsevol classe de joc o aplicació en realitat virtual.

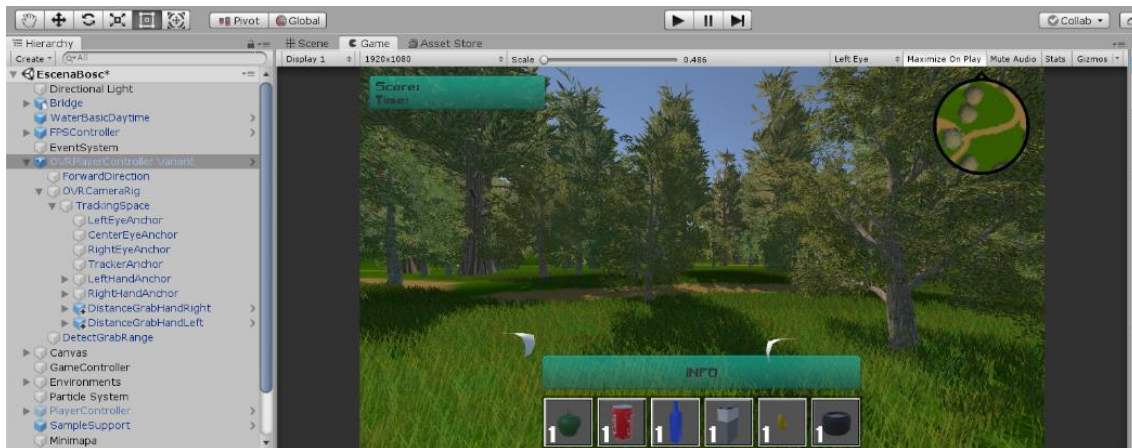


FIGURA 20 Implementació dels controladors de moviment del Oculus dins l'editor de Unity

Per fer funcionar el meu joc d'ordinador des de les ulleres de realitat virtual, algunes mecàniques com la de desplaçar-se per l'entorn i la de recollir i transportar objectes havia de ser modificada perquè pogués ser usada amb tot l'equip de realitat virtual.

D'aquest paquet només en vaig usar una petita part de tot el seu contingut, però que, durant els mesos de feina, tots els exemples que portava m'han servit de guia per aprendre alguns trucs, funcionalitats o de com treballar i desenvolupar correctament l'apartat de les ulleres.

D'aquests punts, el primer que vaig implementar al meu projecte va ser la mecànica de moure's mitjançant tele-transportacions, ja que el mètode tradicional de desplaçament és mitjançant els *joysticks* o caminar de forma real. Aquestes no són unes formes de desplaçar-te compatibles amb el meu projecte, ja que usar els *joysticks* per moure't sol, en moltes persones, causar un mareig per poca estona que s'estigui fent servir aquest sistema, i tampoc podia fer servir la forma tradicional de caminar de forma real, perquè les *Oculus* van connectades mitjançant un cable no massa llarg.

Però, per resoldre aquests inconvenients, vaig fer servir un dels exemples al paquet d'*Oculus* on, en una escena amb obstacles, els podies superar sense haver de moure't del lloc ni creant una sensació de mareig, gràcies a aquesta mecànica de transportar-te. Implementar aquest exemple al meu joc no va ser difícil, però sí que vaig necessitar bastant de temps per aprendre exactament com funcionava i quins eren tots els components necessaris que necessitava importar i modificar per fer-ho funcionar al meu projecte.

L'altra mecànica que necessitava una revisió per poder ser utilitzada mitjançant la realitat virtual, era la de ser capaç de recollir els objectes i després dipositar-los als contenidors pertinents. Com en l'apartat d'ordinador, el jugador és capaç de mitjançant un clic dels botons dels controladors d'*Oculus*, atraure l'objecte que s'està apuntant amb la mà per tal d'agafar-lo i poder dipositar-lo dins el contenidor. Aquesta mecànica, com en l'apartat anterior vaig trobar que dins el paquet d'exemples d'*Oculus* n'hi havia un que tenia una funció similar. Per afegir aquesta funcionalitat al meu joc, primer s'havien d'editar els objectes de la sala dels contenidors i afegir-li un *script* de codi que ens permet atraure els objectes des de la distància a la nostra mà dins l'entorn virtual. També vaig haver d'afegir i configurar tots els components necessaris per poder, mitjançant el *RayCast* de les ulleres, seleccionar els objectes a recollir.

PUNTS FORTS I FEBLES

En aquest breu apartat parlo dels punts forts i dels punts febles que personalment veig que compleix el meu projecte final:

PUNTS FORTS	PUNTS FEBLES
Hi ha versió de PC i realitat virtual i seria molt senzill portar-ho a <i>smartphones</i> i consoles	
L'escenari del bosc és molt immersiu en realitat virtual	Actualment només hi ha un escenari disponible
La <i>UI</i> és molt comprensible i fàcil d'interpretar	En la realitat virtual, la falta d'una <i>UI</i> tan elaborada com en PC, pot crear confusió
És un producte que es pot portar a les aules o altres centres educatius per ensenyar de la importància del reciclatge	Es necessita un aparell de realitat virtual per tal de fer-lo servir correctament i moltes escoles no disposen d'aquest equip
És un joc àgil i per jugar-lo no és necessitem més de 15 minuts	
El joc és relativament lleuger i pot ser jugat des de qualsevol ordinador recent	
Molt senzill de jugar i no es necessita experiència prèvia en realitat virtual per entendre els controls	Al ser tan senzill pot arribar a perdre la gràcia si es vol jugar més d'una vegada.

CONCLUSIONS I FUTURES MILLORES

Com s'ha pogut veure en aquesta memòria, on s'explica el funcionament i el que ha estat tot el desenvolupament que he necessitat per tal de crear aquest projecte de final de grau, un *serious game* en realitat virtual per ajudar a aprendre i conscienciar sobre la importància del reciclatge entre els més petits.

En la major part he complert amb la totalitat dels objectius que m'havia proposat a l'hora de començar aquest projecte. En algun moment donat del desenvolupament vaig arribar a pensar que no em donaria temps d'acabar-lo o que si ho aconseguia el resultat final seria mediocre i no seria presentable, però a l'haver complet la gran majoria dels objectius, el resultat del treball del projecte és més que satisfactori. El joc no és perfecte, com en l'actualitat cap ho és, però els petits errors que segueixen dins el joc o algunes funcionalitats que podrien ser millorades són molt petites en comparació amb la part de fites assolides.

Una de les preocupacions que sempre havia tingut al llarg de tot el grau era que no volia fer un treball de fi de gran en el qual, en portar dues setmanes treballant, perdés la motivació de seguir-hi treballant. En acabar aquest projecte puc dir que no he perdut la motivació ni les ganes de seguir-hi treballant en un futur, i és que, durant aquests mesos de dura feina, el més important és que m'ho he passat bé treballant-hi.

Un dels motius pels quals vaig voler estudiar multimèdia va ser per l'àmbit de creació i desenvolupament de videojocs, però a part d'alguna assignatura mai havia treballat en un projecte tan complex com aquest. Finalment, gràcies a aquest treball de final de grau m'he adonat que, si em surt l'oportunitat, és al que m'agradaria dedicar-me en un futur no molt llunyà.

Durant el desenvolupament d'aquest joc m'he adonat de la importància d'haver triat el grau de Multimèdia, ja que la majoria de les disciplines presents en la creació dels videojocs, des del disseny gràfic de la interfície d'usuari fins a la programació del codi pel seu funcionament, han estat aspectes que hem estudiat, practicat i après durant la carrera. Tots aquests coneixements que he

adquirit per separat m'han sigut molt útils en el moment de treballar en aquest projecte.

Per altra banda, també he après per què les grans empreses de videojocs tenen estudis de desenvolupament de centenars, i fins i tot, milers de persones treballant en els seus jocs, ja que crear un joc des de zero no és gens fàcil. En acabar aquest projecte he pogut observar que es necessita la implicació de moltes hores i que hi ha moltíssima més feina de la que m'havia imaginat al principi del desenvolupament.

Com a conclusió final, he pogut veure que el nombre de punts positius en comparació al nombre de punts negatius que ha suposat aquest treball és, amb diferència, molt més gran. Finalment, puc assegurar que he gaudit molt aquest projecte i que, espero poder seguir treballant en el món dels videojocs en un futur.

Com ja he donat a entendre abans, el desenvolupament d'aquest joc no finalitzarà un cop l'hagi entregat, sinó que m'agradaria poder seguir treballant-hi de forma personal i cada cop tenir un producte més complet. És per això que m'agradaria fer un seguit de futures implementacions o millores.

Primer de tot, m'agradaria millorar la implementació de les ulleres de realitat virtual, que tot i funcionar correctament, si no se sap prèviament com funcionen aquests aparells o és el primer cop que es fan servir, el joc pot tenir uns primers moments de dificultat per culpa de la falta de pràctica en aquest tipus d'entorn. M'agradaria poder polir alguns errors que, com en tots els jocs, succeeixen de manera esporàdica.

També m'agradaria, ja que la base està acabada, crear els dos mapes extra que volia crear al principi del desenvolupament, un mapa de platja i una petita ciutat amb grans edificis. Aquest fet seria bastant senzill, ja que només necessitaria més hores de feia per muntar aquests escenaris, però no s'hauria de programar res, ja que podria importar totes les mecàniques necessàries del mapa del bosc. Tot i això, seria necessari seguir treballant en el mapa principal, ja que, tot i ser un escenari bastant complet, amb molts elements diferents, sempre hi ha marge de millora per poder aconseguir un mapa molt més diferent.

En menció a la carrera, voldria implementar molts més elements multimèdia per tal de tenir un joc més finit, com per exemple afegir vídeos creats per mi sobre la importància del reciclatge i l'impacte positiu d'aquest al medi ambient.

Els objectes a recollir actualment són, simplement, models tridimensionals amb textures o materials senzills, és per això que una de les futures millores seria tornar a crear aquests objectes des de zero i fer-los més realistes. A part dels ja existents, s'hauria de crear molts més tipus de residus per fer el joc més complicat a l'hora de reciclar i que, per tant, sigui més educatiu.

Com a última futura millora que m'agradaria implementar en el meu joc seria la inserció d'animals animats en l'escena del bosc. Aquests no haurien de ser grans ni molt complexos, però els voldria afegir per donar al mapa una sensació d'acompanyament i vida. També complirien la funció d'ensenyar que també, gràcies al grau cursat, he après animació i de forma autodidàctica sabria programar la intel·ligència artificial d'aquests éssers, ja que actualment, per falta de temps no he pogut introduir-los, així com les millores prèviament esmentades.

AGRAÏMENTS

Després d'aquests mesos de treball, m'agradaria reconèixer i agrair a totes aquelles persones que en algun moment m'han ajudat a seguir endavant amb aquest projecte.

Primer de tot, m'agradaria agrair als meus dos tutors de treball de fi de grau per haver estat sempre molt atents amb mi, en ajudar-me i guiar-me per tal de poder realitzar aquest treball i poder-lo completar satisfactòriament.

En segon lloc, m'agradaria agrair als meus companys de classe per totes les hores que hem passat al taller de multimèdia treballant, intercanviant consells i opinions, i sobretot, ajudant-nos entre nosaltres com sempre hem fet.

Finalment, m'agradaria mencionar i agrair tot el suport rebut per part dels meus pares, el meu germà i la meva parella, els qui, igual que tothom mencionat anteriorment, m'han ajudat a trobar en alguns moments la motivació per poder seguir esforçant-m'hi.

WEBGRAFIA

Advanced Game Sounds (2018) Recuperat de 01/06/2019

<https://assetstore.unity.com/packages/audio/sound-fx/advanced-game-sounds-37339>

Brackeys (24, Gener 2016) LIGHTING in Unity (2018) Recuperat de

01/06/2019 <https://www.youtube.com/watch?v=VnG2gOKV9dw>

Highlight Glow System (2018) Recuperat de 01/06/2019

<https://assetstore.unity.com/packages/tools/particles-effects/highlight-glow-system-20166>

Introduction To Illustrator (2019) Recuperat de 01/06/2019

<https://helpx.adobe.com/illustrator/user-guide.html>

Lam, Jonathan(18, Maig 2017) Designing game UI Asset in Adobe Illustrator

Recuperat de 01/06/2019 <https://design.tutsplus.com/courses/designing-game-ui-assets-in-adobe-illustrator/lessons/blocking-out-the-shapes>

Lurony (4, desembre 2016) Unity 5 – Grab and Throw Objects Recuperat de

01/06/2019 <https://www.youtube.com/watch?v=Xv-c3-IONM0&t=>

Manual Blender 2.8 (2019) Recuperat de 01/06/2019

<https://docs.blender.org/manual/es/dev/>

Oculus developers (2019) Recuperat de 01/06/2019

<https://developer.oculus.com/develop/>

Oculus Integration(2019) Recuperat de 01/06/2019

<https://assetstore.unity.com/packages/tools/integration/oculus-integration-82022>

SpeedTree for Unity Manual (2019) Recuperat de 01/06/2019

https://docs.speedtree.com/doku.php?id=unity_introduction

The Elder Scrolls V: Skyrim VR (2019) Recuperat de 01/06/2019

https://store.steampowered.com/app/611670/The_Elder_Scrolls_V_Skyrim_VR/

Tutorial: Build Your First VR App (n.f.) Recuperat de 01/06/2019

<https://developer.oculus.com/documentation/unity/latest/concepts/unity-tutorial/>

Unity Dashboard Cloud (2019) Recuperat de 01/06/2019

<https://developer.cloud.unity3d.com/projects/>

Unity manual UI(2019) Recuperat de 01/06/2019

<https://docs.unity3d.com/Manual/UIToolkits.html>

Unity VR (Oculus) manual (2019) Recuperat de 01/06/2019

<https://docs.unity3d.com/es/current/Manual/VRDevices-Oculus.html>

WildLife VR (2016) Recuperat de 01/06/2019

https://store.steampowered.com/app/500140/Wildlife_VR/