



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Centre de la Imatge i la Tecnologia Multimèdia

Disseny i Prototipatge d'un Videojoc per a Realitat Virtual

Treball Final de Grau

Grau en Disseny i Desenvolupament de Videojocs

Cognoms: Borrell Vives Nom: Aleix

Pla: 2014

Director: Duque Ros, Josep Maria

RESUM

El següent document tracta sobre el disseny i desenvolupament d'un prototip de videojoc en realitat virtual, en el qual s'intenta resoldre l'actual problema per als jugadors *hardcore* en què, per a aprofitar la jugabilitat que ofereix el dispositiu, els hi requereix un gran esforç físic.

Per aconseguir-ho, la jugabilitat del prototip es basarà en conducció, on tot i estar el jugador assegut, pot utilitzar les mecàniques disponibles en la realitat virtual per mirar l'entorn.

Paraules clau

Realitat virtual, prototip, conducció, videojoc, disseny, programació.

Índex

Glossari	6
1. Introducció	7
1.1 Motivació	7
1.2 Formulació del problema	7
1.3 Objectius generals del TFG	8
1.4 Objectius específics del TFG	8
1.5 Abast del projecte	8
2. Estat de l'art	10
2.1 Tecnologia i dispositius	10
2.2 Productes	12
2.3 Diferenciació	13
3. Planificació	14
3.1 Anàlisi DAFO, riscos i pla de contingències	14
3.2 Anàlisi inicial dels costos	15
4. Metodologia	18
4.1 Eines per al seguiment del projecte	18
4.2 Eines de validació	18
5. Desenvolupament del projecte	20
5.1 Preproducció	20
5.1.1 Eines per al desenvolupament en Realitat Virtual	20
5.1.1.1 Motor per a Videojocs	20
5.1.1.2 Dispositiu de Realitat Virtual	21
5.1.2 Recursos inicials	21
5.1.2.1 Ciutat	21
5.1.2.2 Vehicle base	23
5.2 Producció	23
5.2.1 Primera "milestone". Producte Viable Mínim	23
5.2.2 Segona "milestone"	25
5.2.2.1 GPS	25
5.2.2.2 Mareig i malestar	27
5.2.2.2.1 Sistema Vestibular	27
5.2.2.2.2 Optimització	29
5.2.2.3 Testeig, millores i resultat	31
5.2.3 Última "milestone"	33
5.2.3.1 Intel·ligència artificial dels vehicles	33
5.2.3.2 Cinemàtica final	38
5.3 Postproducció	41

6. Conclusions i treballs futurs	42
7. Bibliografia	43
8. Descàrrega del prototip	44
9. Annexos	45
9.1 GDD: document de disseny	45

Índex de taules i figures

Taula 2.1: Dispositius de realitat virtual	11
Taula 3.1: Anàlisi DAFO	14
Taula 3.1: Salari mitjà a Espanya.	15
Taula 3.2: Cost de l'equipament.	16
Taula 3.3: Col·laboradors i paquets de treball.	16
Taula 3.4: Pressupost.	17
Imatge 5.1: Captura del recurs sense cap modificació.	22
Imatge 5.2: Captura del recurs després d'editar els shaders per actualitzar-los.	22
Imatge 5.3: Muntatge de l'escena per a la primera "milestone".	24
Imatge 5.4: Muntatge del vehicle per a la primera "milestone".	24
Imatge 5.5: Muntatge de la càmera que generarà la textura per al GPS.	26
Imatge 5.6: Resultat obtingut en implementar el GPS amb el camí.	27
Imatge 5.7: Sistema Vestibular	28
Imatge 5.8: Il·luminació anterior, dura i en temps real.	31
Imatge 5.9: mapa d'il·luminació, més suau i realista.	31
Imatge 5.10: Implementació de les noves malles.	32
Imatge 5.11: Resultat obtingut en la segona "milestone".	33
Imatge 5.12: Configuració dels costos i resultat de la malla de navegació.	34
Imatge 5.13: Exemple d'una intersecció amb les dades especificades.	35
Imatge 5.14: Exemple d'una intel·ligència artificial amb les dades especificades.	36
Imatge 5.15: Exemple d'un muntatge de l'script que detecta les col·lisions amb les dades especificades.	37
Imatge 5.16: Col·locació dels disparadors i col·lisionadors per a la cinemàtica.	39
Imatge 5.17: Col·locació del vehicle amb el sistema de partícules.	39
Imatge 5.18: Resultat de la cinemàtica on veiem els vehicles creuant la intersecció i el recorregut del GPS fins al destí.	40

Glossari

Jugador *hardcore*: jugador més apassionat als videojocs, sol dedicar diverses hores en les seves sessions a jocs usualment més llargs i elaborats.

FPS: *frames per second*, imatges mostrades en pantalla per segon.

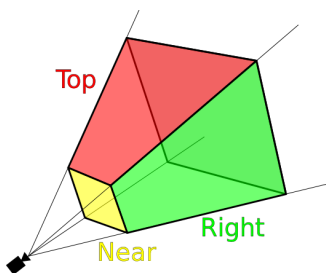
Videojoc *casual*: orientat per a jugar-lo en moments lliures o d'espera de la vida quotidiana, en sessions curtes de pocs minuts, com per exemple en un autobús o tren.

Script en programació: *Llista de comandes que són executades per cert programa o motor de scripting. Solen ser utilitzats per automatitzar processos.*¹ En els motors de videojocs solen ser la manera que té l'usuari per a programar els comportaments.

Shader: *Codi independent, escrits en llenguatges de programació específics, que serveixen per a funcions relacionades amb l'apartat gràfic.*²

Renderitzat: *procés en el qual es genera un producte final digital, una imatge, a partir de, per exemple, models en dues o tres dimensions. El procés es fa en les gràfiques en temps real.*³

Frustum: Figura geomètrica que serveix per a representar la visibilitat d'una càmera.



Culling: Relacionat amb el *renderitzat*, és una tècnica emprada per a evitar *renderitzar* elements que no estaran a la vista de l'usuari, estalviant-se així bastants recursos.

¹ Extret i traduït de la pàgina anglesa *TechTerms*.

² Extret de la pàgina sobre *Shader* de la Viquipèdia en anglès.

³ Extret i traduït de la pàgina anglesa *TechTerms*.

1. Introducció

1.1 Motivació

Des de ben petit he estat interessat i apassionat pels videojocs, gràcies a la meva mare que ens va iniciar al meu germà i a mi, i de petits ens alternàvem el temps lliure al seu ordinador. Llavors, vam tenir la Game Cube i la Wii per petició nostra, aquesta última consola de sortida.

La Wii, la qual venia amb el videojoc *Wii Sports*, requeria un esforç físic constant, nosaltres, al ser petits, l'aguantàvem fins a cansar-nos de jugar. Tanmateix, amb la sortida d'alguna entrega de la saga *The Legend of Zelda*, que vam descobrir amb la Game Cube, on podies utilitzar l'espasa amb el moviment del comandament de la Wii, però, a la llarga era esgotador, ja que en aquest tipus de jocs el nostre interès era l'aventura i història que plantejava. Així que finalment, acabàvem jugant aquest tipus de joc sense utilitzar aquesta mecànica del comandament de la Wii i en el seu lloc ho fèiem com amb un comandament corrent.

En veure els jocs de realitat virtual i provar-ne algun, ràpidament em va recordar aquest problema que vam detectar de petits per a aprofitar la jugabilitat que ofería el dispositiu en jocs més enfocats per a un públic *hardcore*, Per aquesta raó, al ser també un producte innovador, vull col·laborar amb aquest prototip a resoldre aquest problema tant amb un disseny com dut a la pràctica.

1.2 Formulació del problema

El problema és el requeriment físic en els videojocs de realitat virtual dissenyats per a un públic *hardcore*, que va lligat a l'ús del dispositiu. De manera que es sol reduir l'esforç físic disminuint l'ús i aprofitament de les mecàniques que ofereix la realitat virtual.

El problema ja està mínimament resolt de la forma més simple, prescindint de la major part de jugabilitat que ens proporciona el dispositiu i solament utilitzar-lo com una pantalla més i una forma diferent de girar el cap del personatge. Per exemple, adaptar un videojoc existent únicament incloent les ulleres de realitat virtual, com en el cas de *Subnàutica*⁴. L'altra variant

⁴ [En línia] Pàgina web, URL <<http://store.steampowered.com/app/264710/Subnautica>> [Consulta el 25 de febrer de 2018].

és, en aquesta adaptació, degut a les limitacions del dispositiu, fer el moviment amb teletransports en lloc de dinàmicament per evitar mareig a causa de la baixada de FPS, com per exemple *DOOM VFR*⁵, *L.A. Noire: The VR Case Files*⁶ o *Fallout 4 VR*⁷.

Tot i que també hi ha algun videojoc de realitat virtual de conducció que analitzarem en l'estat de l'art, en aquest prototip s'intentarà un nou disseny on s'aprofiti més la jugabilitat del dispositiu de realitat virtual, amb una conducció propera a arcade però donant una raó perquè el jugador miri al voltant, sense exigir-li un esforç físic.

1.3 Objectius generals del TFG

L'objectiu general d'aquest treball de fi de grau és la creació d'un prototip per a realitat virtual amb un disseny i jugabilitat orientat als jugadors *hardcore*, on puguis jugar assegut sessions de mitja o llarga durada sense que impliqui un esforç físic per a l'usuari i, alhora, s'aprofiti la jugabilitat que ens brinda el dispositiu de realitat virtual.

1.4 Objectius específics del TFG

- Dissenyar un prototip de videojoc per a realitat virtual on es tingui en compte les necessitats dels jugadors amb un perfil més *hardcore*.
- Investigar i aprendre les limitacions i necessitats del desenvolupament per a realitat virtual, així com solucionar-les.

1.5 Abast del projecte

L'abast del projecte aglutina la realització pràctica en un prototip dels objectius específics anomenats. Elaborant així un document de disseny, el prototip i aquest document on quedarà plasmat els resultats i aprenentatges del transcurs d'aquest projecte.

Els possibles obstacles són les limitacions i necessitats tècniques del dispositiu de realitat virtual, així com les limitacions i necessitats que sorgeixen al dur a la pràctica el disseny.

⁵ [En línia] Pàgina web, URL <http://store.steampowered.com/app/650000/DOOM_VFR> [Consulta el 25 de febrer de 2018].

⁶ [En línia] Pàgina web, URL <http://store.steampowered.com/app/722230/LA_Noire_The_VR_Case_Files> [Consulta el 25 de febrer de 2018].

⁷ [En línia] Pàgina web, URL <http://store.steampowered.com/app/611660/Fallout_4_VR> [Consulta el 25 de febrer de 2018].

El producte va dirigit als usuaris *hardcore* de videojocs, que els hi agrada la conducció i en tenen coneixements. Per altra banda, qui es beneficiarà del resultat són els desenvolupadors de la indústria del videojoc que busquin idees i un disseny dut a la pràctica orientat als jugadors *hardcore*, així com els que busquin un exemple realitzat amb una jugabilitat amb moviment dinàmic. A més, utilitzaran aquest prototip tant aquests jugadors que trobin i s'interessin pel prototip, com aquests desenvolupadors mencionats.

2. Estat de l'art

2.1 Tecnologia i dispositius

Actualment, tot i haver-hi desenes de dispositius de realitat virtual, els orientats per a videojocs es poden separar en dos grans grups:

- els que es connecten per cable al teu ordinador o consola, la qual és qui executa l'aplicació.
- els que utilitzen el teu mòbil per executar les aplicacions i, per tant, van sense cable.

Els videojocs per a mòbils són, majoritàriament, *casuals*, degut a la potència i propietats del dispositiu. Per això, els videojocs fets per a jugar amb els dispositius de realitat virtual que requereixen un mòbil són també *casuals*. Per aquesta raó, els que són interessants per als usuaris *hardcore* són els que es connecten a un dispositiu extern, on, en la indústria dels videojocs, tenim principalment tres dispositius: *Oculus Rift*, *HTC Vive* i *Sony PlayStation VR*. Descartarem però, l'últim atès que ens interessin especialment els que es connecten a un ordinador, ja que les consoles també són una limitació tècnica major que un ordinador i, a més, suposa també una barrera d'entrada major per al desenvolupador, a causa dels requisits que demana el propietari de la consola per a esdevenir-ne desenvolupador, sumant el cost i accés a un equip de desenvolupament.






És cert que hi ha la capacitat de connectar a un ordinador, tant els dispositius de realitat virtual per a mòbils com per a consoles, amb *TrinusVR*⁸ per exemple, però llavors tenim igualment la limitació del dispositiu i un augment d'intermediaris, que podria esdevenir en un augment significatiu del retard en la comunicació entre l'ordinador que executaria el videojoc i el dispositiu de realitat virtual. Per això, és més pràctic centrar-se en *HTC Vive* i *Oculus Rift*, ja que és on tenim el màxim potencial de realitat virtual.

A continuació, mostro una taula amb les especificacions tècniques dels dispositius més coneguts i utilitzats, on estan inclosos els anomenats anteriorment i, a més, dos dispositius per a mòbils.

⁸ [En línia] Pàgina web, URL <<http://www.trinusvirtualreality.com>> [Consulta l'1 de març de 2018]

Taula 2.1

Dispositius de realitat virtual

	Sony PlayStation VR	HTC Vive	Oculus Rift	Google Daydream View (2017)	Samsung Gear VR (2017)
Product					
Headset Type	Tethered	Tethered	Tethered	Mobile	Mobile
Connections	HDMI, USB 2.0	HDMI, USB 3.0	USB 3.0	None	USB 2.0, USB 3.0
Resolution	960 by 1,080 (per eye)	1,080 by 1,200 (per eye)	1,080 by 1,200 (per eye)	Native to phone	Native to phone
Refresh Rate (Hz)	120	90	90	60	Native to phone
Field of View (degrees)	100	110	110	Not Specified	101
Sensors	Motion, external visual positioning	Motion, camera, external motion tracking	Motion, external visual positioning	Motion	Motion
Controls	DualShock 4, PlayStation Move	HTC Vive motion controllers	Oculus Touch, Xbox One gamepad	Handheld remote	Handheld remote, touchpad on headset
Hardware Platform	PlayStation 4	PC	PC	Google Daydream	Samsung Gear VR powered by Oculus
Software Platform	PlayStation 4	SteamVR	Oculus	Android	Android

Taula 2.1: [En línia] (2017, 5, desembre). Pàgina web, URL

<<https://www.pcmag.com/article/342537/the-best-virtual-reality-vr-headsets>> [Consulta l'1 de març de 2018].

Com podem veure, els dos punters per a videojocs d'ordinador, *HTC Vive*⁹ i *Oculus Rift*¹⁰, no es diferencien gaire entre ells i, a més, a l'hora de desenvolupar es pot fer de manera que sigui compatible en els dos dispositius i així també augmentem l'accessibilitat del producte. Per tant, no ens és necessari una extensa comparativa entre els dos dispositius.

Centrant-nos en el comandament, tant *Oculus Rift* com *HTC Vive* tenen comandaments específics, que serveixen per a simular la posició de les dues mans per separat dins de l'entorn en 3D del videojoc. Per altra banda, també hi ha volants per a la simulació amb tecnologia hàptica. En aquest prototip no en farem ús, ja que ni busquem que l'usuari mogui les mans per separat ni la intenció és fer un simulador realista de conducció. En el nostre cas, un comandament com el de la Xbox 360 ens és suficient per al tipus de conducció que busquem on, així i tot, també podem aprofitar aquest comandament per a generar sensacions hàptiques amb la seva vibració. Per a aconseguir-ho, si el motor no té les eines suficients, hi han llibreries gratuïtes i de codi obert per augmentar-ne el control, com per exemple una que hi ha a *Github* anomenada *XInput*¹¹.

2.2 Productes

Com he comentat anteriorment: a causa de la limitació tècnica del dispositiu, per evitar el desplaçament del personatge dinàmicament i també augmentar el rendiment, molts videojocs han optat a moure el personatge fent teletransports, de manera que l'usuari apunta a un lloc i clicant un botó es teletransporta allí. Dins d'aquest grup, un dels que ho ha resolt d'una manera més adequada és *L.A. Noire: The VR Case Files*³, on en lloc de fer una fosa i el teletransport, canvia la càmera del jugador de primera a tercera persona i executa una animació on el personatge fa el desplaçament. Però, per a poder apuntar d'aquesta manera, aprofiten els comandaments del dispositiu de realitat virtual.

Per altra banda, entre els que l'usuari no s'ha de desplaçar físicament, hi ha els que directament permeten el moviment lliure, girant el personatge amb el gir del cap de l'usuari, on també hauria de girar el cos estant de peu per a permetre el gir en 360°; o, el gir del cap serveix per a mirar al voltant i es gira i mou el personatge amb el comandament, on llavors el jugador té facilitat per a jugar també assgut.

⁹ [En línia] Pàgina web, URL <<https://www.vive.com/us/product/vive-virtual-reality-system>> [Consulta l'1 de març de 2018].

¹⁰ [En línia] Pàgina web, URL <<https://www.oculus.com/rift>> [Consulta l'1 de març de 2018].

¹¹ [En línia] Pàgina web, URL <<https://github.com/speps/XInputDotNet>> [Consulta el 4 de març de 2018].

Ja centrant-nos en els videojocs que són o hi ha conducció:

- el ja anomenat *L.A. Noire: The VR Case Files*³, té parts de conducció però amb una jugabilitat extremadament arcade, sense cap tipus de penalitzacions per mala conducció ni cap tipus de dificultat.
- també tenim una nova versió del videojoc Desert Bus de 1998: *Desert Bus VR*¹², on també amb una jugabilitat totalment arcade simplement conduïm un autobús seguint una simple carretera.
- en l'altre extrem trobem ja simuladors més realistes de conducció que han fet el seu videojoc compatible amb els dispositius de realitat virtual, com el *Dirt Rally*¹³ o *Project Cars 2*¹⁴.

2.3 Diferenciació

Per a diferenciar aquest disseny i desenvolupament de la resta, la jugabilitat del prototip serà orientada als usuaris *hardcore*, ni sent completament arcade ni arribant al punt de ser un simulador, sinó que els controls seran més propers als videojocs arcade però el jugador haurà de complir mínimament la normativa de trànsit per evitar accidents.

¹² [En línia] Pàgina web, URL <http://store.steampowered.com/app/638110/Desert_Bus_VR> [Consulta l'1 de març de 2018].

¹³ [En línia] Pàgina web, URL <http://store.steampowered.com/app/310560/DiRT_Rally> [Consulta l'1 de març de 2018].

¹⁴[En línia] Pàgina web, URL <http://store.steampowered.com/app/378860/Project_CARS_2> [Consulta l' 1de març de 2018].

3. Planificació

Tal com es fa actualment en la majoria de casos dins de la indústria de videojocs, per a planificar el projecte, en fer-se ús d'una metodologia àgil per al desenvolupament, anomenat "scrum", on en lloc de planificar exactament pas a pas el procés, es va desenvolupant de forma àgil i iterativa, obert a canvis i amb l'objectiu final en ment amb unes iteracions anomenades "sprints", d'una durada de dues setmanes. En ser un projecte individual, no són necessaris els rols específics, ja que solament es fan reunions entre el tutor del projecte i l'alumne per avaluar l'avanç. Tanmateix, tal com també es sol fer en molts casos, s'han marcat unes "milestones", o fites on s'hauria d'haver assolit un cert punt:

- 09/02/2018: [document de disseny, GDD](#). Una setmana després del lliurament de la primera Rúbrica.
- 20/04/2018: MVP, producte viable mínim, on hi haurà la jugabilitat del producte i es podrà analitzar i avaluar per a verificar el disseny o decidir canvis. Una setmana abans del lliurament de la segona Rúbrica.
- 01/06/2018: versió beta avançada. Una setmana abans del lliurament de la tercera Rúbrica.
- 15/06/2018: versió final del prototip. Una setmana abans del lliurament del TFG.

3.1 Anàlisi DAFO, riscos i pla de contingències

Taula 3.1

Anàlisi DAFO

<p>Debilitats</p> <p>Solament és un prototip, on es prova un nou disseny i, així i tot, poden sortir imprevistos com en qualsevol desenvolupament.</p>	<p>Amenaces</p> <p>Entre ser solament un prototip i amb la quantitat de productes a l'abast gràcies a la facilitat de publicació, és difícil arribar a l'usuari a qui va dirigit. A més, la realitat virtual està encara introduint-se al mercat i el públic no és tan extens com en altres plataformes.</p>
<p>Fortaleses</p> <p>La producció àgil en videojocs permet adaptar-se fàcilment als imprevistos i essent solament un prototip, és llavors, un cop fet, quan es podrà avaluar el resultat i és llavors poden fer moltes millores coneixent millor cap a quina direcció enfocar el producte.</p>	<p>Oportunitats</p> <p>La realitat virtual és un producte innovador que ara està començant a introduir-se millor al mercat amb millors preus i ofertes i encara hi ha pocs productes.</p>

Per a parlar de riscos i contingències em referiré a les tres àrees del desenvolupament de videojocs:

- Disseny: és possible que al llarg del desenvolupament, gràcies als *sprints*, s'arribi a la conclusió que el disseny no és viable. Llavors s'haurà de valorar el problema i analitzar si amb el que es té en aquell moment hi ha alguna solució de disseny viable.
- Programació: és possible que sorgeixin dificultats en programar alguna mecànica o característica, i/o que això produeixi un augment molt elevat en el volum de treball, com també podria passar per culpa de la necessitat d'optimització per a evitar molèsties als usuaris de realitat virtual. Per això s'ha decidit que el prototip solament abarcarà una part de la jugabilitat similar a una demostració, l'anomenat "demo" en el sector.
- Art: és possible que sorgeixin dificultats a causa de les limitacions del dispositiu, per això i també atès a l'objectiu de solament fer el prototip, és possible que l'art no acabi sent el que hauria de ser en un producte final i hi hagi algun marcador de posició, l'anomenat "placeholder" en el sector.

3.2 Anàlisi inicial dels costos

Per a fer una anàlisi inicial dels costos, ja que el principal cost en el desenvolupament de videojocs és el salari, primer s'ha buscat el salari mitjà a Espanya de les professions que estarien relacionades amb el projecte, per a fer-ne una estimació: gestor i productor, dissenyador, programador i artista, tal com mostra la següent taula.

Taula 3.1

Salari mitjà a Espanya.

Job	Average annual Salary	Aprox. amount of € per hour
<i>Junior Programmer</i>	17.662€	10€
<i>Product Manager</i>	26.318	15€
<i>UX Designer</i>	25.888€	14€
<i>Videogame Artist</i>	aprox 14.000€	8€

Taula 3.1: Extret del fitxer adjunt anomenat *BorrellA_TFG_Pressupost.xlsx*.

Llavors s'ha fet una aproximació dels costos del material, on no ha estat necessari fer una llista extensa, ja que a part del material físic, el software necessari té versions gratuïtes per estudiants, tal com mostra la següent taula.

Taula 3.2

Cost de l'equipament.

Equipment	Price
PC with VR requirements	aprox. 1000€
VR device	aprox 500€
Development programs	student version 0€

Taula 3.2: Extret del fitxer adjunt anomenat *BorrellA_TFG_Pressupost.xlsx*.

A l'hora de fer el pressupost, en ser un projecte unipersonal, s'ha tingut en compte que hi treballaria una sola persona encara que ho faci en disciplines diferents, tal com es pot veure a la taula següent.

Taula 3.3

Col·laboradors i paquets de treball.

Partners						
Partner	Short name	Type	Country	Average cost (€)	Funding rate	
1	Aleix Borrell	Profit	Spain	10,77 €	100%	
Work packages						
	Type	WP leader	WP Title			
WP1	Management	Partner 1	Planificació i gestió del projecte			
WP2	Design	Partner 1	Disseny del prototip			
WP3	Technical	Partner 1	Desenvolupament, programació del prototip			
WP4	Art	Partner 1	Art del prototip			

Taula 3.3: Extret del fitxer adjunt anomenat *BorrellA_TFG_Pressupost.xlsx*.

Finalment, amb tota la informació anterior, s'ha calculat el pressupost:

- Com a cost de personal obtenim, a partir dels salaris anteriors i amb una aproximació de les hores que s'hauran d'invertir en cada disciplina, un cost de 3.230 €.
- Per a calcular l'amortització, s'hi ha comptabilitzat en quins paquets de treball es fa servir l'equipament i s'ha aproximat el cost d'amortització per als 5 mesos que durarà el projecte, sumant un cost de 625 €.

4. Metodologia

Com ja s'ha comentat a la planificació, s'emprarà per al desenvolupament una metodologia àgil anomenada "scrum", on es va desenvolupant de forma dinàmica i iterativa, amb l'objectiu final en ment, organitzat en unes iteracions anomenades "sprints", d'una durada de dues setmanes. En ser un projecte individual, no són necessaris els rols específics, ja que les avaluacions dels "sprints" són unipersonals i solament es fan reunions entre el tutor del projecte i l'alumne per avaluar l'avanç, a disponibilitat d'aquests.

En ser un projecte unipersonal, no s'ha adaptat les tasques per a utilitzar una eina de gestió, ja que aquestes estan orientades a grups i en aquest cas no es creu necessari, solament amb les "milestones" ja planificades i amb els objectius i la següent "milestone" en ment, es farà els anomenats "sprints".

Un cop ficada la metodologia en pràctica, s'ha vist que a causa de naturalesa d'aquesta i la pluridisciplinarietat de l'alumne, tot i seguir la planificació, no s'han pogut realitzar els "sprints" al peu de la lletra cada dues setmanes. Els "sprints" estan pensats per a un desenvolupament continuat i en aquest Treball de Final de Grau unipersonal, l'alumne ha hagut de compaginar-lo amb l'elaboració de la memòria, la recerca i amb el desenvolupament artístic, ja sigui en la fase inicial cercant recursos com editant-los perquè siguin més adients al prototip.

4.1 Eines per al seguiment del projecte

Durant el desenvolupament, per a guardar els documents del projecte s'utilitza *Google Drive*, i per a guardar les versions del codi s'utilitza *Github*, plataforma que també serveix per a oferir disponibilitat al públic de, tant les versions executables i finals del producte, com del codi complet. Ambdues són gratuïtes per a l'usuari.

4.2 Eines de validació

Per a validar tant el desenvolupament com el resultat, després de les "milestones" i d'alguns "sprints", es generarà una versió executable i es publicarà a *Github*, de manera que estarà disponible per a provar-se tant pel tutor del projecte, per l'alumne com per a diversos possibles usuaris que serviran de testers voluntaris, per a obtenir una visió i crítica externa al projecte a partir d'un qüestionari.

Un cop realitzada la primera “milestone”, juntament amb els inconvenients inicials amb el dispositiu de Realitat Virtual i les necessitats de software per a facilitar el testeig, s’ha pres la decisió d’iniciar-lo a partir d’aquest punt. A més, la naturalesa d’aquesta fita està més encarada a una validació interna on provar la jugabilitat bàsica i verificar la viabilitat del prototip des d’un punt de vista més tècnic. Per tant, la jugabilitat en aquest punt no està totalment acabada ni el prototip optimitzat per a l’usuari final, desencadenant en una experiència incompleta i curta per a aquest.

Finalment, a causa dels inconvenients i requeriments trobats, tal com s’explica en l’apartat “5.2.2.3 Testeig, millores i resultat”, juntament amb la coincidència amb els exàmens finals del centre, només ha estat possible fer un testeig extern amb el dispositiu. A l’haver estat un testeig individual i directe amb un company proper de classe, no ha estat necessari un qüestionari, gràcies a la facilitat de parlar directament les sensacions i opinions sobre el prototip.

5. Desenvolupament del projecte

5.1 Preproducció

5.1.1 Eines per al desenvolupament en Realitat Virtual

5.1.1.1 Motor per a Videojocs

Al mercat, de forma gratuïta per a ús educatiu, hi ha una gran quantitat de diferents motors de videojocs que ja tenen integrades eines per a desenvolupar orientat a Realitat Virtual. Entre ells, els més coneguts són: *Unity*, *Unreal Engine*, *Godot Engine*, *CryEngine*.

Ja que no és necessària una gran llibreria de Realitat Virtual, perquè no es farà ús a fons dels sensors ni utilitzarem els comandaments específics dels dispositius per a simular les mans per separat. Un cop ens centrem en els motors en tres dimensions amb una mínima integració, ja que aquests ofereixen un servei similar i tenint en compte del temps que es disposa sent un treball unipersonal, la millor opció és elegir el motor que sigui més còmode i es tingui més experiència.

Per les raons mencionades, s'ha fet una preselecció centrant-se entre *Unreal Engine*¹⁵ i *Unity*¹⁶, els quals s'han vist a classe durant el grau en Disseny i Desenvolupament de Videojocs, sobretot l'últim, i són ara mateix els més accessibles i utilitzats, i per tant, hi ha més informació, ajuda, guies i documentació a Internet. A més, els dos tenen llibreries actualitzades i integració dinàmica amb realitat virtual, per a poder provar el dispositiu ràpidament quan es desenvolupa i es fan proves, on el motor automàticament modifica la càmera principal per a adaptar-se al dispositiu, modificant la vista, matrius de projecció, perspectiva i posició. D'aquesta manera, no és necessari haver de crear una versió executable cada vegada que es vulgui provar un canvi.

Ja que ofereixen un servei similar, amb petits avantatges i desavantatges ambdós, sense que cap faciliti aconseguir els objectius del treball per sobre de l'altre, finalment s'ha decidit utilitzar *Unity* per al prototip. Aquesta elecció ha estat basant-se en que és el motor al qual l'alumne està més habituat i el que també utilitza a l'empresa on està fent pràctiques.

¹⁵ [En línia] Pàgina web, URL <<https://docs.unrealengine.com/en-us/Platforms/VR>> [Consulta el 25 d'abril de 2018].

¹⁶ [En línia] Pàgina web, URL <<https://docs.unity3d.com/Manual/XR.html>> [Consulta el 25 d'abril de 2018].

D'aquesta manera, tenint en compte les limitacions i objectius, és la millor eina a disposició per assolir el resultat desitjat satisfactòriament.

5.1.1.2 Dispositiu de Realitat Virtual

Com ja s'ha investigat i comentat en l'estat de l'art, els dispositius als quals va dirigit el prototip són *Oculus Rift* i *HTC Vive*.

Pel que fa al dispositiu per al desenvolupament i per al testeig, gràcies a la disponibilitat que ofereix el Centre de la Imatge i la Tecnologia Multimèdia, hi ha com a material de préstec els equips de desenvolupament d'*Oculus Rift* en la seva primera i segona versió. Tanmateix, després dels inconvenients trobats a la pràctica, ens centrarem en la segona versió, ja que la primera ha resultat estar ja obsoleta. El primer equip de desenvolupament d'*Oculus Rift*, a part d'haver-se quedat enrere tècnicament, no té suport i està desactualitzat, generant problemes a l'hora d'intentar utilitzar-lo i obligant a instal·lar versions antigues dels controladors del dispositiu.

5.1.2 Recursos inicials

Per a facilitar el desenvolupament centrat en Realitat Virtual, per a començar, en lloc de fer els recursos i texturitzar-los l'alumne sol, que esdevindria una càrrega d'hores inabastable, s'ha fet una recerca i tria de recursos per a crear l'entorn i les bases del prototip, per a després ja editar-les i modificar-les segons les necessitats del prototip. En la recerca de recursos s'ha buscat bàsicament l'entorn, una ciutat, i el vehicle base.

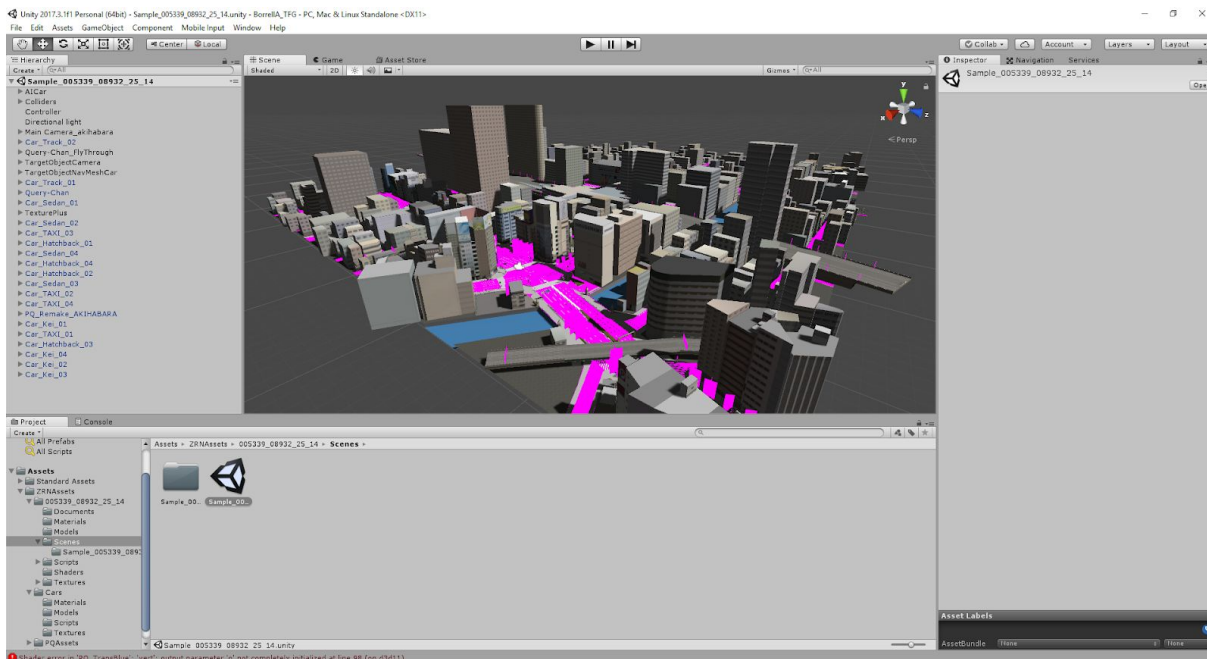
5.1.2.1 Ciutat

Per a la ciutat, primer s'ha intentat buscar recursos d'edificis per a després generar manualment la ciutat. Després de la primera recerca on no s'ha trobat recursos adients, ja que entre ells variaven massa en tots els aspectes i llavors seria impossible compaginar-los per a crear un entorn a mà coherent, s'ha intentat buscar una ciutat o un conjunt de recursos suficients. Finalment, s'ha trobat una ciutat completa que simula Akihabara¹⁷, no estava actualitzada a les últimes versions, però solament fent uns canvis en els *shaders* i manualment canviant la configuració d'aquests dins del motor s'ha solucionat. També s'ha

¹⁷ [En línia] Pàgina web, URL
<<https://assetstore.unity.com/packages/3d/environments/urban/japanese-otaku-city-20359>> [Consulta el 28 d'abril de 2018].

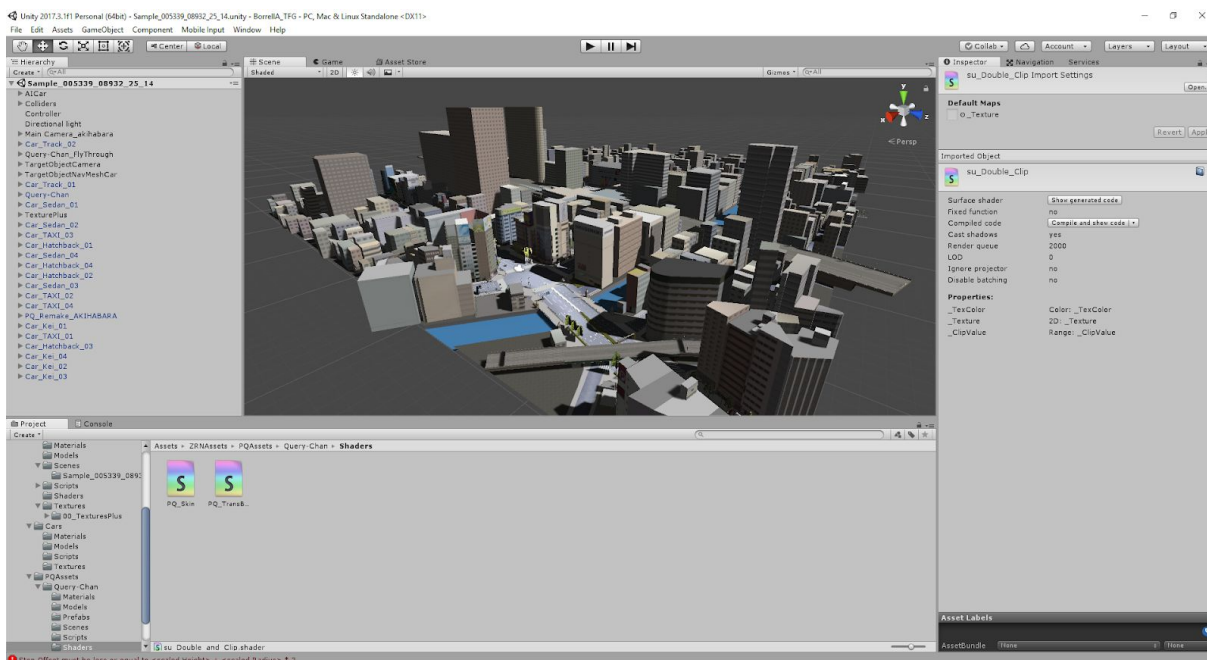
hagut de fer de nou la malla de navegació, que servirà per als vehicles que no controlarà l'usuari, i afegir les físiques necessàries perquè xoqui el vehicle.

Imatge 5.1



Imatge 5.1: Captura del recurs sense cap modificació.

Imatge 5.2



Imatge 5.2: Captura del recurs després d'editar els shaders per actualitzar-los.

5.1.2.2 Vehicle base

Per al vehicle base, inicialment s'ha mirat els recursos que ja té *Unity* per a aquests. Així i tot, a l'haver trobat durant la recerca uns recursos més complets, s'ha optat per a utilitzar "MS Vehicle System"¹⁸. Un cop implementat aquest recurs, únicament s'ha hagut de canviar i adaptar la càmera per a Realitat virtual, juntament amb els controls del vehicle per a adaptar-los als desitjats pel prototip, fent els canvis adequats en la fase de producció.

Durant la fase de producció, també s'han de modificar els retrovisors i afegir-hi la reflexió amb efecte de mirall. Aquesta reflexió s'ha aconseguit gràcies a un *script* i un *shader* de la wiki d'*Unity*¹⁹. En la seva implementació, pel fet que s'obté una reflexió en l'eix y positiu, i és necessari que l'origen sigui un objecte amb la posició local a zero, s'ha editat l'*script* per a tenir la possibilitat d'utilitzar la transformació del pare de l'objecte. D'aquesta manera, introduint cada mirall com a fill d'un objecte orientat correctament, s'obté la reflexió desitjada.

5.2 Producció

5.2.1 Primera "milestone". Producte Viable Mínim

Per a aquesta fita, on l'objectiu era tenir una aproximació funcional de la jugabilitat del jugador amb el seu vehicle i un entorn, s'ha utilitzat els anteriors recursos de la fase de preproducció, editats amb els canvis inicials comentats, per a muntar l'escena d'entrada amb el menú d'inici i accés a l'escena principal, on hi haurà la jugabilitat.

¹⁸ [En línia] Pàgina web, URL <<https://assetstore.unity.com/packages/tools/physics/ms-vehicle-system-free-version-90214>> [Consulta el 28 d'abril de 2018].

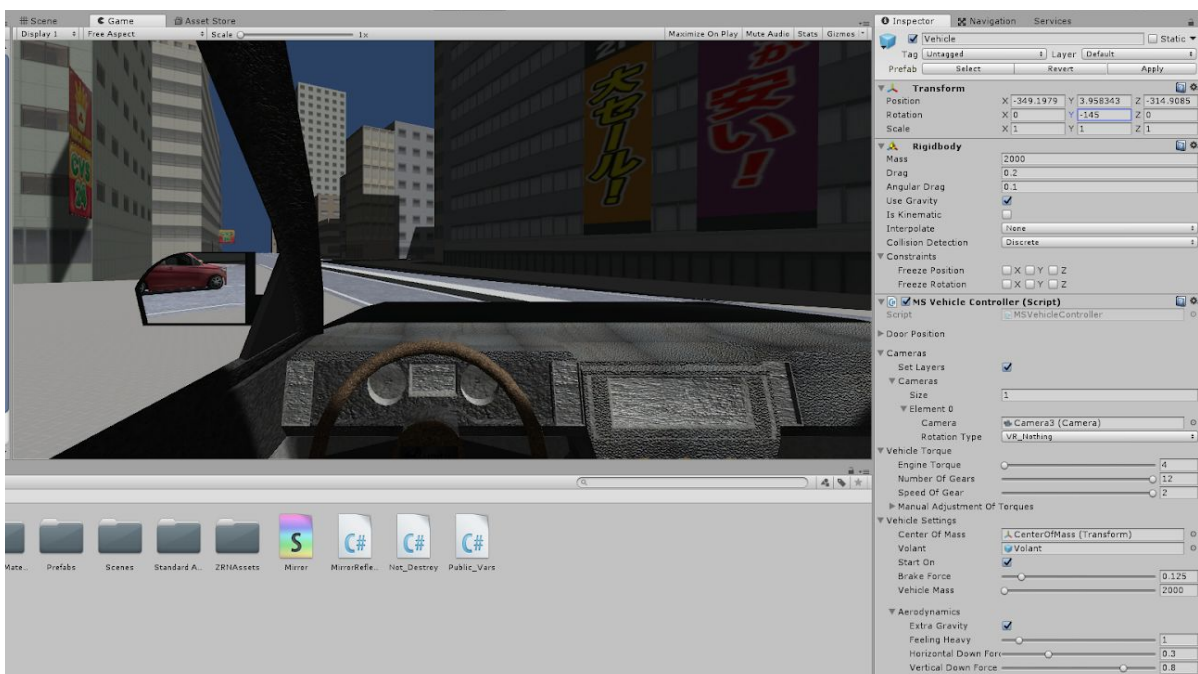
¹⁹ [En línia] Pàgina web, URL <<http://wiki.unity3d.com/index.php/MirrorReflection4>> [Consulta el 28 d'abril de 2018].

Imatge 5.3



Imatge 5.3: Muntatge de l'escena per a la primera "milestone".

Imatge 5.4



Imatge 5.4: Muntatge del vehicle per a la primera "milestone".

Per a tenir una bona accessibilitat, s'ha habilitat l'ús i control del prototip tant amb teclat i ratolí i pantalla com amb comandament i ulleres de realitat virtual, encara que l'experiència estigui orientada i dissenyada per a aquests últims. Per a aconseguir-ho, gràcies a les

funcions que té *Unity*, es pot detectar quan es connecta o desconnecta un comandament o un dispositiu de realitat virtual, i també es pot habilitar o deshabilitar l'ús d'aquest. D'aquesta manera, s'ofereix llibertat a l'usuari a l'hora de fer ús del prototip.

Per a elaborar aquests menús mantenint l'accessibilitat mencionada, s'ha elaborat un sistema per a detectar els diferents dispositius i guardar les preferències de l'usuari triades en el menú, així com habilitar diferents versions d'aquest menú, orientades als diferents modes, ja que aquests no tenen les mateixes necessitats. Amb teclat i ratolí l'usuari ha de poder fer clic en els botons i, en canvi, amb comandament s'ha d'habilitar la navegació entre aquests botons. Pel que fa al dispositiu per a mostrar l'univers gràfic al jugador, aquests tenen diferents resolucions i, sobretot, diferents funcions. Així com en una pantalla es pot mostrar el menú estàtic omplint tota la pantalla superposadament a l'escenari, amb el dispositiu de realitat virtual, al poder moure el cap amb llibertat, és indispensable fer aquesta interfície diegètica per a evitar el mareig, configurant els menús per a estar localitzats físicament en l'escenari.

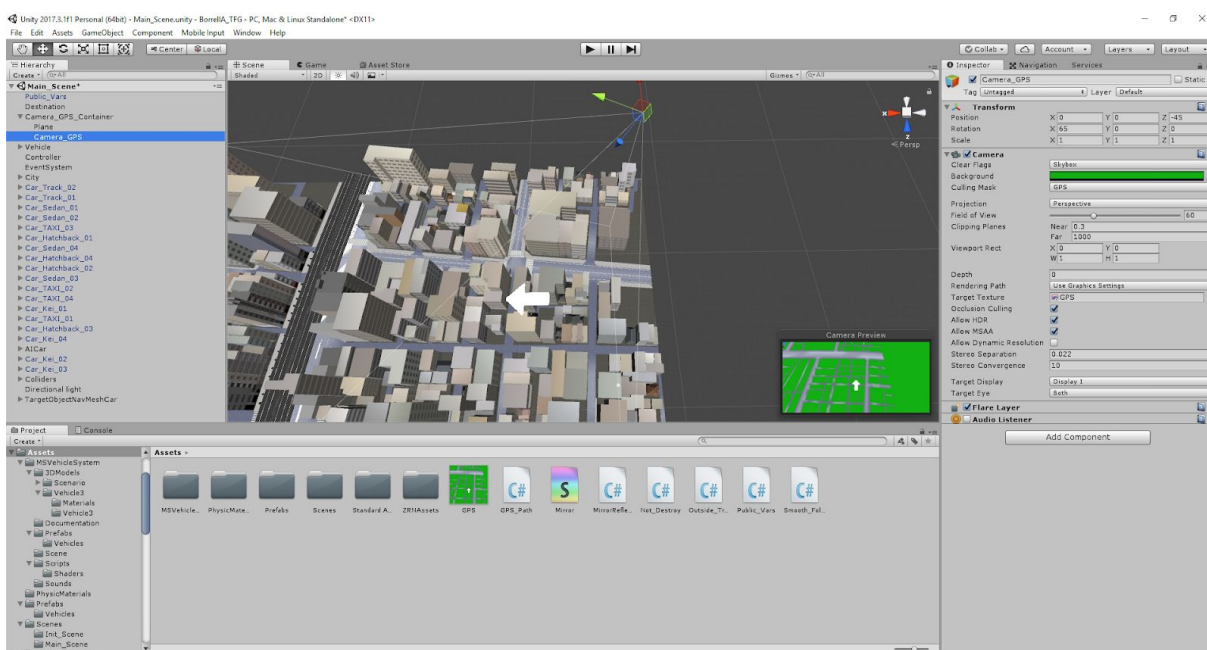
5.2.2 Segona "milestone"

5.2.2.1 GPS

Un cop la jugabilitat bàsica del personatge està resolta, per a orientar el jugador cap al seu objectiu, s'ha desenvolupat un GPS. Fer-ho ha estat facilitat gràcies al motor, ja que ofereix la funcionalitat per a guardar i actualitzar una imatge des d'un altre punt de vista en temps real, i a més filtrar els tipus d'objectes que es vol visualitzar.

Aquesta imatge, que es generarà en format de textura, s'obté a partir d'una nova càmera que filtrarà els objectes que etiquetem per al GPS, com són el riu, les carreteres duplicades sense físiques o una imatge amb una fletxa, la qual tindrà la posició lligada a la càmera i ens servirà per a marcar la posició del jugador.

Imatge 5.5



Imatge 5.5: Muntatge de la càmera que generarà la textura per al GPS.

Per a pintar el camí fins al destí, s'ha implementat en un *script*, aprofitant el sistema propi de navegació del motor per a obtenir el recorregut, un pintat en aquest filtre que s'actualitza imitant un dispositiu real. Sobre aquest sistema de navegació, se'n profunditzarà més en [l'apartat 5.2.3.1 Intel·ligència artificial dels vehicles](#). També s'ha adaptat la posició de la càmera que crea la imatge perquè no es desplaci instantàniament a la posició del jugador, a partir d'un *script* que inclou *Unity* amb l'objectiu de fer que la càmera segueixi al jugador. S'ha editat a les necessitats per a imitar el funcionament d'un GPS, com mantenir bloquejada l'alçada i actualitzar la posició amb retard sense actualitzar la rotació mirant al jugador, únicament fer-ho en l'eix y copiant la rotació del jugador en aquest eix, on també es fa amb retard.

Imatge 5.6



Imatge 5.6: Resultat obtingut en implementar el GPS amb el camí.

5.2.2.2 Mareig i malestar

Amb l'objectiu per al jugador ja indicat, per a poder fer un testeig viable, s'ha començat a investigar i implementar eines i pràctiques de desenvolupament per a optimitzar el producte i evitar el mareig del jugador.

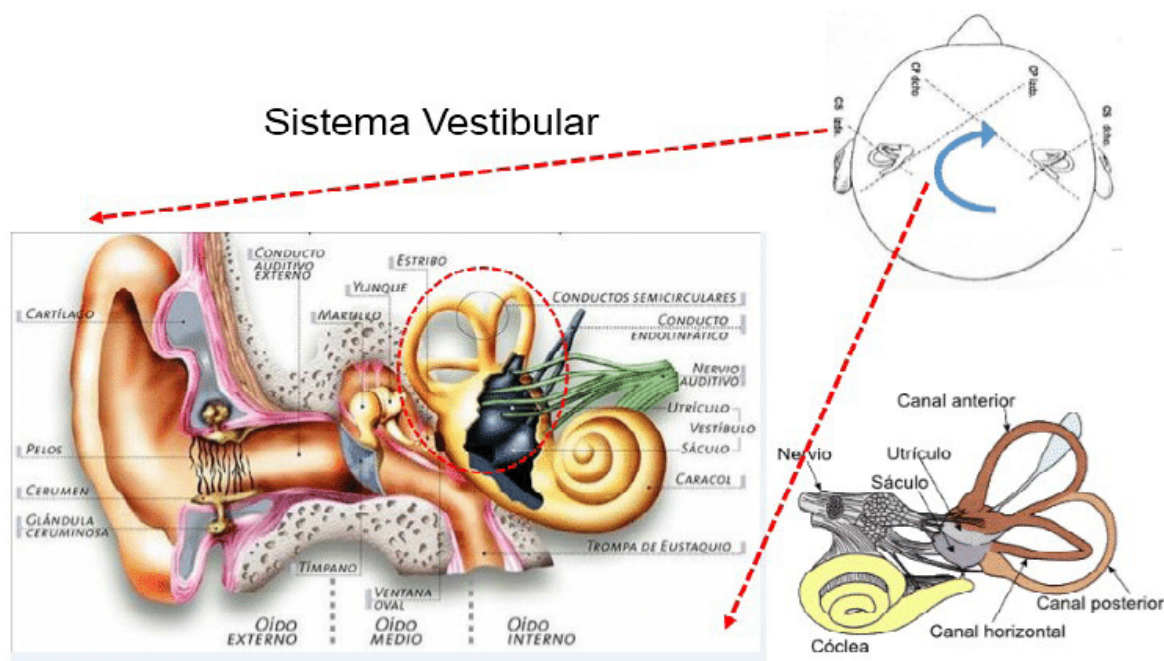
5.2.2.2.1 Sistema Vestibular

Concretament, tal com ens explica la guia per al moviment en realitat virtual del motor de videojocs Unity²⁰, la causa d'aquest mareig és la discordança entre la informació visual i la del sistema vestibular, localitzat a les orelles, el qual informa el cervell sobre la posició i moviment del cap. Si la informació que proporciona el sistema vestibular no correspon al que ahora envia la vista al cervell, aquest assumeix que el nostre cos ha estat enverinat, originant un avís en forma de mareig i malestar.

²⁰ [En línia] Pàgina web, URL <<https://unity3d.com/es/learn/tutorials/topics/virtual-reality/movement-vr>> [Consulta el 19 de maig de 2018].

Imatge 5.7

Sistema Vestibular



Taula 5.7: [En línia] (2017, gener). Pàgina web, Caracterización de la adquisición de nistagmos asociados al estudio del sistema vestibular, imatge científica extreta de ResearchGate, URL

<https://www.researchgate.net/Representacion-de-la-estructura-del-oido-y-del-sistema-vestibular-1_fig1_312661529> [Consulta el 5 de juny de 2018].

Per a minimitzar aquest problema, ja que tenim el vehicle base del jugador implementat, el qual ja ens serveix de punt de referència per a també reduir el mareig, s'ha investigat quins valors per a l'acceleració i el fre són els més adequats. Centrar-nos en aquest apartat ha estat basant-se en que principalment és en aquests dos moments, en accelerar i frenar, on el cervell rep informacions que no concorden, ja que el sistema vestibular no percep la velocitat constant.

Investigant sobre com actuar davant d'aquest problema, s'han trobat informacions contradictòries, inclús en les entrevistes fetes al mateix fundador i inventor de les *Oculus Rift*, Palmer Luckey. A finals del 2013²¹, en l'entrevista comenta que el nostre cos no assimila bé la gran acceleració instantània que hi ha en alguns jocs, i que per a

²¹ [En línia] Pàgina web, URL

<<https://www.techradar.com/news/gaming/oculus-rift-creator-xbox-one-and-ps4-are-far-too-limited-for-what-we-re-planning-1198420>> [Consulta el 19 de maig de 2018].

desenvolupar en realitat virtual s'hauria de tenir en compte valors que siguem capaços d'assimilar en la realitat. En canvi, en una altra entrevista realitzada a principis de 2016²², explica el fet que la velocitat no és el problema, sinó que ho és la constant acceleració i desacceleració, on per exemple, una acceleració realment instantània de zero a cent, crea malestar a ben poca gent comparant-ho amb l'acceleració gradual, que crea malestar en la gran majoria.

A partir d'aquestes informacions contradictòries i basant-se també en l'opinió directa del públic²³, s'ha arribat a la conclusió que aquest tema recau en les diferències físiques de cada persona. Per tant, un mateix sistema amb acceleracions instantànies pot fer sentir malestar a una part del públic que no ho sentirà amb les graduals però, en canvi, una altra part del públic pot sentir malestar amb aquestes acceleracions graduals però no amb les instantànies.

Per a poder arribar a tots els tipus d'usuaris, s'ha habilitat una opció en el prototip per a donar la possibilitat al jugador a elegir entre les dues acceleracions. En una tindrà acceleració i fre de manera més instantània i llavors es mantindrà la velocitat constant, i en l'altra, acceleració i fre més suau i la fricció amb el terra habilitat, que esdevindrà en un fre suau de forma passiva.

5.2.2.2 Optimització

Desenvolupant per a realitat virtual, l'optimització guanya més importància, ja que, tal com es veu en [la Taula 2.1: Dispositius de realitat virtual](#), aquests dispositius tenen una bona resolució per ull, el que seria equivalent a duplicar la típica pantalla. A més, mantenir uns FPS alts a 60 estables també és molt important, ja que la immersió visual ens fa molt més sensibles a una mala optimització, i juntament amb tenir la imatge tan propera a l'ull, si no mantenim aquests mínims, l'usuari es marejarà i notarà malestar ràpidament.

Amb l'ajuda de les eines pròpies del motor Unity, com el "profiler", que serveix per a veure en temps real els FPS i què consumeix més recursos en cada instant, es pot analitzar què

²² [En línia] Pàgina web, URL
<<https://www.technologyreview.com/s/601052/oculus-founder-palmer-luckey-on-what-it-will-take-to-make-virtual-reality-really-big>> [Consulta el 19 de maig de 2018].

²³ [En línia] Pàgina web, URL
<https://www.reddit.com/r/oculus/comments/2h8g7n/how_do_you_feel_about_acceleration> [Consulta el 19 de maig de 2018].

hauria de ser la prioritat a optimitzar. A més, amb la documentació i tutorials, ja s'obté una guia per a optimitzar orientat a realitat virtual²⁴.

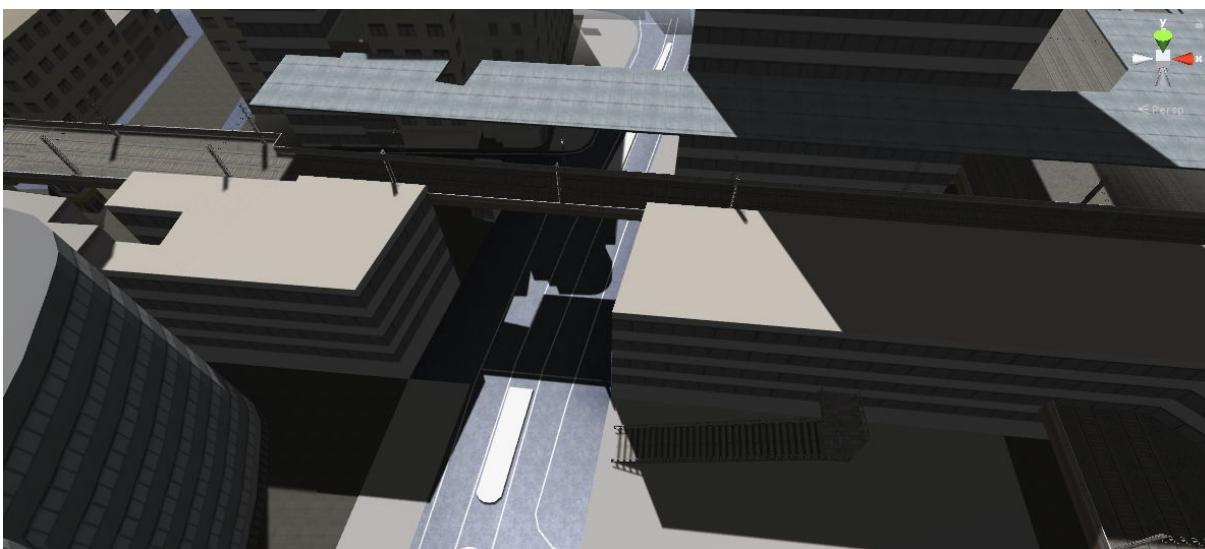
Per a començar, a causa de la resolució del dispositiu, la prioritat és centrar-se en el *renderitzat*, ja que serà el que consumirà una major quantitat de recursos amb més probabilitat. Per això, primer s'ha configurat adequadament la càmera, limitant el pla llunyà per a evitar *renderitzar* elements que no veurem, el que s'anomena *frustum culling*. Hi ha la possibilitat d'optimitzar-ho més, afegint boira, i així poder limitar més el pla llunyà de la càmera, però en el prototip no ha estat ni necessari ni possible utilitzant els *shaders* propis d'*Unity*, ja que per als miralls es modifica la matriu de projecció. Per a seguir optimitzant en aquest aspecte, llavors hi ha l'*occlusion culling*, que serveix per a evitar *renderitzar* elements que estan dins la vista de la càmera però tapats per algun objecte més proper a aquesta. En el cas de tenir un escenari més tancat amb algun embut en el recorregut com un passadís o porta, es pot preparar l'escena per a directament evitar *renderitzar* zones que sabem que estan fora de visió.

Relacionat amb la distància de *renderitzat*, també és important utilitzar els LOD, nivells de detall, en cas de tenir un escenari obert i recursos elaborats, per a utilitzar-ne amb menys triangles segons la distància. Tampoc ha sigut indispensable al tenir bàsicament edificis, ja que aquests són en la seva major part cubs sense malles complexes.

La part d'optimització que també s'ha implementat, ha estat etiquetar els elements estàtics per a simplificar el *renderitzat* que farà *Unity*, així com configurar aquests objectes per a posteriorment fer un mapatge de la il·luminació. D'aquesta manera, a costa d'augmentar la mida dels recursos, podem aconseguir una il·luminació més realista i menys costosa per als elements estàtics.

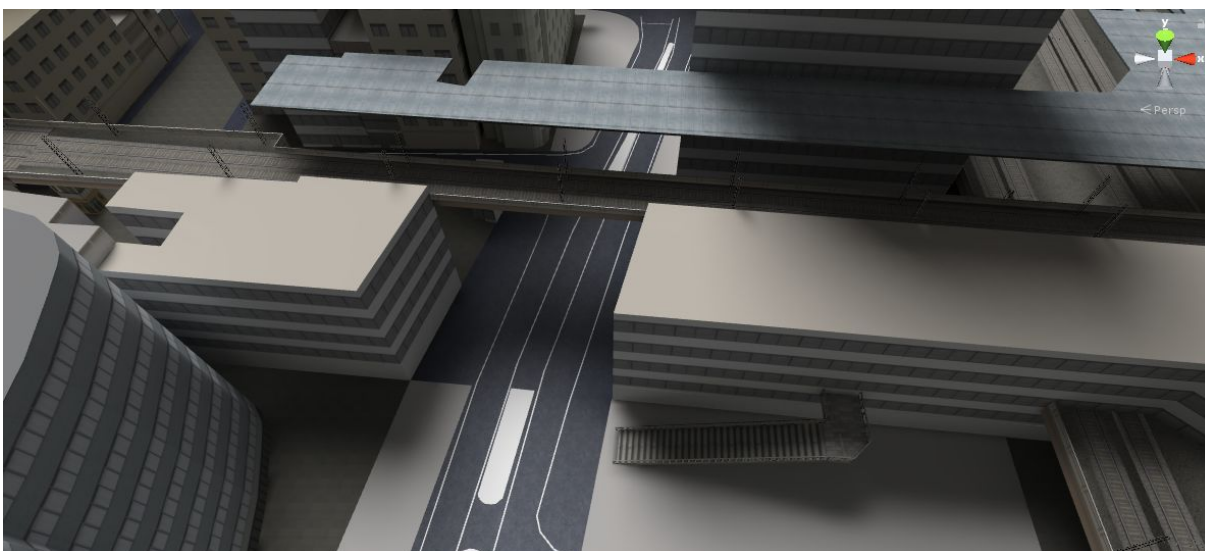
²⁴ [En línia] Pàgina web, URL <<https://unity3d.com/es/learn/tutorials/topics/virtual-reality/optimisation-vr-unity>> [Consulta el 2 de juny de 2018].

Imatge 5.8



Imatge 5.8: Il·luminació anterior, dura i en temps real.

Imatge 5.9

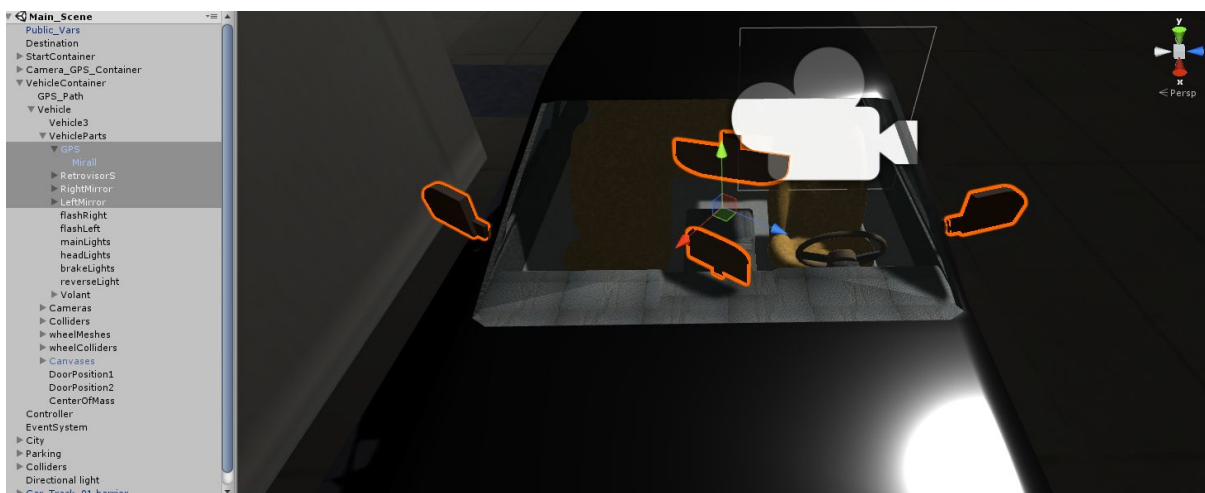


Imatge 5.9: mapa d'il·luminació, més suau i realista.

5.2.2.3 Testeig, millores i resultat

Per a aquesta fita, al ja estar més orientada a fer testeig, s'ha millorat l'accessibilitat, simplificant els menús i adaptant-los millor a la navegació amb comandament, així com limitant al jugador dins de la ciutat. També s'ha afegit el retrovisor central i el panell per al GPS, així com s'han editat les malles dels retrovisors per a poder-los editar independentment, amb l'objectiu d'obtenir una rotació més adequada per a cadascun d'ells.

Imatge 5.10



Imatge 5.10: Implementació de les noves malles.

Per a implementar les sensacions hàptiques que tenim a l'abast, amb l'ajut d'XInput¹⁰, s'ha aconseguit controlar la vibració del comandament i s'ha fet que concordi amb el motor del vehicle, així com a reacció en xocar. D'aquesta manera, fàcilment amb la funció `GamePad.SetVibration(PlayerIndex playerIndex, float leftMotor, float rightMotor)`, es pot controlar a quina potència es desitja fer vibrar el comandament. Un cop s'ha controlat la vibració, també s'ha de tenir en compte aturar aquesta, ja que si no es crida altre cop la funció anomenada anteriorment passant per valor zero al motor esquerre i dret, el comandament no deixarà de vibrar encara que l'aplicació es tanqui. Per aquesta raó, tant en obrir els menús dins de l'aplicació com en tancar aquesta, s'ha de cessar la vibració. En Unity, és fàcil aconseguir-ho gràcies a la funció que es crida en tancar l'aplicació, `OnApplicationQuit()`, es pot executar el codi desitjat abans que l'aplicació es tanqui.

A causa de la necessitat d'un processador amb suport a les instruccions SSE 4.2 per a instal·lar els elements necessaris per a desenvolupar i utilitzar el dispositiu de realitat virtual, el testeig a estat limitat. Així i tot, s'ha pogut realitzar un testeig amb un alumne extern al treball de fi de grau, el qual ha provat diverses vegades seguides el prototip, provant les dues opcions de velocitat, sense mostrar signes de mareig. En el seu cas no notava un gran mareig en cap de les dues opcions de velocitat, a diferència de l'alumne que desenvolupa el prototip, que nota major mareig amb l'acceleració instantània.

A conseqüència del testeig, gràcies a la impressió de tant el director com l'alumne extern, s'ha millorat el control de la direcció del vehicle, suavitzant-ne tant la velocitat com la reacció en el gir.

Imatge 5.11



Imatge 5.11: Resultat obtingut en la segona “milestone”.

5.2.3 Última “milestone”

En aquesta fita, l'objectiu ha estat deixar el prototip el més tancat possible, programant la cinemàtica final, tal com es va planejar en [el document de disseny, el GDD adjunt en l'annex 9.1](#). Ja que en la cinemàtica final intervenen altres vehicles, i també per a donar una sensació de producte més acabat, s'ha programat una intel·ligència artificial per als vehicles que es desplaçaran aleatòriament per la ciutat, juntament amb establir els semàfors i encreuaments que determinaran el seu moviment.

5.2.3.1 Intel·ligència artificial dels vehicles

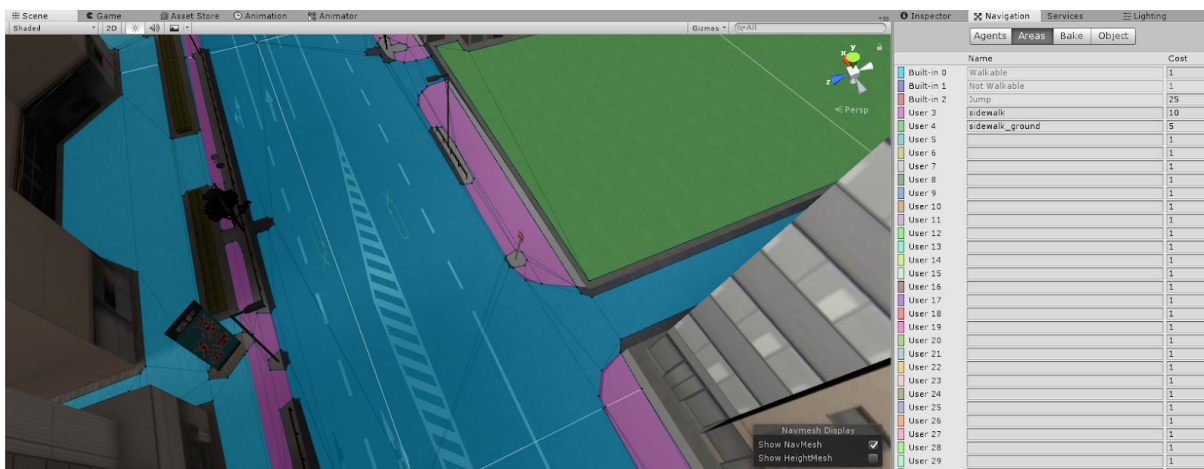
S'ha afegit als vehicles un component d'Unity, “NavMesh Agent”²⁵, que serveix per a aconseguir que els objectes es desplacin cap a un destí establert segons els seus valors i generant el recorregut per sobre de la “NavMesh”²⁶. Aquesta malla de navegació es genera

²⁵ [En línia] Pàgina web, URL <<https://docs.unity3d.com/Manual/class-NavMeshAgent.html>> [Consulta el 23 de juny de 2018].

²⁶ [En línia] Pàgina web, URL <<https://docs.unity3d.com/Manual/nav-BuildingNavMesh.html>> [Consulta el 23 de juny de 2018].

dins del motor i se li poden establir els costos que tenen navegar per sobre de cada objecte. D'aquesta manera s'obté la possibilitat de prioritzar el recorregut per sobre de la carretera, sense prohibir trepitjar la vorera o el terra a l'altre costat d'aquesta.

Imatge 5.12

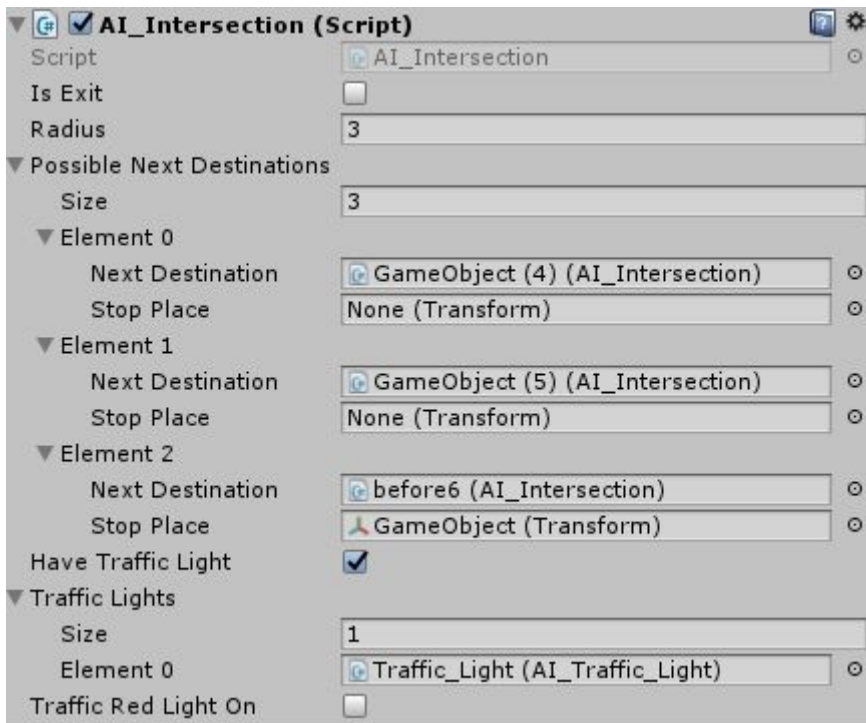


Imatge 5.12: Configuració dels costos i resultat de la malla de navegació.

Un cop assolits aquests requisits, podem controlar el comportament del component en un *script*²⁷. Per al desplaçament, s'utilitza com a base un comú sistema que consisteix a determinar el recorregut establint els punts clau de manera que en arribar al destí, se li dona a l'agent el següent de la llista. Llavors, per a fer un sistema aleatori i automàtic, amb moviments més realistes, s'ha programat un sistema on aquests punts claus esdevenen en les interseccions. En aquestes interseccions s'ha afegit un *script* creat en què s'especifiquen els possibles següents destins, el radi de detecció, si té un stop, els semàfors juntament amb la informació sobre si estan en vermell i si aquesta posició és una sortida del mapa. Aquest *script* també gestiona el temporitzador dels semàfors per a canviar-ne el color guardant la informació per a facilitar-la als agents. A més, l'*script* també té funcions per a rebre aleatòriament una destinació de la llista sobre les possibles, així com per a saber el radi de detecció, si els semàfors estan en vermell i el temps que tardaran a canviar de color.

²⁷ [En línia] Pàgina web, URL <<https://docs.unity3d.com/ScriptReference/AI.NavMeshAgent.html>> [Consulta el 23 de juny de 2018].

Imatge 5.13

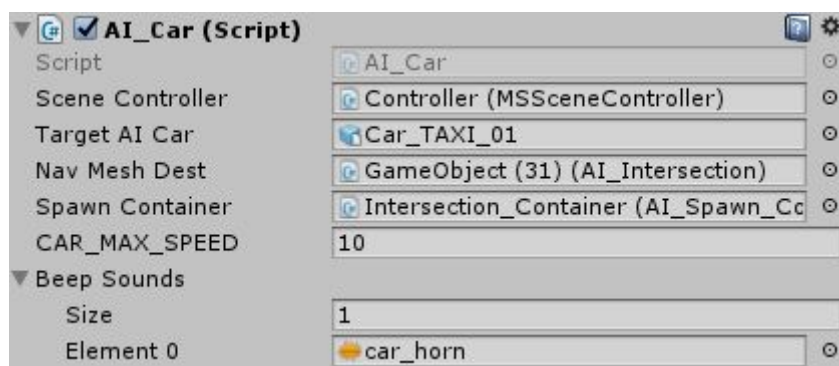


Imatge 5.13: Exemple d'una intersecció amb les dades especificades.

El codi de l'script, juntament amb l'historial de versions, el podem trobar en el següent enllaç: https://github.com/AleixBV/BorrellA_TFG/blob/master/Assets/Scripts/AI_Intersection.cs.

Finalment, on hi ha el comportament de la intel·ligència artificial, és en l'script per als vehicles, on s'especifica el controlador de l'escena, per a poder accedir al vehicle del jugador, el model del vehicle, la següent intersecció com a destinació, l'script que conté les interseccions que serveixen de punt d'aparició i en retorna un aleatòriament, la velocitat màxima del vehicle i els sons per a quan hagi de fer sona la botzina.

Imatge 5.14



Imatge 5.14: Exemple d'una intel·ligència artificial amb les dades especificades.

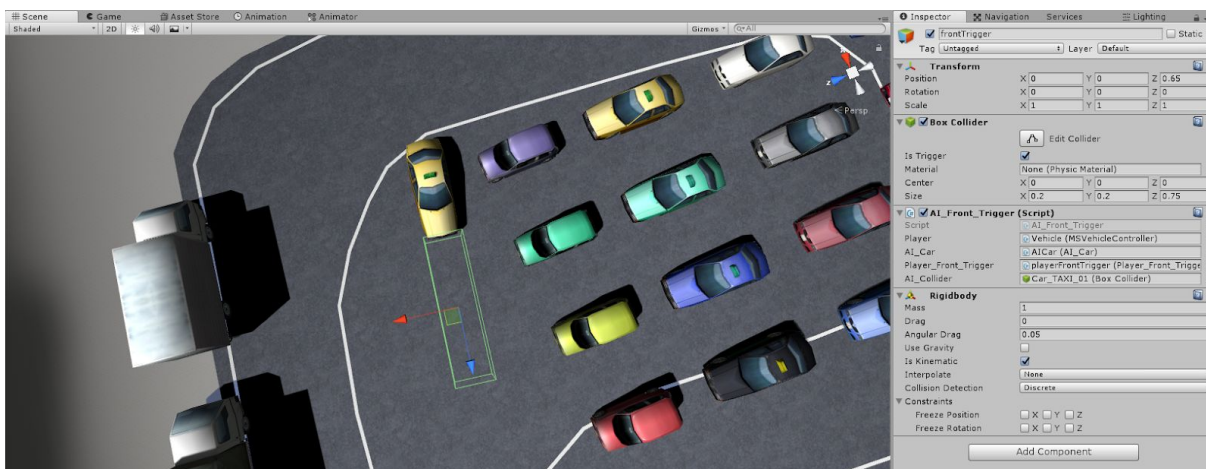
El codi de l'script, juntament amb l'historial de versions, el podem trobar en el següent enllaç: https://github.com/AleixBV/BorrellA_TFG/blob/master/Assets/Scripts/AI_Car.cs.

Aquest script, a part de diverses funcions per a llegir o modificar variables i parar o engegar el vehicle, principalment té tota la funcionalitat per a determinar la següent intersecció, així com detectar si hi ha un stop o un semàfor en vermell per a actuar, dins de la funció *FixedUpdate()*. Aquesta funció proporcionada per *Unity*, es crida de forma estable independentment dels FPS. El primer pas és comprovar que l'agent té el recorregut calculat i si el vehicle hi ha arribat tenint en compte el radi de detecció. Llavors es fan les comprovacions adequades per parar el vehicle si el semàfor està en vermell, frenar si aquest destí era un stop, o aparèixer en un punt d'aparició si era una sortida del mapa. Després se li assigna la següent destinació, que rebem de l'*script* de la intersecció que s'ha guardat com a destí actual, on també comprovem si aquesta conté un stop o és un punt intermedi per a aconseguir un recorregut més natural. En aquest últim cas on utilitzem com a destí un punt intermedi, es desactivarà que l'agent freni automàticament en apropar-se al destí, canviant el booleà que conté anomenat "autoBraking" a fals. Posteriorment es recuperarà l'estat anterior en arribar en aquest destí intermedi.

Un cop s'ha aconseguit la navegació dels vehicles, es vol evitar que aquests col·lideixin entre ells o amb el jugador però, tampoc es vol que evitin l'obstacle automàticament com fa l'agent de navegació del motor, ja que podria realitzar alguna infracció com trepitjar la vorera o envair el carril contrari. Per aquesta raó, s'ha desactivat l'evasió automàtica d'aquests obstacles i s'ha gestionat el comportament a partir d'un *script*, juntament amb un objecte que ens servirà per a detectar col·lisions davant dels vehicles. A aquest se li haurà d'especificar

el controlador del vehicle del jugador, l'*script* de la intel·ligència artificial i el col·lisionador del vehicle.

Imatge 5.15



Imatge 5.15: Exemple d'un muntatge de l'*script* que detecta les col·lisions amb les dades especificades.

El codi de l'*script*, juntament amb l'historial de versions, el podem trobar en el següent enllaç: https://github.com/AleixBV/BorrellA_TFG/blob/master/Assets/Scripts/AI_Front_Trigger.cs.

Aquest *script*, en estar al mateix objecte que el col·lisionador frontal amb el booleà "Is Trigger" activat, tal com es pot veure en la imatge anterior, es poden utilitzar unes funcions d'*Unity* que s'executen quan un altre col·lisionador entra, està o surt d'aquest: `OnTriggerEnter(Collider other)`, `OnTriggerStay(Collider other)` i `OnTriggerExit(Collider other)`, respectivament.

Per a detectar i distingir els col·lisionadors dels altres vehicles o del jugador, s'utilitzen les etiquetes del motor. Així, dins de les tres funcions anomenades, primer es comprova si la col·lisió és amb una de les dues etiquetes que es volen controlar. En entrar en col·lisió, únicament es guarda qui s'ha detectat i s'actualitza la velocitat màxima per a aconseguir que el vehicle amb intel·ligència artificial redueixi a una velocitat inferior. En mantenir-se dins del col·lisionador, si el vehicle no està evitant un obstacle, es segueix actualitzant la velocitat màxima o inclús frenant el vehicle en cas de detectar massa proximitat. A més, en cas de verificar que dues intel·ligències artificials es detecten entre si, procedeixen a activar temporalment l'evasió automàtica, ja que hi ha la possibilitat que es trobin i detectin entre si en un encreuament. En sortir del col·lisionador, es deixa de guardar aquesta col·lisió i es

recupera la velocitat normal. Llavors, en la funció *FixedUpdate()*, es comprova si la detecció és amb el jugador, i després d'uns segons, es fa sonar la botzina del vehicle.

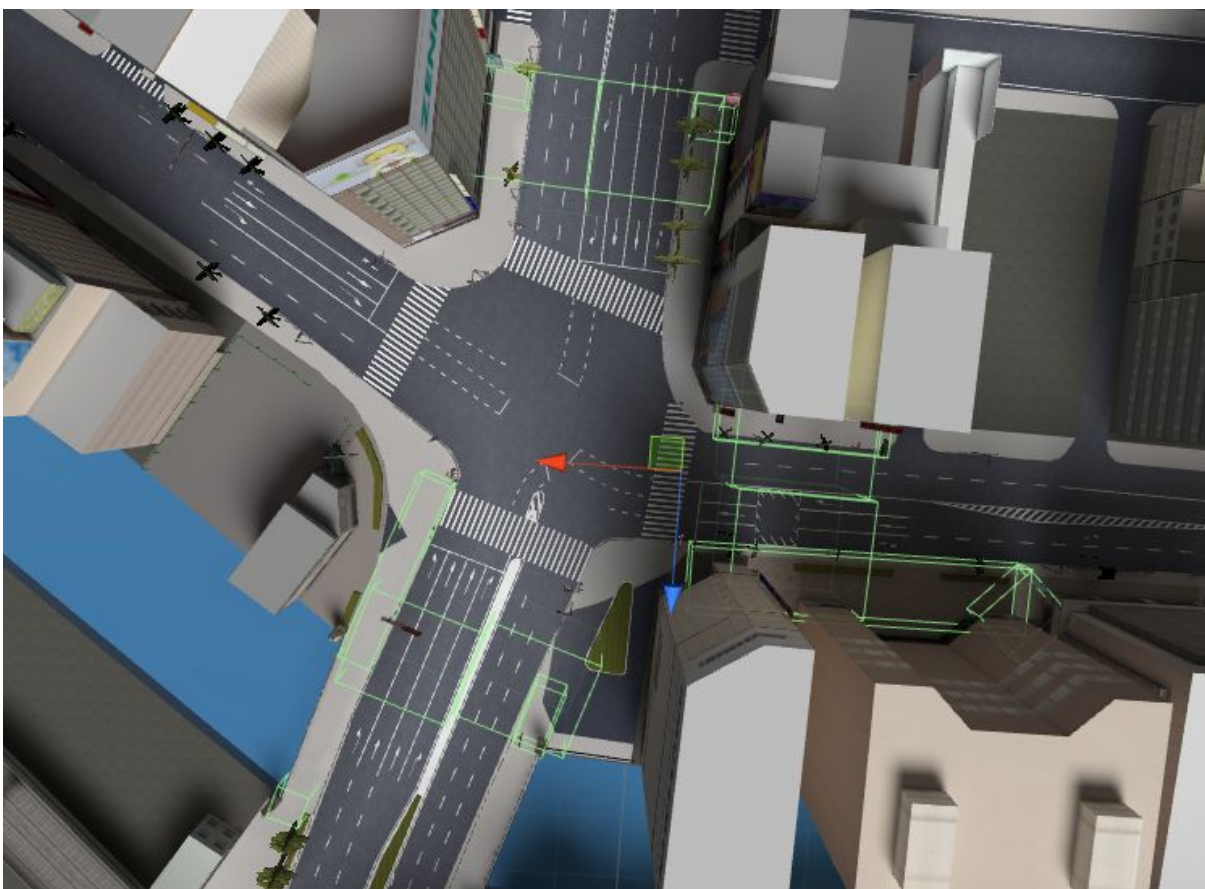
5.2.3.2 Cinemàtica final

La naturalesa de la cinemàtica que es vol que aconseguir, és donar la llibertat al jugador d'arribar a la intersecció per qualsevol de les tres carreteres restants, amb velocitat i direcció lliure. A més, tenint en compte que ha d'estar orientat a realitat virtual, on és important aprofitar la immersió. Per això, crear una animació o utilitzar la navegació automàtica, no s'adequa a les necessitats, ja que l'objectiu és que el jugador visqui la cinemàtica en primera persona i ho senti real.

Per aquestes raons, la millor opció és aconseguir el resultat manualment, controlant el vehicle a través d'un *script* en el moment en què el jugador s'apropi a la intersecció. D'aquesta manera la cinemàtica serà completament immersiva on ens assegurem que el jugador perd el control del vehicle però encara té la llibertat de moure el cap per a veure al seu voltant utilitzant el dispositiu de realitat virtual.

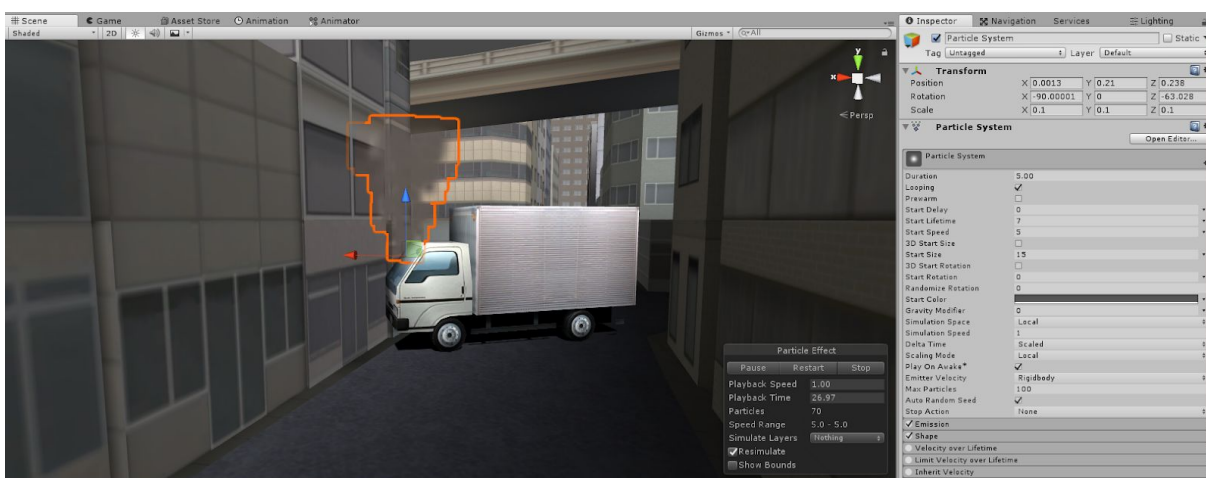
Per a començar, s'han col·locat els disparadors de la cinemàtica, així com col·lisionadors per a evitar que el jugador hi entri per on no es desitja, amagant-los amb elements com cartells o pancartes. Llavors, per a acabar la preparació, a causa del fet que hi ha un petit carrer pel qual el jugador podria evitar passar per la intersecció, s'hi ha afegit un camió simulant un accident, on amb un sistema de partícules propi d'*Unity* s'hi ha simulat el fum sortint del capó.

Imatge 5.16



Imatge 5.16: Col·locació dels disparadors i col·lisionadors per a la cinemàtica.

Imatge 5.17



Imatge 5.17: Col·locació del vehicle amb el sistema de partícules.

Un cop l'escenari preparat, s'hi han afegit dos grups de vehicles, en què creuen la intersecció horitzontalment o verticalment, segons per quina intersecció arriba el jugador. Aquests vehicles amb una intel·ligència artificial simplificada, on segueixen una petita llista de destins i, en acabar, reapareixen en el seu inici per a repetir el recorregut. Aconseguint com a resultat que, en la intersecció, el jugador es trobarà un seguit de vehicles infinits que el creuen perpendicularment a la seva direcció. A més, aquests vehicles contenen un col·lisionador en què en detectar el jugador, activaran el so de xoc entre vehicles juntament amb una transició a color blanc per a carregar el menú d'inici finalitzant el prototip.

Imatge 5.18



Imatge 5.18: Resultat de la cinemàtica on veiem els vehicles creuant la intersecció i el recorregut del GPS fins al destí.

Finalment, en l'*script* que controla la cinemàtica, és on es sobreescrirà els controls del jugador, calculant a partir de la intersecció d'entrada, l'angle restant per aconseguir la direcció al mig de la carretera, i així obtenir el gir de volant desitjat. També s'ha creat un sistema per a rectificar en cas de col·lisió invertint la direcció i activant la marxa enrere. D'aquesta manera, s'assegura l'arribada del jugador a la intersecció encara que entri en la cinemàtica a punt de xocar amb una vorera. A més, també es revisa la direcció del vehicle per a tenir en compte inclús la possibilitat que el jugador entri en la cinemàtica conduint el vehicle cap enrere.

5.3 Postproducció

Després de l'última "milestone" i gràcies al testeig fet en cerca d'errors per tant el director del treball com l'alumne, s'ha procedit a fer millores al prototip i moltes correccions d'errors.

Algunes de les correccions són sobretot en la cinemàtica i la intel·ligència artificial, però per exemple, també s'ha corregit el càlcul del GPS per a assegurar que es recalculi sempre el recorregut i la vibració per aconseguir que s'aturi quan l'usuari no es centri en el prototip i així poder activar que l'aplicació també s'aturi quan l'usuari el minimitzi o canviï de finestra.

Pel que fa a les millores, s'ha afegit diversos inicis aleatoris per al jugador, incentivant així una possible rejugabilitat del prototip. També s'ha millorat tota la intel·ligència artificial, on juntament amb el col·lisionador frontal ja explicat per a evitar col·lisions, s'ha afegit un sistema per a comprovar si el jugador està en un semàfor vermell o darrere un vehicle aturat en aquest. D'aquesta forma s'evita que la intel·ligència artificial faci sonar la botzina i intenti evitar al jugador en aquests casos.

6. Conclusions i treballs futurs

Durant el desenvolupament del treball, s'ha pogut observar que l'ús de les noves tecnologies, com són els dispositius de realitat virtual, està molt facilitat tant pel distribuïdor, facilitant les eines necessàries, com pels motors i la comunitat, ajudant en la pràctica. Per exemple, per a poder utilitzar les *Oculus Rift*, solament és necessari instal·lar l'aplicació pròpia, la qual ja inclou els elements necessaris. A l'hora de desenvolupar, un cop funcionant el dispositiu en l'equip, únicament és necessari en la configuració d'un motor com *Unity*, activar el desenvolupament per a realitat virtual i l'SDK adequat.

Aquesta preparació és extremadament ja senzilla i facilitada per tots els actors, així i tot, si tenim a disposició una de les primeres versions com el primer equip de desenvolupament d'*Oculus Rift*, aquest procés es complica pel fet que ja està obsolet i no en mantenen el suport. Per això, s'ha observat que treballant amb tecnologies innovadores, el desenvolupador s'ha de mantenir constantment actualitzat, en aquest cas tant de software com de hardware.

En ja tenir-ho tot preparat per al desenvolupament, llavors ja es troben les diferències entre un desenvolupament estàndard i un orientat a realitat virtual. S'han de tenir en compte els temes que s'han investigat i posat en pràctica durant el desenvolupament del prototip, com tenir en compte la llibertat de moviment de l'usuari creant uns menús adequats, el canvi que farà el motor a la càmera i llavors quins altres canvis tindrà al seu abast el desenvolupador sobre aquesta. Finalment, el que possiblement és el tema més important, evitar el mareig, on per aconseguir-ho és indispensable una perfecta optimització juntament amb uns controls adequats, així com tenir en compte el sistema vestibular.

Un cop acabat aquest treball de fi de grau sobre el disseny i prototipatge d'un videojoc per a realitat virtual, resten una gran quantitat de vies de millora. Per exemple, el producte es podria millorar tenint una major quantitat de temps i desenvolupadors, aconseguint afegir més contingut i millor acabat, així com realitzant un testeig més exhaustiu amb una anàlisi posterior per experts. Per evitar el mareig, sempre es pot seguir optimitzant, i per a reduir el problema que suposa el sistema vestibular hi hauria la possibilitat de simular el vehicle amb una cadira hàptica o un simulador de cabina. D'aquesta manera es podria aconseguir un producte totalment acabat i inclús preparat per a sortir al mercat.

7. Bibliografia

Pàgina web amb ofertes de treball on tenen informació sobre la mitjana salarial a Espanya.

[En línia] <https://www.indeed.es>

Pàgina web d'*Unity* on es poden comprar i publicar recursos, també n'hi ha de gratuïts.

[En línia] <https://assetstore.unity.com>

Pàgines web d'*Unity* on tenen tutorials, documentació i qüestions preguntades i resoltes per usuaris sobre temes relacionats en l'ús del motor.

[En línia] <https://unity3d.com/es/learn/tutorials>

[En línia] <https://docs.unity3d.com/es/current/Manual/index.html>

[En línia] <https://answers.unity.com>

Pàgina web amb definicions de termes tecnològics, en anglès.

[En línia] <https://techterms.com>

8. Descàrrega del prototip

Un executable de cada "milestone", juntament amb la possibilitat de descarregar-ne el codi inclòs, està disponible entrant en el següent enllaç.

https://github.com/AleixBV/BorrellA_TFG/releases

9. Annexos

9.1 GDD: document de disseny

Prototip en Realitat Virtual sobre Conducció GDD

Treball de Fi de Grau

Aleix Borrell Vives

Introducció

Amb la finalitat de resoldre, amb un estil de joc diferent, el problema per als jugadors “hardcore” de no poder estar varies hores immers en un joc que els exigeixi un esforç físic. Com ja passava amb consoles com la Wii o dispositius com el Kinect, la solució era fer els controls compatibles per a jugar amb els comandaments assegut sense necessitat de moviment.

En el cas de la Realitat Virtual, pel fet que les ulleres són indispensables i no aprofitar-ne la jugabilitat i immersió és un malbaratament, l'objectiu d'aquest disseny és aconseguir fer un joc immersiu per a Realitat Virtual, on aprofitant-ne les mecàniques no requereixi un esforç físic per l'usuari, el qual pot jugar-lo assegut.

Per això, el prototip es basarà en una experiència de conducció, on l'usuari amb el comandament conduirà el vehicle i, amb la jugabilitat de les ulleres de Realitat Virtual, podrà mirar al seu voltant així com als retrovisors. Aconseguint una immersió realista de conducció amb un gameplay més arcade allunyant-se d'un simulador.

Argument

El prototip i demo comença amb el jugador en un vehicle i amb un objectiu que veu en el seu GPS. En estar a prop del seu objectiu, el jugador trobarà un semàfor vermell i en intentar frenar veurà els frens bloquejats, tenint un accident en l'encreuament i finalitzant així el prototip/demo.

Però, realment, el personatge està fent memòria del que va succeir llavors, ja que va tenir l'accident poc abans d'arribar al seu objectiu.

Idees addicionals i possibilitats.

- Sense GPS o amb un objectiu poc clar, i en cert temps de joc activar que succeeixi l'accident en algun semàfor que creua el jugador.
- El personatge pot ser un detectiu que recorda el que va succeir per a posteriorment buscar el culpable de l'accident, ja que diverses pistes indiquen que va ser intencionat.
- Introducció d'un joc sobre gestió d'una empresa de transports / taxi, on el jugador comença amb vehicle d'alta gamma amb màximes prestacions i acaba perdent-lo en l'accident per a començar des de baix de nou.

Gameplay

Comandament d'Xbox 360:

- Gallet dret: accelerar dependent de la pressió en aquest.
- Gallet esquerre: frenar dependent de la pressió en aquest, així com accelerar marxa enrere.
- Joystick esquerre: navegar en els menús i (cap a la dreta i esquerra) moure el volant dependent de la pressió en aquest.
- Joystick dret: moure la càmera per a mirar al voltant si no està activat el dispositiu de RV.
- Botó Y: fre de mà.
- Botó Start: Obrir i tancar el menú.
- Botó Select: Engregar i apagar el motor.
- Pitar?
- LLums?

Teclat:

- Moviment: W, A, S i D.
- Fre de mà: Espai.
- Obrir i tancar el menú: Escape.
- Engregar i apagar el motor: F.

