

FisEout

ALVARO MARTINEZ ARROYO

DANIEL GARCIA ROMERO



ÍNDEX

1. Descripció	02
2. Part comuna	12
3. Funcionalitats	14
4. Treball en equip	17
5. Bibliografia	22

1. Descripció

Hem desenvolupat un videojoc de carreres de naus de caire futurista que es caracteritza per l'anti-gravetat d'aquestes. Ens hem basat en dues sagas de videojocs de característiques molt similars: *Wipeout* i *F-Zero*.

La mecànica del joc, doncs, és ben senzilla: el jugador controlarà una nau i haurà de competir amb tres naus enemigues en una cursa que guanyarà aquella nau que completi tres voltes abans que les altres.

1.1 Naus seleccionables

- *FEISAR*



Figura 1. *FEISAR F7200 (Wipeout 3)*

- *ASSEGA*



Figura 2. *ASSEGA F7200 (Wipeout 3)*

Establint una comparativa entre les dues naus seleccionables, podem dir que *FEISAR* té una resistència als míssils enemics superior i disposa de més míssils a l'inici de la cursa (tres míssils un, l'altre només un), mentre que *ASSEGAI* destaca per la seva velocitat màxima i velocitat de gir.









CARACTERÍSTICA	NAU RECOMENADA
 Velocitat màxima	 <i>ASSEGAI</i>
 Velocitat de gir	 <i>ASSEGAI</i>
 Nombre inicial de míssils	 <i>FEISAR</i>
 Resistència als míssils	 <i>FEISAR</i>

Figura 3. Comparativa entre naus seleccionables

1.2 Pistes seleccionables

- *TROY CIRCUIT*



Figura 4. *TROY CIRCUIT*

- *LAMBDA CIRCUIT*

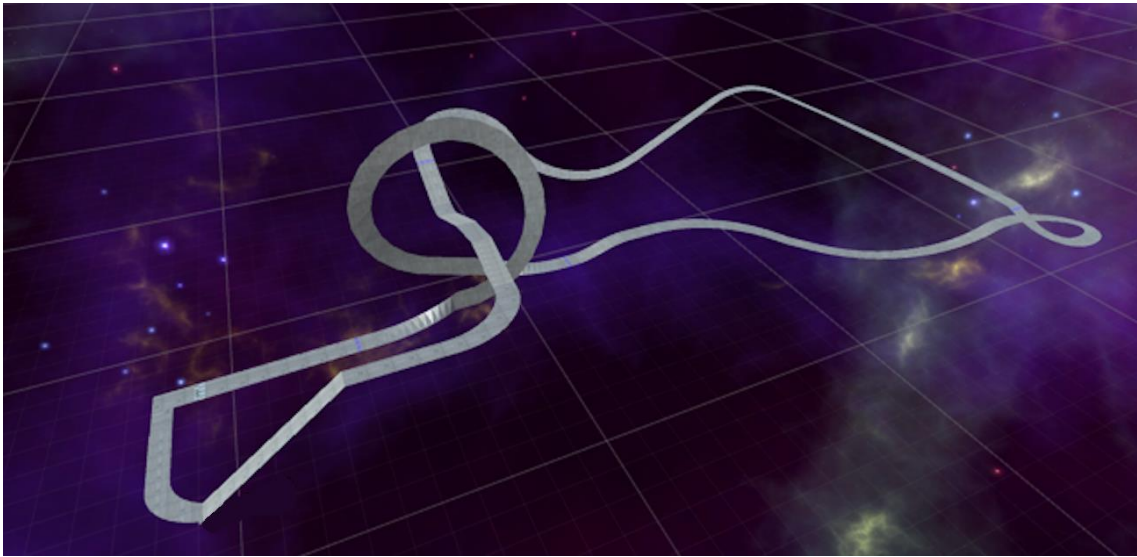


Figura 5. *LAMBDA CIRCUIT*

Les curses sobre *Troy Circuit* tenen lloc a la Terra i l'ambientació és la d'un dia solejat, mentre que les curses sobre *Lambda Circuit* es produeixen a l'espai.

Troy Circuit es caracteritza per tenir rectes llargues que faciliten el llançament i col·lisió de míssils contra els enemics. Disposa d'un nombre superior de bases de velocitat i comparteix nombre de bases recarregadores de míssils amb *Lambda Circuit*.

Lambda Circuit té més corbes i, per tant, triar *ASSEGAI* en aquesta pista seria una bona opció perquè és la nau seleccionable amb major velocitat de gir.

Ambdues pistes tenen els mateixos tipus de bases, que són els següents:

- Acceleradors. Quan una nau passi per sobre d'un d'ells, se l'incrementarà progressivament la velocitat fins a arribar a una velocitat límit i superior a la velocitat màxima de la nau.
- Recarregadors de míssils. Quan una nau passi per sobre d'un d'ells, s'incrementarà en una unitat el seu nombre de míssils.



Figura 6. Els dos tipus de bases a *Troy Circuit*



Figura 7. Els dos tipus de bases a *Lambda Circuit*

1.3 Naus enemigues

Totes les naus enemigues d'una mateixa pista tenen característiques comunes: mateixa velocitat màxima, velocitat de gir, nombre inicial de míssils (tres en ambdues pistes) i resistència a aquests. En relació al player, tenen característiques molt semblants per tal de garantir la dificultat del joc.

TROY CIRCUIT

- *ICARAS*



Figura 8. *ICARAS SUNGOD XVI (Wipeout 3)*

- *PIRHANA*



Figura 9. *PIRHANA F7200 (Wipeout 3)* amb textura verda

- AG-SYS

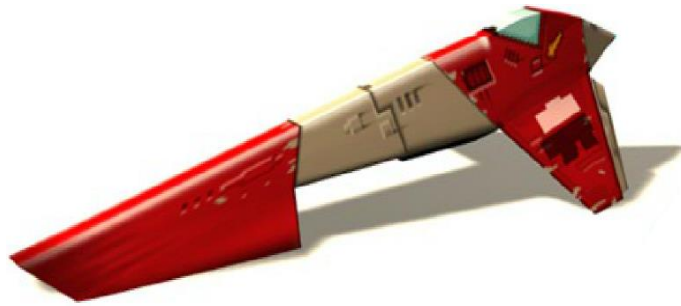


Figura 10. AG SYSTEMS F7200 (Wipeout 3)

LAMBDA CIRCUIT

- QIREX



Figura 11. QIREX F7200 (Wipeout 3)

- GOTEKI



Figura 12. GOTEKI 45 F7200 (Wipeout 3)

- *AURICOM*



Figura 13. *AURICOM F7200 (Wipeout 3)*

1.4 Armes disponibles

Tant la nau que controla el jugador com les naus enemigues poden llançar míssils per tal que aquests col·lisionin amb altres naus i es redueixi la velocitat d'aquestes com a conseqüència de l'impacte. La direcció dels míssils ve determinada per l'orientació de la nau, és a dir, per la direcció cap on mira aquesta. Els míssils es mouran en línia recta fins a col·lisionar amb una nau enemiga, amb el terra o amb les tanques, moment en el qual explotaran. En cas que passin deu segons i no hagin col·lisionat amb cap dels elements esmenats anteriorment, explotaran automàticament.

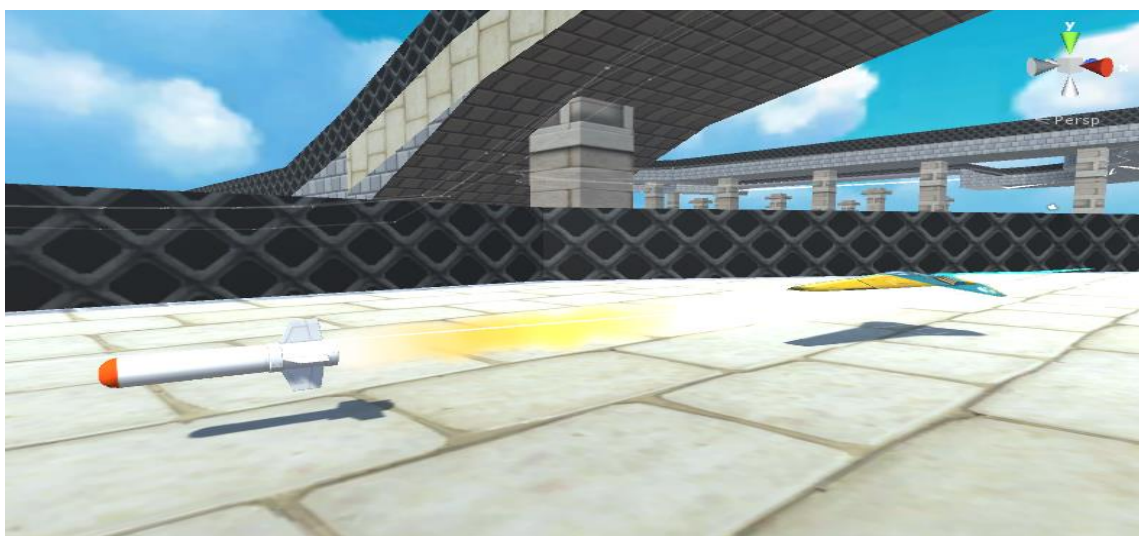


Figura 14. Llançament d'un míssil

1.5 Controls

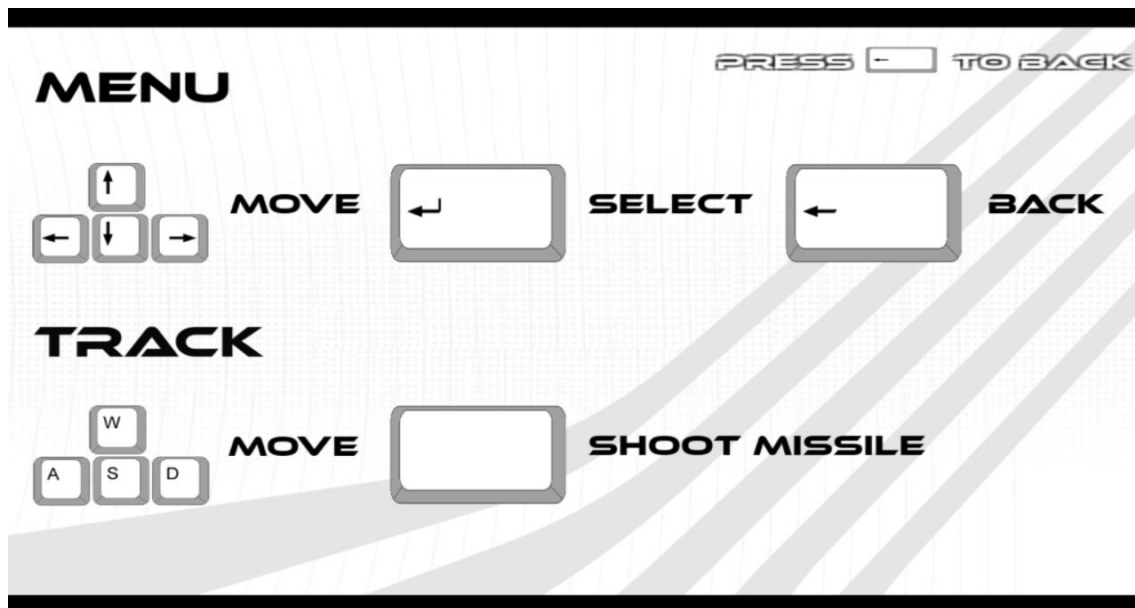


Figura 15. Controls

1.6 Escenes

- Menú principal. És la pantalla que veiem quan obrim el joc. Des d'aquí podrem anar a les pantalles de Seleccionar pista, Controls i Crèdits i també tindrem la opció de sortir del joc.



Figura 16. Menú principal

- Controls. En aquesta pantalla (figura 15) apareixeran les tecles que hem de polsar per tal de poder jugar i moure'ns a través de les diferents escenes.
- Crèdits. Aquí trobarem els noms i cognoms dels desenvolupadors.



Figura 17. Crèdits

- Seleccionar pista. En aquesta pantalla podrem escollir sobre quina pista volem realitzar la cursa un cop analitzada la informació que se'ns proporciona de cadascuna d'elles.

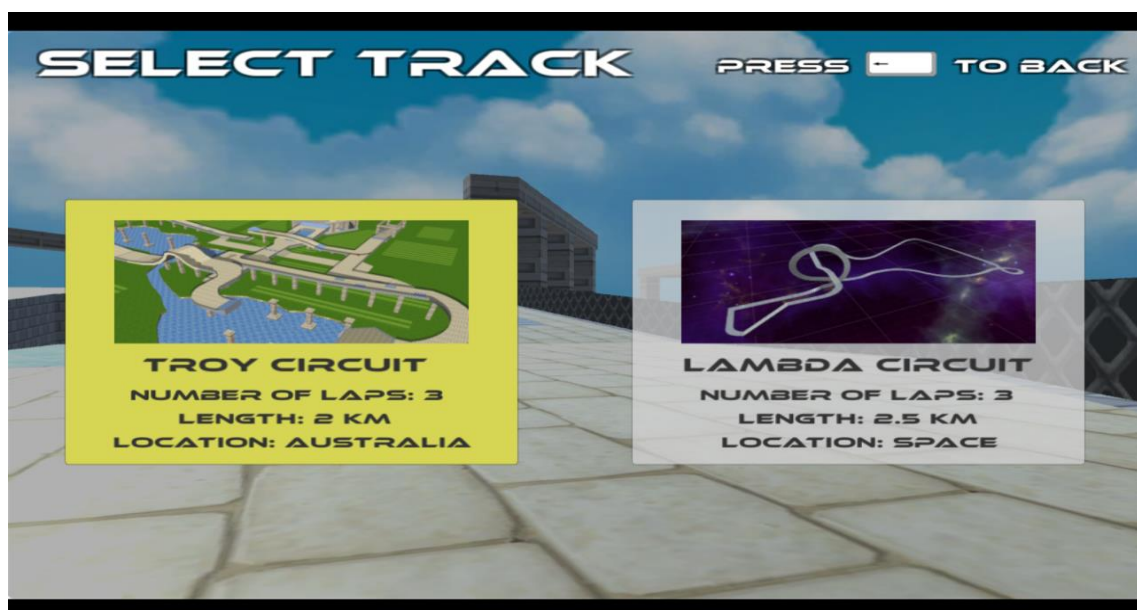


Figura 18. Seleccionar pista

- Seleccionar nau. Aquesta és l'última pantalla prèvia a començar una cursa. Anàlogament a la pantalla anterior, veurem les característiques de les naus de les quals hem parlat a l'apartat 1.1 d'aquesta memòria i n'escollirem una.

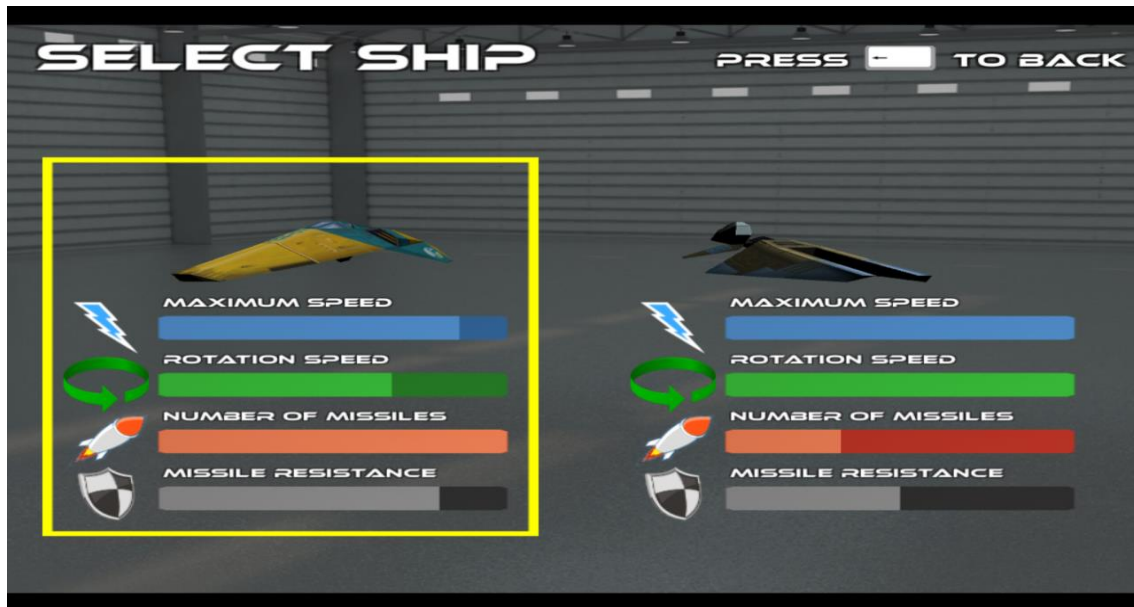


Figura 19. Seleccionar nau

- Cursa. En aquesta pantalla controlarem la nau prèviament seleccionada sobre la pista que haguem escollit i competirem amb tres naus enemigues, l'aparença i característiques de les quals dependrà de la pista, com ja hem comentat anteriorment.



Figura 20. Inici d'una cursa en *Troy Circuit*

2. Part comuna

MENÚ

- Tenim dues naus seleccionables, *FEISAR* i *ASSEGAI*, amb models i característiques ben distintes (apartat 1.1).

Implementació: Per a la comunicació entre escenes i poder saber en tot moment quina nau hem seleccionat, utilitzem PlayerPrefs amb la paraula clau nau.

- Tenim dues pistes seleccionables prou diferenciades l'una de l'altra (apartat 1.2).

Implementació: De manera anàloga a la implementació de seleccionar nau.

CARRERA

- Moviment i rotació de la nau del player.

Implementació: Per tal de mantenir-nos a una alçada constant del terra, utilitzem Raycast. El moviment de la nau consisteix en fer transform.Translate amb un valor d'acceleració que va canviant multiplicat pel deltaTime. Pel que fa a la rotació, utilitzem transform.Rotate sobre els eixos y i z per no només girar la nau si no també per donar-li una sensació de realisme.

- Col·lisions.

Implementació: Per tal de detectar col·lisions amb enemics, tanques i terra, utilitzem OnCollisionEnter i les naus tenen un Rigidbody i un BoxCollider i les tanques i el terra un MeshCollider.

- Competim amb tres naus enemigues, l'aparença i característiques de les quals dependrà de la pista seleccionada (apartat 1.3).

Implementació: Carreguem uns models o uns altres en funció del valor de PlayerPrefs amb la paraula clau mapa.

- Moviment i rotació de les naus enemigues.

Implementació: Es mouen per la pista seguint uns Waypoints propis de cada nau amb una velocitat constant i similar a la velocitat màxima del player. Quant a la rotació, calculem la rotació desitjada amb una crida a `Quaternion.LookRotation` tenint en compte el següent Waypoint i la posició de la nau i modifiquem la rotació d'aquesta mitjançant `Quaternion.Slerp`, al qual l'indiquem la rotació actual de la nau, la rotació desitjada a la qual volem arribar i en un temps `deltaTime` multiplicat per una velocitat de rotació constant que li assignem nosaltres.

- Podem disparar míssils a les naus enemigues i viceversa (apartat 1.4).

Implementació: El player pot llançar míssils sempre i quan en tingui de disponibles. Pel que fa als enemics, només instanciaran míssils si detecten mitjançant Raycast l'enemic (figura 21) a una distància ni molt propera ni molt llunyana, si li queden míssils i si fa deu segons o més des que va llançar el darrer míssil.

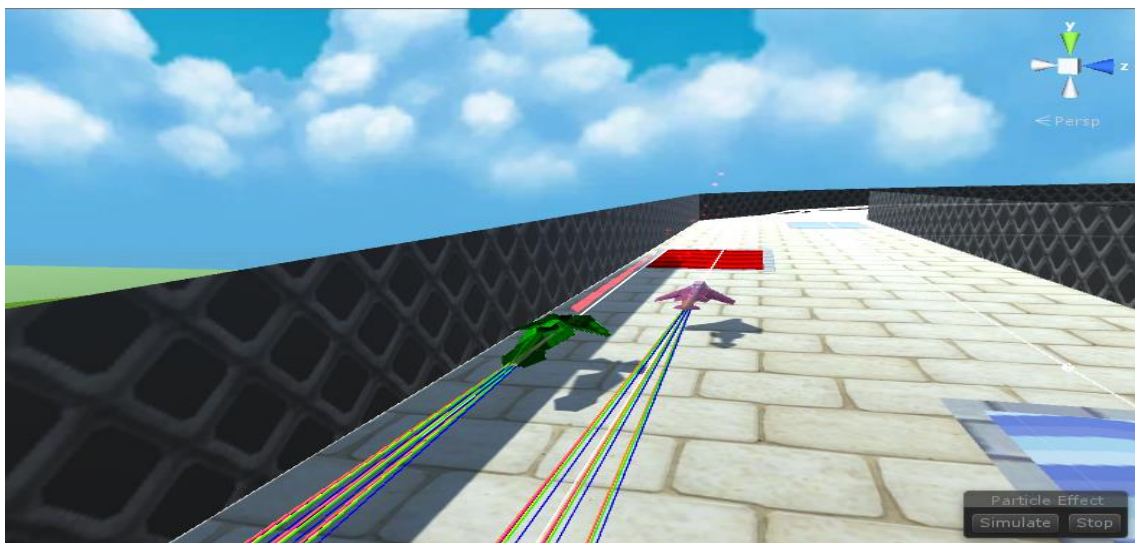


Figura 21. Raycast de les naus enemigues

- Donem la possibilitat al player de passar per sobre d'unes bases que li proporcionen avantatges (apartat 1.2).

Implementació: Per detectar quan passem per sobre d'una base, ho fem mitjançant Raycast i els tags dels `GameObjects` que conformen la pista, i en funció d'aquesta base, incrementem la velocitat o el nombre de míssils de la nau.

3. Funcionalitats

- Sistemes de partícules. Malgrat que els *Meshes* (3D) i *Sprites* (2D) són la manera ideal de representar a Unity objectes sòlids amb una figura ben definida, hi ha altres entitats en els jocs que són fluïdes i intangibles en la naturalesa i, per tant, són difícil de representar utilitzant *Meshes* o *Sprites*. Per a efectes com a líquids en moviment, fum, núvols, flames i encanteris màgics, s'utilitzen sistemes de partícules.

Implementació: Hem utilitzat sistemes de partícules per representar que la nau del player està encesa (a través d'una petita flama al seu tub d'escapament), que el míssil està en moviment (mitjançant una petita flama similar a la de la nau) i que explota (mitjançant un efecte de foc acompanyat per un de fum) i també per donar una animació vistosa a les bases de recarregar míssils. Per crear aquests efectes, hem afegit components de sistemes de partícules de Unity i els hem modificat al Inspector fins a obtenir el resultat desitjat.

- TrailRenderer. Deixa una estela darrere el GameObject al qual està assignat quan aquest es mou.

Implementació: Hem utilitzat un TrailRenderer per cada nau i ho hem fet mitjançant el component TrailRenderer de Unity i modificant els seus valors al Inspector fins a obtenir el resultat desitjat.

- Skybox. És una eina que proporciona Unity per poder posar una imatge de fons a una escena.

Implementació: Hem utilitzat dos skybox, un per cada pista. Per tal d'assignar-li una imatge de fons a una escena, arrosseguem el material corresponent a l'escena.

- So. Els sounds effects es reproduïxen via script mitjançant la crida a PlayClipAtPoint, mentre que la música de fons del menú i de cada pista requereix ser arrossegada a un component AudioSource d'un GameObject de l'escena i hem de marcar la opció Loop.

Per tal d'esperar a que determinats sfx acabin abans que comencin d'altres, hem utilitzat contadors i booleans, i per poder mantenir la mateixa cançó malgrat anar canviant d'escenes del menú, hem hagut d'assignar-li un script al GameObject que conté l'AudioSource per a que aquest no sigui destruït al carregar una nova escena (DontDestroyOnLoad(gameObject)) i només fer un audio.Stop() si l'escena actual no és cap del menú, és a dir, si l'escena actual és la de la pista 1 o la de la pista 2.

- Evitar que el player faci trampa creuant la línia de meta sense haver fet la volta sencera (no se li sumarà una volta més i no afectarà a la seva posició actual en la cursa).

Implementació: Mitjançant triggers repartits estratègicament per la pista.

- UI Cursa. Abans que comenci una cursa, es mostrarà el clàssic 3,2,1 ... GO! i les naus no podran moure's. Durant el transcurs de la cursa, podem veure la posició del player (primer, segon, tercer o quart), el temps transcorregut, en quina volta es troba i la velocitat que duu la seva nau. Un cop el player travessi la línia de meta en la seva última volta, apareixerà a la pantalla un You win! si queda primer i, en cas contrari, un You lose!, i posteriorment un ranking amb la posició, el nom i una imatge representativa de cada nau participant.



Figura 22. Final de cursa amb victòria en *Troy Circuit*



Figura 23. Final de cursa amb derrota en *Lambda Circuit*

Implementació: Durant el temps que dura el 3,2,1 ... GO!, deshabilitem els scripts associats a les naus i després els activem. Per calcular la posició de cada nau en un instant qualsevol, mirem la seva volta actual, el Waypoint al qual es dirigeix i la distància entre la nau i aquest Waypoint, i ho comparem amb la resta de naus. Per saber la volta actual de cada nau, hem utilitzat triggers repartits estratègicament per la pista.

- UI Seleccionar pista. En funció de la pista seleccionada, activem una càmera o una altra que va donant voltes per la pista en qüestió i ens mostra una perspectiva similar a la que veuria el player.
- UI Seleccionar nau. En funció de la nau seleccionada, rotem una o l'altra.
- God Mode. Si en qualsevol moment de la cursa polsem la tecla P, tornem al menú principal.

4. Treball en equip

4.1 Metodologia de treball

Vam optar pel *pair programming* perquè ja ens havia anat bé així en altres projectes que hem hagut de fer junts al llarg de la carrera i també per la disponibilitat horària dels dos, ja que estem matriculats gairebé de les mateixes assignatures i som de la mateixa especialitat. Tenint en compte això i el fet que cap dels dos treballem encara i que aquest quadrimestre l'especialitat d'enginyeria del software té horari de tardes, hem quedat tots els matins de dilluns a divendres a la FIB exceptuant aquells dies en els quals hem hagut de dedicar hores a altres assignatures. Els caps de setmana, malgrat que no ens hem reunit físicament, hem estat en contacte via Facebook i pràcticament tots els diumenges hem fet videoconferències a través de Skype.

Els avantatges que hem trobat en relació a altres companys de l'assignatura que es repartien la feina i només es veien els dies de laboratori són els següents:

- El compromís amb el projecte i el respecte envers el company estan garantits pel fet de quedar quasi tots els dies i estar sempre en contacte. D'aquesta manera evitem el mal ambient que es genera en cas de repartir tasques i que un dels dos o bé no les acabi en el temps estimat per manca de dedicació o bé generi dubtes sobre si ha fet ell el codi, si l'ha entès almenys o si directament l'ha copiat d'Internet o d'algú altre.
- En persona és més fàcil i ràpid argumentar, posar-nos d'acord en temes com, per exemple, el disseny de les naus i de les pistes.
- Evitem problemes d'ajuntar en un sol projecte de Unity el que ha fet cadascú per separat.

- Dos ulls veuen més que un i dos caps pensen més que un, o dit d'una altra manera, el fet d'ajudar-nos entre nosaltres fa que sigui més senzill detectar errors i fer millors codis.

Pel que fa al control de les tasques realitzades, tasques pendents, hores de dedicació, bibliografia, etc., hem utilitzat Google Drive, i quant a les còpies que hem anat generant de les diferents versions del projecte, les hem pujat a Mega i a Dropbox i, per si de cas, també les hem guardat als nostres pendrives.

4.2 Tasques realitzades

Tasca	Temps de dedicació
1. Informar-nos sobre les mecàniques bàsiques del Wipeout	2 hores
2. Buscar models de naus pel player	2 hores
3. Aprenentatge bàsic de Unity	6 hores
4. Buscar models de circuits i importació del model d'un circuit	2 hores
5. Fer que la nau es mogui i detecti col·lisions dins del circuit	4 hores
6. Aprenentatge i provatures del Raycast amb un cub dins d'un terrain	4 hores
7. Provatures del Raycast aplicat a la nau dins del circuit	4 hores

8. Modificació del circuit (construcció de rampes i tanques) i afegir tags a les peces del circuit	4 hores
9. Acceleradors, sistemes de partícules pels motors	3 hores
10. Buscar models de naus enemigues	1 hora
11. TrailRenderer pel deixant (estela) de la nau	2 hores
12. Evitar que la càmera surti del circuit	2 hores
13. Waypoints per la IA de les naus enemigues	10 hores
14. TrailRenderer i acceleració de les naus enemigues	4 hores
15. Millorar el moviment de la nau del player	1 hora
16. Comptador de voltes i comptador regressiu 3,2,1...GO! abans de començar una cursa	4 hores
17. Càlcul de la posició (1r, 2n, 3è, 4t) de cada nau en qualsevol instant d'una cursa	2 hores
18. Checkpoints per saber la posició del player	8 hores
19. Evitar que l'usuari faci trampa creuant la línia de meta sense haver fet la volta sencera	1 hora
20. Implementació i disseny d'un velocímetre	2 hores
21. UI Cursa (mostrar comptador regressiu, voltes, posició, temps i el velocímetre)	3 hores

22. UI del final d'una cursa (You win/You Lose, temps final parpellejant i ranking amb posició, nom i imatge de cada nau)	4 hores
23. Menú principal, pantalla de seleccionar pista i pantalla de seleccionar nau	6 hores
24. Disseny mapa 2	12 hores
25. Míssil player i enemics	12 hores
26. Replicar les funcionalitats del mapa 1 al mapa 2	4 hores
27. So	6 hores
28. Memòria	10 hores
29. Refer el UI (canvas) perquè s'ajusti a qualsevol resolució de pantalla	6 hores
30. Detecció i solució de bugs	2 hores
31. Video	3 hores

Figura 25. Tasques realitzades

4.3 Diagrama de Gantt

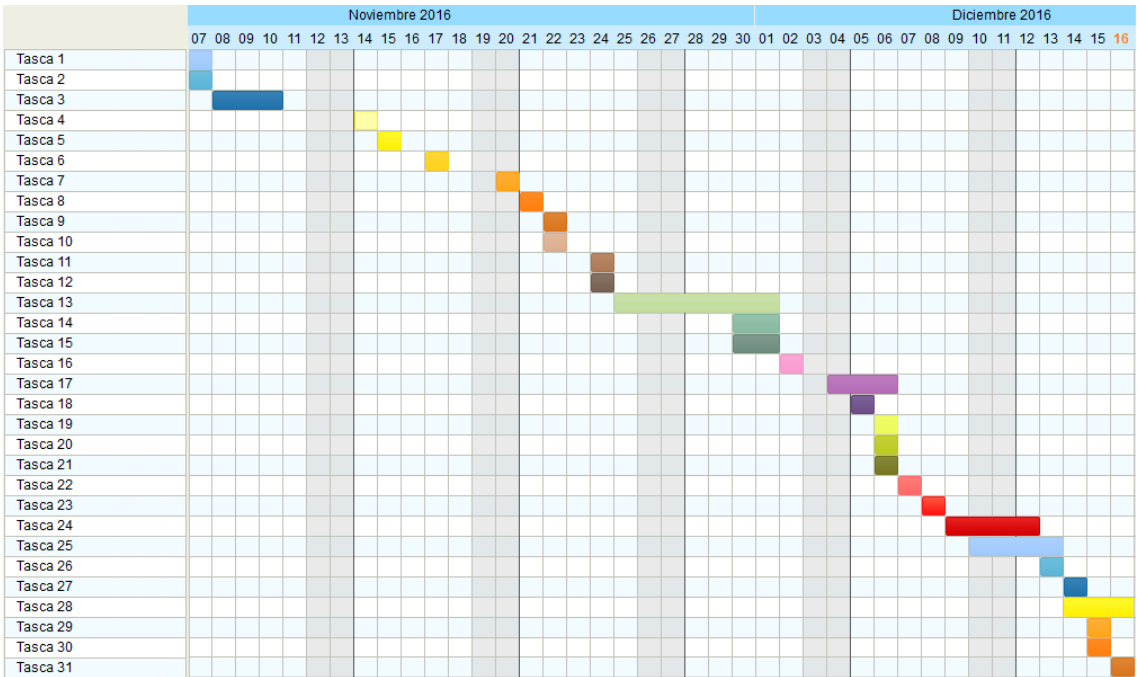


Figura 26. Diagrama de Gantt

5. Bibliografia

- **Documentació oficial de Unity**

<https://unity3d.com/es/learn/tutorials>

- **Unity Answers**

<http://answers.unity3d.com>

- **Tutorials de YouTube**

<https://www.youtube.com>

- **Models 3D**

- **Pista 1 i peces pista 2**

<https://3dwarehouse.sketchup.com>

- **Naus**

<http://steamcommunity.com/sharedfiles/filedetails/?id=714072377>

- **Míssil**

<https://www.assetstore.unity3d.com/en/#!/content/72692>

- **Fonts**

<http://www.dafont.com/es>

http://www.mariomayhem.com/downloads/super_mario_fonts

- **Wiki del Wipeout 3**

http://wipeout.wikia.com/wiki/Wipeout_3

- **SFX del F-Zero**

<https://www.youtube.com/watch?v=mvOfBsZp9wc>

- **OST del Wipeout**

<https://www.youtube.com/watch?v=V974MkN7v94>

<https://www.youtube.com/watch?v=4uQnXvRndcE>

<https://www.youtube.com/watch?v=tVHAEB1REBY>