

Creació d'un videojoc arcade. NeoTrooper.

Luis M. Rosillo Expósito
Grau d'Enginyeria Informàtica
Àrea de Videojocs

Joel Servitja Feu
Javier Luis Cánovas Izquierdo

Juny 2.019



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FITXA DEL TREBALL FINAL

| | |
|--------------------------------------|--|
| Títol del treball: | <i>Creació d'un videojoc arcade. NeoTrooper.</i> |
| Nom de l'autor: | <i>Luis Miguel Rosillo Expósito</i> |
| Nom del consultor/a: | <i>Joel Servitja Feu</i> |
| Nom del PRA: | <i>Javier Luis Cánovas Izquierdo</i> |
| Data de lliurament (mm/aaaa): | <i>06/2019</i> |
| Titulació o programa: | <i>Grau d'Enginyeria Informàtica (Itinerari en Tecnologies de la informació)</i> |
| Àrea del Treball Final: | <i>Videojocs</i> |
| Idioma del treball: | <i>Català</i> |
| Paraules clau | <i>Videojoc, Android, Arcade</i> |

Resum del Treball (màxim 250 paraules): *Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball*

La present memòria de treball de final de grau descriu el procés de disseny, desenvolupament, publicació i proves d'un videojoc arcade realitzat per plataforma mòbil Android. Aquest és un treball de l'àrea de videojocs del Grau d'Enginyeria Informàtica de la Universitat Oberta de Catalunya (UOC). La dedicació estimada és 12 crèdits (300 hores).

Cal remarcar que donada la naturalesa del treball, a banda de les restriccions temporals, existia una restricció econòmica, de forma que s'han utilitzat exclusivament recursos, com gràfics o efectes sonors, gratuïts.

L'esmentat videojoc està basat en el joc clàssic Paratrooper de 1.982, desenvolupat per Greg Kuperberg. El principal entregable, a més de la present documentació, és una aplicació en format .apk que es pot executar en dispositius Android.

La dinàmica de joc és senzilla, l'usuari manega un canó situat a la part inferior de la pantalla, i el defensa dels enemics. A la versió final, NeoTrooper consta de dues pantalles o nivells de joc que emulen els nivells presents a l'aplicació en la qual es basa.

Amb referència a la plataforma de desenvolupament, a l'inici del TFG s'analitzen les alternatives existents, i s'opta per Unity, amb C# com a llenguatge de programació.

La conclusió principal és que els '*game engines*' actuals simplifiquen de forma molt rellevant la vessant tècnica de la creació de videojocs. El creador

es pot concentrar en la part artística o de disseny de la història, contràriament a com es feia abans de l'existència d'aquestes eines.

Abstract (in English, 250 words or less):

This report describes the process of designing, developing, publishing and testing an arcade video game made for Android mobile platform. This is the final project in the videogame specialization area of the Bachelor's Degree in Computer Engineering of the Universitat Oberta de Catalunya (UOC). The estimated effort is 12 credits (300 hours).

Given the nature of the work, in addition to temporary restrictions, there is an economical restriction, so, the additional resources, such as graphs or sound effects, had to be publicly available assets.

The video game is based on the classic 1982 Paratrooper game, developed by Greg Kuperberg. The main deliverable, in addition to this documentation, is an application in .apk format for Android mobile devices.

The game dynamics are simple, the user controls a cannon placed at the bottom of the screen and must defend it from the enemies. In the final version, NeoTrooper consists of two game levels that emulate the levels present in the original videogame.

In regard to the development platform, the available options are analyzed at the beginning of the current document, and Unity, with C # as a programming language is the tool that it is finally used.

The main conclusion of this project is that today's game engines reduce dramatically the technical difficulty of the video game creation process. The creator can focus on the artistic and history design work, contrary to how it was done before the existence of these tools.

Índex

| | |
|--|----|
| 1. Introducció | 1 |
| 1.1 Context i justificació del Treball..... | 1 |
| 1.2 Objectius del Treball..... | 2 |
| 1.3 Enfocament i mètode seguit..... | 2 |
| 1.4 Planificació del Treball | 3 |
| 1.5 Breu sumari de productes obtinguts..... | 5 |
| 1.6 Breu descripció dels altres capítols de la memòria | 5 |
| 2. Estat de l'art..... | 7 |
| 2.1 El videojoc com a indústria..... | 7 |
| 2.2 Tendències..... | 7 |
| 3. Definició del joc..... | 9 |
| 3.1 Descripció i objectius..... | 9 |
| 3.2 Elements del joc..... | 9 |
| 3.3 Interacció de l'usuari amb els elements del joc | 9 |
| 3.4 Puntuació | 10 |
| 3.5 Final de la partida..... | 10 |
| 4. Disseny de nivells | 11 |
| 4.1 Fases del joc | 11 |
| 4.2 Gameplay i nivells del joc..... | 11 |
| 5. Fases del desenvolupament..... | 13 |
| 5.1 Definició i elecció dels elements | 13 |
| 5.2 Creació del projecte Unity | 13 |
| 5.3 Creació dels Gameobjects a Unity | 14 |
| 5.4 Programació..... | 14 |
| 5.5 Publicació..... | 16 |
| 5.6 Proves | 17 |
| 6. Disseny tècnic..... | 21 |
| 6.1 Estructura del projecte d'Unity | 21 |
| 6.2 Elements del joc..... | 22 |
| 7. Manual d'usuari | 27 |
| 8. Conclusions | 32 |
| 9. Glossari..... | 34 |
| 10. Bibliografia i enllaços relacionats | 36 |
| 10.1 Recursos relatius a NeoTrooper | 36 |
| 10.2 Documentació utilitzada:..... | 36 |
| 10.3 Websites: | 38 |
| 10.4 Assets per Unity:..... | 39 |

1. Introducció

1.1 Context i justificació del Treball

Motivació i objectius del TFG

Els videojocs són, per la majoria dels usuaris de noves tecnologies, una de les portes d'accés a la informàtica, donat l'atractiu per tot tipus d'usuaris, especialment pels sectors més joves de la població. Dins de les activitats formatives del grau en Informàtica hi ha poques matèries que permetin aprofundir en el procés de desenvolupament de videojocs. És una àrea amb tècniques, llenguatges i metodologies específiques.

Altrament, des del punt de vista professional, és un camp en plena expansió, com es pot observar a les següents dades:

Segons estimacions de l'AEVI (Asociación Española de Videojuegos), a Espanya el sector dels videojocs va tenir una facturació de 1.177 milions d'Euros a l'any 2016 i 1.350 a l'any 2017, amb un volum d'aproximadament 15 milions d'usuaris. Amb aquestes xifres, l'estat està dins dels 10 productors més importants a nivell mundial.

A l'estat hi ha al voltant de 450 empreses i estudis de desenvolupament registrats i la previsió es que el nombre de treballadors del sector es tripliqui entre els anys 2.015 i 2.020. (11.420 vs. 4.460).

Així doncs, l'objectiu és seguir el procés sencer – tot i que simplificat - de desenvolupament d'un videojoc, utilitzant algunes de les eines pròpies d'aquesta disciplina, per crear un videojoc complet, jugable i idealment amè.

Elecció del joc

Tot i les restriccions de recursos pròpies d'un treball de final de grau com són la dedicació d'un sol dissenyador/desenvolupador en un termini d'un quadrimestre (aprox. 300 hores), es persegueix l'objectiu de crear un producte amb una certa coherència i jugabilitat, conseqüentment, el joc a desenvolupar ha de ser senzill.

Es busca un videojoc amb un entorn gràfic simple, per les mencionades restriccions, i per desconeixement personal de les eines de modelatge i disseny en 2D i 3D.

Per aquestes raons, el present TFG es planteja amb la intenció de crear una versió renovada del videojoc de 1.982 Paratrooper, desenvolupat per Greg Kuperberg. Paratrooper era un shooter que es basava al seu torn en un altre joc previ, Sabotage, de 1.981, desenvolupat per Mark Allen. Durant la primera fase de desenvolupament es realitzarà una nova versió amb el mateix funcionament que el joc original. Per fases posteriors, es proposen alguns canvis, amb elements addicionals, nous modes de joc, etc...

És també l'objectiu retre un humil homenatge a aquest i altres jocs similars que tenint en compte les limitades possibilitats gràfiques dels ordinadors personals i els sistemes operatius disponibles en aquell moment, abans de que es generalitzés l'ús de la informàtica personal en general i els entorns gràfics en particular.

El nom escollit per la nova versió és NeoTrooper.

Plataforma

Per la època en la qual es va desenvolupar els jocs originals (Sabotage, ParaTrooper, etc...) les plataformes disponibles eren ordinadors personals o consoles (PC, Apple, Atari), però tenint en compte les característiques del joc; interacció d'usuari bàsica, entorn gràfic senzill i partides curtes, és un videojoc que s'adapta perfectament a les plataformes mòbils. Per aquest motiu, NeoTrooper estarà dissenyat i desenvolupat per Android.

1.2 Objectius del Treball

L'objectiu principal del projecte és experimentar el procés complert, encara que simplificat, de creació d'un videojoc per a plataformes mòbils Android utilitzant els entorns de desenvolupament que actualment es troben disponibles al mercat.

Com a objectius addicionals, però no menys importants, es poden destacar:

- Anàlisi de l'estat i tendències actuals al sector dels videojocs
- Valorar dels diferents *game engines* disponibles al mercat.
- Utilitzar les metodologies i llenguatges de programació emprades en aquesta àrea.
- Avaluar si es possible realitzar un videojoc plenament funcional amb les restriccions de recursos pròpies d'un TFG.

1.3 Enfocament i mètode seguit

Eina de desenvolupament

A l'inici d'un projecte de desenvolupament d'un videojoc, una de les decisions importants a prendre és si s'utilitzarà un dels productes de programari disponibles per ajudar en aquesta tasca, i en aquest cas, quin es selecciona.

Aquests programes faciliten d'una forma molt notable el procés de creació, eliminant la necessitat de desenvolupar tots els aspectes del joc: gestió total dels elements (moviment, física,...), música i só, entorn gràfic, etc...

Actualment, fins i tot els productors de videojocs comercials en fan us d'aquestes eines, consegüentment no hi ha dubte sobre la qualitat dels jocs que es poden generar.

Amb respecte a les diferents opcions, hi ha diferents criteris que poden ajudar en l'elecció. Com són el cost del producte, les plataformes per les quals es poden desenvolupar els videojocs, la facilitat d'us, la documentació disponible o els llenguatges de programació (si cal utilitzar-los) que es poden fer servir.

Hi ha multitud de productes disponibles al mercat. Alguns dels més populars són:

| Nom producte | Cost llicència (per Android si està suportat) | Llenguatges suportats | Plataformes |
|--------------------|---|-----------------------------|--|
| Quest | Gratuït | N/A | Només mode text |
| Twine | Gratuït | N/A | Només mode text |
| Stencyl | 200 \$ /any | N/A | iOS, Android, Web, Desktop |
| Construct 3 | 90 \$ /any | N/A | iOs, Android, Xbox, Web, Desktop |
| Gamesalad | 25\$ / mes | N/A | iOs, Android, Web, Desktop |
| GameMaker Studio 2 | Depen de les plataformes. 130\$ /any per Android | N/A | iOs, Android, Xbox, PS4, Web, Desktop... |
| Unity | Versió personal gratuïta | C#, Javascript | iOs, Android, Xbox, PS4, Nintendo, Web, Desktop, ... |
| Unreal | Versió personal gratuïta | C++ | iOS, Android, Xbox, PS4, Nintendo, Web, Desktop.... |
| Godot | Gratuït | GDScript (similar a Python) | iOs, Android, Web, Desktop |

En aquest TFG, al no tractar-se d'un producte comercial, és imprescindible que el producte sigui gratuït. Per altra banda, la plataforma de destí escollida és Android.

S'analitza la llista anterior de candidats. Descartant els productes amb llicència per Android no gratuïta, Unity, Unreal i Godot serien els mes apropiats de la llista anterior.

És important ressaltar que Unity té un 48% de quota del mercat de desenvolupament de jocs Android, comparat amb el 13% d'Unreal.

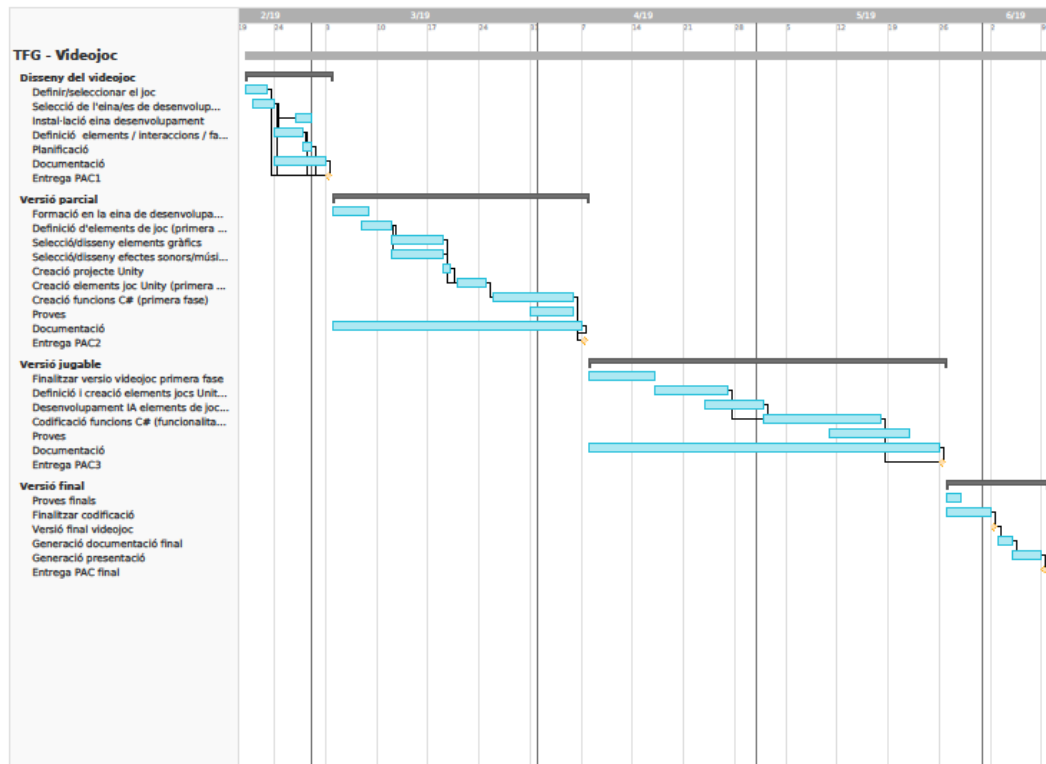
Tenint en compte la documentació disponible, la presència al mercat, la integració amb llenguatges de programació estàndard, que Unreal és un producte de molta qualitat, però complex, Unity és el producte escollit.

1.4 Planificació del Treball

Després de la definició inicial del projecte a realitzar, tenint en compte les dates d'entrega de les diferents parts, i les dues fases de desenvolupament (versió inicial i funcionalitats addicionals), les tasques, fites i dates estimades són:

| Tasca | Tipus | Data Inicial | Data final |
|--|-----------------|--------------|------------|
| Disseny del videojoc | Grup de tasques | 20/02/2019 | 03/03/2019 |
| Definir/seleccionar el joc | Tasca | 20/02/2019 | 22/02/2019 |
| Selecció de l'eina/es de desenvolupament | Tasca | 21/02/2019 | 23/02/2019 |
| Instal·lació eina desenvolupament | Tasca | 27/02/2019 | 28/02/2019 |
| Definició elements / interaccions / fases del joc | Tasca | 24/02/2019 | 27/02/2019 |
| Planificació | Tasca | 28/02/2019 | 28/02/2019 |
| Documentació | Tasca | 24/02/2019 | 02/03/2019 |
| Entrega PAC1 | Fita | 03/03/2019 | 03/03/2019 |
| Versió parcial | Grup de tasques | 04/03/2019 | 07/04/2019 |
| Formació en la eina de desenvolupament | Tasca | 04/03/2019 | 08/03/2019 |
| Definició d'elements de joc (primera fase) | Tasca | 08/03/2019 | 11/03/2019 |
| Selecció/disseny elements gràfics | Tasca | 12/03/2019 | 18/03/2019 |
| Selecció/disseny efectes sonors/música | Tasca | 12/03/2019 | 18/03/2019 |
| Creació projecte Unity | Tasca | 19/03/2019 | 19/03/2019 |
| Creació elements joc Unity (primera fase) | Tasca | 21/03/2019 | 24/03/2019 |
| Creació funcions C# (primera fase) | Tasca | 26/03/2019 | 05/04/2019 |
| Proves | Tasca | 31/03/2019 | 05/04/2019 |
| Documentació | Tasca | 04/03/2019 | 06/04/2019 |
| Entrega PAC2 | Fita | 07/04/2019 | 07/04/2019 |
| Versió jugable | Grup de tasques | 08/04/2019 | 26/05/2019 |
| Finalitzar versió videojoc primera fase | Tasca | 08/04/2019 | 16/04/2019 |
| Definició i creació elements jocs Unity (funcionalitats addicionals) | Tasca | 17/04/2019 | 26/04/2019 |
| Desenvolupament IA elements de joc addicionals | Tasca | 24/04/2019 | 01/05/2019 |
| Codificació funcions C# (funcionalitats addicionals) | Tasca | 02/05/2019 | 17/05/2019 |
| Proves | Tasca | 11/05/2019 | 21/05/2019 |
| Documentació | Tasca | 08/04/2019 | 25/05/2019 |
| Entrega PAC3 | Fita | 26/05/2019 | 26/05/2019 |
| Versió final | Grup de tasques | 27/05/2019 | 09/06/2019 |
| Proves finals | Tasca | 27/05/2019 | 28/05/2019 |
| Finalitzar codificació | Tasca | 27/05/2019 | 01/06/2019 |
| Versió final videojoc | Fita | 02/06/2019 | 02/06/2019 |
| Generació documentació final | Tasca | 03/06/2019 | 04/06/2019 |
| Generació presentació | Tasca | 05/06/2019 | 08/06/2019 |
| Entrega PAC final | Fita | 09/06/2019 | 09/06/2019 |

I el diagrama de Gantt que mostra la distribució de tasques i les relacions de dependència entre elles:



1.5 Breu sumari de productes obtinguts

Seguint els objectius definits pel present TFG, els principals entregables són l'aplicació en format .apk i .aab (Android App Bundle) executable en dispositius mòbils Android i una aplicació executable en sistemes Windows, en plataformes x86 32 i 64 bits.

Adicionalment es realitzarà la publicació de la present memòria, i es crea un vídeo explicatiu sobre el producte final i el seu procés de desenvolupament.

1.6 Breu descripció dels altres capítols de la memòria

Després d'aquest capítol introductori, a la resta de la memòria es documenten amb més detall l'entorn en el qual es desenvolupa el treball, el videojoc objecte del TFG i els recursos emprats.

Al capítol 2 s'explica l'estat actual de la indústria del videojoc i les tendències del mercat. Al capítol 3 es detalla la mecànica i els elements que componen el videojoc, per, a continuació explicar les fases i els nivells entre les quals es desenvolupa l'acció del mateix al capítol següent.

Als capítols 5 i 6 es concreten l'enfocament que s'ha seguit per fer el desenvolupament, des de les fases més inicials fins arribar al producte final, i els components del videojoc des del punt de vista més tècnic (estructura d'objectes i classes de Unity).

Al capítol 7 “Manual d’usuari” cobreix les diferents pantalles, la mecànica de NeoTrooper i com és la interacció entre l’usuari i el videojoc.

Per concloure, als capítols 8,9 i 10 es documenten les conclusions del projecte, i s’aporten el glossari amb la definició dels termes més importants utilitzats i la bibliografia, amb el llistat de recursos externs utilitzats en el desenvolupament del videojoc, a les presentacions i en aquest document.

2. Estat de l'art

2.1 El videojoc com a indústria

Com s'ha comentat al capítol d'introducció, Espanya està dins del 10 productors de videojocs més importants a nivell mundial.

Mes en concret, Catalunya acull el 31% de les empreses del sector, amb un total del 52% de la facturació estatal.

Barcelona ha anat guanyant pes, creant empreses de referència com Social Point, i atraient grans estudis internacionals com Ubisoft, Gameloft, King, Riot Games, etc...

2.2 Tendències

Segons l'AEVI (Asociación Española de Videojuegos), a Espanya el consum de videojocs ha experimentat un creixement de més del 43% enguany, i el consum per plataforma està molt repartit entre els dispositius mòbils i les videoconsoles.

En altres regions, la utilització de les diferents plataformes és desigual. A EEUU les videoconsoles mantenen la majoria del mercat, mentre que a Asia els dispositius mòbils són més utilitzats (font: AEVI).

En lo que respecta a tendències de tipologies de joc, les millors tècniques i llançaments com Fortnite o PUBG Mobile durant l'any 2018, fan que la experiència de joc en dispositius mòbils estigui cada cop més a prop de videoconsoles o PC.

La realitat virtual (VR) semblava que havia de ser la tecnologia predominant en els nous mètodes d'interacció amb els videojocs, però almenys a curt termini, jocs com Pokemon Go! han introduït el camí de la realitat augmentada, superant tots els precedents en quan a popularitat o consum d'un videojoc.

Una altra tendència és la migració de franquícies exitoses d'una plataforma a una altra, passant de PC/videoconsola a plataforma mòbil. En el futur, l'existència de serveis de joc per demanda, com ara Google Stadia, XBOX Game Pass i PlayStation Now, facilitaran aquesta portabilitat.

L'eliminació de les barreres entre plataformes facilitarà la proliferació de jocs multiplayer multiplataforma en temps real, com els mencionats PUBG, Fornite o altres joc de tipus Battle Royale.

Per lo que respecta als diferents models de pagament, encara coexisteixen els diferents tipus:

- Jocs de pagament

- Jocs gratuïts o de baix cost amb pagaments per millores
- Jocs gratuïts o de baix cost amb pagaments per contingut exclusiu
- Publicitat
- Pagament per utilització (p.e. season pass)
- Serveis de joc sota demanda.

Factors com un públic potencial cada cop més gran, superproduccions amb grans costos de creació, i problemes legals en alguns països, condicionaran els models a implantar en els propers anys.

3. Definició del joc

3.1 Descripció i objectius

El videojoc consta únicament d'un escenari, en el que van variant alguns dels elements. Es representa un camp de batalla, amb un canó centrat a la part inferior. L'usuari del joc controla aquest canó, pot moure l'angle de tir i disparar projectils.



Pantalla del joc de PC original de 1.982

El camp de batalla es sobrevolat per helicòpters i avions, que deixen caure paracaigudistes i bombes respectivament.

Hi ha dos objectius, per una banda aconseguir punts destruint els helicòpters, avions, bombes o matant els paracaigudistes. El segon objectiu és evitar que els enemics destrueixin el canó.

A la línia inferior es mostra informació sobre el desenvolupament del joc; al joc original es mostrava la puntuació aconseguida i la puntuació màxima i a NeoTrooper es mostren els punts aconseguits, el nivell actual i la dificultat.

3.2 Elements del joc

L'element central del joc és el **canó** que el jugador controla, de fet, la seva destrucció marca el final de la partida.

El **camp de batalla** està format per una pla horitzontal a la part inferior de la pantalla, és el destí final dels soldats que salten dels helicòpters. El canó la divideix en dues zones.

Hi ha **helicòpters** i **avions** que volen per la banda superior de la pantalla, creuant-la en tots dos sentits.

Des dels helicòpters es llencen **soldats**, que tenen com a objectiu aterrar al camp de batalla i aproximar-se al canó per destruir-lo.

Els avions llencen **bombes**, que si no són destruïdes i impacten al canó, el destrueixen.

3.3 Interacció de l'usuari amb els elements del joc

El canó del joc pot executar tres accions: pivotar a l'esquerra, pivotar a la dreta i disparar. La interacció de l'usuari amb el joc es limita a executar-les aquestes operacions amb les fletxes de moviment del teclat (en el joc original per PC): Fletxa dreta i esquerra (moviment del canó) i fletxa amunt (disparar).

El jugador pot disparar als helicòpters, als paracaigudistes o als paracaigudes durant el seu descens, als avions i a les bombes que aquests llencen. Helicòpters, avions i bombes són destruïdes si reben un impacte. Els soldats moren tant si reben un impacte directe o si el projectil toca el seu paracaigudes. En aquest últim cas, fa una caiguda lliure fins al terra i mor.

La velocitat dels projectils i el moviment del canó són deliberadament lents, de forma que l'usuari ha de calcular en quin moment ha de disparar per fer coincidir el seu tret amb la trajectòria dels elements de l'enemic, i en algunes ocasions ha d'escollir un objectiu, perquè la velocitat de moviment no permet abatre tots els que hi ha en pantalla. El canó està situat a sobre d'una torre, i té una inclinació mínima per cada banda, de forma que quant els soldats estan molt a prop de terra, o ja han aterrat, el canó no els pot tocar.

3.4 Puntuació

L'usuari comença la partida amb zero punts. Cada cop que destrueix un element enemic, aconsegueix punts. 10 punts per un helicòpter o un avió, 5 per un paracaigudista i 30 si intercepta una bomba. Per cada projectil que es llença des del canó, es perd un punt.

3.5 Final de la partida

La partida acaba quan l'enemic aconsegueix destruir el nostre canó. Només hi ha dues possibilitats de que això succeeixi:

- Han arribat quatre o més soldats vius a una de les zones del camp de batalla (esquerra o dreta). En aquest moment es reagrupen, i destrueixen el canó.
- Una bomba llençada des d'un avió impacta contra el canó.

4. Disseny de nivells

4.1 Fases del joc

Atac d'helicòpters

Dels extrems esquerre i dret de la pantalla sorgeixen helicòpters que la creuen, en ambdós sentits. Hi ha dues files, de forma que en un determinat moment poden haver-hi alguns que volin cap a un extrem i alguns en sentit contrari.

Des dels helicòpters es poden llençar paracaigudistes en qualsevol moment (un paracaigudista per helicòpter), aproximadament a la meitat del seu descens, s'obre el paracaigudes.

Com s'ha comentat, l'usuari pot moure l'angle del canó i disparar per destruir els helicòpters, els paracaigudistes o els paracaigudes, abans de que arribin al camp de batalla.

Atac d'avions

En aquesta fase la pantalla es creuada per avions, també en ambdós sentits. Els avions llencen projectils que van dirigits contra el canó. Són llençats amb diferents graus de caiguda.

L'usuari pot disparar als avions o a les bombes que aquests llencen, per interceptar-les durant el vol.

Destrucció del canó

Com s'ha descrit a l'apartat anterior, si quatre paracaigudistes arriben a una de les zones del camp de batalla o una bomba d'un avió impacta contra el canó, aquest es destrueix i s'acaba la partida.

4.2 Gameplay i nivells del joc

A partir de les fases anteriors del joc, es compon el gameplay de la versió definitiva de NeoTrooper. Cadascun dels nivells comprèn una fase d'atac d'helicòpters i una fase d'atac d'avions. Després de cada nivell, s'incrementa la dificultat del joc.

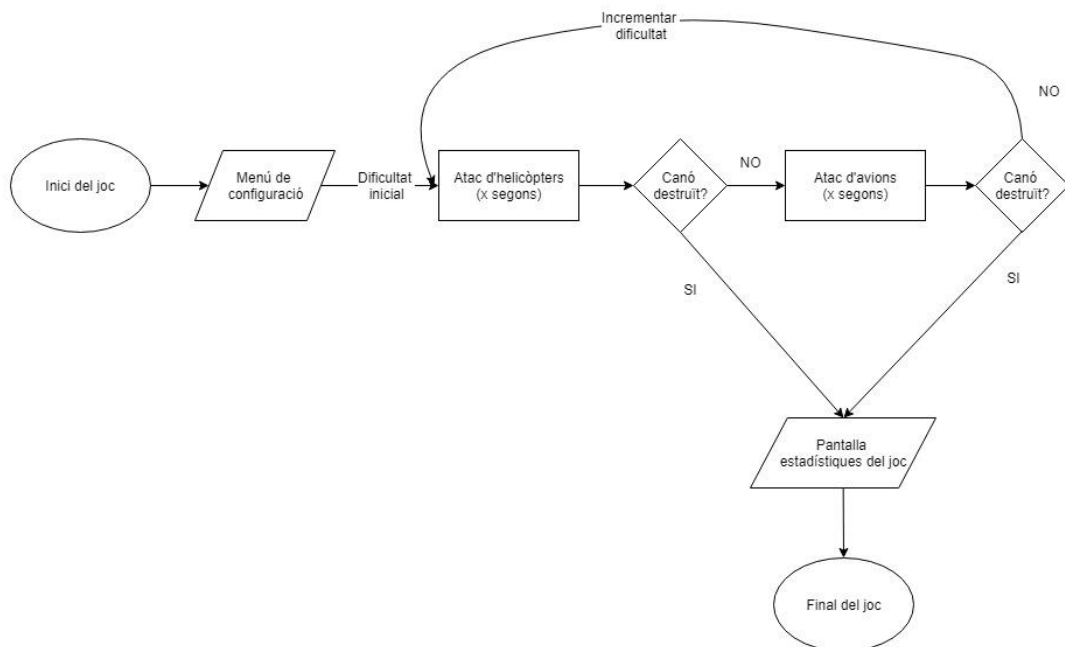
Aquest increment en la dificultat del joc s'implementa a nivell tècnic amb la modificació dels paràmetres que determinen com es comporten els diferents elements de joc:

- Angle mínim de rotació del canó.
- Angle màxim de rotació del canó.
- Velocitat de rotació del canó.
- Velocitat dels projectils que llença el canó.
- Temps de demora abans de poder disparar un altre projectil.

- Temps de demora entre dos d'helicòpters o avions.
- Velocitat dels helicòpters o avions.
- Nombre de paracaigudistes que llença cada helicòpter.
- Velocitat de les bombes que llencen els avions.

Abans de cada partida s'escull el nivell de dificultat. Aquesta elecció instaura uns determinats valors inicials per cadascun dels paràmetres anteriors. Després de la finalització de cada nivell de joc, es produeix una modificació de cadascun d'aquests valors, de forma que gradualment el control del canó és més difícil, el nombre de projectils és més baix i més lent i els enemics són més nombrosos.

Resumint, i tal i com es pot observar al següent diagrama, els atacs d'helicòpters i avions es succeeixen, amb grau de dificultat creixent fins que es produeix la destrucció del canó:



5. Fases del desenvolupament

5.1 Definició i elecció dels elements

Partint del concepte del joc original, es van definir els elements de joc que formaven part de la història del joc i el seus comportaments, començant pel moviment, atac, interaccions amb altres elements de joc i interacció amb l'usuari, en cas d'aplicar.

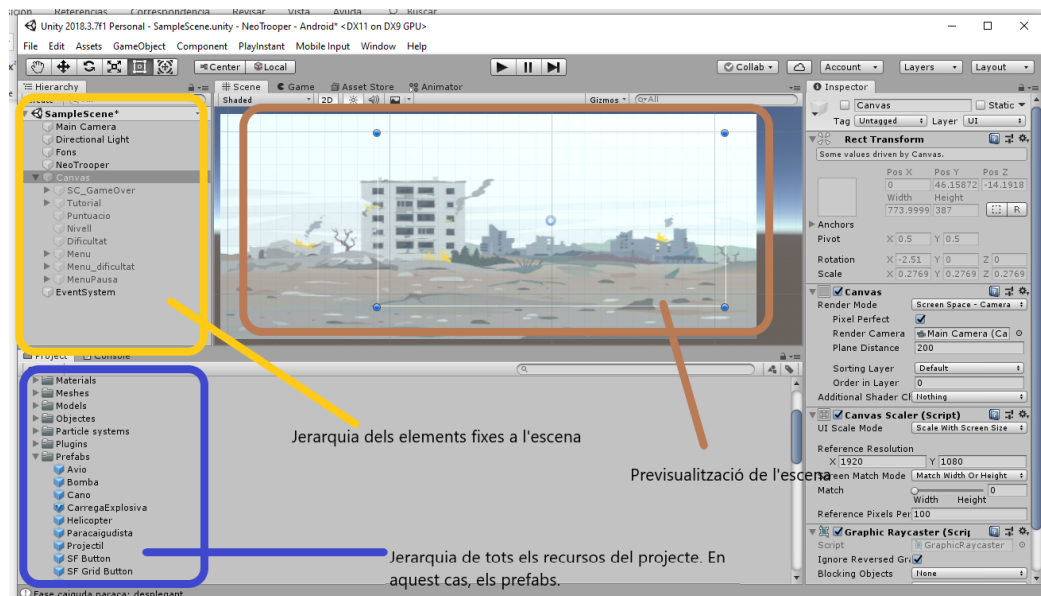
El següent pas va ser trobar els assets de Unity necessaris per representar aquests elements, escollir els efectes sonors, i identificar les animacions. En aquesta fase, el principal recurs va ser l'Asset Store de Unity. L'asset store és un repositori de components disponibles per la creació de videojocs amb Unity. Hi ha des de components gratuïts creats per Unity o la comunitat de dissenyadors i desenvolupadors fins a elements de qualitat professional de pagament, oferts per tercers.

El ventall de recursos va des de efectes sonors, fins a personatges, passant per animacions, fons, components per crear interfícies al joc, etc...

Els elements escollits estan detallats al capítol 6.2 d'aquesta memòria.

5.2 Creació del projecte Unity

Un cop triades les representacions gràfiques, per poder utilitzar-les dins del joc, aquestes s'han d'incloure dins d'un projecte de Unity.



Pantalla principal de Unity. Elements bàsics.

Un projecte de Unity és la col·lecció d'escenes, elements de joc, scripts i recursos en general que seran necessaris pel desenvolupament i la execució del videojoc.

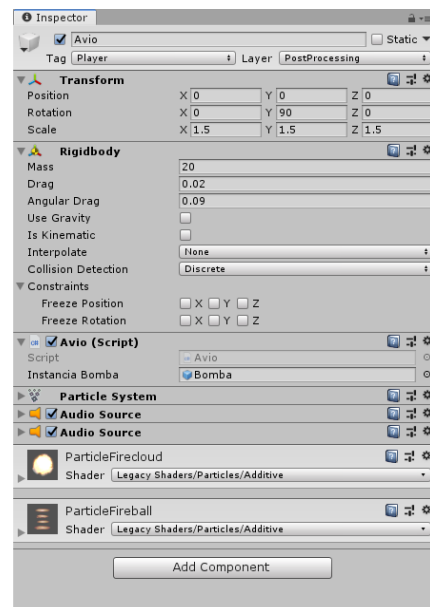
El component bàsic de tot videojoc és la escena, i a dins d'aquesta són imprescindibles la càmera, que determina dins del món 3D del joc el punt de vista que tindrà l'usuari i els llums, que com en el món real, són imprescindibles per que els objectes del joc siguin visibles.

La resta dels elements representats al joc podran estar fixes a l'escena o seran plantilles d'elements que es poden instanciar a demanda. Per exemple, en el cas de NeoTrooper, la imatge de fons és un element fixe de l'escena, però els helicòpters, avions, bombes, etc... que es generen quan la lògica del videojoc ho requereix. Aquestes plantilles a Unity s'anomenen *prefabs*.

5.3 Creació dels Gameobjects a Unity

Un cop triades les representacions gràfiques, per poder utilitzar-les dins del joc, aquestes s'han d'associar a un GameObject de Unity. Un GameObject és qualsevol element de l'acció del videojoc. En el moment de la creació dels GameObjects, s'adjunten els components de Unity que determinaran i complementaran la seva representació a pantalla (scripts, animacions, efectes sonors, etc...)

Els GameObjects, prefabs i components associats utilitzats s'expliquen de forma més detallada als capítols 6.1 i 6.2.



Finestra de definició d'un GameObject de Unity i els seus components associats.

5.4 Programació

Un dels objectius del present TFG era la utilització d'un *game engine* i l'aprenentatge de les tècniques de programació pròpies del desenvolupament de videojocs d'aquest entorn. Els punts més notables a tenir en compte en aquesta fase eren:

- **Utilització de llenguatge C#:** El llenguatge de programació de propòsit general utilitzat generalment al grau en Informàtica és Java, de forma que personalment la exposició a C# havia estat nul·la. Tot i tenir una sintaxi molt similar, hi ha petites diferències en alguns tipus, canvis en l'herència, sobreescritura de mètodes, etc...

- **Adaptació a l'entorn de Unity:** Unity és un entorn que facilita de forma molt significativa el desenvolupament de videojocs. A l'estar dissenyat explícitament per aquest propòsit, incorpora llibreries per les funcions que típicament seran necessàries durant la creació d'un videojoc. La utilització d'animacions, la reproducció de efectes sonors, la funcionalitat del motor físic i l'emulador de gravetat o la interacció amb l'usuari són implementables molt fàcilment mitjançant l'API de Unity. Tot i aquesta facilitat, i que la corba d'aprenentatge és molt ascendent, hi ha un període d'adaptació necessari. Són bons exemples d'aquesta adquisició de coneixements necessaris el tractament dels vectors en 3D, manegar angles de rotació, la utilització d'animacions, etc...

Probablement una de les adaptacions més importants de la metodologia de programació és la utilització de les funcions Start() i Update(): en el moment de la instanciació d'un objecte, Unity executa la funció Start() dels scripts associats a aquest objecte. Posteriorment, durant tota la vida de l'objecte, Unity cridarà la funció Update() cada frame, es a dir, que s'executarà nombroses vegades per segon.

Això fa que la funció Start() s'utilitzi per inicialitzacions, i que la majoria de la lògica de programa es codifiqui a les corresponents funcions Update(), controlant el fluxe de l'execució a través de variables que determinin l'estat dels objectes en cada moment. Per exemple, si es vol realitzar una determinada operació només en una ocasió, s'haurà d'utilitzar una variable per determinar si aquella operació ja ha estat executada, per evitar-ne la repetició.

- **Relació entre els gameobjects i els scripts que gestionen cadascun dels objectes:** Com ja s'ha mencionat, a qualsevol GameObject de Unity s'associen els components que determinen el seu comportament. Un d'aquests components serà el/s script/s en C# que contindrà la codificació de les operacions a realitzar per respondre a determinats events del gameobject (creació, col·lisió, actualització, etc...). Aquest script podrà accedir de forma senzilla als altres elements del GameObject, com la seva posició, les animacions associades, els efectes sonors...
- **Instànciació d'altres gameobjects, accés a les seves variables, execució de funcions:** Els prefabs hauran de ser instanciats i manipulats des de Game Manager o script principal de gestió del joc o bé des dels

scripts associats als altres objectes. Per tal de facilitar aquesta interacció s'hauran de definir correctament la visibilitat de les variables (publiques, privades, protegides) i la seva tipologia. Per exemple, totes les variables “globals” que emmagatzemen la informació bàsica del joc (punts, fase actual, nombre de paracaigudistes a terra,...), hauran de ser públiques i estàtiques.

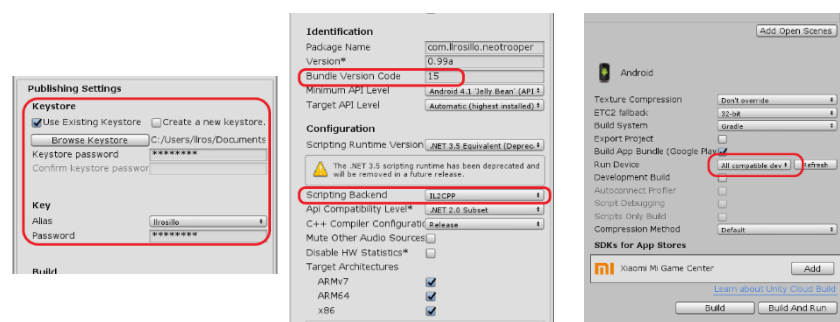
5.5 Publicació

Durant les fases inicials del desenvolupament, les necessitats de compartició dels fonts i els binaris es podien resoldre amb github o la utilització puntual de serveis temporals de transferència de fitxers.

Amb el propòsit d'arribar a un grup més elevat de provadors, que, per exemple, no tenien coneixements suficients de configuració d'Android per permetre l'execució d'apps provinents de fonts desconegudes, es va decidir crear un perfil d'Android developer, i la distribució del videojoc a través de la Google Play Store.

Convé subratllar que els requeriments d'aquesta plataforma, enriqueixen els coneixements adquirits durant el present TFG. Entre d'altres, es van haver d'afrontar els següents requisits:

- **Obtenció d'una clau de desenvolupador i signatura de l'app:** Google requereix que totes les aplicacions estiguin signades digitalment pel seu desenvolupador. Unity, dins de les opcions de compilació i execució, permet crear un magatzem de claus i una clau. En cada compilació s'hauran de proveir les contrasenyes del magatzem i de la clau. (*Build Settings | Player Settings | Publishing Settings|Keystore password i Key password*)



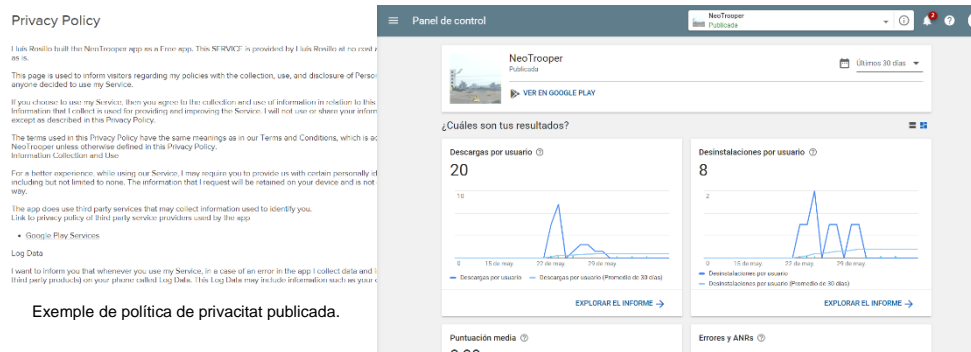
Diferents aspectes de la configuració de compilació a Unity.

- **Requeriment d'especificar un version code diferent a cada nova versió publicada de l'aplicació:** Cada nova versió del binari de l'aplicació que es publica a Google Play Store, ha de tenir un codi de versió

diferent. (*Build Settings | Player Settings | Other Settings | Bundle Version Code*)

- **Necessitat de proveir les llibreries de 32 i 64 bits per les plataformes suportades:** A partir d'Agost de 2019, les normes de publicació d'aplicacions a Google Play han de ser compatibles amb arquitectures de 64 bits. A Unity, cal definir IL2CPP com a scripting backend (*Build Settings | Player Settings | Other Settings | Scripting Backend*) i a la opció de compilació *Run Device*, seleccionar “All compatible devices” (*Build Settings*)
- **Política de privacitat publicada i disponible online:** Encara que no és obligatori en el moment de la primera publicació de l'aplicació, Google revisa que cada aplicació disposi d'una política de privacitat publicada. Existeixen websites generadors de polítiques de privacitat personalitzables que es poden utilitzar per aquest propòsit (p.e. <https://app-privacy-policy-generator.firebaseio.com/>). Aquests serveis proporcionen una pàgina html que posteriorment s'ha de publicar. La política de privacitat de NeoTrooper està publicada a: <https://lrosillo.wixsite.com/privacypolicy>.

Finalment, la publicació de l'app, permet accedir a la Google Play Console, aprendre les seves possibilitats de visualitzar les estadístiques, KPIs, ressenyes, etc... i potencialment utilitzar-les per avaluar l'acollida de les diferents versions del videojoc.



Panell de control de Google Play Console.

5.6 Proves

Per tal d'aconseguir un producte final de més qualitat, més jugable i més adaptat a una ampla varietat de potencials usuaris, després de desenvolupar la funcionalitat per permetre el joc des de dispositius mòbils, implementar la

lògica dels diferents nivells i crear el tutorial, es va començar la distribució de la versió alpha del producte entre diferents provadors, per obtenir informació molt valuosa per la finalització del desenvolupament. Els aspectes bàsics sobre els quals es demanava informació als provadors van ser:

- Model i marca del mòbil. Per assegurar el funcionament en el ventall més ampli possible de dispositius.
- Impressió sobre la dificultat i jugabilitat del joc.
- Impressió sobre l'aspecte gràfic, tenint en compte que donades les restriccions de cost i temps del desenvolupament.
- Opinió sobre si representa una experiència amena o si pel contrari és avorrit.
- Feedback sobre errors en el funcionament general del joc.

El grup de provadors estava comprés per persones de diferent sexe, edat i experiència/interès en l'ús de videojocs.

En general l'experiència va ser molt positiva pel desenvolupament del joc, tant per la identificació de bugs que no s'havien detectat durant el desenvolupament i les proves directament derivades del mateix com pels suggeriments de millora, alguns dels quals són viables pel present cicle de desenvolupament de NeoTrooper i alguns altres queden pendents per potencials noves versions del videojoc.

Cal comentar que el fet d'utilitzar Google Play Store des de les primeres versions de proves, va agilitzar molt el procés de detecció i correcció de bugs.

Dispositius provats

Gràcies a les proves realitzades, es va comprovar el correcte funcionament de NeoTrooper a nombrosos terminals de diferents marques. Entre ells:

- Xiaomi Mi5, Mi6, A1, A2 lite, Mix 2, Note6 pro, Redmi 4A
- Lenovo P2
- Huawei P8 lite, P9, P20 lite, Honor 7, Honor 9
- Samsung Galaxy S6
- Bluestacks 4 (Emulador Android per Windows)

Durant les proves no es va detectar cap incompatibilitat amb cap dispositiu dels utilitzats. Unicament es van observar problemes de rendiment puntuals en els dispositius més antics o de gama més baixa.

Bugs detectats

Tot i les proves realitzades durant el procés de desenvolupament, el grup de proves, especialment durant les proves de les primeres versions publicades, detecta alguns errors, especialment de navegació. Per exemple:

- El menú de pausa es pot invocar des de qualsevol pantalla, no només des de la pantalla de joc.
- Si s'intenta accedir dos vegades de forma consecutiva al tutorial, el segon intent no funciona.

Funcionalitats proposades

Una de les qüestions sobre les quals eren preguntats els provadors era sobre possibles noves funcionalitats que veurien positives. Tot i que fora de l'abast del present projecte, val la pena remarcar com a mostra, alguns dels suggeriments:

- Possibilitat d'escollir la música de fons del joc.
- Possibilitat de personalitzar els colors del canó, paracaigudes, avions, etc...
- Afegir algun efecte sonor per reflectir la comunicació entre paracaigudistes i helicòpters.
- Utilitzar un slider per escollir el nivell de dificultat del joc, afegint més possibilitats d'elecció.
- Afegir una barra per indicar el percentatge de vida disponible del canó. En aquest cas s'hauria de modificar la mecànica del joc, ja que actualment un sol atac al canó el destrueix.
- Dins del tutorial, afegir la possibilitat de retrocedir a la pantalla anterior.

Feedback rebut

A banda dels bugs i funcionalitats, el feedback rebut sobre l'aplicació, encara que en general positiu, estava molt condicionat per les preferències dels provadors en termes de videojocs i de la seva experiència.

Encara que no unànimes, es reben alguns comentaris amb respecte a la qualitat dels recursos gràfics, animacions i efectes sonors. Com ja s'ha comentat, donades les limitacions de pressupost, els recursos gràfics i de so havien de ser gratuïts i aquest aspecte del videojoc es veu penalitzat.

En alguns casos els provadors troben que el joc és poc variat, però també en aquest cas una possible explicació es troba en el fet de per l'abast inicial del projecte s'han desenvolupat únicament dues pantalles de joc.

En el grup de provadors no tant acostumats a utilitzar videojocs en dispositius mòbils en general i arcades en particular, s'observa que és imprescindible la existència d'un tutorial descrivint la funcionalitat bàsica, ja que per aquest grup d'usuaris la mecànica d'interacció no és evident.

Altrament, en el grup de provadors més avesats a videojocs, els comentaris estaven més orientats a aspectes de jugabilitat o de possibilitats de personalització.

Amb respecte a l'experiència en general i la opinió sobre si era o no amè, la resposta normal era positiva, essent més valorat el joc pel col·lectiu dels jugadors habituals.

6. Disseny tècnic

6.1 Estructura del projecte d'Unity

El projecte d'Unity consta únicament d'una càmera, una llum i una escena com a elements fixes, que es a on es desenvolupa l'acció.

Existeix un Gameobject anomenat NeoTrooper, amb un script associat que es carrega en el moment d'iniciar l'aplicació, el Game Controller.

A la escena també hi ha definit un objecte de tipus Canvas, i depenent d'ell, els gameobjects que fan possible la representació a pantalla dels diferents marcadors, el menú principal i de pausa, el tutorial i la pantalla de resum que es mostra després de la finalització de cada partida.

La resta d'elements del joc són *prefabs* de Unity, es a dir, plantilles que modelen objectes i que es poden instanciar a demanda. Així, són prefabs els avions i helicòpters, el canó, els projectils, les bombes, etc...

Seguint el paradigma de programació orientada a objectes, hi ha una estructura de classes que s'instancia a demanda, depenent del transcurs del joc. Cal distingir la classe NeoTrooper, de la qual només hi haurà una instància que existirà mentre l'app estigui en execució. Aquesta classe conté variables públiques estàtiques que seran consultades i modificades per tots els components del joc i que fixaran en temps d'execució en quina fase està el joc, quants punts té l'usuari, quins objectes de tipus avió, helicòpter, ... existeixen, etc...

Composició dels prefabs

Depenent de la funció dels prefabs mencionats anteriorment, tindran diversos components de Unity associats:

- **Programes codificats en C#:** defineixen el comportament de l'objecte en concret, en funció de l'estat de les variables o dels *inputs* rebuts de l'usuari.
- **Animacions:** En el cas d'objectes com el soldat (Ethan) o els helicòpters, els assets disposen de la definició dels moviments que poden realitzar els objectes en 3D, que permeten simular un comportament més realista o facilitar la transició a nivell gràfic entre dos estats. Són exemples d'animació l'efecte de córrer del soldat o el moviment dels rotors de l'helicòpter.
- **Particle System:** És una característica de Unity que permet definir efectes per simular explosions. Cadascun dels objectes del joc

susceptible d'explotar conté un element Particle System que es pot invocar des de l'script C# associat.

- **Colliders:** És el mecanisme que proporciona Unity per detectar quan un objecte ha col·lidit amb un altre. És necessari proporcionar una figura 3D que s'associï a l'objecte i que sigui lo més propera a la representació en 3D de l'objecte que es mostra a pantalla. Quan el motor físic de Unity detecta que algun altre objecte ha entrat dins d'aquesta figura proporcionada, generarà un event que es pot capturar des de l'script en C# associat al gameobject. Una mostra és en el prefab corresponent a l'avió, quan es captura aquest event es perquè algun altre Gameobject coincideix a l'espai amb l'avió (p.e. un projectil llençat pel canó).
- **Efectes sonors:** Dins de la definició del prefab es poden adjuntar fitxers de so que anàlogament als particle Systems es poden invocar des dels scripts C#. Aquests efectes sonors es poden executar durant tota la "vida" de l'objecte, com el soroll dels motors dels avions, o de forma puntual, com les explosions produïdes quan un projectil els intercepta.

6.2 Elements del joc

Game controller

NeoTrooper.cs és una classe java que fa les funcions de gestor del joc. Fa les crides a la pantalla inicial de configuració del joc, mostra el tutorial, defineix i fa les transicions entre nivells, instància els diferents elements de joc, etc... En aquesta classe es defineixen i es modifiquen les variables estàtiques que determinen el comportament de la resta d'elements.

Fons



És una imatge estàtica d'un entorn de guerra, amb elements de fons lleugerament difuminats, per no treure l'atenció del jugador sobre els elements del joc.

Canó

És l'element central del joc. Està ubicat a l'origen de coordenades i junt amb el fons, és l'únic component fixe a l'escena durant el joc. A nivell de Unity està compost de tres elements; la plataforma, el cos i els canons, que varien la seva rotació en funció de les entrades del jugador. Existeix un script de control, adjunt als canons (Guns.cs) que gestiona la rotació i crea les instàncies dels projectils. Aquest mateix script controla l'event de col·lisió, instanciant el *particle effect* de l'explosió, i destruint els *GameObjects* corresponents al canó. destrucció del canó. L'usuari del joc controla aquest canó, pot moure l'angle de tir i disparar projectils. La partida acaba quan el canó és destruït.



Aeronaus

Els enemics de l'usuari (helicòpters i avions), tenen algunes funcionalitats comuns. Aprofitant les possibilitats de la programació orientada a objectes, hi ha una classe – codificada a *Aeronau.cs* - que defineix els comportaments coincidents, com el moviment, la destrucció de l'objecte quan surt de pantalla o la gestió dels impactes.

Els objectes aeronau volen d'esquerra a dreta o a l'inrevés, sobrevolant l'escena, desapareixent al sortir dels límits de pantalla. En el moment de ser impactats, es reproduïx el *particle system* associat a cada tipus d'aeronau.

Helicòpters

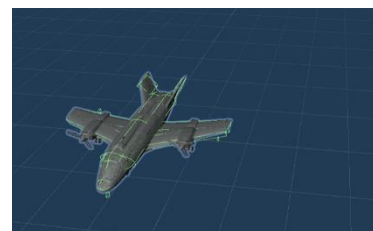
La representació gràfica és un *asset* d'aspecte molt acurat de l'asset Store de Unity. Consta d'una animació, que fa moure les hèlices principals i les del rotor de cua.



L'script associat (*Helicopter.cs*) defineix una classe que hereta d'*Aeronau*. Sobreescrui de forma parcial *Start* i *Update*, afegint la lògica de programa relativa al llançament dels paracaigudistes (consultar quants s'han de llençar, instanciar-los en un moment aleatori, etc...)

Avions

Consta d'una representació gràfica que forma part del pack bàsic de Unity. Novament, l'script associat (*Avio.cs*) sobreescrui parcialment els mètodes de la seva superclasse, per fer la gestió del llançament de bombes (que té una lògica diferent al llançament dels paracaigudistes).



Paracaigudistes

La representació gràfica del paracaigudista està composta per tres elements gràfics diferents, per una banda una figura humana disponible al pack bàsic d'assets d'Unity (Ethan) i per altre el paracaigudes en dos posicions diferents, desplegant-se i totalment desplegat. Després de ser instanciats, els paracaigudes comencen el descens gràcies a la simulació de gravetat de Unity.



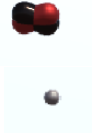
L'obertura i el moviment descendent es simula modificant el paràmetre **drag** del motor de física de Unity i habilitant i deshabilitant els diferents components que formen els paracaigudes. Inicialment "salta" la figura del soldat, a partir d'un cert punt s'activa el gameobject que representa l'inici del desplegament d'un paracaigudes, gairebé de forma immediata es desactiva aquesta primera fase i s'activa el gameobject que representa el paracaigudes sencer, al mateix temps es modifica el mencionat paràmetre **drag**, per reduir la velocitat de descens, a l'arribar a terra, es deshabiliten els gameobjects que representen els paracaigudes. En cas de que un projectil toqui el paracaigudes o el soldat durant el descens, el soldat inicia una caiguda lliure, que provoca la seva mort.

Per tal d'identificar clarament els paracaigudistes que han estat morts, a l'arribar al camp de batalla, es gira cap endavant la figura i queda estesa al terra. Uns segons després, aquesta figura desapareix i es destrueix l'objecte de joc.

Si arriben quatre o més paracaigudistes il·lesos a una de les dues bandes del canó, corren fins a la base d'aquest, col·loquen una càrrega explosiva i fugen. La càrrega explota, destruint el canó i finalitzant el joc.

Bombes i projectils

Son respectivament la representació de les bombes llençades des dels avions i els projectils que disparen els canons. Estan realitzades a partir d'elements 3D bàsics de Unity (esferes i càpsules). Els scripts Bomba.cs i Projectil.cs defineixen respectivament els comportaments relatius a impactes, sortides de pantalla, etc...



Càrregues explosives



És la representació de la bomba que col·loquen els paracaigudistes en el moment que ataquen el canó. Aquests la col·loquen al costat del canó, i explota després d'uns segons, provocant la destrucció del canó i el final de la partida. A nivell de Unity és un asset gratuït disponible a la asset Store. Té un script C# associat, que únicament carrega i

executa el particle System i l'efecte sonor en el moment que es produeix la explosió.

Elements del Canvas

Com s'ha mencionat a l'apartat anterior, un dels components fixes a l'escena de Unity és un GameObject de tipus Canvas. Aquest tipus d'objectes són necessaris per mostrar blocs de text a la pantalla del joc.

A l'escena de NeoTrooper es mostren, depenent del moment del joc o de les opcions escollides per l'usuari els següents elements:

Menu inicial



És la primera pantalla que es mostra quan comença el joc o quan acaba una partida. Permet iniciar una partida, activar i desactivar el so del videojoc o sortir al sistema operatiu tancant l'aplicació.

Menu elecció nivell



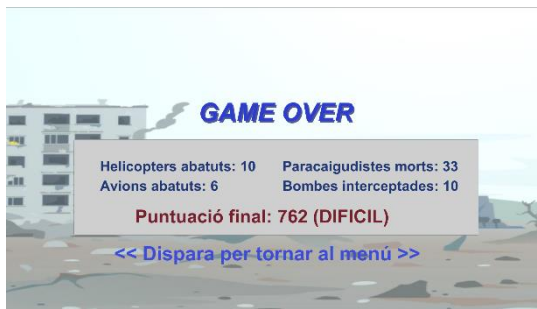
Al seleccionar la opció de jugar al menú principal, es mostra una nova pantalla, que llista els diferents nivells de joc, oferint una explicació bàsica i demanant que l'usuari trii una de les possibilitats.

Menu pausa



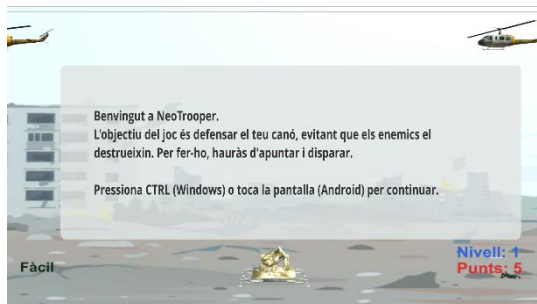
Si durant el transcurs del joc es polsa la tecla Escape (Windows) o el botó de retrocedir (Android), s'atura l'acció i es mostra aquest menú en el qual se li dona la opció a l'usuari de reprendre el joc, sortir al menú principal o sortir de l'aplicació.

Pantalla GameOver



Després de la finalització d'una partida, és la pantalla que mostra el resum dels objectius aconseguits per l'usuari: avions i helicòpters abatuts, paracaigudistes morts, bombes interceptades, i total de punts, tenint en compte la dificultat de joc escollida.

Tutorial



Des del menú inicial es pot arribar a aquesta successió de pantalles que mostren la mecànica bàsica del joc i els controls que permeten la interacció del usuari.

Puntuació i número de nivell

Durant la partida, a la part inferior dreta de la pantalla es mostren els punts que ha aconseguit l'usuari fins al moment, i el nivell de joc que ha assolit.

Grau de dificultat

De la mateixa forma que la puntuació i el número del nivell, a la part inferior esquerra es mostra el grau de dificultat escollit per l'usuari per la partida actual.



7. Manual d'usuari

Menú principal

A l'inici de l'execució del videojoc, o després d'una partida, es mostra un menú en pantalla que permet començar un nou joc, mostrar un tutorial amb instruccions sobre el joc, desactivar o tornar a activar la música de fons i els efectes sonors, o sortir del videojoc.



Menu elecció dificultat del joc

Després de pressionar la opció de jugar en el menú inicial, s'accedeix a un altre menú en el qual es mostren els diferents nivells de joc disponibles, amb una breu explicació de l'increment de dificultat entre ells.

Aquí es permet triar entre els nivells fàcil, mitjà, difícil i suïcida. Just després de fer la selecció, s'inicia l'acció.



Objectiu del joc

L'objectiu del jugador de NeoTrooper és preservar la vida del canó, atacant als objectius que el sobrevolaran durant les diferents pantalles del joc. L'usuari ha d'intentar destruir el màxim nombre d'enemics, evitant que els enemics ataquin el canó. El canó es destruirà per l'efecte de les bombes o els paracaigudistes enemics.

Pantalles de NeoTrooper

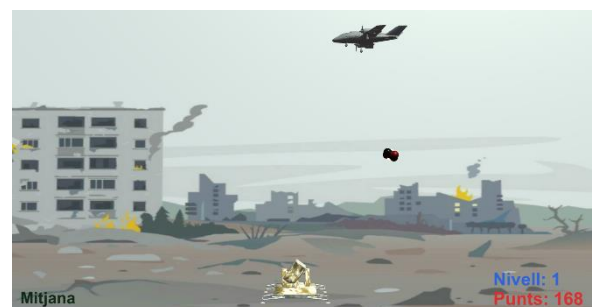
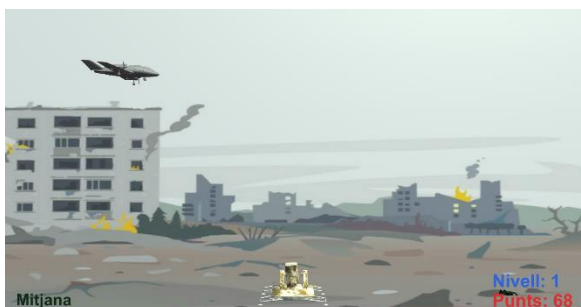
Durant el desenvolupament del joc, diferents enemics intentaran destruir la nostra arma.

Inicialment helicòpters enemics sobrevolaran el canó, dels que saltaran paracaigudistes. Si quatre d'aquests paracaigudistes del mateix costat del canó aconseguixen arribar a terra il·lesos, atacaran el canó i el destruiran.

L'usuari pot disparar als helicòpters per destruir-los o als paracaigudistes durant el seu descens, per evitar que arribin a terra.



A la següent fase del videojoc, avions bombarders atacaran el canó. Llançaran bombes que en cas de tocar el canó el destruiran. L'usuari ha de llençar projectils per impactar als avions o interceptar les bombes en la seva trajectòria fins al canó.



Un nivell està compost per un atac d'helicòpters i un atac d'avions. Després de cada nivell, s'incrementarà la dificultat del joc. Augmentant el nombre d'aeronaus atacants i la seva velocitat, el nombre de paracaigudistes, la velocitat de les bombes, i reduint la mobilitat del canó i la seva capacitat de disparar.

Controls de joc

Per impactar sobre els objectius enemics, l'usuari haurà de girar el canó per apuntar i disparar, tenint en compte la trajectòria i velocitat del projectil i de l'objecte al que està disparant.

En plataforma Windows, el canó es girarà amb les fletxes esquerra i dreta del teclat. En dispositius Android, l'usuari haurà de girar verticalment el dispositiu per apuntar.

Per disparar, s'haurà de pressionar la tecla "Control" a Windows i tocar la pantalla amb dispositius Android.

En qualsevol moment del joc, per accedir al menú de pausa, es podrà pitjar la tecla "Escape" a Windows o el botó de retrocedir a Android.

Nivells de dificultat

A l'inici de cada partida, el jugador pot escollir el nivell de dificultat (fàcil, mitjana, difícil o suïcida). El videojoc modificarà el comportament del canó i dels enemics. El nivell de dificultat afecta a:

- Comportament del canó:
 - Amplitud de l'angle de rotació del canó.
 - Velocitat de rotació del canó.
 - Velocitat dels projectils que llença el canó.
 - Quantitat de projectils que pot llençar.

- Comportament dels enemics:
 - Nombre d'helicòpters i avions.
 - Velocitat dels helicòpters i avions.
 - Nombre de paracaigudistes que es llencen.
 - Velocitat de les bombes que llencen els avions.

Puntuació

El jugador comença la partida amb zero punts. Aconsegueix punts destruint objectius enemics:

- Helicòpter: 10 punts.
- Avió: 10 punts.
- Paracaigudista: 5 punts.
- Bomba: 30 punts.

D'altra banda, cada projectil disparat resta un punt del marcador.

La dificultat de joc escollida canvia la puntuació final:

- Nivell fàcil: Puntuació aconseguida.
- Nivell mitjà: Puntuació aconseguida + 20%
- Nivell difícil: Puntuació aconseguida + 40%
- Nivell suïcida: Puntuació aconseguida + 80%

Menú de pausa

Si durant el transcurs del joc, en qualsevol de les seves pantalles, es pressiona la tecla Escape (Windows) o el botó de tornada enrere en un dispositiu Android, s'atura l'execució del joc, i l'usuari tindrà la possibilitat de reprendre el joc, tornar al menú principal o sortir de l'aplicació.



Pantalla resum de partida

Al final de cada partida, es mostra un panell amb els objectius abatuts i els punts aconseguits. La puntuació final té en compte el nivell de joc escollit, segons les regles dels apartats anteriors.



8. Conclusions

L'objectiu principal plantejat a l'inici d'aquest treball era seguir el procés sencer de desenvolupament d'un videojoc, utilitzant les eines pròpies de l'àrea i amb la intenció final d'aconseguir un joc complet, jugable i idealment amè.

Podem dir que l'objectiu inicial s'ha complert, un dels entregables és la versió totalment funcional d'un videojoc per Android, amb la implementació del gameplay de la primera fase. A tenor del feedback rebut, el videojoc és jugable i proporciona una experiència de joc agradable.

La planificació inicial contemplava les tasques requerides per aquesta fase, deixant com a opcionals el desenvolupament de funcionalitats addicionals (diferents enemics, powerups, etc...) A excepció d'alguns canvis menors (p.e. programació dels efectes sonors) la execució de les tasques ha coincidit amb la planificació, i les diferents entregues contenien els desenvolupaments previstos.

Les lliçons apreses han sigut nombroses, a tall d'exemple, es destaquen les principals:

- Coneixement en general de les diferents eines disponibles al mercat i coneixement molt més detallat de Unity.
- En aquest sentit, aprendre com Unity crea una capa d'abstracció que simplifica enormement les tasques necessàries per la programació gràfica i la gestió de la interacció amb l'usuari.
- Experimentar tot el procés de desenvolupament d'un videojoc, des de l'elecció de la temàtica, definició dels objectius, la interacció de l'usuari, selecció dels elements gràfics, programació de la lògica de programa, proves, correcció de bugs, redefinició dels comportaments per ajustar la dificultat esperada, etc...
- Tenir una primera presa de contacte amb C# i l'API de Unity.
- Aprendre els requisits de publicació a Google Play Store i la gestió de les versions d'una app.

En quan a la metodologia, es tracta d'un primer projecte amb unes eines desconegudes, és evident que en un potencial nou treball, la forma d'afrontar alguns reptes seria diferent. Ha sigut necessari un temps addicional per aprendre alguns aspectes molt concrets ja mencionats (interacció amb l'usuari, manipulació d'animacions, etc...).

Cal ressaltar, per exemple, que una gran part de la lògica de la gestió del moviment o l'atac dels paracaigudistes es va haver de reformular, ja que el primer desenvolupament era excessivament complex. Un coneixement més a fons de les possibilitats d'interacció entre els diferents elements, com la

possibilitat/necessitat de crear, manipular i consultar variables globals, simplifica el desenvolupament.

Tot i que la corba d'aprenentatge amb Unity i C# es impressionantment elevada, els següents videojocs desenvolupats tindrien una millor qualitat i es realitzarien en menys temps.

La conclusió principal és que Unity simplifica la vessant tècnica de creació de videojocs, permetent que el creador d'un videojoc es pugui concentrar en la part artística o de disseny de la història, contràriament a el que era necessari fa uns anys.

Com ja s'ha comentat, NeoTrooper és un joc funcional, però la versió actual està condicionada per les limitacions pròpies d'un TFG. Les funcionalitats més importants pendents per a una propera versió, serien:

- Desenvolupament de nivells addicionals de joc, amb diferents atacants.
- Creació d'una botiga del joc, amb powerups o customitzacions addicionals.
- Support multilinguatge.
- Joc multijugador en temps real.

9. Glossari

- Android:** Sistema operatiu per dispositius mòbils amb pantalla tàctic, desenvolupat per Google, basat en Linux.
- API:** Conjunt de funcions i procediments que permeten la creació d'aplicacions que accedeixen a les dades o funcionalitats d'un sistema operatiu, aplicació o servei.
- Arcade:** Terme que inicialment feia referència a les màquines recreatives de videojocs que es basaven en la perícia del jugador. Posteriorment, per extensió, es va utilitzar per anomenar als llocs a on estaven ubicades i finalment als jocs.
- Asset de Unity:** Recurs gràfic, sonor o d'algun altre tipus que s'utilitza pel desenvolupament de videojocs al *game engine* Unity.
- Battle Royale:** Fa referència als llocs de lluita entre diferents jugadors en els quals només un d'ells pot sobreviure. El terme es va popularitzar per la novel·la japonesa homònima, escrita per Koushun Takami al 1.999.
- C#:** Llenguatge de programació orientat a objectes desenvolupat per Microsoft. La seva sintaxi deriva de C/C++ i està també inspirat en Java.
- Particle System:** És una funcionalitat de Unity que permet definir el comportament de conjunts de petites imatges (partícules) que es mostren i es mouen creant un efecte.
- Format .apk:** És un paquet (Android Application Package) que s'utilitza per distribuir i instal·lar components per la plataforma Android.
- Format .aab:** És un format de paquet per components Android (Android App Bundle) que s'utilitza per la publicació d'aplicacions a Google Play. Un únic fitxer conté diferents versions de l'aplicació per diferents arquitectures.
- Frame:** Un frame (fotograma), es cadascuna de les imatges estàtiques que componen una seqüència de vídeo, i que essent visualitzades amb una determinada freqüència, generen la sensació de moviment a l'espectador.
- Game engine:** És un entorn de desenvolupament de software dissenyat per construir videojocs.
- GameObject:** Un GameObject és qualsevol element de l'acció d'un videojoc dins de l'entorn de desenvolupament de Unity.

- Gameplay:** És la forma específica mitjançant la que un usuari interactua amb un videojoc. El Gameplay descriu les regles i els objectius del mateix.
- Google play Store:** Botiga digital oficial d'aplicacions Android de Google. És l'aparador a on la comunitat de desenvolupadors Android ofereixen les seves apps, gratuïtes o de pagament.
- Prefab:** Són plantilles que modelen els GameObjects de Unity i que poden ser instanciades a demanda en qualsevol moment del joc.
- Script:** És el fitxer que conté la codificació en un determinat llenguatge de programació de la seqüència d'operacions que conformen un programa.
- TFG:** Treball de Final de Grau
- Unity:** Game Engine desenvolupat per Unity Technologies. Actualment disposa de suport per més de 20 plataformes. Es pot utilitzar per construir tant jocs en dos com en tres dimensions.
- Unity asset Store:** Repositori oficial d'assets per Unity de Unity Technologies. Ofereix un gran ventall de recursos gratuïts i de pagament.

10. Bibliografia i enllaços relacionats

10.1 Recursos relatius a NeoTrooper

- NeoTrooper a Google Play Store:
<https://play.google.com/store/apps/details?id=com.llrosillo.neotrooper>
- Video resum TFG a Youtube:
<https://youtu.be/pVIUh7Uj8ds>
- Repositori de NeoTrooper a Github:
<https://github.com/llrosillo/NeoTrooper>
- Perfil de Lluís Rosillo a Unity Connect:
<https://connect.unity.com/u/5c7e6367edbc2a0daedc7120>
- Showcase de NeoTrooper a Unity Connect:
<https://connect.unity.com/p/neotrooper>

10.2 Documentació utilitzada:

- **Gibson, Jeremy** (2017). "Introduction to Game design, prototyping, and development". Addison-Wesley.
- **Asociación Española de Videojuegos (AEVI)** (2018). "El sector de los videojuegos en España: impacto económico y escenarios fiscales". Gener 2018. Disponible online a: "http://www.aevi.org.es/web/wp-content/uploads/2018/01/1801_AEVI_EstudioEconomico.pdf"
- **Escuela Universitaria de Diseño Innovación y Tecnología** (2018). "La industria española del videojuego entra en el 'top 10' mundial en volumen de negocio". 3 de Setembre. Disponible online a: "<https://www.esne.es/la-industria-espanola-del-videojuego-entra-en-el-top-10-mundial-en-volumen-de-negocio/>"
- **Desarrollo Español de Videojuegos (DEV)** (2017). "Libro blanco del desarrollo Español de videojuegos 2017". Disponible online a: "https://www.europacreativamedia.cat/rcs_auth/convocatories/libro_blanco_dev_2017.pdf"

- **Lee, Joel** (2018). “5 Free Game Development Software Tools to Make Your Own Games”. Make Use Of 25 d’Octubre. Disponible online a: [“https://www.makeuseof.com/tag/five-free-game-development-tools-make-your-own-games/”](https://www.makeuseof.com/tag/five-free-game-development-tools-make-your-own-games/)
- **Game Designing** (2018). “Essential Software for Game Designers”. 21 de Desembre. Disponible online a: [“https://www.gamedesigning.org/career/software/”](https://www.gamedesigning.org/career/software/)
- **Patel, Nisha** (2018). “Unity vs Unreal Engine? No more Confusion for Game development”. 10 d’Abril. Disponible online a: <https://www.linkedin.com/pulse/unity-vs-unreal-engine-more-confusion-game-nisha-patel/>
- **Corpier, Kendra** (2018). “How to Publish Your Game to Google Play Store with Unity”. 13 de Juliol. Disponible online a: <https://youngstowngamedevelopers.com/how-to-publish-your-game-to-google-play-store-with-unity/>
- **Plarium** (2018). “Top 7 Video Game Trends of 2018: Battle Royale, Diversity and the End of Loot Boxes”. 21 de Juny. Plarium Blog, disponible online a: <https://plarium.com/en/blog/video-game-trends-2018/>
- **Plarium** (2018). “How Battle Royale Is Changing Online Gaming”. 11 de Febrer. Plarium Blog, disponible online a: <https://plarium.com/en/blog/battle-royale/>
- **Plarium** (2018). “Microtransactions and Loot Boxes in AAA Games”. 14 de Gener. Plarium Blog, Disponible online a: <https://plarium.com/en/blog/triple-a-games-microtransactions-loot-boxes/>
- **Gerken, Tom** (2018). “Video game loot boxes declared illegal under Belgium gambling laws”. BBC News, 26 d’Abril. Disponible online a: <https://www.bbc.com/news/technology-43906306>
- **Marin, Eduardo** (2018). “El Netflix de Videojuegos ya llegó y ni nos dimos cuenta”. Gizmodo, 29 de Novembre. Disponible online a: <https://es.gizmodo.com/el-netflix-de-videojuegos-ya-llego-y-ni-nos-dimos-cuent-1830750986>

- **Horti, Samuel** (2018). “EA wants cross-platform play for its big franchises”. PCGamer, 28 de Juliol. Disponible online a:
<https://www.pcgamer.com/ea-wants-cross-platform-play-for-its-big-franchises/>

10.3 Websites:

- Versió online ParaTrooper:
<https://classicreload.com/paratrooper.html>
- Pàgina de ParaTrooper a Wikipedia:
[https://en.wikipedia.org/wiki/Paratrooper_\(video_game\)](https://en.wikipedia.org/wiki/Paratrooper_(video_game))
- Pàgina de Sabotage a Wikipedia:
[https://en.wikipedia.org/wiki/Sabotage_\(video_game\)](https://en.wikipedia.org/wiki/Sabotage_(video_game))
- Google Developer Documentation:
<https://developer.android.com/docs>
- Google Play Console:
<https://developer.android.com/distribute/console?hl=es>
- Generador de polítiques de privacitat personalitzades per aplicacions:
<https://app-privacy-policy-generator.firebaseio.com/>
- Servei de publicació gratuïta de pàgines web:
<https://es.wix.com/>
- Unity Scripting Reference:
<https://docs.unity3d.com/ScriptReference/index.html>
- Unity Tutorials:
<https://unity3d.com/learn/tutorials>
- Unity Asset Store:
<https://assetstore.unity.com/>

- 10 Trends in Game Design:
<https://www.gamedesigning.org/gaming/trends/>
- UOC. “La dedicación al estudio”.
<http://cv.uoc.edu/estudiant/acollida/es/grau/pindoles/dedicacio.html>
- Web de l’EEES (Espacio Europeo de Educación Superior).
<http://www.eees.es/es/eees>

10.4 Assets per Unity:

- Missile Launcher (Duane’s Mind).
<https://assetstore.unity.com/packages/3d/props/weapons/3759>
- Military Helicopter II(Duane’s Mind).
<https://assetstore.unity.com/packages/3d/vehicles/air/9472>
- Standard Assets (Unity Technologies).
<https://assetstore.unity.com/packages/essentials/asset-packs/32351>
- Parachute.3ds a Turbosquid. (JensPR)
<https://www.turbosquid.com/3d-models/free-parachute-s-e-3d-model/357093>
- Càrrega Explosiva (Yughues Free Bombs).
<https://assetstore.unity.com/packages/3d/props/weapons/yughues-free-bombs-13147>
- Efectes sonors. Freesound.org.
 - o Paracaigudes. Parachute (Benboncan)
<https://freesound.org/people/Benboncan/sounds/72853/>
 - o Canó. Bomb – Small (Zangrutz)
<https://freesound.org/people/Zangrutz/sounds/155235/>
 - o Helicòpter. Military helicòpter (Benboncan)
<https://freesound.org/people/Benboncan/sounds/74242/>
 - o Avió. Plane (Uncle Sigmund)
<https://freesound.org/people/UncleSigmund/sounds/36929/>
 - o Paracaigudista ferit. Grunt (Joel Audio)
<https://freesound.org/people/JoelAudio/sounds/135855/>

- Caiguda paracaigudista. Scream (Team Masaka)
<https://freesound.org/people/TeamMasaka/sounds/197429/>
- Bomba o explosió. HugeExplosion (CGEffex)
<https://freesound.org/people/CGEffex/sounds/93835/>
- Musica de fons. Absolutely Free Music. (Vertex Studio)
<https://assetstore.unity.com/packages/audio/music/absolutely-free-music-4883>