

# **Projecte/Treball Fi de Carrera**

**Estudi:** Eng. Tècn. Informàtica de Gestió. Pla 2001

**Títol:** Desenvolupament de prototips  
(Android/LibGDX i Unity), per a la producció d'un  
videojoc d'acció lateral amb plataformes (Ouroboros)

**Document:** Resum

**Alumne:** Jaume Calm Serra

**Director/Tutor:** Gustavo Patow

**Departament:** Informàtica i Matemàtica Aplicada

**Àrea:** LSI

**Convocatòria:** Juny 2014

# 1 Introducció

Actualment la indústria dels videojocs mou més diners que la indústria del cinema i la música junts. Per exemple, als EEUU la música va arribar gairebé als 7 bilions de USD(\$), el cinema 10.6 i els videojocs amb gairebé 23; tot l'any 2013. En tot el món, els ingressos vas ascendir a més de 70 bilions amb una audiència estimada de 1.231 milions de jugadors.

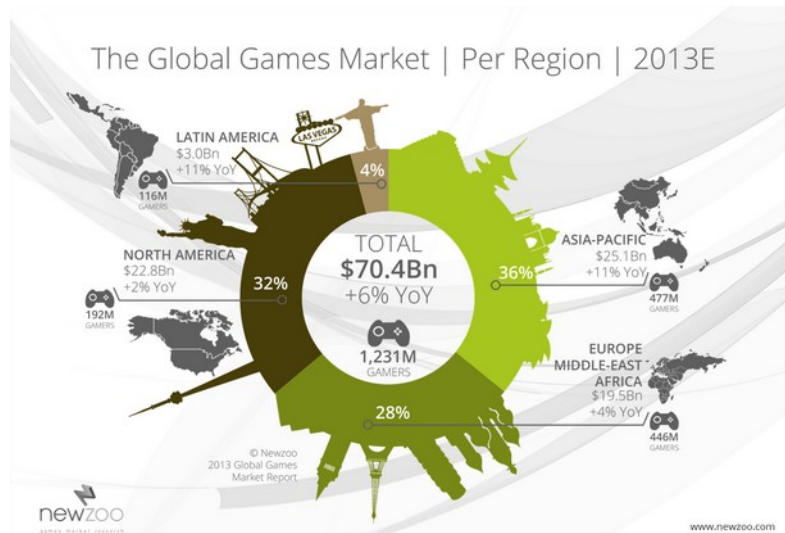


Figura 1: El mercat global de videojocs per regions

Els jocs d'acció són un gènere de videojocs que emfatitzen desafiaments, com per exemple, la coordinació ull-mà i el temps de reacció.

En un joc d'acció, el jugador controla un **avatar** del protagonista. Aquest avatar ha de navegar per un **nivell**, superant obstacles i batallant contra enemics amb diferents atacs. Al final del nivell, o d'una sèrie de nivells, el jugador sovint s'ha d'enfrontar a un gran enemic, típicament anomenat "**Boss**", que és més complexe i desafiant que altres enemics. Els atacs dels enemics i els obstacles redueixen la vida, o el nombre de vides de l'avatar, i el joc s'acaba quan el jugador es queda sense vides, o quan el jugador supera amb èxit tots els nivells del joc.

## 2 Propòsit i objectius

L'objectiu d'aquest projecte és la creació de dos **prototips**, un per **Android** i l'altre per **Unity**, establint les bases per a la producció d'un **videojoc** d'acció lateral (Beat 'em up) amb plataformes (puzles) anomenat "**Ouroboros**".

Android és un sistema operatiu basat en Linux, dissenyat primerament per mòbils tàctils (smartphones) i tabletetes. En concret s'utilitzarà el SDK (Software Development Kit) dins de l'entorn de programació **Eclipse** amb llenguatge **Java**, i les bases d'un **framework** anomenat **LibGDX**.

Unity, en canvi, és un motor de videojocs multi-plataforma amb un entorn de desenvolupament integrat, del que nosaltres utilitzarem la versió en **Javascript**.

Es volen explorar les dues plataformes per tal d'esbrinar quina de les dues vies és la més idònia de cares a la producció final d'un joc.

### 3 Motivació Personal

Des de que era molt petit sempre m'han fascinat els videojocs, segurament pel fet de poder explorar i controlar entitats d'un món imaginari ple d'elements atípics i sorprenents.

Ja més endavant m'han continuat fascinant, sobretot pel fet que agrupen les dues professions o camps que més m'han agradat: la informàtica i el disseny artístic.

El meu objectiu, algun dia, és de poder crear la meua pròpia empresa i poder-me guanyar la vida dissenyant videojocs.



*Figura 2: Logo de l'estudi de videojocs que s'està creant*

### 4 Marc de treball i conceptes previs

En el desenvolupament de videojocs normalment s'utilitza un motor de joc (Game Engine) degudament adaptat a les necessitats dels desenvolupadors, de manera que puguin optimitzar la creació del videojoc sense preocupar-se de com el joc interactuarà amb el hardware del dispositiu pel que s'estigui creant.

Per aquest projecte s'han diferenciat clarament dues vies o opcions a seguir:

- Desenvolupar un motor específic pel videojoc que volem crear (LibGDX).
- Utilitzar un motor ja establert sobre el que treballar (Unity).

#### **4.1 LibGDX**

Aquest framework, en el seu moment, només constava d'una sèrie de llibreries o classes, donant com a punt de partida un parell d'aplicacions d'exemple, i és basat en el popular sistema operatiu per a mòbils Android. D'aquesta manera es va començar a treballar; es va agafar l'exemple d'un senzill joc de plataformes i es va començar a desenvolupar, estudiant-lo i modificant-lo. Aquesta via ens permetrà obtenir una vista completa de tot el procés de creació del joc, ja que partirem pràcticament de zero i haurem de construir/modificar bona part del motor per adequar-lo a les nostres necessitats.

#### **4.2 Unity**

Aquesta opció ens permetrà no haver-nos de preocupar pel desenvolupament de parts específiques del motor, i centrar-nos purament en la codificació del joc. Disposa d'un sistema molt fàcil per generar prototips, gràcies a una interfície gràfica molt visual, on es disposen tots els elements simplement arrastrant-los a l'escena i editant les seves característiques posteriorment amb l'ajuda d'scripts.

### **5 Diagrames de classes (Android/Libgdx)**

El diagrama de classes mostrat (Figura 3), conté les classes més destacades del prototip. S'han remarcat amb un fons blavós aquelles que formen part del prototip creat, mentre que la resta provenen de les llibreries auxiliars utilitzades.

### **6 Diagrames de classes (Unity)**

Unity és un motor de disseny de videojocs complet i no hem entrat en la modificació de la seva arquitectura a diferència del que hem realitzat amb LibGDX. Unity es basa en la **composició d'objecte** i tots els elements de Unity son **Game Objects**, als quals se'ls hi afegeixen diferents components per modificar-ne el comportament. Aquests components estan basats en arxius **Javascript** (Figura 4).

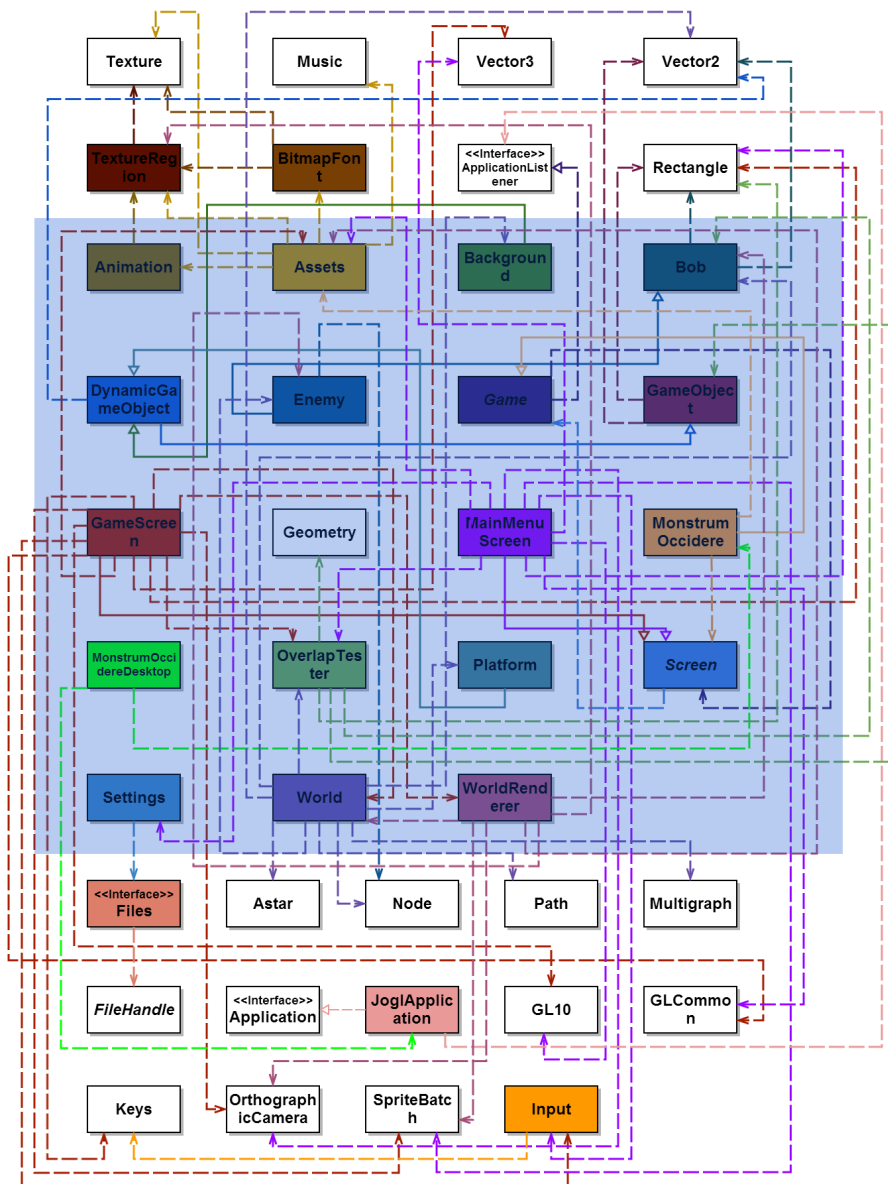


Figura 3: Diagrama de classes, Android/LibGDX

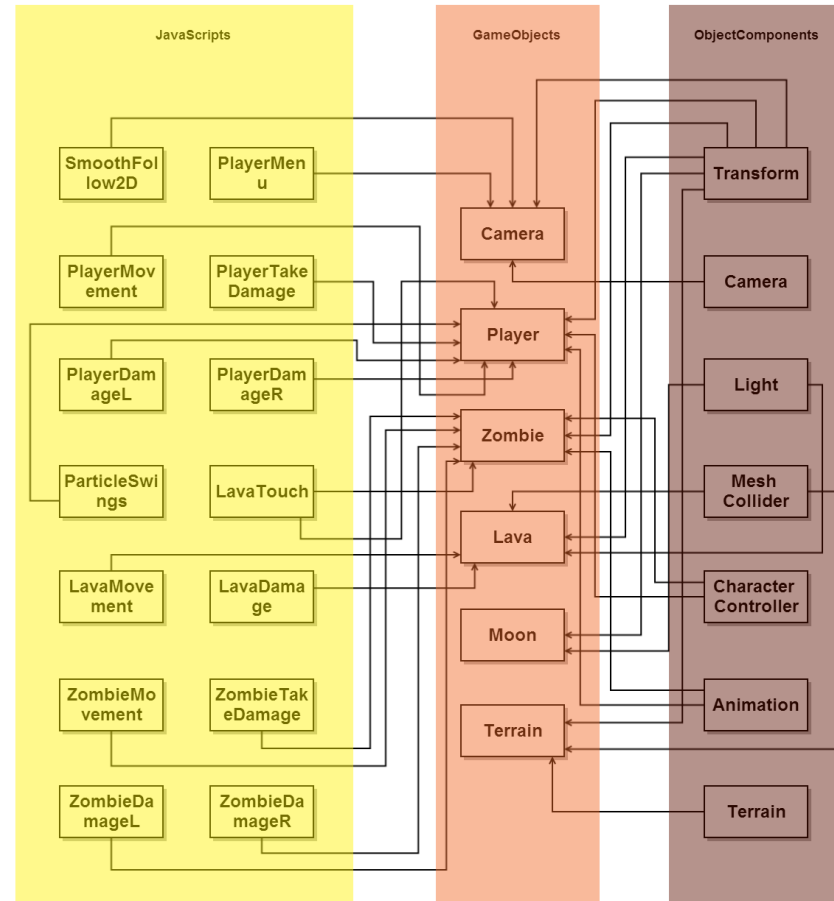


Figura 4: Relacions entre Javascripts, GameObjects i ObjectComponents de Unity

## 7 Resultats

A continuació es mostren dues pantalles dels prototips realitzats.



*Figura 5: Pantalla de joc LibGDx, atacant un enemic*



*Figura 6: Pantalla de joc, enfrontament contra enemics*

## 8 Conclusions

En el dos casos que ens ocupen, podem veure que pel prototip fet amb LibGDx no necessitem d'una infraestructura i uns recursos que vagin més enllà del temps que dediquem a generar dissenys nous (sprites) i animacions, ja que no són més que dibuixos. En canvi pel prototip de Unity necessitaríem d'una infraestructura de programari i instal·lacions que suposarien una gran inversió, ja que requeriríem d'una sala de captura de moviment i un programari molt costós, però que seria suficientment versàtil per arribar a desenvolupar un producte similar al realitzat amb LibGDx sense tanta inversió.

En quant a eficàcia, el prototip LibGDx és ideal per poder controlar al 100% el comportament del nostre personatge, però per contrapartida el motor de col·lisions dissenyat per ell no ha resultat ésser útil per més d'una instància en el cas dels enemics. Per altre banda, Unity ofereix un sistema que compleix amb les especificacions del sistema de col·lisions que havíem pensat, i és totalment robust.

Al final decidim quedar-nos amb la plataforma Unity per ser un sistema compacte en constant actualització, per donar moltes facilitats a l'hora de crear contingut nou, per ser versàtil quan es vol crear un joc per diferents plataformes i per la facilitat de resoldre dubtes i problemes gràcies a la gran comunitat d'usuaris que té al darrere.