|  |
| --- |
| [Escriba el nombre de la compañía] |
| Trabajo de Fin de Grado |
| Aplicación de los videojuegos a la medida de la inteligencia |
|  |
| **Alejandro Ortega Álvarez** |
| **Curso 2019/2020** |

|  |
| --- |
| TFG del grado de Desarrollo de Videojuegos de la Universidad Complutense de Madrid |



# Indice

Objetivo

2.Planificacion

2.1 El juego:Edge

3.Desarrollo

# Objetivo

Una gran parte de los videojuegos diseñados para plataformas móviles (sobre todo) están destinados a resolver puzzles o niveles donde se requiere habilidad y precisión. Estos juegos pueden ser muy distintos entre sí mismo pero todos tienen una característica propia común: necesitan un avance progresivo notable en la dificultad de los desafíos.

Este aspecto hace que el jugador tenga que ir memorizando ciertos patrones propios del juego, con los que poder superar los nuevos retos que vayan apareciendo. Se podría decir, pues, que el jugador tendrá que desarrollar cierta inteligencia si quiere afrontar desafíos más complejos.

El objetivo de este proyecto es crear un videojuego desde 0 en Unity ya existente para alguna plataforma (para no tener que preocuparse del diseño de los niveles) que tenga este aspecto ya mencionado, con el fin de realizar un pequeño experimento en el que participen jugadores reales para ir observando y midiendo el desarrollo de la inteligencia mientras juegan.

# Planificación

Somos 3 integrantes en este proyecto de fin de grado. Lo primero de todo era darnos a conocer el objetivo y las distintas partes de proyecto que íbamos a llevar a cabo.

Después, tocaba elaborar una pequeña planificación para que pudiéramos repartir el trabajo lo más equitativamente posible entre los integrantes del grupo y entre el tiempo del que disponíamos. La primera parte fue fácil, ya que cada integrante debe desarrollar un juego por separado, por lo que esa tarea repartía ya bastante equitativamente trabajo para un considerable periodo de tiempo.

Para la segunda parte también fue fácil, de hecho no hizo falta planificarla demasiado. Al disponer de 2 cuatrimestres para realizar el TFG, nuestra idea fue la siguiente:

* El primer mes lo empleamos en planificación, investigación de fuentes que nos pudieran resultar útiles en el desarrollo; y en la elección de cada uno de los juegos a desarrollar. Para facilitarnos el desarrollo, los juegos a desarrollar serían juegos que ya existen, y que podrían ser de distintas plataformas (ordenador, Tablets, móviles, Wii); con el objetivo de ahorrarnos tiempo en diseño de niveles, al tener que reconstruir tal cual los de los juegos ya existentes y no tener que inventárnoslos.
* El resto del primer cuatrimestre nos centraríamos principalmente en el desarrollo del juego elegido para Unity, con tutorías cada 2-3 semanas con los profesores para ir informando del avance del juego y poder ir resolviendo los posibles inconvenientes que surgieran con la funcionalidad del juego o con programación. Esta tarea, al ser tan extensa, la teníamos que subdividir en varias sub-tareas que serían las siguientes:

1. Lo primero de todo plantear cómo iba a organizarze el proyecto de Unity de forma cómoda para que fuera fácil, dependiendo del juego, añadir niveles, funcionalidades nuevas o elementos.

Con ello podíamos determinar, por ejemplo, cómo iban a ser las escenas, cuántas serían necesarias y por qué, cuántos managers serían imprescindibles, cuáles de ellos globalmente, cómo se iban a comunicar las escenas entre sí, cómo se iban a guardar los posibles datos, etc.

Aunque posiblemente cambiara gran parte en el desarrollo del juego, pero es bueno ir teniendo una idea en la que centrarse.

1. Una vez fijada la idea, comenzar el desarrollo en base a esa idea. Aunque los juegos no eran especialmente complicados, siempre iban surgiendo obstáculos que requerían un cambio en alguna de las ideas o incluso, en ocasiones, tener que prescindir de funcionalidades o elementos del juego si su coste era elevado y vencía en peso a su importancia en el juego.
2. Una vez desarrollada una parte considerable del juego, tendríamos que programar la parte de guardado de datos que nos indicara la psicóloga que nos ayuda con el proyecto. Estos datos son las variables que nos interesan registrar en el juego para la medición del objetivo del proyecto sobre el juego.
3. Al tener el juego “acabado” (los niveles que se nos indique que sean necesarios), debemos realizar una pequeña fase de testeo para resolver todos los posibles bugs que pudieran afectar al jugar e introducir, si da tiempo, algunos detalles visuales o arreglar elementos estéticos que no hubiera dado tiempo en el desarrollo.

* Antes de comenzar el experimento, lo del server y “testeo” para ver que funciona bien el juego y tal.
* Experimento

# Elección de los juegos

Nuestro profesor nos proporcionó información con una lista de juegos que podíamos desarrollar para este proyecto.

Tal como nos indicó la psicóloga, hay tres aspectos a tratar en la medida de la inteligencia:

- Gf: fluid reasoning o razonamiento fluido, es la capacidad de pensar con flexibilidad (fluidez) a la hora de resolver problemas.

- Gv: visuospatial ability o habilidad visoespacial es la capacidad de representar, analizar y manipular objetos en la mente.

- Gs: processing speed o velocidad de procesamiento, referido a la capacidad de poder realizar acciones con mayor velocidad cuando son aprendidas por repetición.

En cada uno de los juegos, hay al menos una de estas características posibles que corresponde con la que tendríamos que medir posteriormente en el análisis de resultados.

Cada integrante debería elegir 1 distinto con total libertad, siempre cuando se acondicionara a :

* Cada integrante debería desarrollar un juego distinto.
* Cada integrante debía escoger un juego que tuviera un aspecto a medir que no tuviera otro compañero, para así poder medir todos los aspectos.
* Cada integrante debía considerar las posibles complicaciones que pudiera tener el juego para el desarrollo en Unity, ya que el construir el juego desde 0, al ser lo que más tiempo nos lleve y no la parte esencial del proyecto, debería tener el mínimo de complicaciones y poder hacerse sin problema alguno. Con este punto tuvimos que descartar más de un juego de la lista.

**Los juegos**

La lista de juegos que se nos proporcionó fue la siguiente:

* **Blek (Gf):** Juego 2D para móviles donde el jugador ha de realizar trazas con el dedo para poder resolver puzles.

[http://blekgame.com](http://blekgame.com/)

* **Rail maze (Gf):** Juego 2D para móviles donde el jugador tiene que completar un escenario de vías de tren y otros elementospara que el tren pueda llegar al destino.

<https://apps.apple.com/es/app/rail-maze-train-puzzler/id445853367>

* **Ski jumping pro (Gs):** Juego 3D para móviles que es un simulador de saltos de esquí.

<https://apps.apple.com/us/app/ski-jumping-pro/id585599497>

* **Unpossible (Gs):** Juego 3D para móviles y tablets donde el jugador se desplaza por un circuito cilíndrico tratando de esquivar obstáculos girando la cámara mientras va aumentando la velocidad.

<https://apps.apple.com/us/app/unpossible/id583577503>

* **Art of Balance (Gf/Gv):** Juego 3D para WiiU que consiste en tratar de encontrar el equilibrio apilando varios objetos sobre una plataforma.

<https://www.nintenderos.com/2014/10/analisis-art-of-balance-eshop-wii-u/>

* **Crazy pool (GV):**    Juego 2D de física simulador de billar.

<https://apps.apple.com/us/app/crazy-pool-3d-free/id391267734>

* **Edge (Gv):** Juego 3D para WiiU y móviles donde el jugador ha de atravesar un escenario manejando un cubo.

<https://en.wikipedia.org/wiki/Edge_(video_game)>

Nuestras elecciones fueron las siguientes:

-Ming: Unpossible, versión para móvil (Gs)

-Pablo: Blek, versión para móvil (Gf)

-Yo: Edge, versión para ordenador (Gv)

# El juego: Edge

Edge es un juego originario de Wii U que también se ha desarrollado para móviles y Tablets.

El juego consta de varios niveles o desafíos en los que el jugador controla un cubo que puede moverse en 4 direcciones posibles. El mundo es un escenario 3D compuesto por casillas discretas por las que irá desplazándose el cubo. El objetivo del jugador es tratar de llegar a la casilla final del nivel en el menor tiempo posible consiguiendo unos items o prismas que estarán distribuidos por todo mapa. Al final de cada nivel sale un menú de información con las estadísticas conseguidas dependiendo de la cantidad de prismas cogidos con respecto al total que hubiera, y el tiempo. También se tiene en cuenta el número de “muertes” que haya habido desde que comenzó el nivel (el jugador “muere” cuando cae por un borde del escenario y vuelve a aparecer desde un checkpoint o punto de guardado).

Todos los datos se quedarán guardados y podrán ser superados si se vuelve a repetir el nivel. Se irán desbloqueando nuevos niveles a medida que se vayan completando.

Más datos sobre el juego:

-El cubo puede subir escalones, pero sólo de uno en uno y si no hay obstáculos encima.

-El cubo no podrá moverse si tiene un bloque justo encima de él.

-El cubo puede ser desplazado por obstáculos movibles.

-Hay plataformas que pueden ser accionadas con activadores cuando se cae sobre ellos.

-Existen otros activadores que empujan al cubo unas cuantas casillas en una dirección determinada.

-Hay bloques de suelo quebradizo que se caen al poco tiempo de pasar por encima de ellos. Estos bloques sólo volverán a recolocarse si el jugador muere.

-El nivel puede reiniciarse en cualquier momento, perdiendo el progreso que se tuviera.

-Se podrá ver en todo momento el cronómetro con el tiempo desde que comenzó el nivel.

**Problemas en el desarrollo**

-A nivel de programación, lo único complejo fue el movimiento rotatorio del cubo, ya que era muy particular y dependiente de obstáculos el que se pudiera o no mover. El movimiento del cubo es una rotación sobre uno de sus bordes que hace que se desplace exactamente su longitud en una de las direcciones, de este modo tenemos movimientos discretos. Pero en Unity no era tan sencillo, era necesario hacer correcciones de posición constantemente porque al llevar un tiempo jugando, el jugador al final llegaba a “salirse” de las casillas, sobre todo al subir o bajar peldaños.

-El gran problema al que me enfrenté no fue en cuanto a código, sino un problema con las físicas. El juego se podría decir que carece completamente de físicas (toda comprobación de si es posible o no un movimiento se realiza mediante rayos o raycast), excepto a la hora de introducir obstáculos movibles. Estos obstáculos hacían que el cubo constantemente fuera empujado o desplazado en cualquier dirección de forma no discreta, y no hubiera habido problema alguno si el juego no requiriera ser discretizado para poder resolver algunos puzles (hay una gran cantidad de casillas de las mismas dimensiones que el tamaño del cubo, a las que no se podía acceder en cuanto el cubo se saliera lo más mínimo de su rango dentro de las casillas). Y no servía colocar colliders para que las plataformas empujaran al cubo porque si se usan las físicas de Unity se han de usar para todo el juego, y el resto del juego carece de estas físicas como hemos mencionado. Lo que originó el tener que depender de unas físicas distintas a las de Unity, y tener que estar corrigiendo posiciones constantemente al entrar y salir de plataformas movibles, y tener que prescindir de algunas plataformas que originaban algún efecto complicado de solucionar (tener que escalar plataformas mientras se estaban moviendo, por ejemplo).

**Referencias y enlaces**

Página de información del juego: <https://en.wikipedia.org/wiki/Edge_(video_game)>

Página oficial de Unity: <https://unity.com/es>

Youtube (referencias de vídeos de gameplay): <https://www.youtube.com/>

Enlace de descarga del juego para Android (versión gratuita-demo): <https://play.google.com/store/apps/details?id=net.mobigame.edge.demo&hl=es>

Enlace de descarga del juego para Android (versión de pago): <https://play.google.com/store/apps/details?id=net.mobigame.edge&hl=es>

Enlaces de vídeos de Youtube:

- Trailer (review): <https://www.youtube.com/watch?v=_qmSqiSj61U>

- Niveles 1-10: <https://www.youtube.com/watch?v=Bvyzrr9PeUI&t=195s>

- Niveles 23-25 que corresponden a los 3-5 de este proyecto: <https://www.youtube.com/watch?v=6ZXVhhyHJn8&list=WL&index=5&t=147s>

Enlace de perfil de Github: <https://github.com/alex97ortega>

Enlace del TFG del juego en Github: <https://github.com/alex97ortega/Edge>