
Control Remoto de Videojuegos con Smartphones



MEMORIA DE TRABAJO DE FIN DE GRADO

Pablo Gómez Calvo
Sergio J. Higuera Velasco

Grado de Desarrollo de Videojuegos
Facultad de Informática
Universidad Complutense de Madrid

Mayo 2019

Documento maquetado con T_EX^S v.1.0+.

Este documento está preparado para ser imprimido a doble cara.

Control Remoto de Videojuegos con Smartphones

Memoria de Trabajo de Fin de Grado
Grado de Desarrollo de Videojuegos
Mayo 2019
Director: Carlos León Aznar
Director: Pedro Pablo Gómez Martín

Versión 1.0+

Grado de Desarrollo de Videojuegos
Facultad de Informática
Universidad Complutense de Madrid

Mayo 2019

Copyright © Pablo Gómez Calvo y Sergio J. Higuera Velasco

A todo aquel que confió en nosotros

Agradecimientos

Nadie es innecesario.

Yitán, Final Fantasy IX

El primer agradecimiento hay que dárselo a la Universidad Complutense por aceptar la creación de este grado, un grado que demuestra la importancia del mundo de los videojuegos en la sociedad actual. Con este grado se han conseguido romper muchas barreras, entre ellas está poder especializarse y adoptar los videojuegos como nuestra profesión.

Dar gracias a los profesores que nos han acompañado estos años y que han contribuido en el desarrollo del grado. Una mención aparte para las dos personas que han hecho posible la realización de este Trabajo de Fin de Grado, Carlos León Aznar y Pedro Pablo Gómez Martín.

Nuestro último agradecimiento va dirigido a nuestras familias por soportarnos en todos nuestros momentos durante el grado.

Resumen

¡No estáis preparados!

Illidan Tempestira, World of Warcraft

Índice

Agradecimientos	VII
Resumen	IX
1. Introducción	1
1.1. Introducción	1
1.2. Proposición de la idea	2
1.3. Estructura de capítulos	2
2. Estado del arte	3
2.1. Herramientas para el desarrollo	3
2.1.1. Unity	3
2.1.2. Android Studio	4
2.2. Librerías usadas	5
2.2.1. ZXing	5
2.3. Proyectos Similares	5
2.3.1. Wii U	5
2.3.2. Joystick	5
2.3.3. PlayLink	6
3. Objetivos y especificación	7
3.1. Objetivos	7
3.2. Plan de trabajo	7
3.3. Metodologías	8
3.4. Herramientas utilizadas	8
4. Desarrollo de la aplicación de PC	11
4.1. Arquitectura	11
4.2. Protocolo de conexión Unity y Android Studio	11
4.3. El uso de los QR para inicio de conexión	11
4.4. Pantalla secundaria y controlador	11
4.5. Tecnologías involucradas	11

4.6. Prototipos	11
5. Desarrollo de la aplicación de Android	13
5.1. Pruebas	13
5.2. Resultado	13
5.3. Demo técnica	13
6. Comunicación PC-Android	15
6.1. Discusión	15
7. Conclusiones	17
7.1. Conclusiones Pablo	17
7.2. Conclusiones Sergio	17

Índice de figuras

Índice de Tablas

Capítulo 1

Introducción

RESUMEN: Este capítulo presenta una breve introducción al proyecto. El proyecto consiste en usar un móvil como mando, ha sido desarrollado en Unity ,motor de videojuegos altamente popular. También se encuentra aquí una descripción del resto de capítulos del manual.

1.1. Introducción

La industria del videojuego desde su inicio nos ha enseñado que la innovación a la hora de crear experiencias nuevas para los usuarios es algo que enriquece a muchos jugadores, se está ayudando a que la experiencia de juego sea cada vez más inmersiva.

Es por esta razón que empresas como Electronic Arts, Ubisoft, Kunos Simulazioni y Polyphony Digital,entre otras, dedican gran cantidad de sus recursos a hacer realidad muchas experiencias que los usuarios quieren tener como, por ejemplo, jugar a juegos deportivos realistas como en FIFA o conducir automoviles de competición por un circuito famoso como en Assetto Corsa.

Pero para invertir en crear este nuevo tipo de estilos de juego se necesita una gran financiación, cosa que estudios pequeños no tienen. La mayor fuente de ideas para estos proyectos de creación de nuevas experiencias para los usuarios suelen ser los estudios independientes ya que se suele trabajar en un ámbito donde las ideas son más dadas a aflorar. Estos estudios independientes usan motores como Unity3D o Unreal Engine 4. Estos motores ,cuyo pago es porcentual a las ganancias obtenidas por tu juego o en algunos casos mensual,

ofrecen una gran cantidad de herramientas a disposición de sus usuarios para que los desarrolladores puedan ahorrar tiempo de implementación de una nueva feature y lo utilicen para que su juego siga creciendo. El desarrollo para dispositivos móviles en estudios independientes ha crecido de manera exponencial gracias a que motores como Unity lo hacen bastante accesible.

Entonces, ¿Y si un juego se pudiese jugar en el teléfono móvil pero en verdad el juego se estuviese ejecutando en el ordenador?

1.2. Proposición de la idea

Este trabajo pretende usar un móvil como mando inalámbrico usando como motor Unity. Se busca crear un híbrido entre un juego convencional de ordenador junto con una tecnología, la cual la mayor parte de los usuarios tiene, como son los teléfonos móviles.

La herramienta propuesta se apoyará sobre Unity, motor usado en el 34 por ciento del top mil de juegos de móviles y en el 53 por ciento de juegos de Oculus Rift en su salida, sirviendo como una prueba de concepto para integrar el uso de un dispositivo móvil a un juego convencional de ordenador.

Una vez terminada, la herramienta podrá ser usada por cualquier desarrollador que lo desee. La herramienta será de licencia libre para su posible ampliación.

1.3. Estructura de capítulos

Este documento está estructurado en los siguientes capítulos:

- El capítulo 2 expone una revisión del estado del arte de las tecnologías involucradas en el desarrollo de este trabajo.
- El capítulo 3 se centra en los objetivos a cumplir en este proyecto junto con la especificación y las metodologías que se van a seguir para la completa realización de este trabajo.
- El capítulo 4 explica todo lo relacionado con el diseño de la herramienta y la posterior implementación de la misma. También incluye la explicación de los prototipos llevados a cabo durante el desarrollo del proyecto.
- El capítulo 5 aborda los resultados obtenidos y la demostración de lo conseguido.
- El capítulo 6 expone la discusión y las conclusiones.

Capítulo 2

Estado del arte

RESUMEN: Este capítulo explica las herramientas que han sido utilizadas en el desarrollo de este trabajo. También se ofrecen unos cuantos ejemplos en los cuales nos hemos basado y nos han llevado a pensar que la idea de este proyecto es algo útil.

2.1. Herramientas para el desarrollo

En este proyecto se ha unificado el uso de varias tecnologías, en principio independientes, para formar una herramienta que una el desarrollo de videojuegos para ordenador y el uso de una aplicación de Android que nos facilite la conexión del PC con Android. Estas tecnologías/herramientas han sido:

2.1.1. Unity

Unity es un motor de videojuegos multiplataforma creado por Unity Technologies. Se encuentra disponible para sistemas Windows, Mac Os X y Linux. Es una de las herramientas de desarrollo de videojuegos más populares actualmente en el mundo de los desarrolladores independientes. También cuenta con una gran cantidad de documentación generada tanto por los diversos usuarios como por sus creadores. Otra opción barajada era Unreal Engine 4; este motor es más potente que Unity y también es altamente usado por las desarrolladoras.

Las principales características buscadas son:

- **Multiplataforma.** El hecho de que Unity y Unreal sean sistemas multiplataforma, te garantiza poder hacer juegos/aplicaciones que puedan ejecutarse en cualquier dispositivo a un coste bastante bajo en cuanto a esfuerzo.
- **Potencia.** Unreal Engine 4 es más potente que Unity gracias a su fotorrealismo y trabajo de gráficos hiperrealista como en la demo de Star Wars creada por los desarrolladores del motor, aun así Unity no es, ni mucho menos, un motor mediocre ya que consigue buenos acabados gracias al uso de diferentes shaders y filtros. Lo decisivo es que Unity es un motor ya dado en la universidad que facilitaba el uso del mismo y, que para el objeto de este proyecto, su potencia ya es más que suficiente.
- **Documentación.** El caso de Unity en cuanto a documentación es algo que los desarrolladores de Unity mantienen funcional de manera continua. Puedes acceder a la documentación de una manera muy intuitiva desde la misma página web de Unity y en los últimos meses están incluyendo proyectos de juegos completos de principio a fin.
- **Comunidad.** Debido a que Unity es un motor de videojuegos potente y gratuito, cuenta con un gran número de desarrolladores que saben utilizarlo, lo que facilita la solución de problemas más específicos.

2.1.2. Android Studio

Android Studio es el entorno de desarrollo oficial de la plataforma Android en contrapartida también cabe la posibilidad de descargar las Android SDK sin necesidad del entorno completo de Android Studio. Fue desarrollada por Google y sustituyó a Eclipse en el desarrollo de aplicaciones para dispositivos Android. Las SDK de Android son imprescindibles para la creación de una aplicación Android, las SDK ofrecen un Emulador para poder ver tu aplicación en ejecución. Android Studio tiene la función de encapsular estas SDK de Android y poner en marcha la aplicación. Este IDE puede utilizarse tanto en Windows, Mac OS X y Linux pero únicamente puede usarse para el desarrollo de aplicaciones para Android.

Las principales características de Android Studio son:

- **Específico.** Si el producto que quieres desarrollar va a ser exclusivo de un sistema Android, con Android Studio te vas a centrar en explotar las funcionalidades de ese sistema al máximo sin tener que lidiar con diferentes sistemas que no te interesan.
- **Editor de diseño.** Android Studio cuenta con un editor visual para poder acomodar el layout de tu aplicación Android de una manera mucho más sencilla.

2.2. Librerías usadas

2.2.1. ZXing

Este es un proyecto del tipo Open-Source. Esta librería de procesamiento de imágenes de códigos de barras y QR's está implementada en Java y tiene diferentes versiones en los distintos lenguajes. En nuestro caso hemos usando la versión de .NET para usarla desde Unity. Las principales características de ZXing son:

- **Versatilidad.** Al ser una librería que tiene soporte en muchos otros lenguajes que no son Java, puedes utilizarla en tus proyectos multiplataforma si estas plataformas usan diferentes leguajes como en nuestro caso (Android Studio y Unity).

2.3. Proyectos Similares

Hay algunas empresas que han invertido mucho en innovar y crear nuevas formas para que los usuarios disfruten de los diferentes juegos. Algunos de los ejemplos que ya existen en el mercado y han servido de inspiración para la realización de este proyecto son:

2.3.1. Wii U

Wii U es una consola de Nintendo de la octava generación. Esta consola de sobremesa incluía una pantalla portátil que servía a la vez de pantalla secundaria y mando. Esta pantalla portátil es táctil y recibe una señal de 480p. A este nuevo mando se le denomina Wii U GamePad. Este nuevo mando permitía a los desarrolladores tener un HUD mucho más limpio dentro del juego ya que, en muchos casos, esta pantalla era utilizada para poner elementos como el minimapa.

2.3.2. Joystick

El joystick es un periférico de entrada que consiste en una palanca que tiene libre movimiento sobre una base. Este informa sobre su ángulo y dirección al dispositivo al que esté conectado. Los joysticks se han utilizado siempre en el mundo de los videojuegos, empezando por la Atari hasta llegar al DualShock de PlayStation que hay actualmente.

2.3.3. PlayLink

PlayStation es una de las compañías que más tráfico de jugadores mueve llegando a la cifra de 90 millones de usuarios activos mensuales, sus consolas son de las más vendidas en todo el mundo y para que esto sea posible siempre tienen que intentar estar a la cabeza de nuevos periféricos, nuevos juegos y, por supuesto, darles a sus usuarios las mejores experiencias posibles. En 2017, Sony PlayStation sacó al mercado una nueva serie de juegos llamados PlayLink. Estos juegos tienen una particularidad con respecto a un juego convencional de consola, estos juegos están hechos para jugarlos con gente y usando un dispositivo móvil como mando/controlador. Conectando la consola y el móvil a la misma red WIFI y conectándolos a través de una aplicación, se pueden conectar de 2 a 8 jugadores, depende de los que admita cada juego, para poder jugar en familia o con amigos.

Capítulo 3

Objetivos y especificación

RESUMEN: Este capítulo se centra en exponer los objetivos a cumplir por este proyecto, tales como la creación de la librería (Unity), la aplicación (Android) y la comunicación entre ambas ayudado por las metodologías y herramientas como Scrum y Github.

3.1. Objetivos

Con este proyecto se han establecido los siguientes objetivos generales:

- Crear una librería para Unity3D que se comunique con una aplicación Android usando una comunicación Cliente-Servidor, Unity3D(Servidor), el cual se encargará de recibir los inputs del Android(Cliente).
- Crear la aplicación Android para usar este como mando virtual.
- Crear un prototipo de lo que hemos sido capaces de desarrollar y demuestre las capacidades de la herramienta desarrollada.

En las próximas subsecciones plantearemos la metodología y el plan de trabajo seguido.

3.2. Plan de trabajo

Para la realización de este TFG se unificarán muchos aspectos que se han ido planteando a lo largo del Grado, tales como Unity3D, Android y

desarrollo de protocolos tipo Cliente-Servidor.

Los primeros meses serán dedicados a la investigación y el estudio de las herramientas ya existentes que exploran los aspectos en común con este TFG. Se usará Github como sistema de control de versiones, donde estarán 2 repositorios: uno para la memoria y otro donde se encontrará el código de la demo.

Las reuniones con los directores serán cada 2 semanas. El objetivo de estas reuniones será llevar un control de lo que se haya avanzado y plantear las próximas 2 semanas de trabajo. Se han marcado unos hitos específicos:

1. **Navidad.** Sobre estas fechas deberemos tener la conexión entre Android y Unity3D establecida. Con esto se pretende que un personaje en Unity3D sea capaz de moverse gracias al Input que recibe desde Android.
2. **Semana Santa.** Se deberá de haber conseguido tener una cámara en Unity3D enviando una imagen via streaming al dispositivo Android mientras se juega con este.
3. **Pre-Exámenes Junio.** Aquí se debe tener una demo que sea capaz de dostrar las capacidades de la herramienta.

3.3. Metodologías

La metodología utilizada para la realización de este proyecto está basada en el desarrollo ágil. Se han marcado pequeños objetivos de 2 semanas de duración llamados sprints hasta llegar a las metas grandes o hitos mencionados en el apartado anterior.

La metodología que se iba a usar estaba definida desde el primer día ya que es la que los miembros del proyecto siguen en todos los trabajos que realizan. Se decidió usar **Scrum**. Scrum es una metodología ágil que adopta una estrategia de desarrollo incremental, en lugar de la planificación y ejecución completa del proyecto en cuestión. La particularidad que se tiene en este proyecto es el constante cambio de las tareas nuevas que se pueden llegar a incluir tras cada reunión.

3.4. Herramientas utilizadas

En la sección 3.1 se ha hablado de las herramientas que se acordaron utilizar para la realización de este proyecto:

1. **MiKTeX y LaTeX**

Las características más apreciables de MiKTeX son su habilidad de actualizarse por sí mismo descargando nuevas versiones de componentes y paquetes instalados previamente, y su fácil proceso de instalación.

2. **GitHub**

GitHub es una plataforma de desarrollo cooperativo en la que se pueden alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza para la creación y almacenamiento de código fuente de manera pública. La herramienta Git se ha usado para mantener un control de versiones sobre 2 repositorios, uno para la demo y otro para la memoria.

Capítulo 4

Desarrollo de la aplicación de PC

RESUMEN: Este capítulo describe todos los aspectos relacionados con las imágenes de los documentos. En particular, describe la estructura de directorios que T_EX_S aconseja, así como los aspectos relacionados con la diferencia entre los formatos esperados cuando se genera el documento final con `latex` y `pdflatex`.

4.1. Arquitectura

4.2. Protocolo de conexión Unity y Android Studio

4.3. El uso de los QR para inicio de conexión

4.4. Pantalla secundaria y controlador

4.5. Tecnologías involucradas

4.6. Prototipos

Capítulo 5

Desarrollo de la aplicación de Android

RESUMEN: Este capítulo describe todos los aspectos relacionados con las imágenes de los documentos. En particular, describe la estructura de directorios que T_EX_S aconseja, así como los aspectos relacionados con la diferencia entre los formatos esperados cuando se genera el documento final con `latex` y `pdflatex`.

5.1. Pruebas

5.2. Resultado

5.3. Demo técnica

Capítulo 6

Comunicación PC-Android

RESUMEN: Este capítulo describe todos los aspectos relacionados con las imágenes de los documentos. En particular, describe la estructura de directorios que `TEXIS` aconseja, así como los aspectos relacionados con la diferencia entre los formatos esperados cuando se genera el documento final con `latex` y `pdflatex`.

6.1. Discusión

Capítulo 7

Conclusiones

RESUMEN: Este capítulo describe todos los aspectos relacionados con las imágenes de los documentos. En particular, describe la estructura de directorios que `TeXIS` aconseja, así como los aspectos relacionados con la diferencia entre los formatos esperados cuando se genera el documento final con `latex` y `pdflatex`.

7.1. Conclusiones Pablo

7.2. Conclusiones Sergio

*–¿Qué te parece desto, Sancho? – Dijo Don Quijote –
Bien podrán los encantadores quitarme la ventura,
pero el esfuerzo y el ánimo, será imposible.*

*Segunda parte del Ingenioso Caballero
Don Quijote de la Mancha
Miguel de Cervantes*

*–Buena está – dijo Sancho –; fírmela vuestra merced.
–No es menester firmarla – dijo Don Quijote–,
sino solamente poner mi rúbrica.*

*Primera parte del Ingenioso Caballero
Don Quijote de la Mancha
Miguel de Cervantes*

