

laSalle

UNIVERSITAT RAMON LLULL

Escola Tècnica Superior d'Enginyeria La Salle

Treball Final de Màster

Màster Universitari en Creació, Disseny i Enginyeria Multimèdia

Drappen's World: programación lógica y gráfica desarrollo para smarthone y base de datos

Alumne
Bartolomé Rosselló Ramos

Professor Ponent
Oscar García Pañella

ACTA DE L'EXAMEN DEL TREBALL FI DE CARRERA

Reunit el Tribunal qualificador en el dia de la data, l'alumne

D. Bartolomé Rosselló Ramos

va exposar el seu Treball de Fi de Carrera, el qual va tractar sobre el tema següent:

Drappen's World: programación lógica y gráfica desarrollo para smarthone y base de datos

Acabada l'exposició i contestades per part de l'alumne les objeccions formulades pels Srs. membres del tribunal, aquest valorà l'esmentat Treball amb la qualificació de

Barcelona,

VOCAL DEL TRIBUNAL

VOCAL DEL TRIBUNAL

PRESIDENT DEL TRIBUNAL

Abstract

Drappen's World es un serious game realizado como proyecto final de máster por TheraGames, un grupo formado por cuatro estudiantes. Este proyecto se realiza en el marco del convenio existente entre la empresa LabPsiTec y la facultad de La Salle, con el objetivo de combatir la obesidad infantil. Drappen's World pretende que el usuario haga ejercicio en casa y también incentivarlo a practicar deporte al aire libre. Se han utilizado una cámara de captura de movimiento llamada Kinect, un GPS embebido en un teléfono móvil y una camiseta sensorial para monitorizar el estado físico del jugador. Este documento describe el proceso de desarrollo de la aplicación, haciendo énfasis especial en la programación lógica y gráfica, programación de móviles de última generación, así como las cuestiones referentes a la base de datos.

Drappen's World és un serious game realitzat com a projecte final de màster per TheraGames, un grup format per quatre estudiants. Aquest projecte es realitza en el marc del conveni existent entre l'empresa LabPsiTec i la facultat de LaSalle, amb l'objectiu de combatre l'obesitat infantil. Drappen's Woeld pretén que l'usuari faci l'exercici a casa i també pretén incentivar-lo a practicar l'esport a l'aire lliure. S'han utilitzat una càmera de captura sensorial anomenada Kinect, un GPS integrat a un telèfon mòbil i una camiseta sensorial per monitoritzar l'estat físic del jugador. Aquest document descriu el procés de desenvolupament d'una aplicació, fent èmfasi especial en la programació lògica i gràfica, programació de mòbils de darrera generació, així com les qüestions referents a la base de dades.

Drappen's World is a serious game produced as a mater thesis by TheraGames, a team of four students. Under the banner 'fighting childhood obesity' the project got developed in the frame of the existing agreement between the company LabPsiTec and the faculty of La Salle. Drappen's World expects the user to do exercises at home and practise open air sports aswell. Several devices are used to monitorize the player's activity: a Kinect motion capture camera, a GPS embedded in a cell phone and a sensory t-shirt. This document explains the development process of the application, emphasizing the logical and graphical programming, the last generation cell phone programming and issues related to the database.

Resumen

Theragames es un grupo de estudiantes de La Salle (Universidad Ramón Llull) formado por Pau Moreno (técnico - scrum master), João Caldas (diseñador de interfaces), Adela Gutiérrez (diseñadora gráfica) y el autor de este documento, Bartolomé Rosselló Ramos (técnico - programador). El grupo tiene el objetivo principal de realizar el proyecto final del Máster en Creación Diseño e Ingeniería multimedia. Para ello, realizan un proyecto para la el Laboratorio de Psicología y Tecnología (LabPsiTec), de la Universidad Jaume I de Castellón y de la Universidad de Valencia, los cuales tienen convenio con la universidad.

El proyecto encargado por LabPsiTec se trata de un prototipo de serious game llamado Drappen's World. El objetivo del juego es combatir la obesidad infantil promocionando la actividad física. El usuario target al que va orientado es a niños de 8 a 14 años, principalmente niños con problemas de obesidad infantil. El serious game va orientado a ser utilizado por terapeutas para incorporarlo en sus terapias contra la obesidad. LabPsiTec pretende obtener un prototipo que le sirva para validar clínicamente el serious game con pacientes reales y poder registrarlos en la aplicación y así poder observar su evolución.

Drappen's World pretende hacer que el usuario realice ejercicio de forma divertida, haciendo que el jugador se olvide en el mayor grado posible del objetivo principal: la práctica de actividad física. El juego está pensado para que el ejercicio se practique dentro de casa y a la vez incentivar al usuario, en varias fases del juego, a salir al aire libre para continuar con el juego y realizar ejercicio en el exterior.

Se ha utilizado una cámara de captura de movimiento llamada Kinect, para controlar en todo momento los movimientos del jugador y comprobar si está realizando bien los ejercicios propuestos por el juego. El ejercicio al aire libre consta de desplazarse caminando o corriendo una cierta distancia y esto es controlado con un GPS embebido en un teléfono móvil. Además también se utiliza otro tipo de tecnología especial, se trata de una camiseta sensorial para monitorizar el cansancio del jugador.

Como se ha comentado, este documento está realizado por el técnico programador de TheraGames, por tanto, describe el proceso de desarrollo de la aplicación, haciendo

énfasis especial en las partes que este ha desarrollado. Estas partes son la programación lógica del serious game, la programación del entorno gráfico, el desarrollo de una aplicación para móvil de última generación, así como las cuestiones referentes a la base de datos.

A continuación, se citan las otras tres memorias que forman parte de la documentación completa de Drappen's World por si el lector tiene interés en profundizar en sus contenidos. Cada una de las memorias tratan del trabajo que ha realizado su autor en Drappen's World. Es importante remarcar que en el momento de desarrollo de este documento las siguientes memorias están el proceso de realización, pero sus temas tratados serán los siguientes:

- Memoria de Pau Moreno (técnico - scrum master): En esta memoria se podrá encontrar toda la parte de gestión de producto, y metodologías usadas para ello, así como, progreso y evolución de la mismas. También se podrá encontrar todo lo referente al periférico Kinect, reconocimiento de gestos, y movimiento del personaje del juego a partir de los datos obtenidos de Kinect.
- Adela Gutiérrez (diseñadora gráfica): En su documento se tratarán los aspectos del diseño gráfico tanto del juego, interfaz y personajes como de la imagen del grupo Theragames. Se justificarán las elecciones de las diferentes estéticas y la psicología de los colores.
- João Caldas (diseñador de interfaces): En esta memoria se hablará del diseño de los distintos mini juegos que componen Drappen's World, no a nivel gráfico ni técnico, sino a nivel conceptual. Se explicará cual es la dinámica de juego de cada mini juego, así como la historia y el guión que hay detrás de Drappen's World. También tratará los aspectos de usabilidad.

Índice

Abstract	1
Resumen	2
Índice	4
1. Introducción	8
1.1. Marco.....	8
1.2. Estado del arte.....	9
1.2.1. Aplicaciones GPS en SmartPhones	9
Pygpslog	10
Google Maps Navigation	11
Copilot.....	12
Runstar	13
Cardio Trainer	14
Foursquare.....	15
SCVNGR.....	16
MotionX GPS	17
MobileTracker	18
Waze.....	19
1.2.2. Sensorización en videojuegos.....	20
Wii Vitality Sensor	21
EA Sports Active 2	22
1.2.3. Conclusiones del estado del arte	23
1.3. Descripción del problema.....	24
1.4. Solución propuesta	25
1.5. Perspectiva general del proyecto	26
2. Fundamentos teóricos.....	29
2.1. Motor de juego.....	29
2.1.1. Motores gráficos propietarios	33

Unreal Engine 3.....	33
Cry Engine 2	36
Anvil Engine	39
Geo-Mod Engine 2.0	41
2.1.2. Motores gráficos open source	43
Irrlicht	43
Ogre 3D	45
jMonkey Engine.....	48
Panda 3D	50
2.1.3. Motor elegido: Irrlicht.....	53
2.2. Lenguaje de programación C++	54
2.2.1. Criterio de Selección	54
2.2.2. Características	54
2.3. Entorno de desarrollo: Visual Studio	55
2.3.1. Criterio de Selección	56
2.3.2. Características	56
2.4. Dispositivo de captura: Camiseta sensorial.....	56
2.4.1. Criterio de selección.....	58
2.4.2. Características	58
2.5. Periférico capturador de movimientos: Kinect	59
2.5.1. Criterio de selección.....	60
2.5.2. Características	61
2.6. Smartphone	66
2.6.1. Android	67
Criterio de selección.....	67
Android market.....	73
Características	73

Entorno de desarrollo: Eclipse.....	81
2.7. Tecnologías de servidor web	82
2.7.1. Apache2 http Server	83
Criterio de selección.....	83
Características	84
2.7.2. MySQL.....	86
Criterio de selección.....	86
Características	86
2.7.3. phpMyAdmin.....	87
Criterio de selección.....	87
Características	88
2.7.4. WAMP.....	88
Criterio de selección.....	89
Características	89
3. Parte práctica.....	91
3.1. Diagrama de arquitectura.....	91
3.2. Aplicación PC.....	92
3.2.1. Diagrama de transición de estados	93
3.2.2. Diagramas de módulos software	102
Diagrama de software general.....	103
Diagrama de software del paquete Gestor_Texturas.....	105
Diagrama de software del paquete Conexion_Web.....	106
Diagrama de software del paquete Gestor_Objeto.....	108
Diagrama de software del paquete Camiseta_Sensorial	109
3.2.3. Camiseta sensorial.....	110
3.3. Aplicación mini juego Smartphone	112
3.4. Servidor: Base de datos.....	113
3.4.1. Conexión con la aplicación PC.....	115

3.4.2. Conexión con la aplicación Android	116
4. Resultados.....	117
5. Conclusiones y líneas de futuro	127
6. Índice de figuras.....	129
Bibliografía.....	132

1. Introducción

1.1. Marco

Esta memoria forma parte de un conjunto de memorias que describen el proceso y el resultado de desarrollar un serious game para promocionar la actividad física. Concretamente, este documento ha sido realizado por el técnico del equipo desarrollador.

Se trata de una aplicación desarrollada por un grupo de 4 estudiantes llamado "TheraGames". El proyecto ha sido realizado para una empresa real llamada "LabPsiTec", Laboratorio de Psicología y Tecnología de la Universidad Jaume I de Castellón y de la Universidad de Valencia. Es un trabajo final del "Máster en Creación, Diseño e Ingeniería Multimedia" (MCDEM) de La Salle, Universidad Ramón Llull.

LabPsiTec investiga las posibilidades que las Nuevas Tecnologías de la Información y la Comunicación (TICs) (como la Realidad Virtual, la Realidad Aumentada, Internet, los sistemas móviles, etc.) pueden ofrecer a la Psicología Clínica, como herramientas de evaluación y de tratamiento, así como el estudio de los procesos psicológicos básicos.

En este caso, encargaron a TheraGames un serious game para promocionar la actividad física en niños, principalmente niños obesos. El juego servirá de terapia para niños con problemas de sobrepeso y estará supervisado por un terapeuta. Se pretende que el niño realice ejercicio tanto dentro como fuera de casa. El terapeuta podrá interactuar con el juego programando su dificultad, duración y tipos de ejercicios. El serious game consta de dos partes bien diferenciadas, una parte ejecutada sobre un PC y otra parte ejecutada sobre un Smartphone.

En cuanto a las necesidades técnicas, la primera parte tiene la necesidad de medir los movimientos físicos del jugador, ya que se pretende que realice ejercicio físico en casa, y obtener su nivel de esfuerzo, para mantener un nivel de esfuerzo físico. También tiene que tener registrados todos los datos del niño necesarios para poder jugar, así como, cual es su estado de evolución en el juego. Los datos principales serán los necesarios para la identificación del niño (usuario y contraseña) y la configuración personalizada del

terapeuta. La segunda parte tiene que cubrir la necesidad de medir el desplazamiento del niño en distancias más largas, fuera de casa. Con esto se pretende hacer salir al niño de casa para hacer ejercicio caminando o corriendo.

Para poder satisfacer estas necesidades, la aplicación del PC se sirve de un dispositivo de captura sensorial (frecuencia cardíaca), un seguidor de movimiento y una base de datos. Por otro lado, la aplicación para Smartphone, se sirve de un dispositivo GPS integrado y la misma base de datos anterior. El autor de este documento y técnico del equipo, se encarga de las partes de sensorización, base de datos, GPS y programación de la lógica de juego.

1.2. Estado del arte

El estado del arte hace referencia al nivel más alto de desarrollo conseguido en un momento determinado sobre cualquier aparato, técnica o campo científico. De las necesidades técnicas comentadas en el marco, en este apartado se ha centrado el análisis del estado del arte en las aplicaciones de Smartphone que utilizan GPS y el tema de la captura sensorial en videojuegos; puesto que eran las partes técnicas más novedosas y más desconocidas para el equipo y para el técnico.

Este análisis sirve de inspiración para llevar a cabo el proyecto y ver cuál es el nivel de desarrollo actual de las aplicaciones GPS de SmartPhones y la sensorización en videojuegos. Además, el estado del arte sirve para analizar si existe algún aplicativo que ya cumpla los requerimientos necesarios para la aplicación a desarrollar y ver si se pueden aprovechar componentes de terceros.

1.2.1. Aplicaciones GPS en SmartPhones

En este apartado se analiza el estado del Arte sobre las aplicaciones de smartphone que utilizan GPS. Se citan aplicaciones que están en el mercado para los diferentes sistemas operativos: Android, Blackberry, iOS (ipad e iphone), Symbian y Windows Phone. La

intención de esta investigación es ver que existe en el mercado, buscar inspiración para obtener ideas propias y ver que se ha llegado hacer en este campo.

Pygpslog



Figura 1 – Pantalla de resultados Pygpslog

Sistema Operativo: Symbian

Descripción

Interprete de señales GPS para registrar puntos y coordenadas. Aparte de mostrar información como latitud, longitud, velocidad, satélites en cobertura y distancia recorrida, también permite nombrar puntos específicos (waypoints) con un nombre personalizado (ver Figura 1) [1].

Ventajas

- Uso de formatos estándar como GPX o OziExplorer, la cual cosa da la posibilidad de importar datos recogidos desde otra aplicación.
- Hace uso de la API interna "location and landmarks", lo cual le permite coexistir con otras aplicaciones como Nokia Maps y estar en ejecución al mismo tiempo.

Inconvenientes

- Interfaz poco amigable, falta diseño.

Google Maps Navigation



Figura 2 – Vista de navegación Google Maps Navigation

Sistema Operativo: Android

Descripción

Navegador GPS que permite buscar puntos de interés en el mapa sin necesidad de conocer la dirección. Permite buscar por el nombre de una empresa o simplemente por el tipo de empresa, tal como se haría en Google (ver Figura 2) [2][3]

Ventajas

- Se le puede indicar el destino mediante voz en lugar de teclas.
- En función del tráfico a lo largo de la ruta, aparece un indicador de color verde, amarillo o rojo.
- Se puede activar o desactivar capas como las que muestran estaciones de servicio, restaurantes, aparcamientos...
- Permite ver las vistas satélite 3D de Google.
- Está integrada con Google Buzz (red social de Google).

Inconvenientes

- Los mapas no se guardan en el dispositivo, siendo necesaria la conexión a Internet.
- No soporta la navegación a múltiples destinos.

Copilot

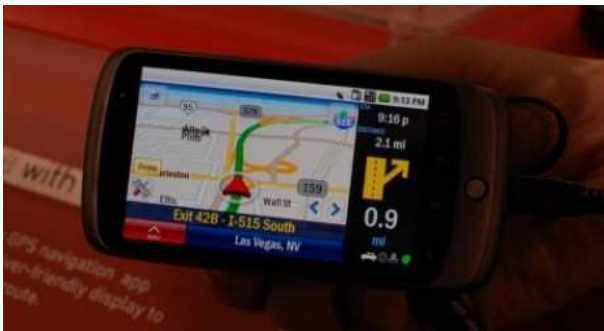


Figura 3 – Interfaz de navegación Copilot

Sistema Operativo: iOS, Android, WindowsMobile

Descripción

Navegador GPS que ofrece una interfaz atractiva y fácil de usar mientras se está conduciendo (ver Figura 3) .[3]

Ventajas

- Funciona con mapas previamente descargados a la tarjeta SD o a la memoria interna del teléfono.
- No necesita conexión a internet.
- Incluye características en vivo del tráfico.
- Soporta navegación con múltiples destinos.

Introducción - Estado del arte

- Funcionamiento Intuitivo.
- Interfaz amigable.

Inconvenientes

- Necesita memoria libre de por lo menos 2gb de espacio.

Runstar



Figura 4 – Diferentes Pantallas de Runstar

Sistema Operativo: Android

Descripción

Aplicación que hace uso del GPS para señalar la ruta que sigue el usuario al correr. Pretende llevar el ejercicio físico a un nivel más divertido y social. Se puede elegir entre varias opciones de carreras disponibles, comparar y compartir resultados con amigos y registrar los tiempos para ir mejorando día a día (ver Figura 4). [4][5]

Ventajas

- Incluye un reproductor mp3 con una función llamada "Power Up" que, al ser activada, reproduce las canciones más motivadoras (previamente almacenadas en el teléfono).

Inconvenientes

- Necesita de suficiente memoria libre en el teléfono para poder funcionar.

Cardio Trainer

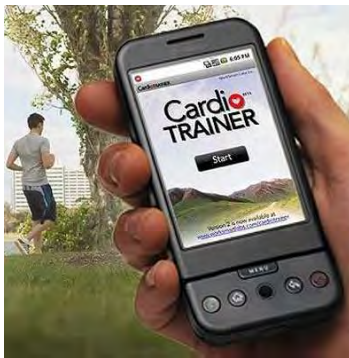


Figura 5 – Pantalla de inicio de Cardio
Trainer

Sistema Operativo: Android

Descripción

Herramienta que combina las prestaciones y las ventajas de un entrenador virtual en el móvil, y un GPS, ya que lleva un localizador que calcula en tiempo real donde se encuentra el usuario y la ruta que estamos siguiendo en cada momento. A la vez, se trata de una guía a la hora de realizar cualquier tipo de actividad aeróbica. Marca exactamente la ruta que está siguiendo, indicando la velocidad, el tiempo transcurrido y los diferentes accidentes del terreno (ver Figura 5).[6]

Ventajas

- Las rutas que se han seguido, quedan grabadas en el móvil.
- Las rutas pueden ser descargadas a cualquier equipo.
- Las rutas pueden ser colgadas en Facebook.

Inconvenientes

- Solamente funciona a partir de la versión 2.0 de Android.
- Necesita de suficiente memoria libre en el teléfono para poder funcionar.

Foursquare



Figura 6 – Pestaña de perfil y pestaña de recomendaciones de Foursquare

Sistema Operativo: Android, Blackberry, iOS

Descripción

Es una plataforma móvil basada en la ubicación cuyo espacio de juego son las ciudades enteras. Se trata de ir haciendo “check-ins”. Cuando un usuario hace un “check-in”, comparte su ubicación con sus amigos mientras gana puntos y símbolos virtuales (ver *Figura 6*). [7]

Ventajas

- Solo revela tu ubicación a tus amigos.
- Los usuarios de Foursquare pueden compartir su localización en Facebook y Twitter.
- Nuevas oportunidades para promover a las empresas.

Introducción - Estado del arte

- Los usuarios pueden hacer recomendaciones del servicio o productos de los lugares que visitan.
- Permite conocer lugares y saber las opiniones de otros clientes o visitantes de un lugar específico

Inconvenientes

- Pocos usuarios comparados con otras redes como Facebook o Twitter.
- Herramienta tecnológica muy nueva y muchos usuarios no los conocen, especialmente de países Latinos.
- Puede ser utilizado como una herramienta para conocer tu ubicación exacta y ser víctima de un asalto o secuestro.

SCVNGR

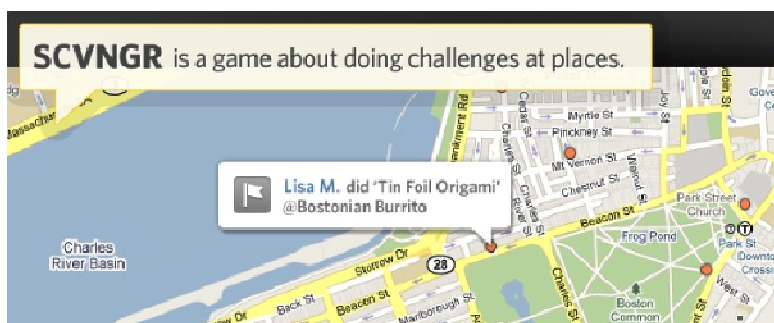


Figura 7 – Vista del mapa de SCVNGR

Sistema Operativo: Android, iOS

Descripción

SCVNGR ofrece un giro al servicio tradicional de geolocalización. Es una aplicación que trata de llevar la evolución de los check-ins a un cambio algo innovador. En adición al servicio normal de check in, los desarrolladores tratan de llevar a cabo una scavenger hunt (o búsqueda del tesoro) mundial, con desafíos para los jugadores (ver Figura 7) [8].

Ventajas

- Todos los desafíos pueden ser creados o inspirados por el usuario, o hechos por las empresas mismas.
- Permite sacar y subir imágenes, y escribir un pequeño texto sobre el lugar en sí.
- Usa la base de datos de Google para llenar su mapa.
- Los usuarios no deben ingresar a sus lugares favoritos en la red, ya que estos ya están mapeados.

Inconvenientes

- Pocos usuarios comparados con otras plataformas como Facebook o Foursquare.

MotionX GPS



Figura 8 – Vista del recorrido realizado y pantalla de información de MotionX GPS

Sistema Operativo: iOS

Descripción

Es una aplicación pensada para deportistas que necesitan algo más que un localizador GPS y que necesitan más precisión en entornos rurales. Proporciona información del tiempo de recorrido, la velocidad puntual, la velocidad media de recorrido, distancia recorrida y la velocidad máxima alcanzada (ver Figura 8) [9].

Ventajas

- Utiliza “OpenStreetMap” como mapas por defecto por lo que encontramos información diferente a la de GoogleMaps.
- Cuenta con mapas de “terrenos” mucho más detallados que los de Google en entornos rurales.
- Cuenta con una brújula de precisión.
- Permite guardar los tracks para ser usados más adelante.
- Permite consultar los tracks en GoogleMaps.

Inconvenientes

- No permite usar tracks descargados de internet para navegar mediante el uso del GPS.

MobileTracker

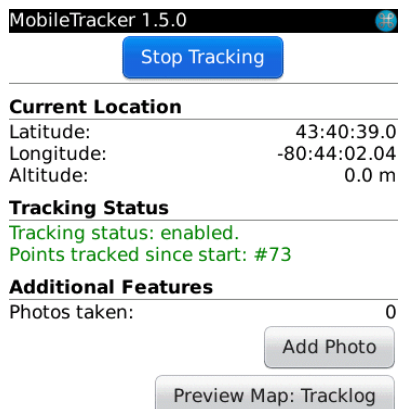


Figura 9 – Información proporcionada por
Mobile Tracker

Sistema Operativo: Blackberry

Descripción

MobileTracker ha sido diseñado para ser tan fácil como sea posible (ver Figura 9) [10].

Con dos pasos fáciles se puede empezar a grabar un track:

1. Inicio de la aplicación
2. Pulsación del botón de inicio para comenzar la grabación.

Ventajas

- Permite seleccionar el retraso que debe haber entre todos los puntos registrados en el track, esto permite ajustar la grabación al movimiento del usuario.
- Toda la información de MobileTracker puede ser visualizada en la vista 3D de Google Earth, sin necesidad de copiar el track desde el dispositivo a un PC.
- Desde la aplicación se puede publicar y compartir a través de Internet, utilizando el software gratuito de Google Maps.

Inconvenientes

- No permite la publicación en redes sociales más usadas, como Facebook o Twitter.

Waze



Figura 10 – Mapa de ruta y pestañas de opciones de Waze

Sistema Operativo: Android, iOS, Symbian, WindowsMobile

Descripción

Aplicación de navegación guiada por voz. Usa informes de tráfico de los usuarios en tiempo real, para ayudar en los viajes cotidianos. Permite que los usuarios trabajen juntos para obtener la información del tráfico más relevante y la ruta más inteligente en cada momento y situación (*ver Figura 10*) [11].

Ventajas

- Con sólo conducir con la aplicación abierta en el terminal móvil, se contribuye pasivamente con la información del tráfico.
- Permite a los usuarios ser activos, e informar a los demás sobre accidentes, controles policiales o cualquier otro obstáculo.
- Permite la integración con Foursquare, Twitter, Facebook y algunos geojuegos.

Inconvenientes

- Algunos mapas no están completos y provoca navegaciones imprecisas.
- Cuando se circula por una calle no pavimentada hay que circular por lo menos dos veces por ella para que la aplicación la empiece a dibujar.

1.2.2. Sensorización en videojuegos

Este apartado trata del estado del arte de la sensorización en videojuegos. Sensorización hace referencia a la captura de datos que proporcionen al juego información sobre el estado físico del jugador como su frecuencia cardíaca.

El objetivo de esta investigación es ver que se ha hecho hasta el momento e incentivar nuestra creatividad. Los usuarios de videojuegos así como los desarrolladores no están acostumbrados a ver juegos que traten con este tipo de tecnología, razón de más para ver que se ha hecho al respecto. Se ha podido comprobar que es un campo que aún está poco explotado en el mundo de los videojuegos y que tiene mucho futuro.

Wii Vitality Sensor



Figura 11 – Imagen del periférico Wii Vitality Sensor

Plataforma: Nintendo Wii

Descripción

Dispositivo presentado en la E3 (Electronic Entertainment Expo) del 2009. Se trata de un periférico capaz de medir las pulsaciones del usuario. El Wii Vitality Sensor es capaz de leer el flujo sanguíneo del jugador con ayuda de unos sensores lumínicos que determinan y miden el torrente sanguíneo (*ver Figura 11*) [12][13][14].

Ventajas

- Capaz de medir el estrés o la relajación del usuario.

Inconvenientes

- Hay varios rumores en la red sobre la posible aparición de juegos que utilizarían este sensor aunque a día de hoy no hay ningún juego que lo utilice.
- Hace casi 2 años que Nintendo presentó este periférico y parece que ha caído en el olvido.

EA Sports Active 2



Figura 12 – Pulsímetro, pantalla de juego y usuario del EA Sports Active 2

Plataforma: Xbox 360, Wii, PlayStation3

Descripción

Aplicación para realizar actividad. Va acompañado de un indicador de pulsaciones (propio del juego) que en todo momento indica al usuario el ritmo al que late su corazón, y una banda elástica para realizar ciertas actividades. En el caso de la videoconsola Xbox, hace uso del periférico Kinect, para el reconocimiento de los movimientos del jugador. En el caso de las otras dos videoconsolas, incluye unos sensores de movimiento, para poder ofrecer una experiencia de libre movimiento en todas las plataformas (ver Figura 12) [15].

Ventajas

- Tiene ejercicios muy variados para todo tipo de usuario.
- Dispone de varios entrenadores personales virtuales, uno para cada disciplina (aeróbico, anaeróbico, relajación, nutrición...).
- Permite la personificación de un avatar personal para cada jugador.
- También se puede entrenar en compañía en modo multijugador.

Inconvenientes

- Requiere de una gran paciencia a la hora de configurar todos los sensores correctamente.

- La versión de Xbox requiere de bastante espacio libre para realizar las actividades (2x3 metros, mínimo).

1.2.3. Conclusiones del estado del arte

La aplicación Smartphone de Drappen's World tiene que ser capaz de leer y modificar registros de una base de datos, así como, calcular la distancia que el usuario recorre con el teléfono. El estado del arte de las aplicaciones GPS en Smartphone ha permitido al equipo ver que existe en el mercado de este tipo de tecnologías y ver que nuestra necesidad es perfectamente capaz de llevarse a cabo. Así mismo, no se ha encontrado ninguna aplicación aprovechable a Drappen's World, puesto que son aplicaciones de las que no se dispone del código fuente y no se pueden readaptar a nuestra base de datos. Aunque se dispusiera del código fuente, se cree que es más rápido desarrollar nuestra aplicación desde cero que modificar el código de las aplicaciones analizadas, estas cumplen muchas más necesidades que la necesidad del caso de Drappen's World.

En cuanto al estado del arte referente a la sensorización en videojuegos, se ha podido comprobar que se trata de un campo tecnológico muy emergente y aún existen pocas alternativas en el mercado. Este es un hecho que juega a favor de Drappen's World que contará con este concepto novedoso de obtener información sobre el estado de alteración o nivel de esfuerzo del jugador. Las aplicaciones existentes en el mercado de este sector son de código cerrado, por tanto no podemos aprovechar módulos de terceros.

Terminado este análisis se concluye que es necesario realizar tanto la aplicación GPS como el módulo de sensorización de Drappen's World desde cero, teniendo en mente las ventajas e inconvenientes de las aplicaciones analizadas

1.3. Descripción del problema

La empresa colaboradora LabPsiTec propuso que se desarrollara un prototipo de serious game para promocionar la actividad física en niños, principalmente niños con problemas de obesidad. El serious game debía servir a terapeutas para ser incluido en los tratamientos terapéuticos en contra de la obesidad. El terapeuta debería poder programar la dificultad del juego para cada jugador. El prototipo funcional tenía que funcionar hasta el punto de poder ser validado clínicamente por terapeutas con pacientes reales.

La aplicación tenía que motivar la realización del ejercicio físico a un target de entre 8 y 14 años. Debía ser divertido y que distrajera al niño del objetivo principal, la práctica de ejercicio físico. El juego tenía que ser jugado en casa, pero a su vez, debía motivar al niño a salir al aire libre a realizar ejercicio en algunas pantallas. Se tenía que incentivar al niño a cansarse físicamente dándole más premios como más se cansase, para lo cual era necesario utilizar la camiseta sensorial que LabPsiTec proporcionaba a TheraGames.

Era muy interesante la posibilidad de que el videojuego fuese en primera persona y que fuese controlado mediante el dispositivo de captura de movimiento Kinect. Era muy positivo integrarlo en redes sociales (preferiblemente Tuenti) e incentivar la competitividad de los jugadores a través de estas. La competitividad debía ser entre pares iguales, con índices de obesidad parecidos. Había que dar feedback positivo sobre el ejercicio después de una experiencia buena y al final del juego. No debían aparecer nunca feedbacks negativos, puesto que es muy fácil desmotivar a este tipo de target, y más aún con el tema del ejercicio físico.

1.4. Solución propuesta

La solución que se propone para los requerimientos que LabPsiTec desea se explican a continuación:

La solución propuesta es la de crear un prototipo de serious game dividido en diferentes etapas, durante las cuales el niño tendrá que realizar movimientos físicos al jugar. Las diferentes pantallas irán proponiendo al jugador diferentes ejercicios, necesarios para progresar en el juego. Para controlar si los movimientos de los ejercicios se hacen de forma correcta se utilizará el periférico de captura de movimiento Kinect.

El juego estará compuesto de varios mini juegos divididos en niveles. Llegado a cierto mini juego, el juego propondrá al jugador salir a la calle para ir a caminar o correr, no será obligado (debido a que algunos niños no tendrán la posibilidad de hacerlo), pero si lo hace tendrá mayor recompensa. La distancia recorrida será controlada por un dispositivo móvil de última generación.

Los mini juegos que se realizan en casa, tendrán la posibilidad de utilizar la camiseta sensorial proporcionada por LabPsiTec, esta se encargará de medir el esfuerzo físico del jugador y en función de la cantidad de esfuerzo se dará más o menos recompensas; teniendo en cuenta que no se pueden dar feedbacks negativos al niño.

El terapeuta podrá registrar nuevos usuarios en la aplicación, programar la dificultad del juego (según el usuario) y consultar la evolución de cada niño, todo mediante un formulario web. Se intentará integrar el ranking de posiciones de los distintos jugadores en alguna red social, preferiblemente Tuenti. En caso de no poderse realizar, se haría una web donde solo los jugadores y el terapeuta podrían ver el ránking.

1.5. Perspectiva general del proyecto

A continuación se comentan cuales son los contenidos de cada apartado de la memoria:

2. Fundamentos teóricos

Se explican conceptos teóricos para que el lector pueda comprender toda la memoria.

2.1. Motor de juego

Se explica el concepto de motor de juego y se analizan algunos motores propietarios y algunos motores opensource disponibles en el mercado. Por último se escoge un motor de juego para desarrollar el proyecto.

2.2. Lenguaje de programación C++

Explicación de que tipo de lenguaje es C++ y sus características. También se argumenta porque ha sido elegido para este proyecto.

2.3. Entorno de desarrollo: Visual Studio

Se comenta el entorno de desarrollo Visual Studio, se analizan sus características principales y se comenta porque ha sido elegido para este desarrollo.

2.4. Dispositivo de captura: Camiseta sensorial

Explicación teórica de la camiseta sensorial que se ha utilizado en Drappe'ns World. Se especifica quienes son los creadores de dicha camiseta y cuales son sus características técnicas. Se argumenta también el porque de la elección de este dispositivo.

2.5. Periférico capturador de movimientos: Kinect

Explicación detallada del periférico kinect, tanto de su constitución y funcionamiento, como de sus especificaciones técnicas. También se exponen las razones que han llevado al equipo a incorporar este periférico.

2.6. Smartphone

Introducción - Perspectiva general del proyecto

Se introduce al lector en el nuevo concepto surgido recientemente de teléfono inteligente o Smartphone. Después se explica el sistema operativo Android que se ha utilizado para incorporar un Smartphone al serious game. Se argumentan las características y criterio de selección de dicho sistema y cuál ha sido el entorno en que se ha programado.

2.7. Tecnologías de servidor web

Se explica el concepto de servidor web y se introduce el protocolo de comunicación web http. También se comenta la necesidad que tiene el juego de incorporar una base de datos.

3. Parte práctica

Se explican todas las partes que ha desarrollado el técnico del equipo TheraGames, se muestran y se analizan algunos diagramas para comprender el comportamiento y la estructura del software de Drappen's World. Se explica también el software desarrollado para la camiseta sensorial y para el Smartphone, además de las cuestiones referentes a la base de datos.

3.1. Diagrama de arquitectura

Se representan los distintos componentes hardware de Drappen's World y como se establece la comunicación entre ellos.

3.2. Aplicación PC

Se exponen las diferentes librerías y tecnologías que necesita la aplicación principal del juego. También se analiza el diagrama de transición de estados de esta aplicación, además de mostrar la estructura de clases y de paquetes del software. Por último se explica cómo se ha desarrollado el software para tratar con la camiseta sensorial.

3.3. Aplicación mini juego Smartphone

Explicación del desarrollo de esta aplicación, dando énfasis a las tres funciones utilizadas más importantes.

3.4. Servidor: Base de datos

Introducción - Perspectiva general del proyecto

Se muestra la estructura de la base de datos y se explican cuales son las tecnologías utilizadas para conectar a ella tanto desde el PC como del teléfono móvil.

4. Resultados

Se exponen cuales han sido los resultados alcanzados a día de hoy al respecto del proyecto Drappen's World.

5. Conclusiones y líneas de futuro

Se comentan los distintos contratiempos que ha habido en el desarrollo de todo el proyecto, así como, las mayores dificultades que se han tenido. También se citan los aspectos de la aplicación que no han podido llegarse a implementar y los que si se han conseguido, centrándose en los temas técnicos y de programación del autor de este documento.

2. Fundamentos teóricos

A continuación, se explican los conceptos teóricos con los que ha tratado el técnico del equipo TheraGames. De esta manera el lector podrá entender mucho mejor la parte práctica de la memoria y ver en que se ha fundamentado el autor de este documento para realizar toda su parte del proyecto.

2.1. Motor de juego

Independientemente del lenguaje de programación utilizado, la mayoría de los videojuegos se presenta en forma gráfica, de manera que es necesario contar con librerías específicas. Actualmente, las APIs gráficas más utilizadas son el OpenGL y el DirectX, las cuales son de bajo nivel e interaccionan directamente con los drivers de las tarjetas gráficas. Para facilitar el trabajo al programador y no tener que programar a tan bajo nivel, el motor de juego le proporciona una serie de rutinas y funciones ya implementadas que permiten el diseño, la creación y la representación de un entorno virtual o de un videojuego.

La funcionalidad básica de un motor es proveer al videojuego de un motor de renderizado para los gráficos 2D y 3D, motor físico o detector de colisiones, sonidos, animación, inteligencia artificial, redes, streaming, administración de memoria y un escenario gráfico. Es importante remarcar que a veces se llama motor gráfico o motor de juego indistintamente, cuando en realidad el motor gráfico (o de render) está incluido en el motor de juego[16].

En los primeros avances de los motores gráficos, se podían observar texturas muy pequeñas que se duplicaban muchas veces con la intención de cubrir espacios muy grandes, modelos 3D con muy baja poligonización y pobre o nulo manejo de la inteligencia artificial. Pero la competencia y el avance en tecnologías de hardware han permitido que los programadores y empresas dedicadas al desarrollo de videojuegos invirtieran parte de su capital en el desarrollo y mejoras de los motores gráficos.

Fundamentos teóricos - Motor de juego

Existen varios lenguajes de programación utilizados para el desarrollo de videojuegos, entre los más utilizados se encuentran C, C++, C#, Java, etc. Hay una diversidad de motores de juegos orientados tanto a consolas como a computadoras personales. Las funciones comunes de un motor de juegos son la renderización, detección de colisiones, animación, manejo de memoria y de escenarios, manejo de sonido; aunque hay otros de última generación que incorporan técnicas más avanzadas como los sistemas de partículas.

La arquitectura del motor de juego permite la portabilidad de varias plataformas, incluyendo sistemas operativos de computadoras de escritorios, consolas de videojuegos y dispositivos móviles. La ventaja de un motor de juegos es que con un comando se realizan rutinas que normalmente llevarían muchísimos comandos en una librería gráfica como OpenGL o DirectX.

Los componentes que forman un motor de juego son los siguientes [17]:



Figura 13 – Componentes de un motor de juego

- El motor gráfico o de render: Proporciona funciones de renderizado 2D, 3D y de sprites. Sus funcionalidades principales son la de mantener la visibilidad, el mapeo de texturas, antialiasing y la gestión de mallas en 3D.

Fundamentos teóricos - Motor de juego

- El grafo de escena: Se encarga de ordenar la representación lógica y espacial de una escena. Una escena se representa con una estructura de datos en forma de árbol, en donde el nodo raíz es el escenario principal (ver *Figura 14*). Los nodos hijos son los objetos que están en el escenario y que a su vez constan de varios hijos.

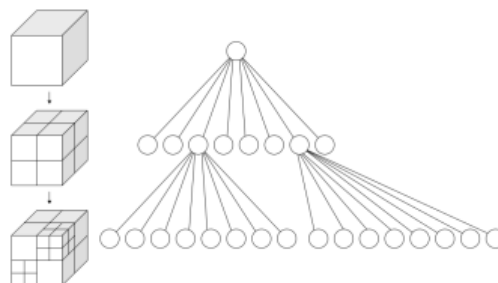


Figura 14 – Grafo de escena

- El motor de física: Simula los modelos del mundo real utilizando variables como la velocidad, masa, gravedad, fuerza, etc. Este tipo de simulaciones provoca un mayor realismo a las acciones del mundo virtual. Algunos de los motores de física más populares son ODE (Open Dynamics Engine), Newton Game Dynamics, PhysX y Havok.
- El detector de colisiones: Se encarga de calcular la cercanía de los objetos dinámicos en un escenario para avisar del momento exacto en que dos o más objetos chocan. Entonces es cuando la respuesta a la colisión debe activarse, pudiendo ser un sonido, una explosión, etc. Estos algoritmos están basados en BV (Bounding Volumes) (ver *Figura 15*), los cuales aproximan la forma de un objeto a una forma geométrica simple y poder así, calcular las distancias entre los objetos.

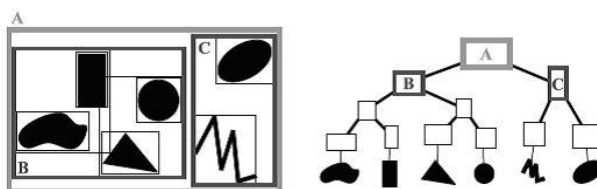


Figura 15 – Grafo de escena con BV

Fundamentos teóricos - Motor de juego

- El motor de Inteligencia Artificial: se encarga de dotar a ciertos elementos del juego de un comportamiento inteligente. Las técnicas empleadas son algoritmos de búsquedas, redes neuronales, algoritmos genéticos, multiagentes, etc.
- El motor de sonido: Reproduce la banda sonora del juego y los efectos de sonido como disparos o explosiones. Existen diferentes APIs para programar el sonido como el DirectSound3D, A3D, EAX y OpenAL. El sonido puede programarse para que tenga posición, velocidad, dirección y ángulos para conos que determinan la manera de viajar del sonido.
- Gestor de Redes: Los juegos en línea son cada vez más populares, de manera que el componente de redes para sincronizar los elementos y jugadores en activo es de cada día más avanzado. Algunos motores de juegos para web son Unity, Vassal, DimensioneX o Java 3D.

El gran avance que han tenido los motores gráficos tienen mucho que ver con el gran avance tecnológico que han tenido las GPUs (Graphics Processing Unit). Las GPUs pueden encargarse de varios trabajos que hace la CPU y poder así minimizar el tiempo de procesamiento. Esto se debe a que cuentan con multiprocesadores que a su vez pueden dividir el trabajo en bloques e hilos. Originalmente las GPUs se usaban para la renderización, generación de luz, sombras y texturas. Actualmente también se usan para diversos algoritmos tanto gráficos como de otras áreas científicas, precisamente por su paralelismo inherente.

Cabe destacar que los motores de juego no solo se utilizan para desarrollar videojuegos, sino que también se utilizan para desarrollar otras aplicaciones gráficas como simuladores, editores, paquetes didácticos, películas, publicidad, etc. En particular, en el campo de los serious games, los motores de juegos se aplican a actividades de áreas como el entrenamiento militar, enseñanza de estudiantes, simulaciones de desastres naturales y otros temas importantes en la vida cotidiana que ayudan a los especialistas en la toma de decisiones.

Hay una gran cantidad de motores gráficos disponibles que facilitan la creación de un entorno virtual, a continuación se analizan algunos ejemplos open source y algunos de propietarios. Los de código libre o abierto (open source) son de libre acceso y no es

necesario pagar por ellos y los segundos son comercializados por casas productoras de videojuegos y son dirigidos a profesionales de la industria.

2.1.1. Motores gráficos propietarios

En esta sección se habla de los cuatro mejores motores gráficos actuales según el ranking realizado por NeoTeo [18]. NeoTeo es una revista digital online española que trata de ciencia, tecnología, Internet, Noticias, Gadgets, Videojuegos, Software, Ciencia, Hardware, etc.

Se hace más hincapié en los dos motores que ocupan los primeros puestos del ranking: el Unreal Engine 3 y el Cry Engine 2.

Unreal Engine 3

Desarrollador: Epic Games

Escrito en: C++

Plataforma: Microsoft Windows, Linux, Macintosh, Dreamcast, Xbox, Playstation 2, Playstation 3, Wii.

Web Oficial: <http://www.unrealtechnology.com/>

Unreal Engine es un motor de juego de PC y consolas creados por la compañía Epic Games. Unreal Engine también ofrece varias herramientas adicionales de gran ayuda para diseñadores y artistas. El Unreal Engine de tercera generación aparece en 2006, diseñado para PC. Es un motor utilizado por grandes compañías de desarrollo de videojuegos, debido a su política de licencias [19][20].

Características principales

- Incluye soporte completo para ambientes interiores y exteriores con iluminación dinámica y sombreado por píxel.

Fundamentos teóricos - Motor de juego

- Dispone del editor visual UnrealKismet que proporciona a los artistas y diseñadores un control prácticamente ilimitado sobre la forma en que se jugará, sin tocar una sola línea de código.
- Dispone del Unreal Editor, una herramienta de creación de contenido que llena el vacío entre XSI, 3D Studio Max y Maya, y el contenido del juego entregable.
- Los desarrolladores utilizan el paradigma de programación "Data-driven", permitiendo a los diseñadores personalizar fácilmente cualquier objeto del juego, y a los programadores exponer nuevas propiedades.
- Implementa la técnica "High Dynamic Range (HDR)" que permite a las imágenes HDR representar con más exactitud el extenso rango de niveles de intensidad encontrados en escenas reales.
- 64-bit y 128-bit HDR
- Nuevo sistema de terrenos basados en LOD (técnica de programación que se utiliza para disminuir el número de polígonos de objetos lejanos).
- El corrector gamma, proporciona excelente precisión de color, y apoya a una amplia gama de efectos post procesamiento.
- Implementa el efecto "bloom light" que produce franjas de la luz alrededor de objetos muy brillantes, ocultando los detalles finos.
- Dispone del filtro "Lenticular Halo" para simular destellos de lentes y efectos similares.
- Da soporte a todo un conjunto de técnicas modernas de renderizado e iluminación por pixel, como la iluminación de phong parametrizada o la generación de mapas normales.
- Genera mapas de la dirección de la luz y auto sombreado automáticamente.
- Utiliza la técnica Voxel rendering que permite la creación de escenarios tridimensionales a partir de unidades mínimas, haciéndolos más detallados.

Fundamentos teóricos - Motor de juego

- Utiliza un “stencil buffered” dinámico para las sombras.
- Utiliza sombras de alta calidad y rendimiento pre computadas.
- Incluye la técnica de mapeados de texturas “displacement mapping” que mejora ampliamente técnicas anteriores como Parallax mapping, Normal mapping o Bump mapping.
- El mapeado de luz direccional permite el sombreado estático y la iluminación difusa normal de un número ilimitado de luces que se calcula previamente, esto permite tener escenas de alto rendimiento.
- Dispone de Efectos ambientales volumétricos implementados, incluyendo niebla de altura.
- Soporta los lenguajes de shaders más extendidos: HLSL, CG y GLSL.
- Saca el máximo provecho a las librerías DirectX 9 de Microsoft.
- Incluye soporte para procesadores multicore.
- Utiliza el motor de físicas Havok.
- Es muy fiable a la hora de jugar en línea
- Incluye efectos de post procesado logrados gracias a DX 11.
- Contiene Plug-ins para 3D Studio Max y Maya.

Algunos títulos realizados

- Gears of War 2



Figura 16 – Capturas de pantalla de Gears Of War 2

- Bioshock

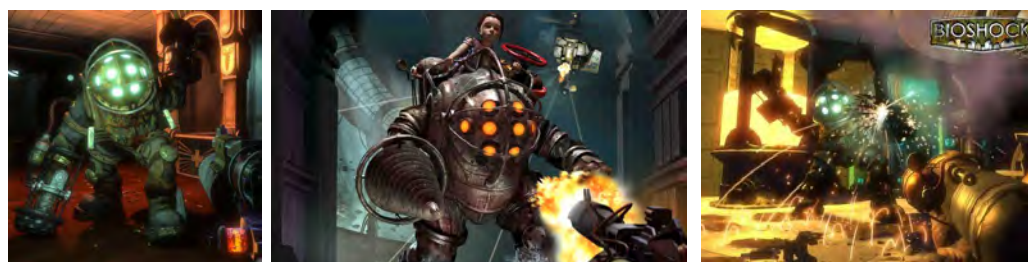


Figura 17 – Tres pantallas de Bioshock

Cry Engine 2

Desarrollador: Crytek

Escrito en: C++

Plataforma: Microsoft Windows, Xbox, PlayStation 3

Web Oficial: www.crytek.com/technology/cryengine

Descripción

CryENGINE es un motor de juego creado el 2003 por la empresa alemana desarrolladora de software Crytek, originalmente era un motor de demostración para la empresa Nvidia. El 30 de marzo de 2006, la totalidad de los derechos de CryENGINE fueron adquiridos por la distribuidora de videojuegos Ubisoft. Se utiliza para crear

videojuegos, pero también proyectos de ingeniería, física, arquitectura, diseño, paisajismo, simulaciones, etc [21]

Características principales

- Dispone de un sistema de edición visual que da a los desarrolladores una intuitiva interfaz para crear y controlar los acontecimientos.
- Incluye el editor sandbox que permite separar en capas un nivel de juego y da la posibilidad de que trabajen varios desarrolladores en una misma capa sin la preocupación de una colisión en lo que haga el compañero.
- Incluye un editor facial que utiliza el análisis del audio para extraer fonemas y así animar los rasgos faciales y proporcionar un movimiento de labios convincente.
- Implementa la técnica "High Dynamic Range (HDR)" que permite a las imágenes HDR representar con más exactitud el extenso rango de niveles de intensidad encontrados en escenas reales.
- Gestiona el nivel de detalle del terreno para optimizar el rendimiento de la CPU. (el escenario próximo tiene más nivel de detalle).
- Permite la posibilidad de cargar terrenos con tamaño prácticamente ilimitado y con un detalle excelente.
- Cuenta con luces con un aspecto muy similar a la luz natural.
- Crea automáticamente rayos de luz, cuando la luz incide sobre algún objeto (como la luz del sol atravesando las nubes).
- Capaz de crear sombras suaves que responden dinámicamente a los movimientos naturales de los objetos que están en la escena.
- Tiene funcionalidades ya implementadas para crear grandes zonas de niebla y nubes.
- Dispone de un sistema de niebla para simular profundidad tanto en un objeto como para diferenciar distancias.

Fundamentos teóricos - Motor de juego

- Sistema de cálculo para determinar que superficies se deben renderizar y cuáles no según la distancia a la que se encuentra y según las oclusiones.
- Calcula pixel a pixel, en tiempo real, la cantidad de luz ambiental que incide en los objetos, dependiendo de las oclusiones del entorno.
- La iluminación HDR permite corregir la luz en todos los tipos de ambiente.
- La luz del sol es volumétrica y soporta todo tipo de partículas y efectos
- Implementa la técnica "Eye Adaptation" para simular la adaptación del ojo humano a los cambios repentinos o extremos en las condiciones de iluminación.
- Es capaz de calcular la difusión y la difracción de la luz, cuando esta incide sobre objetos translúcidos como el hielo.
- Simulación de la superficie del agua en función de la dirección del viento y de las olas.
- Incluye la técnica "Motion Blur" que sirve para dar sensación de movimiento a los objetos móviles desenfocándolos.
- También da la posibilidad de enfocar desde una distancia x hasta otra distancia y ($x < y$), manteniendo las demás profundidades desenfocadas (Depth of Field).
- Soporta múltiples procesadores
- Incluye un sistema de inteligencia artificial de última generación.
- Permite crear fácilmente todo tipo de vehículos con un control intuitivo sobre sus características.
- Está preparado para trabajar con varios núcleos de procesador.
- El paquete de física incluido se puede aplicar a casi todos los objetos, desde edificios hasta el agua, pasando por fuego, etc.
- Contiene herramientas para importar objetos de Maya o 3Ds Max.

Algunos títulos realizados

- Far Cry 2



Figura 16 – Escenas varias del juego Far Cry 2

- Crysis



Figura 18 – Capturas de pantalla del juego Crysis

Anvil Engine

Desarrollador: Ubisoft Montreal

Plataforma: Microsoft Windows, PlayStation 3 y Xbox 360.

Escrito en: n/a

Web Oficial: <http://www.ubi.com>

Descripción

Anvil, llamado provisionalmente Scimitar antes de 2006, es un motor de videojuegos creado en 2007 por la empresa de videojuegos Ubisoft. Ubisoft ha estado utilizando motores gráficos licenciados para casi todos sus títulos, pero cuando llegó el momento

de crear el primer Assassin's Creed nació Anvil Engine. El modelado se realiza en 3DS Max para el entorno y ZBrush para los personajes [22][23].

Características principales

- Incluye la implementación de los ciclos de día y noche.
- Utiliza el middleware Autodesk HumanIK para una animación de personajes dinámica e integrada, en tiempo real.
- Es capaz de mostrar ambientes muy grandes y detallados en tiempo real.
- Es capaz de renderizar objetos situados a gran distancia de la cámara.
- La mayoría de los motores pierden mucha calidad en la representación de los objetos lejanos a la cámara. La pérdida de calidad es proporcional a la distancia focal. Sin embargo, Anvil Engine destaca por no perder tanta calidad en grandes distancias focales.

Algunos títulos realizados

- Assassin's Creed



Figura 19 – Imágenes de Assassin's Creed

- Prince of Persia



Figura 20 – Capturas del videojuego Prince of Persia

Geo-Mod Engine 2.0

Desarrollador: Volition Inc.

Escrito en: C++

Plataformas: PS2, PS3, GameCube, Xbox, Xbox 360,, Windows, N-Gage, Mac.

Web Oficial: n/a

Descripción

Geo-Mod Engine (Geometry Modification Technology), se trata de un motor gráfico introducido por primera vez en 2001 por Volition Inc. Logró equilibrar el motor al punto donde no solo los gráficos y la inteligencia artificial eran avanzados, sino que también permitía destruir todas las estructuras de un mundo abierto con físicas reales.

Actualmente este motor ha evolucionado que en un principio fue llamado RED (Realtime Environment Destruction) pero que al final se ha renombrado a Geo-Mod Engine 2.

Características principales

- Su rendimiento es muy optimizado para poder funcionar de forma muy fluida en las videoconsolas actuales.
- Posibilidad de la modificación del entorno mediante el uso de la técnica Destructible Environment.
- Cambia la geometría del mundo según lo que sucede en él, destrucciones, desprendimientos de partículas, temblores...
- Motor de físicas muy potente que controla la destrucción del entorno, teniendo en cuenta las diferentes partículas que se desprenden.
- Simula por si solo los ciclos de día y noche a medida que pasa el tiempo, controlando la luz ambiental.

Algunos títulos realizados

- Red Faction: Guerrilla



Figura 21 – Imágenes de Red Faction: Guerrilla

- Red Faction: Armageddon



Figura 22 – Escenas del juego Red Faction: Armageddon

2.1.2. Motores gráficos open source

En este apartado se comentan cuatro de los motores gráficos open source más conocidos que existen actualmente: Irrlicht, Ogre 3D, JMonkey y Panda 3D.

Irrlicht

Desarrollador: Nikolaus Gebhardt

Escrito en: C++

Plataforma: Windows 95, 98, NT, 2000, XP, Linux y MacOS.

Web Oficial: <http://irrlicht.sourceforge.net/>

Descripción

Es un motor gráfico 3D multiplataforma creado el año 2002 por Nikolaus Gebhardt quien lo diseñó desde sus inicios. Representa imágenes en tres dimensiones, que se ajusta a las necesidades de los juegos de gran calidad. El motor está en continuo desarrollo y dispone de un foro activo donde resolver dudas [24].

Características principales

- Dispone de una API totalmente documentada con ejemplos y tutoriales.
- Dispone de 5 tipos diferentes de renders: OpenGL, DirectX 9.0c, DirectX 8.1 y 2 de software.
- Utiliza las grillas Octree para particionar un espacio tridimensional, dividiéndolo recursivamente en ocho octantes.
- Tiene un sistema de GUI 2D que permite crear menús utilizando sus funcionalidades.
- Puede ser usado tanto en C++ como en lenguajes .net
- Incluye un básico control de físicas.
- Dispone de un motor de Audio llamado irrKlang.
- Soporta Pixel Shader y Vertex Shader.
- Incluye shaders de HLSL (High Level Shader Language) y GLSL (OpenGL Shading Language).
- Es capaz de recrear tanto escenas de interiores como exteriores.
- Implementa el sistema de animaciones "skeletal" y "morph".
- Utiliza la técnica para simular reflexiones "environment mapping", o "reflection mapping".

Fundamentos teóricos - Motor de juego

- Utiliza "stencil buffer" para el sombreado.
- Dispone de un rápido y fácil sistema de colisiones.
- Lee directamente un amplio formato de archivos comprimidos.
- Tiene un lector XML.

Algunos títulos realizados

- H-Craft Championship



Figura 23 – Capturas de pantalla de H-Craft Championship

- Bloodgiver



Figura 24 – Varias pantallas de Bloodgiver

Ogre 3D

Desarrollador: Equipo de programadores fundado por Steve "sinbad" Streeting

Escrito en: C++

Plataforma: Windows, Linux, Mac OSX

Web Oficial: <http://ogre3d.org/>

Descripción

Ogre 3D O (Object-Oriented Graphics Rendering Engine, “motor de renderizado de gráficos orientado a objetos”) es un motor gráfico 3D cuya primera versión estable estuvo disponible el febrero del 2005. A pesar de ser un motor de licencia libre, se ha utilizado para crear juegos comerciales. Ha sido diseñado para minimizar el esfuerzo que requiere programar videojuego o trabajar con gráficos complejos, pero sin perder potencia [25].

Características principales

- Dispone de amplios ejemplos de muestra en su web y una comunidad muy activa.
- Soporte para shaders de vértices y de fragmentos, ambos a bajo nivel y escritos en ensamblador. También están disponibles en lenguajes de más alto nivel como Cg, DirectX 9 HLSL, o OpenGL GLSL.
- Utiliza las grillas Octree para particionar un espacio tridimensional, dividiéndolo recursivamente en ocho octantes.
- Utiliza el método BSP (Partición Binaria del Espacio) para subdividir recursivamente un espacio y dar lugar a una representación de la escena.
- Dispone de un lenguaje de declaración de materiales que permite mantener varios conjuntos de materiales fuera del código.
- Permite diseñar el material con efectos alternativos para un amplio rango de tarjetas y OGRE automáticamente usa el método mejor soportado.
- Soporte de LOD de material, los materiales reducen el detalle según la distancia u oclusiones.
- Soporte para texturas dinámicas en todos los formatos soportados, para la reproducción eficiente de películas y otro contenido en tiempo real sobre la textura.

Fundamentos teóricos - Motor de juego

- Soporta animación esquelética, incluyendo el mezclado de múltiples animaciones y uso de huesos de tamaño variable.
- Uso de interfaces diseñadas para minimizar el esfuerzo de renderizar escenas 3D, y ser independiente de la implementación 3D.
- Tiene un sistema de superposición que permite construir HUDs (información, puntuaciones, vidas...) y menús usando objetos 2D y 3D.
- Permite configurar la apariencia de niebla.
- Incluye software para llevar a cabo la técnica bump mapping para dar un aspecto rugoso a las superficies de los objetos.
- Utiliza el método "cube mapping" que sirve para encuadrar el entorno de un mundo utilizando 6 imágenes que engloban las diferentes caras de un cubo.
- Dispone de un framework para la integración con otras librerías tales como física y colisión mediante ReferenceAppLayer. Esto da la flexibilidad de usar las librerías que se prefiera, en vez de estar sujeto a una.
- Soporta los sistemas skyboxes, skyplanes y skydomes.
- Soporta archivos exportados desde múltiples herramientas de modelado como 3D Studio Max, Maya o Blender.
- Soporta la carga de archivos comprimidos como ZIP y PK3.

Algunos títulos realizados

- Bolzplatz2006



Figura 25 – Varias vistas de Bolzplatz2006

- Victory: The Age of Racing



Figura 26 – Capturas de pantalla de The Age of Racing

jMonkey Engine

Desarrollador: The jME core team

Escrito en: Java

Plataforma: Multiplataforma mediante Java (JVM)

Web Oficial: <http://jmonkeyengine.com/>

Descripción

Motor escrito completamente en Java que utiliza la librería gráfica OpenGL. Lo inició Mark Powellen el año 2003 para un proyecto personal, donde se quería comprobar si se podía escribir en Java toda una API gráfica. Se trata de uno de los engines 3D más completos escritos en Java [26].

Características principales

- Implementa la técnica “High Dynamic Range (HDR)” que permite a las imágenes HDR representar con más exactitud el extenso rango de niveles de intensidad encontrados en escenas reales.
- Dispone de una biblioteca de shaders.
- Permite la utilización de sus shaders de forma muy intuitiva. Las funciones de shaders utilizan el lenguaje GLSL.
- Implementa multitextura a través de shaders.
- Incluye el efecto “Cartoon”.
- Soporte de iluminación por pixel.
- Implementa la técnica de iluminación de Phong utilizando mapas especiales (diffuse map, alpha map, glow map, bump map, specular map, parallax map).
- Para las físicas se utiliza jBullet, un Puerto de java de la Bullet Physics (potente motor de física en tres dimensiones).
- Incluye simulación de partículas como humo, fuego y explosiones.
- Capaz de simular la reflexión de la luz en el agua.
- Incluye la técnica “Depth of Field” que da la posibilidad de enfocar desde una distancia x hasta otra distancia y ($x < y$), manteniendo las demás profundidades desenfocadas.
- Utiliza Spider Monkey, un motor de creación de redes de alto rendimiento de Java.

Fundamentos teóricos - Motor de juego

- Dispone de múltiples Threads ya implementados para cargar archivos mediante HTTP.
- Permite la carga de objetos de escena mediante archivos ZIP.
- Dispone de gran variedad de formatos para los modelos, texturas y fuentes.

Algunos títulos realizados

- Urban Galaxy



Figura 27 – Imágenes de Urban Galaxy

- Mythruna



Figura 28 – Varias pantallas de Mythruna

Panda 3D

Desarrollador: Disney, ETC

Escrito en: C++

Plataforma: Microsoft Windows, Linux, Mac OS X, FreeBSD

Web Oficial: <http://www.panda3d.org/>

Descripción

Motor desarrollado originalmente por Disney, con la finalidad de crear atracciones para sus parques temáticos. El año 2002 el motor pasó a ser un proyecto open source, cuando Disney decidió colaborar con las universidades en proyectos de investigación de realidad virtual mediante este motor. Entonces, el ETC (Entertainment Technology Center) de la Universidad de Carnegie Mellon, se unió al desarrollo de este motor [27][28].

Características principales

- Se puede emplear cualquier editor de texto, y no hace falta un entorno de desarrollo específico.
- Contiene un conjunto de herramientas de depuración que permiten una gran tolerancia de errores.
- Implementa la técnica de renderizado HDRR (High Dynamic Range rendering).
- El desarrollador dispone del lenguaje de alto nivel Cg o C for Graphics, lenguaje para la programación de vértices y pixel shaders.
- Integración completa para el lenguaje Python.
- Tiene implementados efectos de Luces, partículas y niebla.
- Genera automáticamente Mapas especiales (Normal Map, Gloss Map, Glow Map).
- Tiene clases específicas para operaciones matemáticas (LVector3F, LPoint3F, LMatrix3F, etc.).
- Contiene detector de colisiones.
- Implementa técnicas de triangulación.

Fundamentos teóricos - Motor de juego

- PyODE integrado, un conjunto de enlaces Python a la ODE (Open Dynamics Engine) para la simulación de la dinámica de cuerpos rígidos.
- Su única dependencia externa son los drivers de la controladora gráfica
- Dispone de un potente control de rendimiento y optimización que Identifica cuellos de botella, tanto en la CPU como en la GPU.



Figura 29 – Distintas pantallas de Disney's Pirates of the Caribbean Online

- El tiempo de uso de la CPU está descompuesto en más de 250 categorías.
- Dispone de un sistema para crear entregables comerciales: permite convertir archivos de formatos anticuados, importar archivos de fuente, crear ejecutables y empaquetar juegos para la distribución, etc.

Algunos títulos realizados

- Disney's Pirates of the Caribbean Online
- Angels Fall First: the Second Antarean War



Figura 30 – Pantallas de Angels Fall First: the Second Antarean War

2.1.3. Motor elegido: Irrlicht



Uno de los problemas a los que se enfrentó el equipo de desarrollo en un inicio era la elección de las herramientas software que le permitieran ser lo más eficientes posibles en el desarrollo del proyecto, para poder cumplir con las expectativas. La elección de software más importante era la del motor de juego, ya que repercutiría en la eficiencia a lo largo de todo el desarrollo. Además se necesitaba un motor de juego con ciertas características. Para ello era imprescindible hacer el análisis que se acaba de ver sobre motores de juego.

La elección final fue el motor de juego Irrlicht motivado principalmente porque el equipo técnico ya tenía conocimientos sobre dicho motor y eso hacía que la curva de aprendizaje fuese mucho más rápida.

El motor es open source, que era una premisa importante para el equipo (debido al escaso presupuesto disponible) y además dispone de las siguientes características muy importantes para el desarrollo de Drappen's World:

- Dispone de una API totalmente documentada con ejemplos y tutoriales.
- Tiene un sistema de GUI 2D que permite crear menús utilizando sus funcionalidades.
- Incluye soporte para la aplicación de shaders (pixel shaders y vertex shaders).
- Soporta la animación "skeletal" para la simulación de personajes vertebrados o movimientos musculares.
- Dispone de un lector de XML
- Dispone de una extensa comunidad de desarrolladores.
- Cuenta con foros muy activos dispuestos a ofrecer ayuda ante cualquier duda.

2.2. Lenguaje de programación C++



En cuanto al lenguaje de programación, C++ es un lenguaje de programación diseñado a mediados de los años 1980 por Bjar Stroustrup. La intención de su creación fue el extender al exitoso lenguaje de programación C con mecanismos que permitan la manipulación de objetos. [29]

2.2.1. Criterio de Selección

El criterio de selección estuvo basado en tres razones muy claras:

1. Permitía al equipo de desarrollo utilizar el motor gráfico que se había seleccionado (ver apartado anterior “Motor elegido: Irrlicht”).
2. Es un lenguaje orientado a objetos y permite el uso de técnicas como herencia, abstracción, polimorfismo y encapsulamiento.
3. Era el lenguaje más conocido por el técnico y scrum master del equipo Pau Moreno, puesto que ya había trabajado mucho en irrlicht anteriormente utilizando este lenguaje. El técnico de Theragames y autor de este documento Tolo Rosselló tenía muy poca experiencia en programación gráfica, por lo cual era interesante poder tener soporte de Pau.

2.2.2. Características

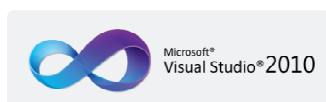
C++ tiene varias características que otros lenguajes de programación no tienen. Las más destacadas son [30]:

- **Programación orientada a objetos:** La posibilidad de orientar la programación a objetos permite al programador diseñar aplicaciones desde un punto de vista más

cercano a la vida real, imaginando el código como módulos u objetos con un estado y unas funciones, que interactúan entre sí. Además, permite la reutilización del código de una manera más lógica y productiva.

- **Portabilidad:** Un código escrito en C++ puede ser compilado en casi todo tipo de ordenadores y sistemas operativos sin hacer apenas cambios.
- **Brevedad:** El código escrito en C++ es muy corto en comparación con otros lenguajes, sobretodo porque en este lenguaje es preferible el uso de caracteres especiales que las "palabras clave".
- **Programación modular:** Un cuerpo de aplicación en C++ puede estar hecho con varios ficheros de código fuente que son compilados por separado y después unidos. Además, esta característica permite unir código en C++ con código producido en otros lenguajes de programación como Ensamblador o el propio C.
- **Velocidad:** El código resultante de una compilación en C++ es muy eficiente, gracias a su capacidad de actuar como lenguaje de alto y bajo nivel y a la reducida medida del lenguaje.

2.3. Entorno de desarrollo: Visual Studio



El entorno de desarrollo del serious game ha sido Microsoft Visual Studio 2010, utilizando el lenguaje de programación C++. Microsoft Visual Studio es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta varios lenguajes de programación tales como C, C++, C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros lenguajes[31].

2.3.1. Criterio de Selección

Se eligió este entorno de desarrollo por ser, con mucha diferencia, el más conocido por el equipo TheraGames, desarrollador de Drappen's World. Además era el entorno más distribuido entre los PC's que el equipo disponía en su sitio de trabajo

2.3.2. Características

En Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión net 2002). Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas web y dispositivos móviles. Hasta ahora, uno de los mayores logros de la versión 2010 de Visual Studio ha sido el de incluir las herramientas para desarrollo de aplicaciones para Windows 7.

Entre sus más destacables características, se encuentran la capacidad para utilizar múltiples monitores, así como la posibilidad de desacoplar las ventanas de su sitio original y acoplarlas en otros sitios de la interfaz de trabajo. Además de esto, aparece una edición que compila las características de todas las ediciones comunes de Visual Studio: Professional, Team Studio, Test, conocida como Visual Studio Ultimate[31].

2.4. Dispositivo de captura: Camiseta sensorial

En este apartado nos referimos a la captura del esfuerzo físico realizado por el jugador de Drappen's World, en función de su frecuencia cardíaca. Se debe diferenciar entre dos conceptos: esfuerzo físico y cantidad de ejercicio físico. El primero es relativo a cada jugador en un momento dado, en cambio el segundo es un concepto que no depende del jugador, ni del momento, ni del cansancio. Dos jugadores distintos, haciendo una misma cantidad de ejercicio (del mismo tipo), pueden estar esforzándose uno más que el otro, dependiendo de su condición física o del cansancio acumulado. Con este dispositivo se es capaz de saber el esfuerzo físico que está realizando un jugador concreto en cada

Fundamentos teóricos - Dispositivo de captura: Camiseta sensorial

momento, con el ejercicio en cuestión. Esto permite dar unos u otros premios al jugador, en función del esfuerzo realizado, y no de la cantidad de ejercicio. De esta forma se motiva al jugador a esforzarse el máximo físicamente.

Esta información es muy útil para conocer el esfuerzo relativo realizado por del jugador. Para ello se utiliza un sensor mucho más sofisticado que un pulsómetro. Se trata de una camiseta sensorial proporcionada por el i3bh (Instituto Interuniversitario de Investigación en Bioingeniería y Tecnología Orientada en el Ser Humano). Ha sido desarrollada por el i3bh en colaboración con la comañia NUUBO. Proporciona más información que el esfuerzo físico aunque para este proyecto con esa información es suficiente.

Se trata de un dispositivo que detecta la posición del cuerpo, la actividad física y el consumo de energía, teniendo en cuenta no alterar la comodidad del usuario (*ver Figura 31*). Esta camiseta dispone de un sensor de actividad fijado en la altura del pecho de la persona. El sistema detecta entre periodos de actividad (vigorosa, leve y moderada), orientación postural (sentado, tumbado, de pie), y detecta acciones tales como caminar o correr. Tiene un grado alto de exactitud. Además de esto, ofrece una estimación del gasto de energía metabólica y obtiene la frecuencia cardíaca [32].



Figura 31 – Usuario con la camiseta sensorial y reverso de la camiseta

2.4.1. Criterio de selección

El criterio de selección del sensor de captura del esfuerzo físico ha venido impuesto por el cliente de TheraGames: LabPsiTec . El Laboratorio de Psicología y Tecnología (LabPsiTec), trabaja conjuntamente con el Instituto Interuniversitario de Investigación en Bioingeniería y Tecnología Orientada en el Ser Humano (i3BH).

Nuubo es una compañía española que desarrolla productos de ingeniería biomédica, con un fuerte acento en la tecnología de materiales. Nuubo, juntamente con i3BH, están desarrollando la camiseta sensorial que nos ha sido proporcionada por LabPsiTec . El cliente de TheraGames impuso el requerimiento de utilizar el prototipo funcional de la camiseta sensorial para realizar Drappen's World.

2.4.2. Características

Las principales propiedades de esta camiseta [32] son:

- Dispone de varios electrodos conductivos dentro de la tela con hilos de metal. Estos sensores capturan las señales vitales del usuario y pueden enviarlas a un teléfono celular, pulsera o router.
- Tiene un módulo electrónico con procesador integrado para la adquisición, procesamiento y transmisión de las señales.
- Acelerómetro de 3 ejes con un rango de aceleraciones de +/- 3G.
- Algoritmo de actividad física y metabólica en tiempo real integrado.
- Memoria de alta capacidad (hasta 20 días).
- El protocolo de comunicación de datos tiene varios modos: modo continuo, modo inteligente (con alarmas), modo programado (resumen de los datos) y modo inalámbrico temporizado de transmisión (Bluetooth or Zigbee).
- Las camisetas están hechas para ambos géneros y hay de varios tamaños para conseguir el mejor confort para el usuario.

- El usuario puede lavar la camiseta sin problemas.

Es importante remarcar que a día de hoy esta camiseta está en fase de desarrollo, por ahora no dispone de ninguna API para ser utilizada. La camiseta envía un flujo de datos a través del puerto serie y estos deben ser filtrados antes de ser utilizados.

2.5. Periférico capturador de movimientos: Kinect



Figura 32 – Jugadores de Kinect en acción

Es un controlador de juego creado por Alex Kipman y desarrollado por Microsoft. Ha sido desarrollado para la videoconsola Xbox 360, y en un futuro tendrá soporte oficial para PC en Windows 8. Fue lanzado por primera vez al mercado el 4 de noviembre del 2010 [33]. Actualmente tiene soporte para PC a través de librerías open source como OpenNII. Se trata de un paso adelante en la forma de interactuar con las máquinas, es el principio de la llamada “computación gestual” que hasta ahora estaba limitada a las interfaces multitáctiles. Kinect reconoce los movimientos y la postura del jugador, permitiendo a este interactuar con el juego sin necesidad de llevar ningún dispositivo extra en el cuerpo. [34]

2.5.1. Criterio de selección

Kinect compite con los sistemas Wiimote con WiiMotionPlus y PlayStation Move, que también controlan el movimiento para las consolas Wii y PlayStation 3, respectivamente; aunque estos últimos sí que precisan contacto físico con algún periférico. Microsoft ha superado las 4 millones de unidades vendidas de su nuevo capturador de movimientos. A finales del 2011 Wii está perdiendo fuerza ante Xbox 360 y PS3 no sólo cae en ventas de consolas sino que además, su control PlayStation Move no ha llegado a vender aún ni un millón de unidades (*ver Figura 33*) [35][36].

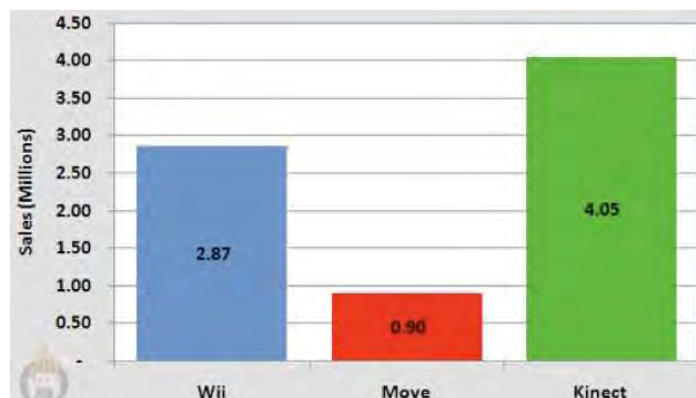


Figura 33 – Ventas de WiiMote, Playstation Move y Kinect
(finales del 2010) (Fuente: VGChartZ)

El dispositivo tiene como objetivo primordial aumentar el uso de la Xbox 360, más allá de la base de jugadores que posee en la actualidad. Las previsiones apuntaban a que Kinect iría asociada a nuevos usuarios que migrarían de otras consolas, pero sólo el 40% de las ventas han sido de packs (consola Xbox + Kinect) [35]. La cual cosa, sugiere que el 60% de los Kinect vendidos han sido adoptados por usuarios antiguos de Xbox o usuarios que quieren utilizar el periférico en PC, a través de librerías externas (como OpenNI).

En todo caso, observamos dos ventajas clave a la hora de elegir este dispositivo de captura de movimiento en comparación con sus principales competidores.

1. La primera y más importante para Drappen's World es que da al usuario total movimiento sin necesidad de llevar ningún dispositivo extra en el cuerpo, convirtiendo la interacción del jugador en algo natural.
2. Es el dispositivo más reciente de los tres en salir al mercado (Move setiembre 2010, Wiimote noviembre 2005 y WiimotePlus noviembre 2010) y desde su salida ya obtiene un mayor índice de ventas (*ver Figura 33*).

2.5.2. Características



Figura 34 – Usuario jugando con interacción totalmente natural

Kinect permite a los usuarios controlar e interactuar videojuegos sin necesidad de tener contacto físico con un controlador de videojuegos (*ver Figura 34*). Interactúa mediante una interfaz natural. Reconoce los gestos del usuario (*ver Figura 36*), comandos de voz, objetos e imágenes [37].

Kinect se trata de una barra horizontal conectada a un pivote (*ver Figura 35*), diseñada para estar en una posición longitudinal. El dispositivo tiene una cámara RGB, un sensor de profundidad y un micrófono multi-array bidireccional que conjuntamente capturan el movimiento de los cuerpos en 3D, además de ofrecer reconocimiento facial y aceptar comandos de voz[38].

Fundamentos teóricos - Periférico capturador de movimientos: Kinect

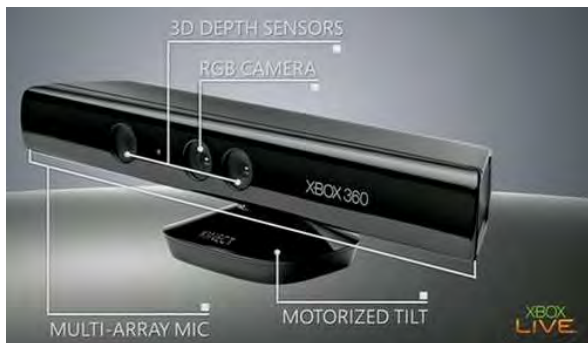


Figura 35 – Periférico Kinect



Figura 36 – Tracking de un jugador realizado por Kinect

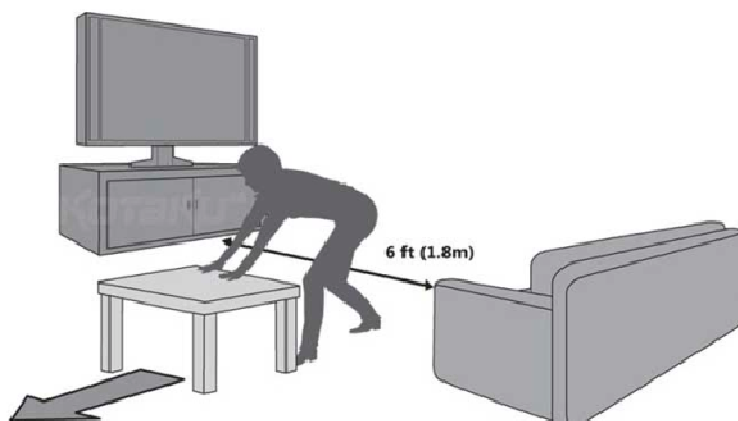


Figura 37 – Distancia mínima operacional para Kinect

Kinect captura video a una frecuencia de 30 Hz, en colores RGB de 32-bit y una resolución VGA de 640x480 pixels, el canal de video monocromo es de 16-bit, resolución QVGA de 320x240 pixels con hasta 65,536 niveles de sensibilidad. Idealmente, Kinect necesita un espacio de 1.8 metros (*ver Figura 37*) [39], si juega un solo jugador y 2.4 metros si juegan dos personas. El límite del rango visual está entre 1.2 y 3.5 metros de distancia, con un ángulo de vista de 57° horizontalmente y un ángulo de 43° verticalmente, mientras que el pivote puede orientarse hacia arriba o abajo ampliando

Fundamentos teóricos - Periférico captador de movimientos: Kinect

hasta 27°. El array del micrófono tiene cuatro cápsulas, y opera con cada canal procesando 16-bit de audio con un ratio de frecuencia de 16 kHz.



Figura 38 – Transformación en escala de grises

La cámara de Kinect funciona con hardware y software de serie para el reconocimiento de imagen. La cámara tiene dos funcionalidades principales, genera un mapa en 3D de la imagen (ver Figura 39) que tiene en su campo visual y reconoce humanos en movimiento entre los objetos de la imagen a partir de diferentes segmentos de las articulaciones del cuerpo y un esquema en escala de grises del rostro (ver

Figura 38). La cámara transmite luz invisible para el ojo humano, cercana en el espectro a los infrarrojos, y puede conocer el tiempo que tarda la luz en volver al sensor tras reflejarse en los objetos.

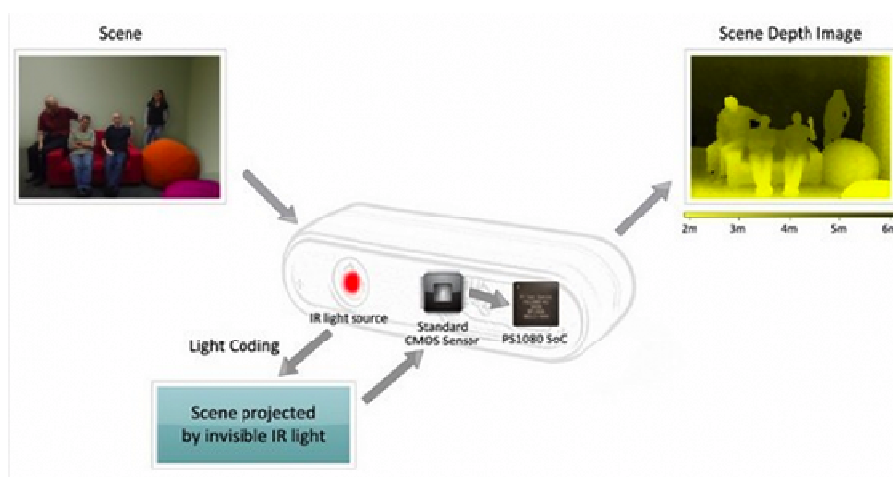


Figura 39 – Transformación de la escena al mapa de profundidad de Kinect

El sensor actúa como un sonar, conociendo el tiempo de salida y llegada de la luz tras reflejarse en un objeto y sabiendo la velocidad absoluta de la luz (ver Figura 40), es capaz de obtener la distancia a la cual se encuentra ese objeto. En un amplio campo visual con objetos, la cámara Kinect trata de reconocer a qué distancia están los objetos, distinguiendo el movimiento en tiempo real. Kinect puede llegar a distinguir la profundidad de cada objeto con diferencias de 1 centímetro y su altura y anchura con diferencias de 3 milímetros. El hardware de Kinect está compuesto por la cámara y el

Fundamentos teóricos - Periférico captador de movimientos: Kinect

proyector de luz infrarroja, añadido al firmware y a un procesador que utiliza algoritmos para procesar las imágenes tridimensionales (ver Figura 39).

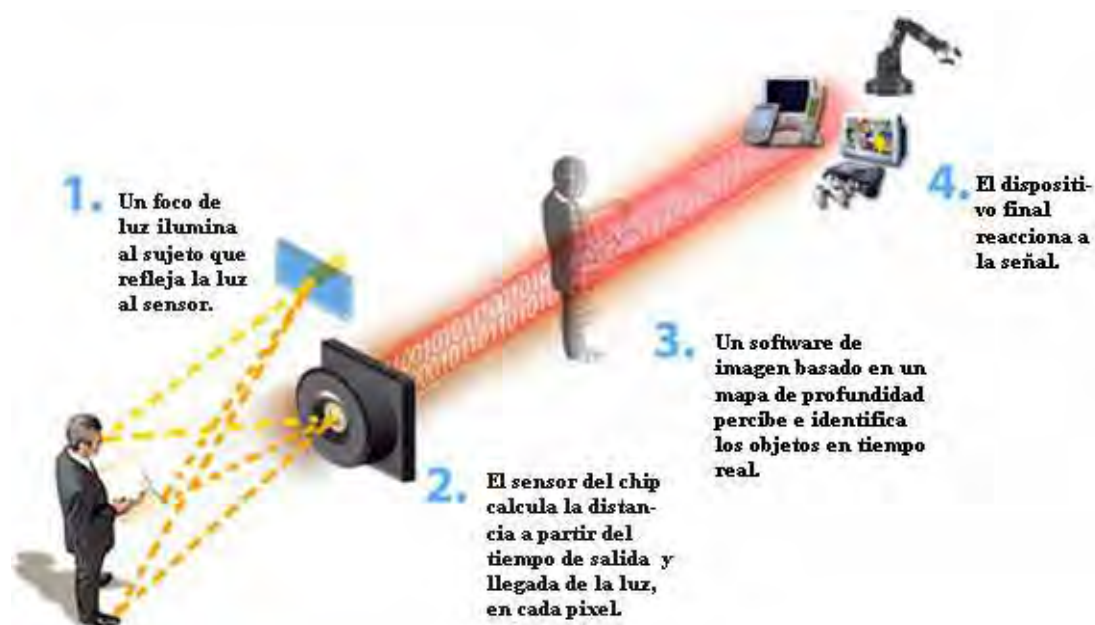


Figura 40 – Funcionamiento del periférico Kinect

El procesador (ver Figura 41) es capaz de interpretar los movimientos que se registran en los objetos capturados por la cámara de Kinect en eventos con significado que aparecen en pantalla. Los movimientos buscados por el algoritmo son contextualizados, si nos encontramos en un juego como Kinect Adventures, donde una balsa desciende por la corriente del río, si este juego requiere movimientos como agacharse o tumbarse, entonces se buscará la identificación de estos movimientos en tiempo real para producir eventos en pantalla. Si el usuario navega por el menú con interfaz gráfica entonces se buscarán movimientos horizontales y verticales con la mano que serán registrados en los fenómenos de pantalla[38].

Fundamentos teóricos - Smartphone

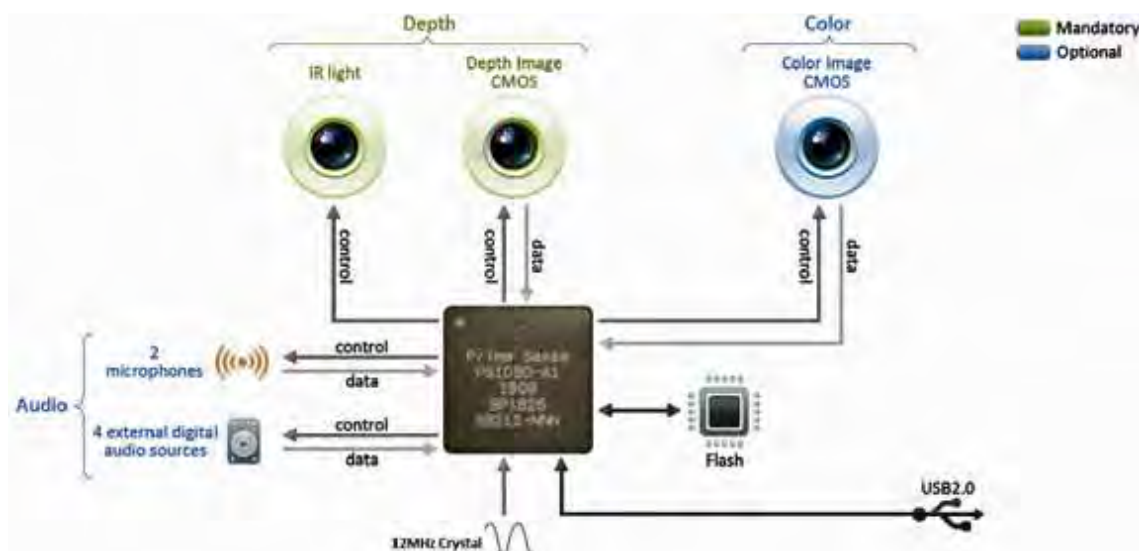


Figura 41 – Diagrama tecnológico de la arquitectura de Kinect

En esta memoria no se habla sobre cómo utilizar Kinect a través de las librerías open source disponibles hoy en día (a día de hoy aún no existe librería oficial de Microsoft), puesto que en este proyecto el tema de Kinect ha sido implementado por el técnico y scrum master del equipo, Pau Moreno. En su memoria, que al finalizar este documento aún estaba en desarrollo, el lector podrá encontrar información detallada sobre la utilización de Kinect a través de las librerías no oficiales (como OpenNII), disponibles a día de hoy.

2.6. Smartphone

El teléfono inteligente (smartphone en inglés) es un término comercial para denominar a un teléfono móvil que ofrece más funciones que un teléfono celular común. El Smartphone usualmente posee hardware para poder conectarse a internet mediante WiFi o 3G, dispone de acelerómetro, receptor GPS, cámara y Bluetooth. Además, la mayoría disponen de pantalla táctil (ver Figura 42).[40]



Figura 42 – Hardware común en los Smartphones

Sobre un Smartphone, corre un sistema operativo encargado de efectuar la gestión del hardware y los procesos básicos del sistema, así como, permitir la normal ejecución de las



Figura 43 – Interacción de un SO

aplicaciones (ver Figura 43). Una característica importante de casi todos los teléfonos inteligentes es que permiten la instalación de programas para incrementar el procesamiento de datos y la conectividad. Estas aplicaciones pueden ser desarrolladas por el fabricante del dispositivo, por el operador o por un tercero. Las aplicaciones que suelen venir ya instaladas en el sistema operativo son: navegador web, cliente de correo electrónico, aplicaciones para soporte multimedia y documentos, mensajería (sms, mms), aplicación de captura fotográfica, administración de contactos y llamada, calendario y agenda.

Los sistemas operativos para Smartphone más usados son : Android (Google), Symbian OS (Nokia), iOS (Apple), Blackberry OS (RIM) y Windows Phone (Microsoft).

2.6.1. Android

Desarrollador: Open Handset Alliance y Google Inc.

Escrito en: C (núcleo), C++ (algunas bibliotecas de terceros), Java (UI).

Plataforma: ARM, x86, MIPS, IBM POWER

Web Oficial: <http://www.android.com/>



Android es un sistema operativo basado en Linux para dispositivos móviles, tales como teléfonos inteligentes (Smartphones) o tablets. Fue desarrollado inicialmente por Android Inc., una firma comprada por Google en 2005. Google liberó la mayoría del código de Android bajo la licencia Apache, una licencia libre y de código abierto. Es el principal producto de la Open Handset Alliance, un conglomerado de fabricantes y desarrolladores de hardware, software y operadores de servicio. Android permite al desarrollador abstraerse del hardware y facilita el desarrollo de aplicaciones para dispositivos de recursos limitados, como es el caso de un Smartphone [41][42].

Criterio de selección

Android cerró el año 2010 con un aumento de ventas de más de un 615% (ver Figura 44). Los smartphones con sistema operativo Blackberry OS (de RIM) y Symbian (de Nokia) aumentaron sus ventas pero perdieron mercado. Por otro lado, iOS (de Apple) mantuvo su cuota de mercado en un 16%. Windows Phone 7 (de Microsoft), duplicó sus ventas pero es la plataforma con menor cuota de mercado.

Fundamentos teóricos - Smartphone

En la actualidad, uno de cada tres smartphones vendidos llevan el sistema operativo Android.

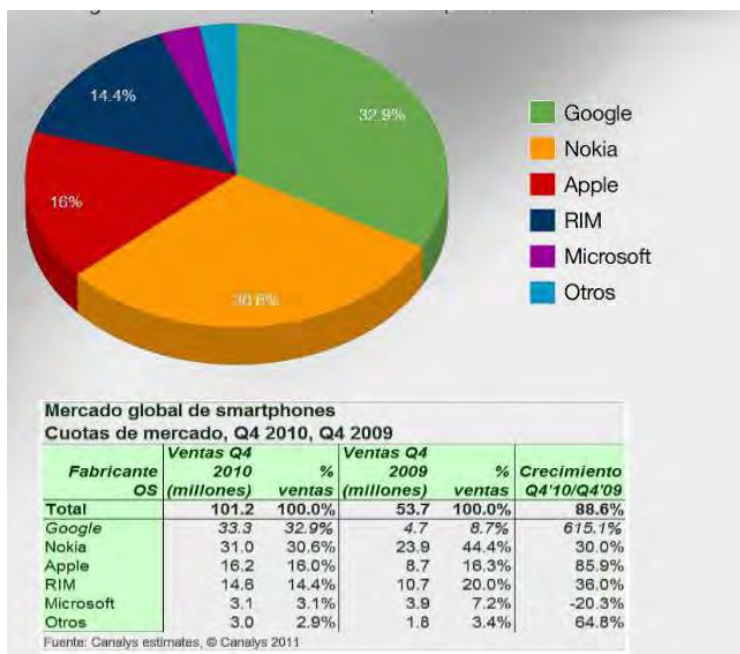


Figura 44 – Cuotas globales de mercado de SmartPhones por marca de SO (Q4 2010) (Fuente: Canalys estimates)

Dentro el año 2010, la progresión de la cuota de mercado de Android fue claramente favorable, superando con creces a su principal competidor iOS (ver Figura 44).

Fundamentos teóricos - Smartphone



Figura 45 – Progresión de cuotas de mercado de SmartPhones por SO en EEUU (2010) (Fuente: The Nielsen Company)

Finalmente, en mayo del 2011, los principales competidores de sistemas operativos para smartphones siguen siendo Android (37%), iOS (27%) y BlackberryOS (22%) (ver Figura 45) (ver Figura 46).

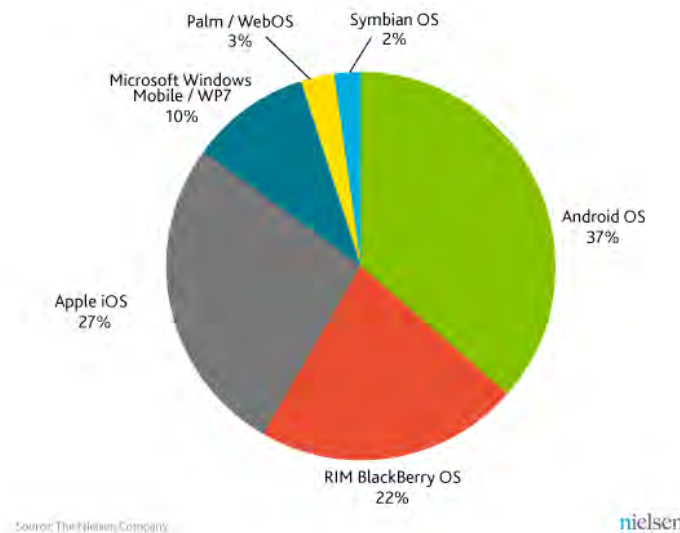


Figura 46 – Cuotas globales de mercado de SmartPhones por SO (2010) (Fuente: The Nielsen Company)

Fundamentos teóricos - Smartphone

A pesar de los tres principales competidores acabados de mencionar, los desarrolladores utilizan en mayor cantidad la plataforma Android, seguido de iOS, JavaME y Symbian. El 60% de los desarrolladores utilizan Android, superando en un 10% a su principal competidor iOS de Apple (*ver Figura 47*).

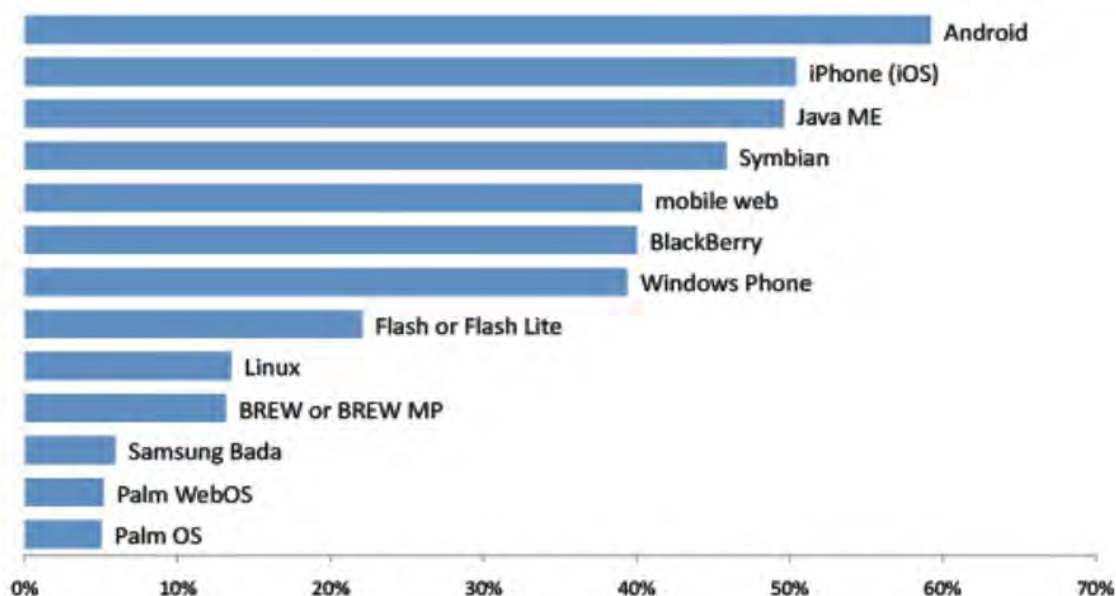


Figura 47 – Plataformas móviles más utilizadas por los desarrolladores (2010) (Fuente: Mobile Developer Economics)

La plataforma que dispone de más aplicaciones para poder ser instaladas es con mucha diferencia el iOS de Apple, seguida de Android (*ver Figura 48*). La cantidad de aplicaciones para Android se vio aumentada en un 18% solamente en el año 2010. A pesar de que iOS dispone de muchas más aplicaciones, su cantidad de aplicaciones gratuitas no es mucho mayor a la de Android. Hay un gran mercado potencial en el mercado de aplicaciones de Android, puesto que el crecimiento de ventas de teléfonos con dicho sistema operativo ha sido muy grande últimamente (*ver Figuras 44, 45 y 46*), pero aún no se han desarrollado todas las aplicaciones que debería implicar ese crecimiento; aunque el último año el crecimiento de aplicaciones Android ya ha realizado un gran aumento (*ver Figura 48*).

Android tiene una gran comunidad de desarrolladores escribiendo aplicaciones para extender la funcionalidad de los dispositivos. A la fecha, existen cerca de 200.000

Fundamentos teóricos - Smartphone

aplicaciones disponibles para Android. Android Market es la tienda de aplicaciones en línea administrada por Google.

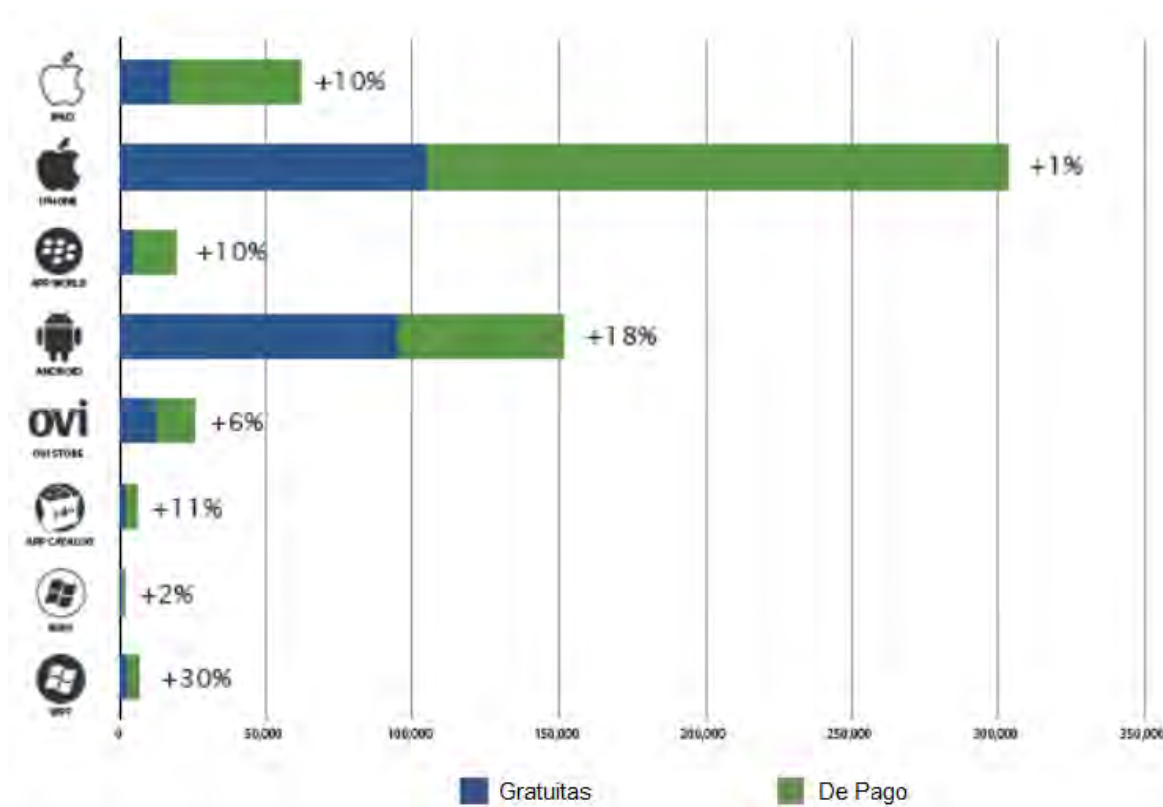


Figura 48 – Aplicaciones disponibles para los distintos SO de Smartphone en EE.UU (Enero 2011) (Fuente: Distimo)

Otro punto a favor de la elección de este sistema operativo, es que hay múltiples marcas que desarrollan modelos de smartphone con sistema operativo Android. Android permite al usuario tener un móvil con este SO sin necesidad de gastarse 600€. Hay teléfonos móviles Android desde unos 150€ hasta unos 700€. (ver Figura 49) El sistema está diseñado para poderse instalar en terminales muy diversos y es capaz de gestionar los diferentes tipos de hardware que esto supone.

Fundamentos teóricos - Smartphone



Figura 49 – Terminales android de gran rango de precios

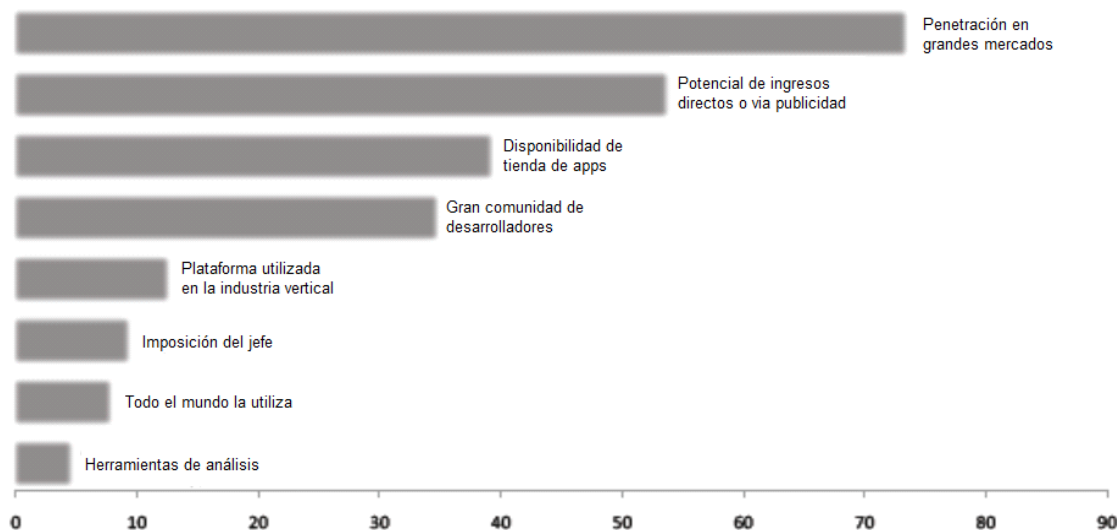


Figura 50 – Razones más consideradas por los desarrolladores al elegir un SO movil (2010) (Fuente: Mobile Developer Economics)

Además de las razones ya comentadas en este apartado, el criterio de selección se ha basado también en las razones que más consideran los desarrolladores de aplicaciones móviles a la hora de elegir un sistema operativo (ver Figura 50). De estas razones, a continuación se comentan las tres más relevantes para este proyecto, las cuales están entre las cuatro primeras posiciones de prioridades:

1. Penetración en grandes mercados: Ya se ha analizado que Android tiene una gran penetración de mercado y de que esta va en aumento ()

2. Disponibilidad de tienda de Apps: Tiene disponible la tienda online Android Market (de la cual se hablará a continuación).
3. Gran comunidad de desarrolladores: Como ya se ha comentado, tiene una gran comunidad de desarrolladores escribiendo aplicaciones para extender la funcionalidad de los dispositivos.

Android market



Android Market es un programa informático basado en un sistema abierto de distribución de contenidos desarrollado por Google para dispositivos basados en el sistema operativo Android, el cual permite a sus usuarios navegar, comprar, instalar y descargar aplicaciones desarrolladas por terceros. Android Market tiene un acceso fácil y rápido a sus aplicaciones. Las aplicaciones son creadas por desarrolladores de todo el mundo y posteriormente puntuadas por los usuarios de Android [43].

Cualquier desarrollador de Android puede publicar en el Android Market, con un previo pago único de 25\$. A diferencia de la AppleStore del iPhone, en el Android Market no se revisan las aplicaciones que se suben. El 70% de los beneficios de las ventas de las aplicaciones subidas son para el desarrollador y el 30% son para Google. Los usuarios pueden elegir que aplicación quieren descargarse a través de las votaciones y los comentarios de los demás usuarios [44].

Características

En el mercado existen diferentes versiones de Android conviviendo al mismo tiempo, la versión que actualmente está más extendida es la 2.2 (ver Figura 51), por lo que se debe tener en cuenta que la aplicación a realizar tiene que tener todas las funcionalidades soportadas por dicha versión [45].

Fundamentos teóricos - Smartphone

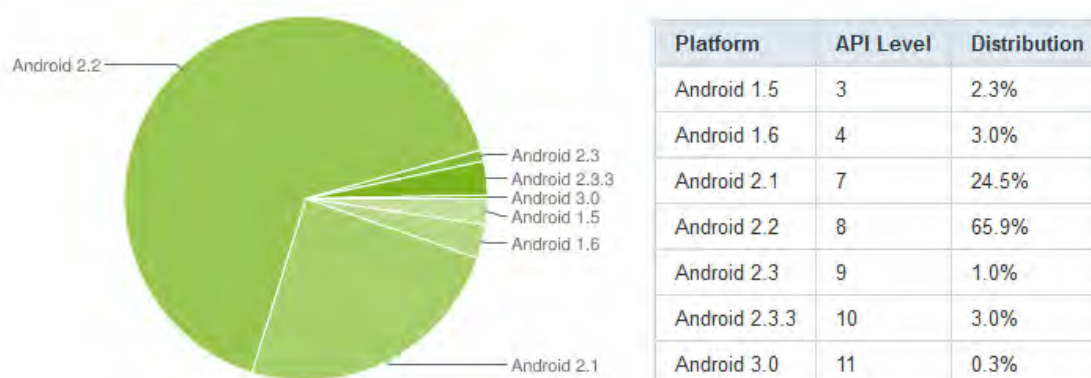


Figura 51 – Porcentaje de distribución de las versiones de Android
(Mayo 2011)(Fuente: Web Android Developers)

El sistema operativo Android incluye las siguientes características:

- **Diseño de dispositivo:** La plataforma es adaptable a pantallas más grandes, VGA, biblioteca de gráficos 2D, biblioteca de gráficos 3D basada en las especificaciones de la OpenGL ES 2.0 y diseño de teléfonos tradicionales.
- **Almacenamiento:** Utiliza SQLite, una base de datos liviana, la cual es usada para propósitos de almacenamiento de datos.
- **Conectividad:** Soporta las siguientes tecnologías de conectividad: GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE y WiMAX.
- **Mensajería :** Soporta las formas de mensajería SMS y MMS, incluyendo mensajería de texto. Además la “Android Cloud to Device Messaging Framework” (C2DM) es parte del servicio de Push Messaging de Android.
- **Navegador web:** El navegador web incluido en Android está basado en el motor de renderizado de código abierto WebKit, emparejado con el motor JavaScript V8 de Google Chrome. El navegador obtuvo una puntuación de 93/100 en el test Acid3.
- **Soporte Java:** Aunque las aplicaciones están escritas en Java, no hay una Máquina Virtual de Java en la plataforma. El código Java no es ejecutado. El

Fundamentos teóricos - Smartphone

código Java se compila en el ejecutable Dalvik y corre en la Máquina Virtual Dalvik. Dalvik es una máquina virtual especializada diseñada específicamente para Android y optimizada para dispositivos móviles que funcionan con batería y que tienen memoria y procesador limitados. El soporte para J2ME puede ser agregado mediante aplicaciones de terceros como el J2ME MIDP Runner.

- Soporte multimedia: Incluye soporte para los siguientes formatos multimedia: WebM, H.263, H.264 (en 3GP o MP4), MPEG-4 SP, AMR, AMR-WB (en un contenedor 3GP), AAC, HE-AAC (en contenedores MP4 o 3GP), MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF y BMP.
- Soporte para streaming: Streaming RTP/RTSP (3GPP PSS, ISMA), descarga progresiva de HTML (HTML5 <video> tag). Adobe Flash Streaming (RTMP) es soportado mediante el Adobe Flash Player. Se planea el soporte de Microsoft Smooth Streaming con el port de Silverlight a Android. Adobe Flash HTTP Dynamic Streaming estará disponible mediante una actualización de Adobe Flash Player.
- Soporte para hardware adicional: Android soporta cámaras de fotos, de vídeo, pantallas táctiles, GPS, acelerómetros, giroscopios, magnetómetros, sensores de proximidad y de presión, termómetro, aceleración 2d y 3d.
- Multi-táctil: Android tiene soporte nativo para pantallas multi-táctiles que inicialmente hicieron su aparición en dispositivos como el HTC Hero.
- Bluetooth: El soporte para A2DP (Advanced Audio Distribution Profile) y AVRCP (Audio/Video Remote Control Profile) fue agregado en la versión 1.5, el envío de archivos (OPP) y la exploración del directorio telefónico fueron agregados en la versión 2.0, y el marcado por voz junto con el envío de contactos entre teléfonos lo fueron en la versión 2.2.
- Videollamada: La versión principal de Android no soporta videollamada. Sin embargo, algunos dispositivos tienen una versión personalizada del sistema operativo que lo soporta, ya sea por la red del operador (como el Samsung i9000 Galaxy S) o sobre IP.

- Multitarea: Multitarea real de aplicaciones está disponible.
- Características basadas en voz: La búsqueda en Google a través de voz está disponible como "Entrada de Búsqueda" desde la versión inicial del sistema.
- Tethering: Android soporta tethering, el cual permite al teléfono ser usado como un punto de acceso a internet (alambico o inalámbrico). Para permitir a un computador portátil usar la conexión 3G del móvil android se podría requerir la instalación de software adicional.

Estructura del sistema operativo

La estructura del sistema operativo Android se compone de aplicaciones que se ejecutan en un framework Java de aplicaciones orientadas a objetos sobre el núcleo de las bibliotecas de Java en una máquina virtual "Dalvik" con compilación en tiempo de ejecución.

Las bibliotecas escritas en lenguaje C incluyen un administrador de interfaz gráfica (surface manager), un framework OpenCore, una base de datos relacional SQLite, una API gráfica OpenGL ES 2.0 3D, un motor de renderizado WebKit, un motor gráfico SGL, SSL y una biblioteca estándar de C Bionic.

El sistema operativo está compuesto por 12 millones de líneas de código, incluyendo 3 millones de líneas de XML, 2,8 millones de líneas de lenguaje C, 2,1 millones de líneas de Java y 1,75 millones de líneas de C++ [46].

Arquitectura

Estos son los componentes mayores del sistema operativo Android (*ver Figura 52*), cada sección se describe en detalle[47]:

Fundamentos teóricos - Smartphone



Figura 52 – Diagrama de Arquitectura Android

- Aplicaciones: Las aplicaciones base incluyen un cliente de email, programa de SMS, calendario, mapas, navegador, contactos, y otros. Todas las aplicaciones están escritas en el lenguaje de programación Java.
- Framework de aplicaciones: Los desarrolladores tienen acceso completo a los mismos APIs del framework usados por las aplicaciones base. La arquitectura está diseñada para simplificar el reuso de componentes. Cualquier aplicación puede publicar sus capacidades y cualquier otra aplicación puede luego hacer uso de esas capacidades (sujeto a reglas de seguridad del framework). Éste mismo mecanismo permite que los componentes sean reemplazados por el programador.
- Librerías: Android incluye un set de librerías C/C++ usadas por varios componentes del sistema Android. Estas capacidades se exponen a los desarrolladores a través del framework de aplicaciones de Android. Algunas son: System C library (implementación librería C standard), librerías de medios, librerías de gráficos 3d, SQLite, entre otras.
- Runtime de Android: Android incluye un set de librerías base que proveen la mayor parte de las funcionalidades disponibles en las librerías base del lenguaje de programación Java. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual Dalvik. Dalvik ha sido escrita de forma que un

dispositivo puede correr múltiples máquinas virtuales de forma eficiente. Dalvik ejecuta archivos en el formato Dalvik Executable (.dex), el cual está optimizado para memoria mínima. La Máquina Virtual está basada en registros, y corre clases compiladas por el compilador de Java que han sido transformadas al formato .dex por la herramienta incluida "dx".

- Núcleo Linux: Android depende de un Linux versión 2.6 para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, stack de red, y modelo de drivers. El núcleo también actúa como una capa de abstracción entre el hardware y el resto del stack de software.

Jerarquía visual de Android

La principal clase de Android es *activity*, un objeto de la clase *android.app.Activity*.

Una actividad hace multitud de cosas, pero por ella misma no presenta nada en la pantalla. Para conseguir que aparezca algo en la pantalla es necesario diseñar el UI (user interface), con *views* y *viewgroups*, que son las clases que se usan para crear la interfaz entre el usuario y la plataforma Android.

- Views: Una *view* es un objeto cuya clase es *android.view.View*. Es una estructura de datos cuyas propiedades contienen los datos de la capa y la información específica del área rectangular de la pantalla. Una *view* tiene: layout, drawing, focus change, scrolling, etc..

La clase *view* es útil como clase base para los widgets, que son unas subclases ya implementadas que dibujan los elementos en la pantalla. Los widgets contienen sus propias medidas, pero puedes usarse para construir la interfaz más rápidamente. La lista de widgets usables incluyen *Text*, *EditText*, *InputMethod*, *MovementMethod*, *Button*, *RadioButton*, *CheckBox*, y *ScrollView*.

- Viewgroups: Un *viewgroup* es un objeto de la clase *android.view.ViewGroup*, como su propio nombre indica, un *viewgroup* es un objeto especial de *view* cuya función es contener y controlar la lista de *views* y de otros *viewgroups*. Los *viewgroups* te permiten añadir estructuras a la interfaz y acumular complejos elementos en la pantalla que son diseccionados por una sola entidad.

La clase *viewgroup* es útil como base de la clase *layouts*, que son subclases implementadas que proveen los tipos más comunes de los *layouts* de pantalla. Los *layouts* proporcionan una manera de construir una estructura para una lista de *views*.

- Árbol estructurado de la interfaz UI: En la plataforma Android se define una *Activity* del UI usando un árbol de nodos *view* y *viewgroups*, (ver Figura 53). El árbol puede ser tan simple o complejo como se necesite, y se puede desarrollar usando los *widgets* y *layouts* que Android proporciona o creando las propias *views* del desarrollador.

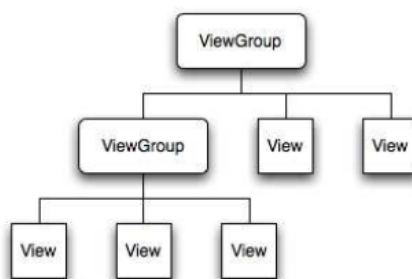


Figura 53 – Árbol de nodos view y viewgroups

Para añadir el árbol a la pantalla, la *Activity* llama al método *setContentView()* y pasa una referencia al objeto nodo principal. Una vez que el sistema Android ha referenciado el objeto nodo principal ya puede trabajar directamente con el nodo para anular, medir y dibujar el árbol. Cuando la *Activity* está activa y recibe el foco el sistema notifica la *activity* y pide al nodo principal medidas y dibuja el árbol. El nodo principal entonces pide que sus nodos hijos se dibujen a sí mismos, a partir de ese momento cada nodo *viewgroup* del árbol es responsable de pintar sus hijos directos.

Como se ha mencionado, cada *viewgroup* es el responsable de tomar medidas sobre el espacio que tienen, preparando a sus hijos y llamando a *Draw()* por cada hijo que se muestra a sí mismo. El hijo hace una petición sobre el tamaño y la localización del padre, pero el objeto padre toma la última decisión sobre el tamaño que cada hijo puede tener.

- *LayoutParams*: Cómo un hijo especifica su posición y su tamaño, todos los *viewgroup* usan como clase anidada una extensión de *ViewGroup.LayoutParams*.

Esta subclase contiene los tipos de propiedades que definen la posición y el tamaño de un hijo, en propiedades apropiadas para la clase de grupo de clases.

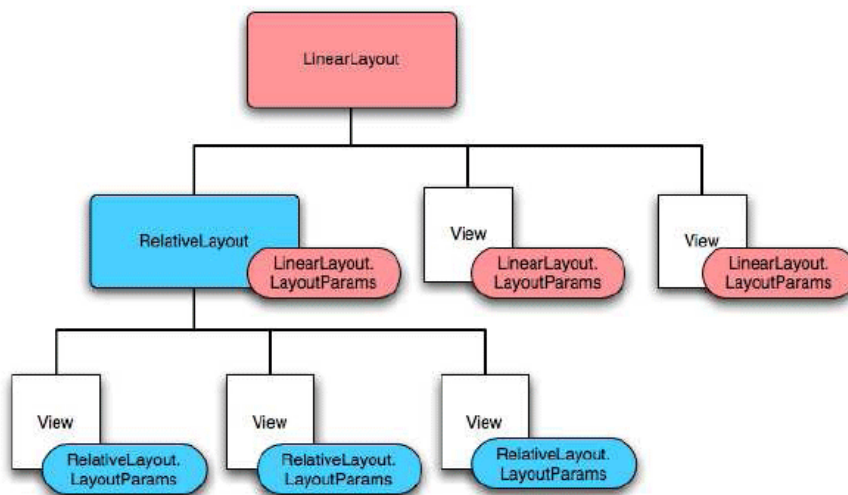


Figura 54 – Arbol de LayoutParams de un árbol de nodos, view y viewgroups.

Hay que reseñar que cada subclase *LayoutParams* tiene su propia sintaxis para cambiar los valores. Cada elemento hijo debe definir unos *LayoutParams* que sean apropiados para su padre, aunque se podrían definir diferentes *LayoutParams* para sus hijos (ver Figura 54).

Todos los *viewgroups* incluyen anchura y altura. Muchos también incluyen márgenes y bordes. Se puede especificar exactamente la altura y la anchura, aunque a menudo esto no se quiere definir. Normalmente se indica a la view que tenga las dimensiones del tamaño de su contenedor, o que llegue a ser tan grande como el contenedor le permita.

Entorno de desarrollo: Eclipse



Figura 55 – Logo del entorno de desarrollo y la SDK

El entorno de desarrollo (IDE) que se ha utilizado es Eclipse [48], que es de código abierto. Utilizando el kit de herramientas software para desarrollar en Android (Android SDK) que incluye un emulador de un sistema Android para poder ejecutar las aplicaciones en el PC sin necesidad de cargarlas en un Smartphone.

Eclipse fue desarrollado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge. Eclipse es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios.

Criterio de selección

La decisión que tomó el técnico de TheraGames por este entorno de desarrollo fue motivada por las siguientes razones:

1. La organización Catdroid (una comunidad de usuarios de Android de Cataluña), proporcionaba a través de su página web el entorno de desarrollo con el SDK de Android integrado y listo para empezar a desarrollar aplicaciones Android, así como algunas aplicaciones de ejemplo [49].
2. Los numerosos proyectos Eclipse de ejemplo, para aprender a desarrollar para Android.
3. Dispone de una comunidad activa de desarrolladores dispuestos a prestar su ayuda.
4. Menor esfuerzo y tiempo de desarrollo [44].
5. Presentación visual de componentes [44].

Características

Eclipse es un entorno de desarrollo integrado de código abierto. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado *Java Development Toolkit* (JDT) y el compilador (ECJ)

que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse).

Eclipse dispone de un Editor de texto con resaltado de sintaxis. La compilación es en tiempo real. Tiene pruebas unitarias con JUnit, control de versiones con CVS, integración con Ant, asistentes (*wizards*) para creación de proyectos, clases, tests, etc., y refactorización. Asimismo, a través de "plugins" libremente disponibles es posible añadir control de versiones con Subversion e integración con Hibernate.

2.7. Tecnologías de servidor web

Un servidor web o servidor HTTP [50] es un programa que procesa cualquier aplicación del lado del servidor realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente. El código recibido por el cliente suele ser compilado y ejecutado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo. Generalmente se utiliza el protocolo HTTP para estas comunicaciones, perteneciente a la capa de aplicación del modelo OSI. El término también se refiere al ordenador que ejecuta el programa.

Para cubrir la necesidad de guardar los datos del jugador Drappen's World y su respectiva configuración realizada por el terapeuta, así como el estado de evolución dentro del juego, es imprescindible tener una base de datos.

Esta base de datos no puede estar instalada en local en el ordenador del jugador, sino que debe estar en internet para que puedan tener acceso a ella tanto las aplicaciones de los diferentes jugadores como el propio terapeuta. La necesidad de tener una base de datos en la nube nos obliga a utilizar un servidor web.

2.7.1. Apache2 http Server



Desarrollador: Apache Software Foundation

Escrito en: C

Plataforma: Windows, Novell NetWare, Mac OS X y los sistemas basados en Unix

Web Oficial: <http://www.apache.org/>

Apache HTTP Server es un servidor web de distribución libre y de código abierto. Apache es usado principalmente para servir páginas web estáticas y dinámicas en la WWW. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o que utilizarán características propias de este servidor web. Apache es el servidor web del popular sistema XAMP (explicado más adelante, en el apartado WAMP).

Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del servidor web NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Apache fue la primera alternativa viable para el servidor web de Netscape Communications, actualmente conocido como Sun Java System Web Server.

Criterio de selección

Es el servidor HTTP más popular del mundo desde abril de 1996. Actualmente tiene una penetración del 60% del total de servidores web del mundo. La principal competencia de Apache es el IIS (Microsoft Internet Information Services) de Microsoft, aunque tiene una cuota de mercado inferior a la mitad de la cuota de Apache. Este servidor web es redistribuido como parte de varios paquetes propietarios de software, incluyendo la base de datos Oracle y el IBM WebSphere application server [51][52].

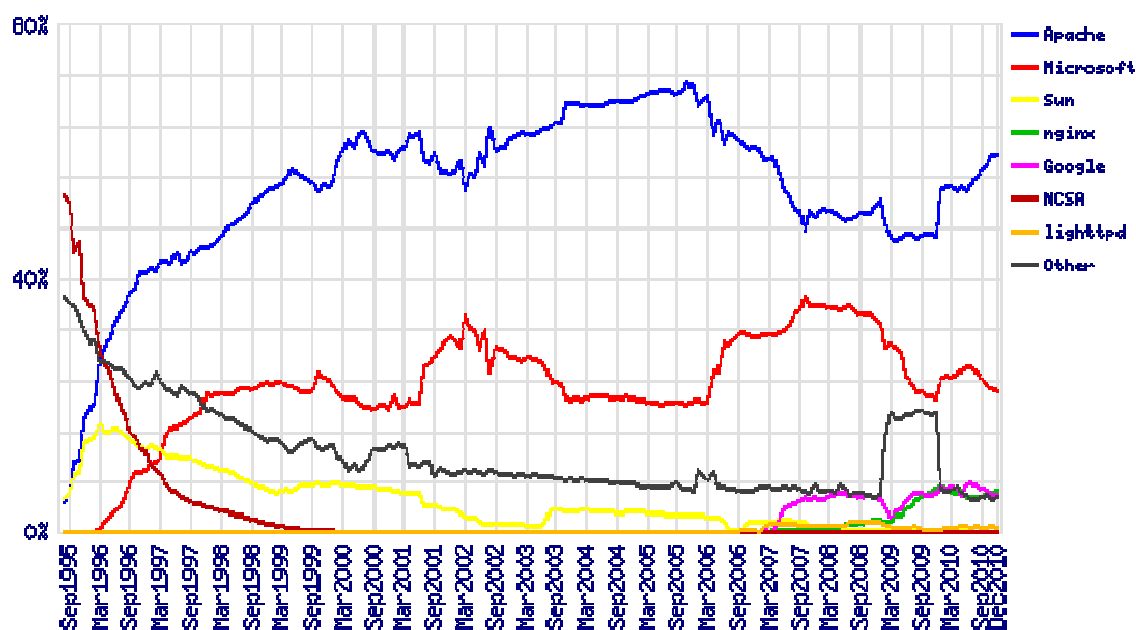


Figura 56 – Estadísticas de los servidores web más utilizados.
(Julio 2009)(Fuente: Netcraft)

Es software libre, y no solo acapara actualmente un 60% de la red compitiendo contra Microsoft, Sun, y varios más. También continúa en crecimiento, dejando obsoletos a servidores desarrollados por empresas que cobran por el uso de sus servers, y a otras que no alcanzan las prestaciones de apache server (ver Figura 56).

El criterio de selección de este tipo de servidor para que soportase la base de datos de Dappen's World estuvo motivado por dos razones:

1. La Facultad de La Salle disponía de este servidor y teníamos acceso a él.
2. Acapara actualmente un 60% del total de servidores de la red de Internet.

Características

Apache ha sido desarrollado y mantenido por una comunidad abierta de desarrolladores dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation. Está escrito en lenguaje C, bajo la "licencia Apache".

La versión 2 del servidor Apache fue una reescritura sustancial de la mayor parte del código de Apache 1.x, enfocándose en una mayor modularización y el desarrollo de una

capa de portabilidad, el Apache Portable Runtime. Apache 2.x incluyó multitarea en UNIX, mejor soporte para plataformas no Unix (como Windows), una nueva API Apache y soporte para IPv6 [53].

Propiedades de Apache

- Soporte para los lenguajes perl, python, tcl y PHP.
- Módulos de autenticación: mod_access, mod_auth y mod_digest.
- Soporte para SSL (protocolo de capa de conexión segura) y TLS (protocolo de seguridad de la capa de transporte).
- Permite la configuración de mensajes de errores personalizados y negociación de contenido.
- Permite autenticación de base de datos basada en SGBD.

Mejoras de la versión Apache2:

- Nuevo sistema de configuración y compilación
- Soporte Multiprotocolo
- Soporte mejorado para las plataformas que no son tipo Unix
- Nueva interfaz de programación (API) de Apache.
- Soporte de IPv6
- Los módulos pueden escribirse para que se comporten como filtros sobre el flujo de contenido.
- Mensajes de error en diferentes idiomas.
- Configuración simplificada.
- En Windows NT usa ahora utf-8 para la codificación de los nombres de fichero.

- Cuando se evalúan las expresiones tipo, se usa siempre la potente sintaxis de Perl 5.

2.7.2. MySQL



Desarrollador: Sun Microsystems

Escrito en: C, C++

Plataforma: Windows, MacOS, Linux

Web Oficial: <http://www.mysql.com/>

Se trata de un sistema gestor de base de datos que hace de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. [54]

Criterio de selección

TheraGames lo ha elegido por su facilidad de uso y estar ya configurado en el servidor de La Salle, lo cual permitía avanzar más rápidamente en la creación de la base de datos de Drappen's World. Puesto que el grupo disponía de poco tiempo para llevar a cabo el proyecto, no encontró oportuna la configuración de otros sistemas.

Características

Las principales características de este gestor de bases de datos son las siguientes [55]:

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).

- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.

2.7.3. phpMyAdmin



Desarrollador: The phpMyAdmin Project

Escrito en: PHP

Plataforma: Windows, MacOS, Linux

Web Oficial: <http://www.phpmyadmin.net>

Es una herramienta que sirve para manejar la administración de MySQL a través de páginas web, utilizando Internet. [56]

Criterio de selección

Este proyecto se encuentra vigente desde el año 1998, siendo el mejor evaluado en la comunidad de descargas de SourceForge.net como la descarga del mes de diciembre del 2002 [57].

La principal razón de la elección de phpMyAdmin por parte del equipo de desarrollo de TheraGames fue su facilidad de uso y el hecho que ya estuviera instalado y configurado en el servidor de La Salle, lo cual nos permitía ganar mucho tiempo.

Características

Actualmente puede crear y eliminar Bases de Datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios, exportar datos en varios formatos y está disponible en 62 idiomas. Todo a través de un navegador web. Se encuentra disponible bajo la licencia GPL[56].

2.7.4. WAMP



Desarrollador: Apache Software Foundation, Sun Microsystems, The phpMyAdmin Project

Escrito en: c, c++, php, Perl, Python

Plataforma: Windows, MacOS, Linux

Web Oficial: <http://www.wampserver.com/>

El uso de WAMP (ver Figura 57) permite servir páginas HTML a internet, además de poder gestionar datos en ellas, al mismo tiempo, WAMP proporciona lenguajes de programación para desarrollar aplicaciones web. Representa la suma de la utilización de varias tecnologías: Windows, Apache, MySQL y PHP [58]:



Figura 57 – Componentes que forman Wamp

Criterio de selección

La utilización de WAMP vino impuesta por las elecciones del sistema operativo Windows, el servidor Apache, la base de datos MySQL y la utilización de php para poder acceder a la base de datos desde la aplicación PC y desde la aplicación Smartphone con conexión http.

Características

XAMP (X Sistema Operativo, Apache, MySQL, PHP, Perl) [59] es el acrónimo usado para describir un sistema de infraestructura de internet que usa las siguientes herramientas:

- Windows (X=**W**) o Linux(X=**L**) o MacOS (X=**M**)
- **A**pache, como servidor web
- **M**ySQL, como gestor de bases de datos
- **P**HP (generalmente), **P**erl, o **P**ython, como lenguajes de programación.



LAMP es el sistema análogo que corre bajo ambiente Linux

WAMP es el sistema análogo que corre bajo ambiente Windows

MAMP es el sistema análogo que corre bajo ambiente Macintosh

Figura 58 – Sistemas Operativos y tecnologías
XAMP

XAMP se distribuye con un paquete que incluye :

- Apache HTTPD 2.2.9 + Openssl 0.9.8h
- MySQL 5.0.51b
- PHP 5.2.6
- PHP 4.4.8

Fundamentos teóricos - Tecnologías de servidor web

- phpMyAdmin 2.11.7
- FileZilla FTP Server 0.9.25
- Mercury Mail Transport System 4.52

3. Parte práctica

En esta sección se explican todas las partes de desarrollo que ha llevado a cabo el técnico de TheraGames y autor de este documento. Se analizan algunos diagramas para comprender el comportamiento y la estructura del software de Drappen's World.

También se entra en detalle sobre el desarrollo software para la camiseta sensorial así como para el mini juego (que se jugará en el exterior de casa) Smartphone que utiliza su GPS integrado. Por último se muestra la estructura de la base de datos, se explica cómo y cuando se accede a ella.

3.1. Diagrama de arquitectura

En el diagrama de arquitectura que se presenta a continuación (*ver Figura 59*) se representan los distintos componentes hardware de Drappen's World y como se establece la comunicación entre ellos.

Drappen's World consta de dos aplicaciones, una se ejecuta en PC y otra en Smartphone (Android). La aplicación más importante con mucha diferencia es la que se ejecuta sobre el PC. Ésta es la que implementa toda la lógica del juego y la que utiliza el motor gráfico para representar las diferentes escenas virtuales. La aplicación para Smartphone forma parte de un solo mini juego del serious game.

Parte práctica - Aplicación PC



Figura 59 – Diagrama de Arquitectura Drappen's World

Analizando el diagrama (ver Figura 59), puede observarse como centro del serious game una aplicación ejecutada en un PC, la cual utiliza el motor gráfico Irrlicht. El PC recibe información de tres partes: desde Kinect (mediante puerto USB), desde una camiseta sensorial con pulsímetro (mediante bluetooth) y de una base de datos (mediante el protocolo http).

A su vez, la aplicación PC envía información a la base de datos por el mismo protocolo y opcionalmente puede ser conectada a un televisor para jugar en una pantalla más grande.

El Smartphone utiliza sistema operativo Android y se conecta a la base de datos utilizando el mismo protocolo que la aplicación ejecutada sobre PC.

3.2. Aplicación PC

Es la aplicación principal de Drappen's World. La otra es la aplicación Smartphone, que forma parte de un solo mini juego y es controlada por esta.

3.2.1. Diagrama de transición de estados

Para comprender los distintos estados del serious game y como se transiciona entre ellos se ha realizado un diagrama de estados.

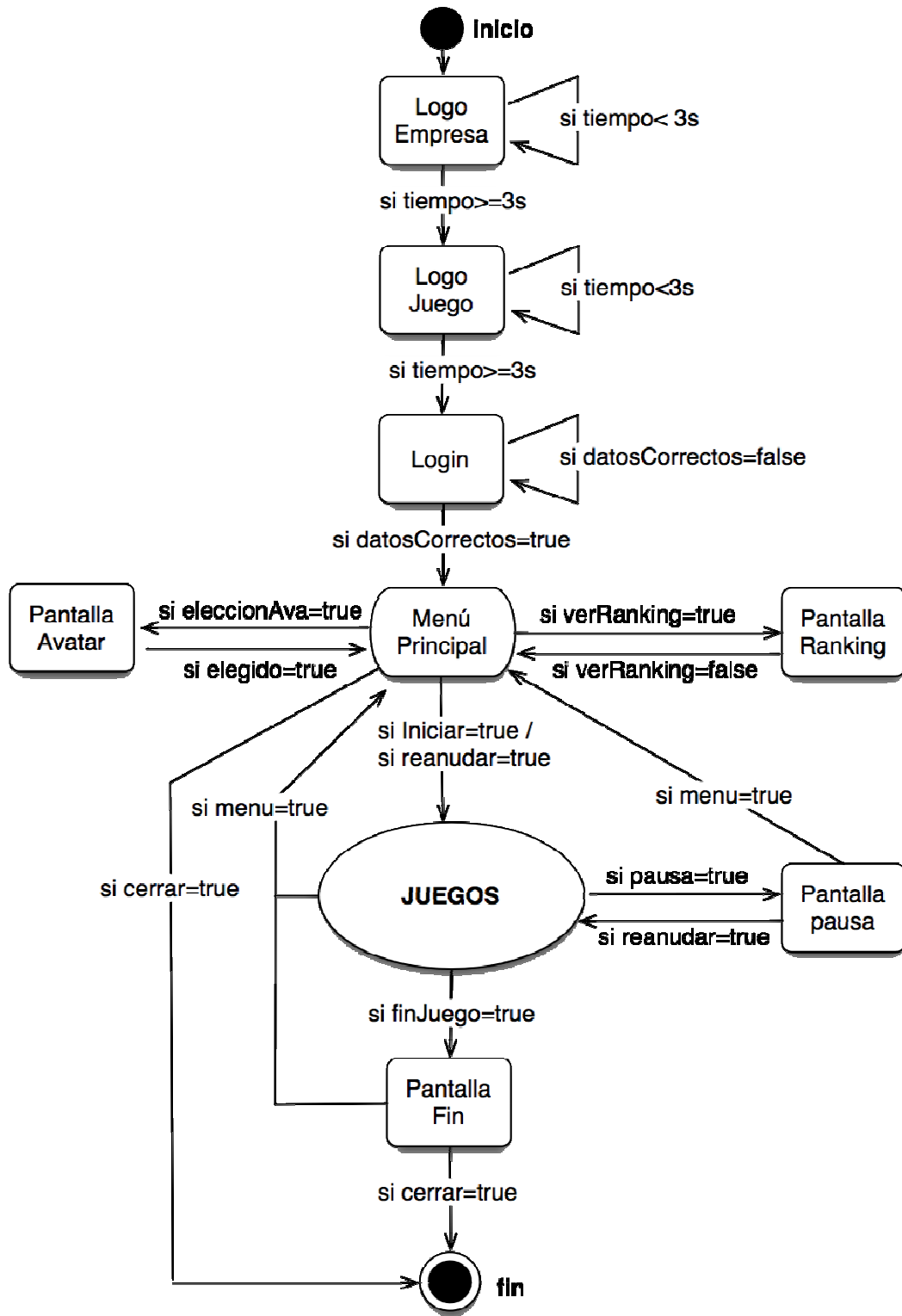


Figura 60 – Diagrama de transición de estados general

Parte práctica - Aplicación PC

A continuación se procede a la explicación del diagrama:

- 1) El serious game empieza con una pantalla con el logo de la empresa seguida de una pantalla con el logo de Drappen's World.



Figura 61 – Pantalla de bienvenida con el logo de la empresa.



Figura 62 – Pantalla de bienvenida con el logo del serious game.

- 2) Entonces, aparece la pantalla de login para iniciar la sesión. El usuario debe ingresar su Nick y contraseña de forma correcta para pasar al siguiente estado.



Figura 63 – Pantalla prototipo para el login

- 3) El siguiente estado es la pantalla de Menú principal. Desde el menú puede pasar a los siguientes estados

Parte práctica - Aplicación PC

- a. Ir a la pantalla donde se puede consultar el ranking de todos los jugadores. Desde la cual se regresa al menú cuando el usuario da la orden.
- b. Ir a la pantalla de elección de avatar. Desde la cual se regresa al menú una vez realizada la elección.
- c. Iniciar un nuevo juego.
- d. Reanudar una partida anterior.

Desde cualquiera de estos estados puede regresar al menú principal.



Figura 64 – Prototipo de la pantalla del menú

- 4) Si el usuario ha elegido iniciar o reanudar una partida, pasa al estado “JUEGOS” (ver Figura x). Este estado se divide en muchos otros subestados que se analiza más adelante con un diagrama que lo desarrolla a más nivel de detalle.

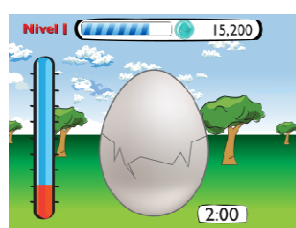


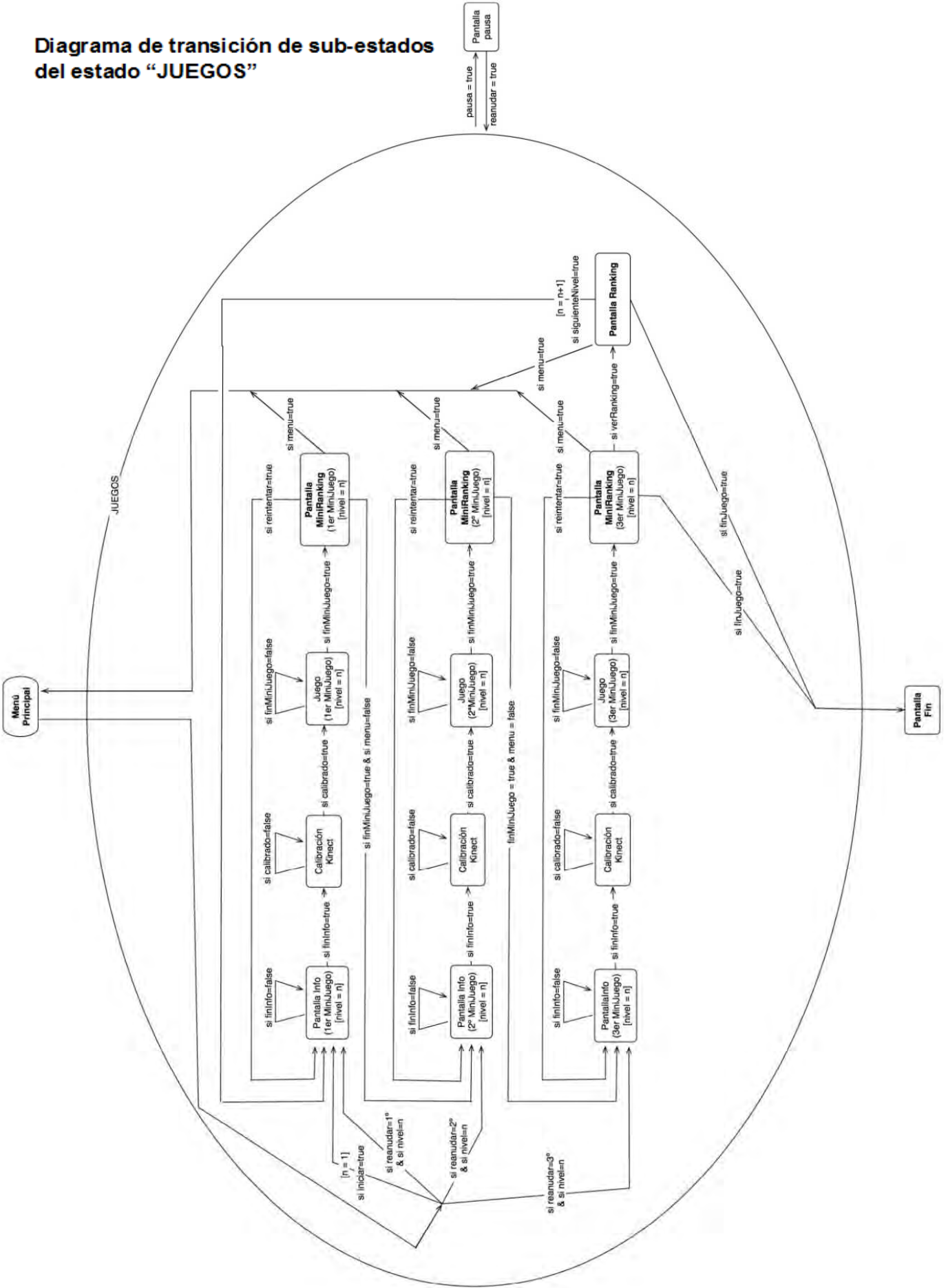
Figura 65 – Prototipo de una de las escenas de un minijuego.

- 5) Desde “JUEGOS” el usuario puede transicionar hasta los siguientes estados:
 - a. Regresa al Menú principal.

Parte práctica - Aplicación PC

- b. Ir a la pantalla de Pausa. De la cual se puede regresar al mismo punto donde se estaba o volver al menú principal.
 - c. Ver la pantalla correspondiente al final del juego, en caso de haberlo completado.
- 6) Desde el estado "Pantalla Fin" (pantalla correspondiente al final del juego), se puede regresar al menú principal o cerrar la aplicación.

Diagrama de transición de sub-estados del estado "JUEGOS"



Parte práctica - Aplicación PC

El estado "JUEGOS" se ha explicado con más detalle con un diagrama aparte para poder entender con más facilidad el comportamiento general de la aplicación y después adentrarse en los estados de los distintos mini juegos.

Drapen's World está compuesto por varios niveles, de los cuales a día de hoy se ha desarrollado uno a modo de prototipo. Cada nivel son 3 mini juegos. De todos los estados los cuatro primeros que se citan a continuación, se repiten en cada mini juego:

1) Pantalla Info.

En este estado se dan las instrucciones al usuario del mini juego concreto.



Figura 66 – Prototipo para las Instrucciones de un minijuego en la pantalla de info.

2) Calibración Kinect

Estado donde se realiza la calibración de kinect. Para que ello sea posible, se explica al usuario como tiene que colocarse. Tiene que colocarse en una posición concreta.



Figura 67 – Prototipo para las instrucciones de calibración de Kinect.

3) Juego

Estado donde el usuario juega al mini juego actual concreto.

Parte práctica - Aplicación PC



Figura 68 – Prototipo de una de las escenas de un minijuego.

4) Pantalla Mini Ranking

Al acabar cada mini juego, el jugador es direccionado al mini ranking. En este estado el jugador ve las puntuaciones y logros conseguidos solo por el mismo en ese minijuego. Hay tantas posiciones en el ránking como.



Figura 69 – Prototipo de la pantalla de mni ranking posterior a un mini juego

De “Pantalla Mini Ranking” se puede transicionar a los siguientes estados:

- a) Menu Principal
- b) Pantalla Info. (del mismo minijuego)

En caso de que el jugador decida volver a jugar al mismo mini juego para superar sus logros y puntuaciones.

- c) Pantalla Info (del siguiente minijuego)

Si el jugador decide avanzar al siguiente juego.

- d) Pantalla Ranking

Parte práctica - Aplicación PC

Solo se transiciona a "Pantalla Ranking" si el jugador estaba en el último mini juego del nivel y da la orden de ir a esa pantalla. Tiene que ir a esa pantalla para ver el ranking y para después ir al siguiente nivel.

e) Pantalla Fin

Si el serious game se ha completado y el jugador no desea ver la pantalla de ranking.

5) Pantalla Ranking



Jugador	Nivel	Tiempo	Puntos
Pablo	1	400	60,350
Sánchez Rey	3	215	70,370
Yelo	2	220	60,000
Shes	2	245	60,000
Super Juan	1	130	61,500
Fernando	2	220	2,000
Walter	3	220	300

Figura 70 – Prototipo de la pantalla de ranking posterior a la finalización de un nivel

Desde este estado se puede ir a los siguientes estados:

- Menú Principal
- Pantalla Info. (del siguiente mini juego del siguiente nivel)
- Pantalla Fin, en el caso de haberse completado todo el serious game.

Desde el estado "JUEGOS" solo se puede llegar al estado "Pantalla Ranking" si se está en el estado "Pantalla Mini Ranking" del último juego del nivel en cuestión.

6) Pantalla Info (del siguiente nivel) Se llega a este estado solo desde el estado "Pantalla Ranking". Quiere decir que el jugador ha superado el nivel y puede aumentar de nivel. El siguiente nivel tendrá tres mini juegos más.

7) Pantalla Fin

Solo se puede llegar a este estado desde los estados "Pantalla Mini Ranking" o "Pantalla Ranking" si el jugador ha completado el serious game

3.2.2. Diagramas de módulos software

Para explicar cómo se estructura el software de Drappen's World se ha realizado un diagrama donde se muestra la interacción entre los distintos módulos software (paquetes y clases) a nivel general (ver Figura 73). Después se entra en detalle en los paquetes, desglosándolos en clases.

A continuación se explica el significado de tres símbolos que se utilizan mucho en los siguientes diagramas y pueden dar lugar a confusión:

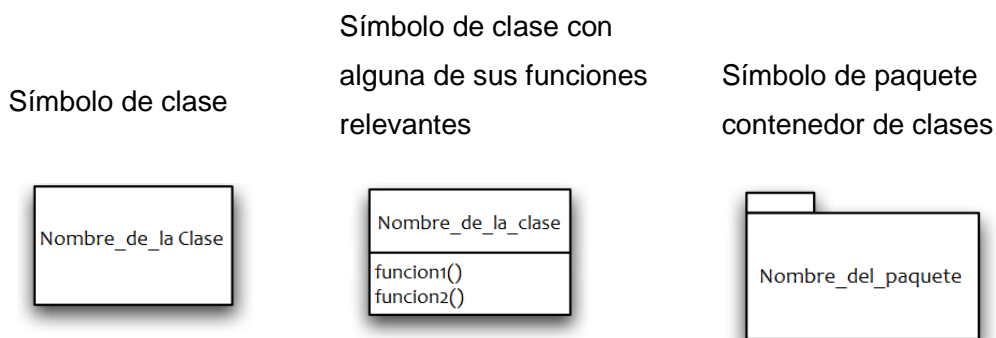


Diagrama de software general

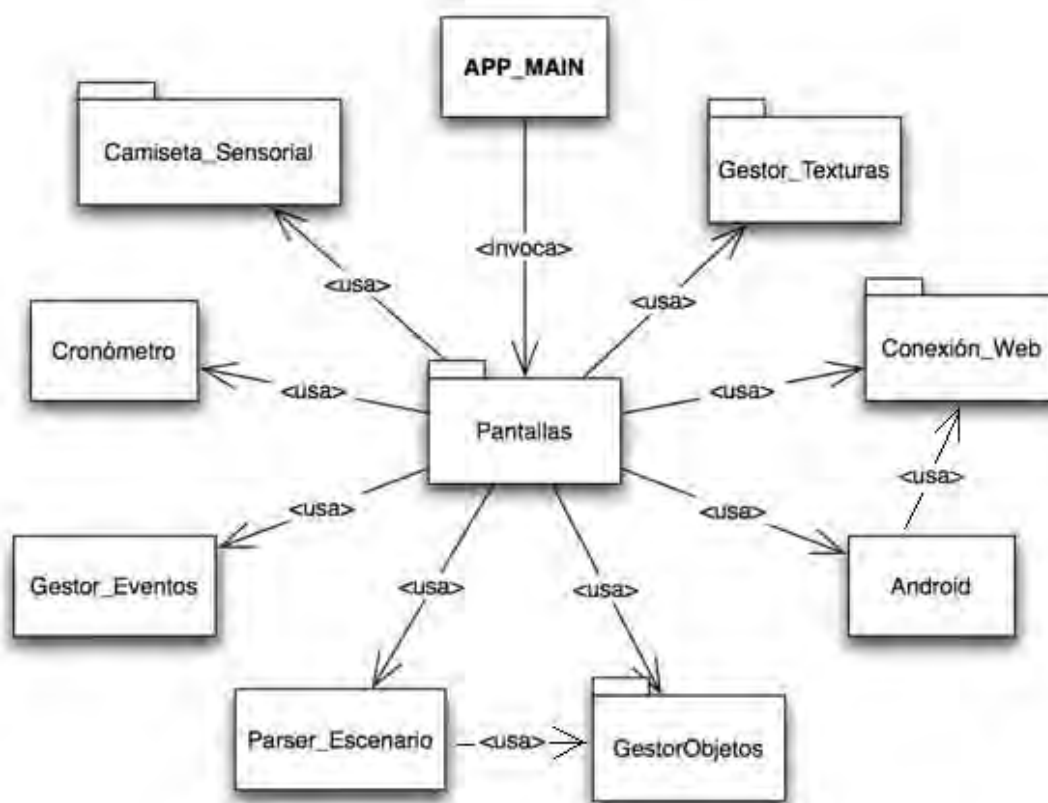


Figura 71 – Diagrama de módulos de software general

- **APP_MAIN:** Es la clase principal, se encarga de inicializar la aplicación y llamar al gestor de texturas.
- **Pantallas:** Es el módulo más importante de la aplicación, se encarga de gestionar que pantalla tiene que mostrarse en cada situación y pintarla (pantalla de login, pantalla de un mini juego determinado, pantalla de pausa, pantalla de ranking, etc...).
- **Gestor_Texturas:** Paquete de clases que nos permite tratar de una forma más sencilla con las texturas que si lo hiciéramos directamente con las funciones del motor gráfico

Parte práctica - Aplicación PC

- **Conexion_Web:** Este paquete contiene las clases necesarias que permiten la conexión web a través del protocolo http. Una vez hecha la conexión permite leer que valores retorna la página.
- **Android:** Es una clase que hace de interfaz entre el Smartphone y la aplicación PC. Lo hace utilizando una base de datos como puente. Esta clase consulta y modifica unos campos que también lee y modifica la aplicación Smartphone. De esta manera se establece una comunicación bidireccional entre la clase Android y la aplicación Smartphone. Utiliza la el paquete conexión web para poder acceder al servidor donde está la base de datos mediante el protocolo http.
- **Gestor_Objeto:** Este paquete gestiona todos los objetos que aparecen en la escena: los objetos que forman el entorno y los objetos que tienen alguna interacción con el usuario o el personaje principal (controlado por Kinect). Se encarga también de controlar las colisiones entre objetos y mantiene una sola instancia de cada tipo de objeto.
- **Parser_Escenario:** Esta clase se encarga de leer desde un fichero XML y pintar el entorno y sus objetos en función de lo que el XML especifica. Crea los objetos con la ayuda de "Gestor_Objeto" El XML especifica que objeto debe pintarse y en qué posición.
- **Gestor_Eventos:** Clase que captura los eventos del teclado y del ratón. Dispone de funciones implementadas para saber si una tecla del teclado o un botón del ratón han sido pulsados, soltados o ambas cosas. Nos permite saber que tecla o botón ha realizado la acción. También dispone de funciones para saber en qué posición (x,y) de la pantalla esta el cursor.
- **Cronometro:** Esta clase nos permite crear un cronómetro de una manera muy sencilla. Dispone de funcionalidades para iniciarlo, resetearlo y obtener el tiempo transcurrido.
- **Camiseta_Sensorial:** Se trata de un paquete de clases encargado de recibir los datos de la camiseta sensorial a través de un puerto serie emulado y procesar los

datos recibidos. La camiseta envía varios parámetros, pero este paquete se encarga de filtrar solo la frecuencia cardíaca.

Diagrama de software del paquete Gestor_Texturas

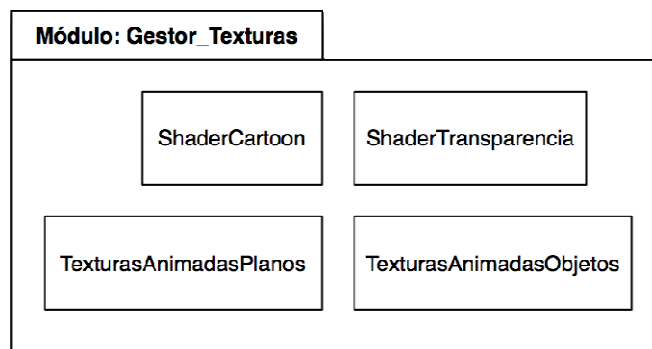


Figura 72 – Diagrama de clases del módulo Gestor_Texturas

- **ShaderCartoon:** Clase para aplicar un cartoon shader.
- **ShaderTransparencia:** Clase para aplicar un shader que permite la detección del canal α de las imágenes y permitir cargar imágenes respetando sus transparencias.
- **TexturasAnimadasObjetos:** Permite aplicar texturas a los nodos de escena de Irrlicht de una manera más sencilla que con las funciones proporcionadas por el motor. Además, permite definir varias texturas para un mismo objeto y hacer que estas se vayan cambiando en secuencia, cada cierto tiempo previamente definido.
- **TexturasAnimadasPlanos:** Permite aplicar texturas sobre planos de dos dimensiones de una manera más sencilla que con las funciones proporcionadas por el motor. Además, permite definir varias texturas para un mismo plano y hacer que estas se vayan cambiando en secuencia, cada cierto tiempo previamente definido.

Diagrama de software del paquete Conexión_Web

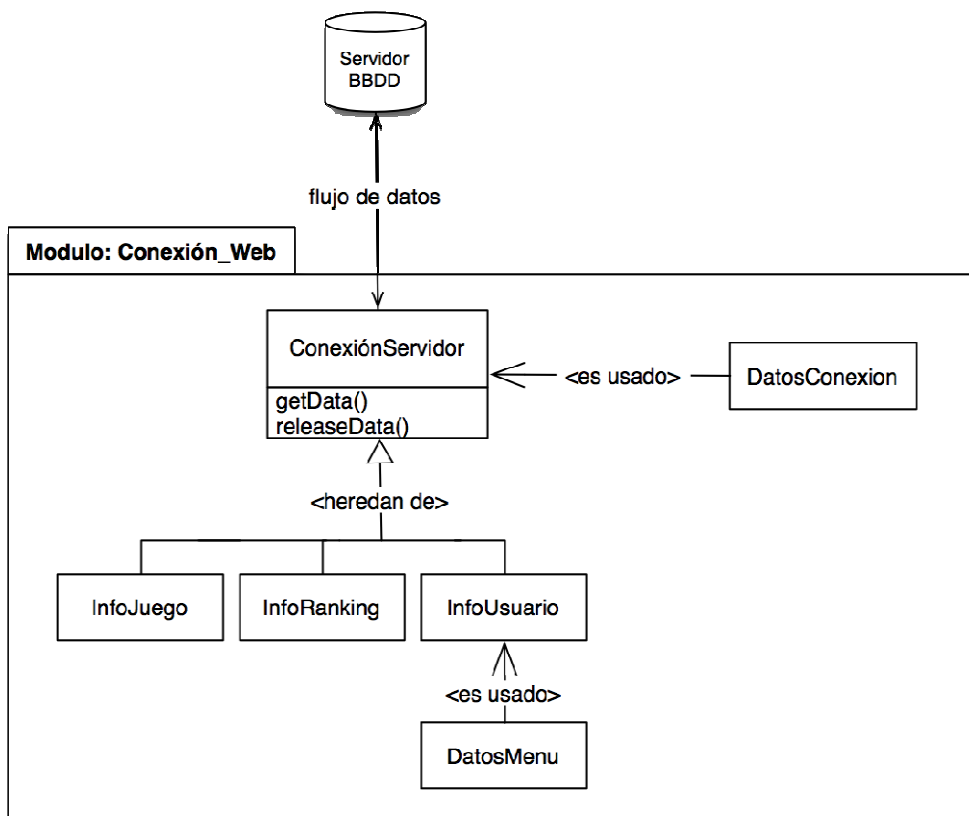


Figura 73 – Diagrama de clases del módulo Conexión_Web

- **ConexionServidor:** Se encarga de realizar las conexiones http al servidor donde se aloja la base de datos de Drappen's World. Lo hace utilizando una librería llamada "libcurl". Tiene dos funciones principales, "getData" se conecta a la url especificada y guarda la información que devuelve esa conexión. "releaseData" es necesaria para vaciar el búfer de contenido de una conexión anterior.
- **DatosConexion:** Dispone de todos los datos necesarios para conectarse al servidor y a la base de datos de Drappen's World. Esta clase se encarga de determinar que archivo php (alojado en el servidor) tiene que llamarse para realizar la operación que se precise sobre la base de datos.

Parte práctica - Aplicación PC

- **InfoJuego:** Registra la información del juego actual: que mini juego se está realizando, en qué nivel está el jugador, cuantos puntos globales lleva, cuantos puntos lleva en este mini juego y que ítems ha conseguido a lo largo del juego, así como en este mini juego. Se encarga de mantener esos datos actualizados en la base de datos.
- **InfoRanking:** Dispone de la información actualizada del ranking global de los jugadores de Drappen's World, nick de cada jugador, el nivel en que se encuentran, los ítems recolectados y los puntos globales en la evolución del serioius game.
- **InfoUsuario:** Contiene toda la información disponible en la base de datos referente al jugador que está conectado en este momento. Esta clase usa la clase DatosMenu para identificar al usuario.
- **DatosMenu:** Clase que se encarga de capturar la información que se introduce en los menús de información del usuario como el Nick y la contraseña.

Diagrama de software del paquete Gestor_Objetos

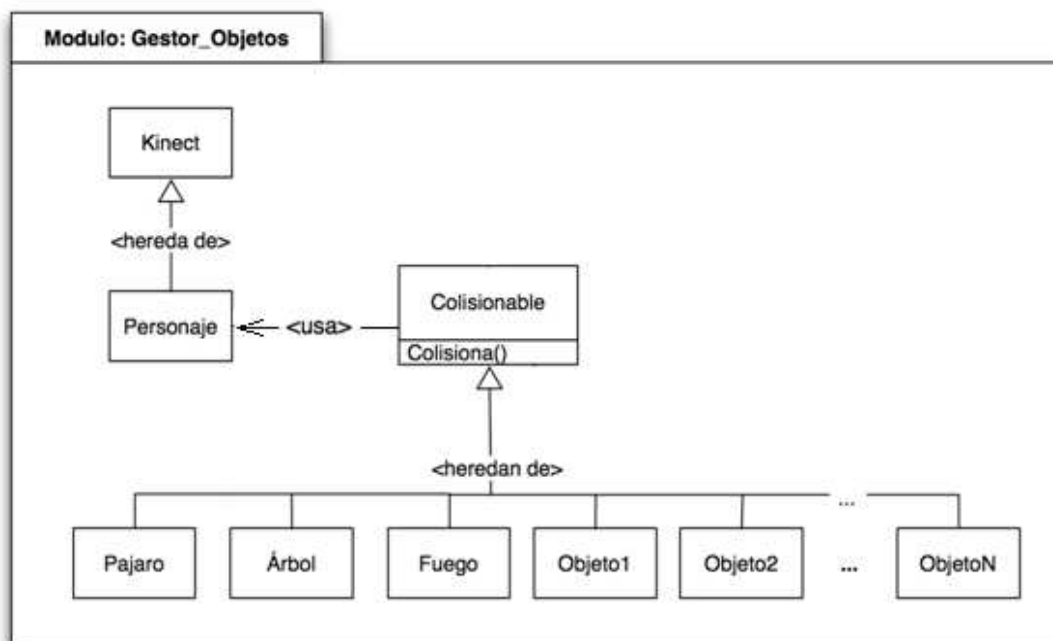


Figura 74 – Diagrama de clases del módulo Gestor_Objetos

- **Kinect:** Clase que interactúa con el periférico Kinect y registra cual es la posición en la que se encuentra el usuario (se podrá encontrar más información al respecto en la memoria de Pau Moreno).
- **Personaje:** Esta clase carga una malla articulada, con los diferentes huesos de las extremidades definidos. Esta malla mantiene la misma pose que el usuario gracias a las funcionalidades heredadas de la clase Kinect. Además, en la clase “Personaje” hay una serie de movimientos definidos y es capaz de detectar si el usuario los hace correctamente.
- **Colisionable:** Clase encargada de detectar si un objeto ha colisionado con el personaje principal.
- **Pajaro, Arbol, Fuego, Objeto1, Objeto2, ..., ObjetoN:** Todas estas clases se encargan de cargar en memoria un objeto concreto, con su malla y su textura.

Además definen que ocurre si el personaje colisiona con el objeto en cuestión. De cada una de estas clases hay una sola instancia.

Diagrama de software del paquete Camiseta_Sensorial

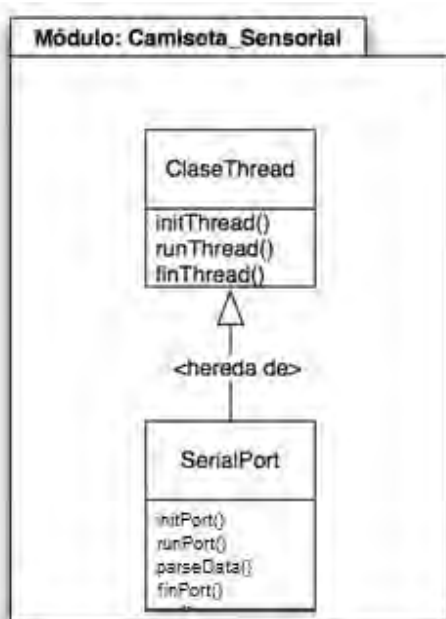


Figura 75 – Diagrama de clases del módulo Camiseta_Sensorial

- **ClaseThread:** Dispone de tres funciones. Una crea un nuevo hilo de ejecución, definiendo la velocidad a la que se tiene que ejecutar según el parámetro que recibe. Otra lo mantiene en funcionamiento y por último hay una función que termina con el hilo de ejecución.
- **SerialPort:** Esta clase se encarga de inicializar un puerto serie virtual por donde se reciben los datos de la camiseta sensorial y va consultando los datos que envía la camiseta. Hace estas consultas cada cierto tiempo mediante un nuevo thread que se ejecuta más lentamente que el hilo de ejecución principal. Con la función “parseData()” se filtra la información que se recibe de la camiseta y se guarda solamente la frecuencia cardíaca. La función “finPort()” permite cerrar el puerto serie y terminar con el thread.

3.2.3. Camiseta sensorial

Como se ha comentado en el análisis del diagrama de clases del módulo *Camiseta_Sensorial* (ver explicación de ese módulo en la sección anterior), se utilizan dos clases para recibir los datos de la camiseta sensorial, ClaseThread i SerialPort. La segunda hereda de la primera.

La camiseta envía un flujo de datos a través de bluetooth, estos datos són capturados por un puerto serie virtual creado en el PC (ver Figura 79), el cual es abierto desde la aplicación Irrlicht en C++ (ver Figura 78). El dispositivo envía información sobre la posición del cuerpo, la actividad física (frecuencia cardíaca) y el consumo de energía (calorías). Para Drappen's World se necesitaba obtener solamente la frecuencia cardíaca. Del flujo de información que es enviado constantemente, se filtra solo la frecuencia cardíaca mediante la función *parseData()* de la clase *SerialPort()*.

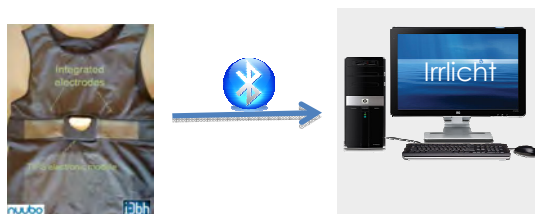


Figura 76 – Comunicación Camiseta Sensorial con Irrlicht

Parte práctica - Aplicación PC

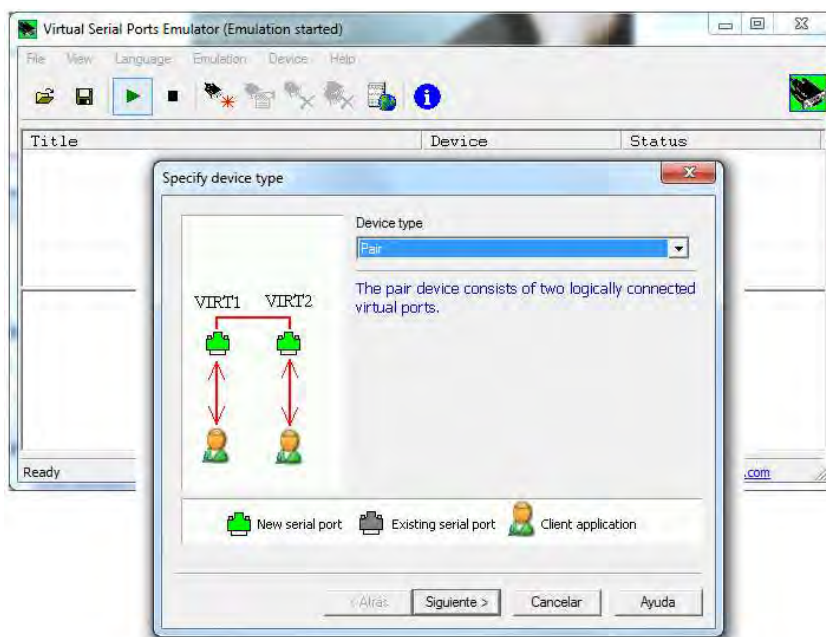


Figura 77 – VSPE: Programa que se utiliza para emular el puerto serie

A día de hoy, aun no se ha podido testear el serious game directamente con la camiseta sensorial y tratarla con el protocolo bluetooth, puesto que aún está bajo proceso de patentes por parte de las empresas desarrolladoras i3bh y nuubo. De momento se utiliza un simulado cedido por labhuman, laboratorio que colabora con el desarrollo de la camiseta. Este simulador es una aplicación software, ejecutada en el mismo ordenador que el serious game, que emula exactamente el comportamiento de la camiseta, enviando directamente al puerto serie virtual el flujo de datos que mandaría la camiseta.

A parte de la creación del nuevo thread para que haya un nuevo hilo de ejecución distinto al principal que funcione en paralelo y con una velocidad determinada (sincronizada con la camiseta), la parte más importante del código referente a la camiseta sensorial es la función *ParseData()*.

Parte práctica - Aplicación mini juego Smartphone

```
void CSerialPort::ParseData(char *szBuffer)
{
    for (int d = 0; d < 256; d++)
    {
        for (int c = 0; c < 25; c++)
        {
            szBuffer[c] = szBuffer[c + 1];
        }

        if (((szBuffer[1] & 0x0F) == 0x01) &&
            ((szBuffer[3] & 0x0F) == 0x02) &&
            ((szBuffer[5] & 0x0F) == 0x03) &&
            ((szBuffer[7] & 0x0F) == 0x04) &&
            ((szBuffer[9] & 0x0F) == 0x05) &&
            ((szBuffer[11] & 0x0F) == 0x06) &&
            ((szBuffer[13] & 0x0F) == 0x07) &&
            ((szBuffer[15] & 0x0F) == 0x08) &&
            ((szBuffer[17] & 0x0F) == 0x09))
        {
            mHealthRate = ((szBuffer[16] << 4) + (szBuffer[17] >> 4));
            cout << "SERIAL PORT: " << mHealthRate << endl;
        }
    }
}
```

Esta función se encarga de filtrar el flujo de datos enviado por la camiseta en bloques de 256 bits. En cada bloque hay información relevante a la camiseta y a sus datos capturados, de los cuales extraemos la frecuencia cardíaca. Este dato se guarda en una variable de clase para poder ser consultada desde el exterior de dicha clase. El código expuesto imprime el dato por pantalla para comprobar que todo funciona correctamente.

Probablemente que en un futuro no se tenga que programar a tan bajo nivel para tratar con este sensor y que se pueda programar a través de una API sin la necesidad de implementar threads, pero a día de hoy el sensor aún está en desarrollo y no dispone de ninguna API de programación.

3.3. Aplicación mini juego Smartphone

Parte práctica - Servidor: Base de datos

Esta aplicación, desarrollada en Android, es un mini juego más de Drappen's World.

Para superarse, el usuario debe recorrer una distancia en el exterior.

La aplicación necesita calcular la distancia recorrida por el jugador. Cuando el jugador ha llegado a recorrer una cierta distancia, la prueba ha sido superada (*ver Figura 80*) y se cambia un valor de la base de datos. Entonces, la aplicación para PC se percata del logro, lo contabiliza, y da los premios pertinentes al jugador, así como, la posibilidad de seguir con el siguiente minijuego.

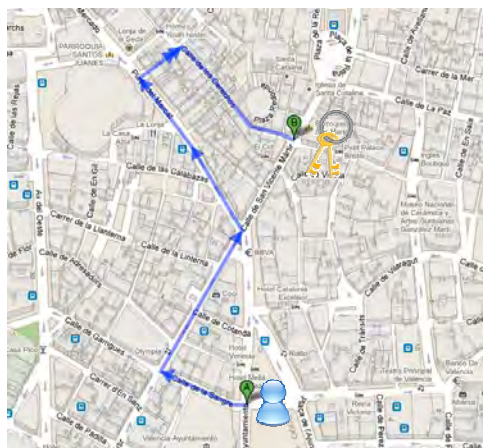


Figura 78 – Imagen de un posible recorrido GPS

Para calcular la distancia recorrida del jugador, se han utilizado principalmente las siguientes funciones que proporciona el API Android. Con ellas se sabe la posición GPS del jugador.

`getLatitude()` – Proporciona la latitud del terminal en ese instante.

`getLongitude()` – Proporciona la Longitud del terminal en ese instante.

Estas funciones se van llamando con una frecuencia de varios segundos. Y cada vez se llama también a la siguiente función:

`distanceBetween()` – Devuelve la distancia entre dos posiciones GPS.

Entonces vamos sumando la distancia entre pares de puntos hasta llegar a la distancia que exige el mini juego concreto. En ese momento, la aplicación cambia un valor de la base de datos que informa a la aplicación PC de que el juego ha terminado.

3.4. Servidor: Base de datos

La base de datos tiene una sola tabla llamada Drappen's World con los siguientes atributos:

Parte práctica - Servidor: Base de datos

Atributo	Tipo	Null
id (atributo clave)	Integer	Not null
nick	VarChar(10)	Not null
contrasena	VarChar(10)	Not null
nombre	VarChar(10)	Null
apellido1	VarChar(20)	Null
apellido2	VarChar(20)	Null
diaNacimiento	TinyInt	Null
mesNacimiento	TinyInt	Null
anyoNacimiento	TinyInt	Null
peso	SmallInt	Null
nivelDificultadTerapeuta	SmallInt	Null
juegoAndroidActivo	Bool	Null
distanciaArecorrer	SmallInt	Null
longitudGPSAndroid	Float	Null
latitudGPSAndroid	Float	Null
nivelJuego	TinyInt	Not Null
numMinijuego	TinyInt	Not Null
puntuacion	MediumInt	Not Null
items	SmallInt	Not Null

3.4.1. Conexión con la aplicación PC

La aplicación PC se conecta a la base de datos mediante el protocolo http. Directamente desde C++ no hay funciones implementadas para tal propósito, por lo que se utilizó la librería libcurl [60]. Libcurl es una librería para ser utilizada desde C++ que permite hacer conexiones mediante protocolo http.

Desde la aplicación, se hace una conexión mediante http al servidor Apache de la salle. Esta conexión invoca a un fichero php alojado en el servidor. Dependiendo de la operación que se quiera hacer sobre la base de datos de invoca a un php u otro. El técnico de theragames ha realizado varios php en el servidor apache, uno para cada posible operación a realizar sobre la base de datos.

El script php es el encargado de conectarse a la base de datos MySQL y hacerle las consultas, o modificaciones pertinentes (ver Figura 81).

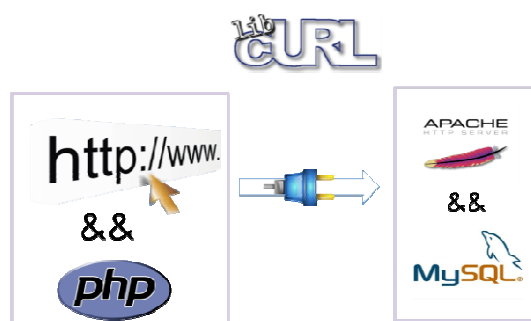


Figura 79 – Diagrama de conexión Aplicacion PC y BBDD

La aplicación PC necesita acceso a la base de datos para:

- Iniciar o restablecer el juego en el punto donde se dejó, mediante un login (con user y password).
- Consultar la información personal del jugador.
- Consultar las puntuaciones y logros de los demás jugadores para visualizar los rankings.

Parte práctica - Servidor: Base de datos

- Consultar en qué punto se había pausado o parado el juego para reanudarlo desde el mismo punto.
- Indicar al Smartphone que ya puede iniciarse su mini juego.
- Consultar cuando se ha completado el mini juego del Smartphone, es decir, consultar si el usuario ha completado la distancia a recorrer fuera de casa.
- Consultar el nivel de dificultad establecido por el terapeuta.
- Guardar el progreso de un jugador a la hora de cerrar el juego.

3.4.2. Conexión con la aplicación Android

Para conectarse a la base de datos desde la aplicación Android, no hace falta la utilización de ninguna librería externa como ocurre con la aplicación PC. Java ya dispone de clases implementadas para poder hacer conexiones http.

Entonces, de la misma manera que con la aplicación PC, se hace una conexión http al servidor Apache de La Salle y se invoca un php que es el encargado de interactuar con la base de datos MySQL (ver Figura 82).

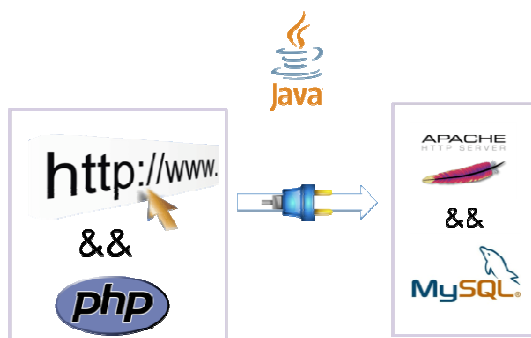


Figura 80 – Diagrama de conexión Android y BBDD

La aplicación Smartphone necesita acceso a la base de datos para:

- Consultar si ya puede iniciar su mini juego (la aplicación PC debe haber dado la orden, modificando el campo de la base de datos "juegoAndroidActivo" a true).

Resultados - Servidor: Base de datos

- Informar de que ya se ha completado el mini juego, habiéndose recorrido la distancia determinada.

4. Resultados

En la fecha de finalización de esta memoria, el proyecto encargado a Theragames aún no está finalizado, puesto que la fecha de entrega de esta memoria es un mes anterior a la fecha de entrega del proyecto acabado. Entonces, a falta de un mes para la finalización del proyecto, faltan cubrir ciertos aspectos.

A día de hoy el proyecto ya está muy avanzado, disponiendo de un mini juego realizado y funcionando completamente con Kinect. A partir de este mini juego será mucho más sencillo realizar los demás, puesto que es muy escalable y adaptable. El juego ya dispone de la secuencia de pantallas que se muestra en las siguientes imágenes. Ya hay varios usuarios registrados en la base de datos y cualquiera de ellos puede hacer login desde la aplicación siendo identificado al instante. Las cuestiones de la base de datos están resueltas y existe un prototipo funcional del aplicativo Smartphone. Este es capaz de calcular distancias recorridas, faltando aún depurar su precisión.



Figura 81 – Pantalla de bienvenida con el logo de la empresa.



Figura 82 – Pantalla de bienvenida con el logo del serious game.



Figura 83 – Pantalla prototipo para el login



Figura 84 – Prototipo de la pantalla del menú

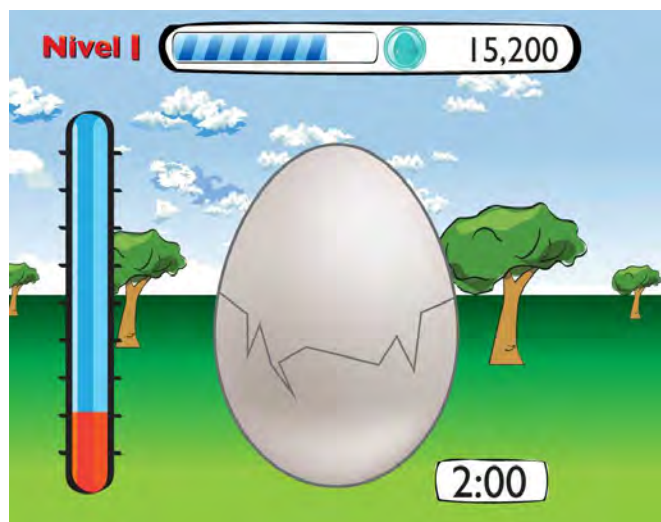


Figura 85 – Prototipo de diseño de una de las escenas de un minijuego.



Figura 86 – Prototipo para las Instrucciones de un minijuego en la pantalla de info.



Figura 87 – Prototipo para las Instrucciones de un minijuego en la pantalla de info.



Figura 88 – Prototipo para las Instrucciones de un minijuego en la pantalla de info.



Figura 89 – Prototipo para las instrucciones de calibración de Kinect.



Figura 90 – Prototipo de diseño de una de las escenas de un minijuego.

Resultados - Servidor: Base de datos



Figura 91 – Prototipo funcionando en irrliect, pantalla de mini juego.

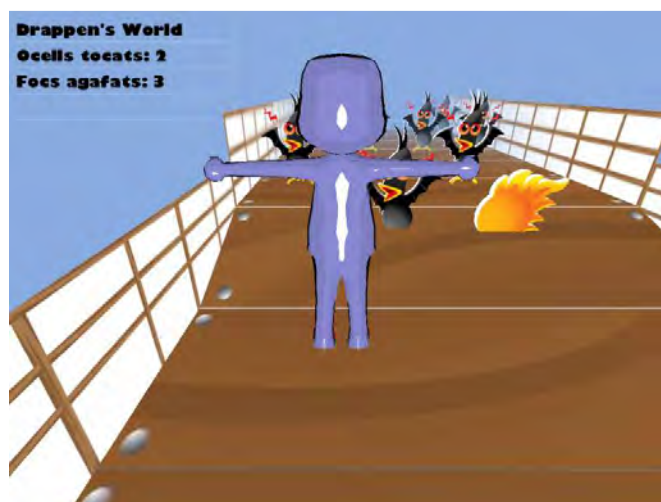


Figura 92 – Prototipo funcionando en irrliect, pantalla de mini juego.



Figura 93 – Prototipo de la pantalla de mni ranking posterior a un mini juego



Figura 94 – Prototipo de la pantalla de ranking posterior a la finalización de un nivel

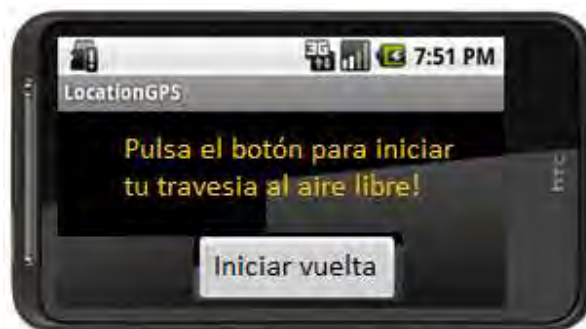


Figura 95 – Aplicación del Smartphone ejecutándose



Figura 96 – Simulador de la camiseta sensorial.

Resultados - Servidor: Base de datos

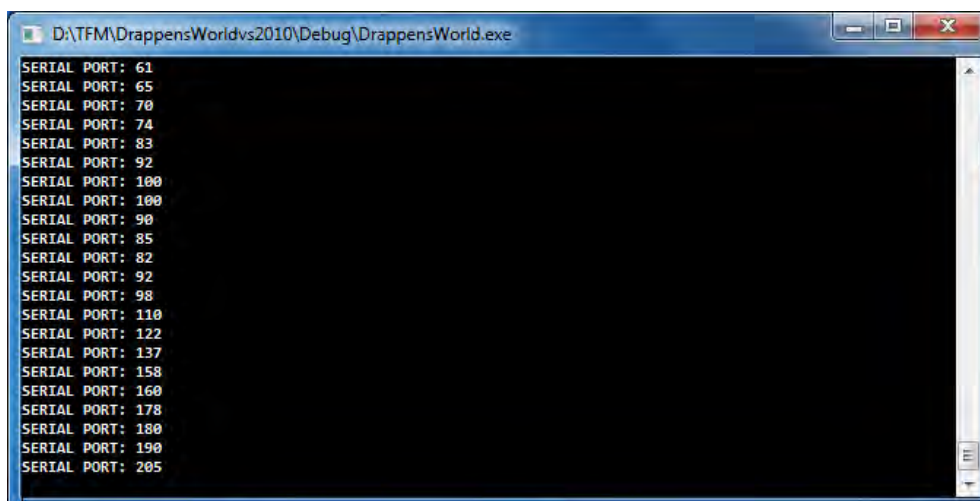


Figura 97 – Lectura de la frecuencia cardíaca del simulador desde Irrlich.

5. Conclusiones y líneas de futuro

Teniendo en cuenta que en el momento de este redactado falta un mes para la entrega del proyecto (puesto que la fecha de entrega de esta memoria es un mes anterior a la fecha de entrega del proyecto acabado), puede concluirse que el proyecto está muy avanzado con la mayoría objetivos cumplidos. El único objetivo que ha tenido que descartarse es la integración con Tuenti, debido a la imposibilidad de conseguir contactar con el personal de la red social.

A continuación se comentan las complicaciones más importantes que ha habido en el desarrollo de Drappen's World y por último las líneas de futuro.

Las principales complicaciones surgidas durante el desarrollo fueron las siguientes:

- La conexión entre la aplicación PC (realizada en C++) con la base de datos situada en un host externo al propio PC. Finalmente se solucionó con una conexión http la librería libcurl.
- Muchos esfuerzos para conseguir compilar la librería libcurl.
- Theragames había acordado con el personal técnico del laboratorio Labhuman, (que colaboran con LabPsiTec), que les proporcionaría ayuda con la complicada tarea de procesar los datos de Kinect. Al parecer hubo un mal entendido y el equipo tuvo que encargarse de todo ese trabajo.
- Muchos problemas para integrar la librería OpenNI para tratar con los datos de Kinect.
- Los problemas técnicos como el de libcurl o los de Kinect provocaron que el técnico – scrum master, Pau Moreno, tuviese que dedicar mucho más tiempo del previsto a aspectos técnicos; pudiendo dedicar menos tiempo a la gestión.
- Se tuvo que dedicar mucho más tiempo del esperado para tratar con los datos de la camiseta sensorial, al tener que crear un nuevo thread y adaptar una rutina

Conclusiones y líneas de futuro - Servidor: Base de datos

(proporcionada por Labhuman) del lenguaje C# a C++, proceso que llevó mucho esfuerzo y tiempo.

- Fue imposible contactar con el personal de Tuenti para que nos proporcionaran algún tipo de API para poder integrar Drappen's World con la red social.

Las líneas de futuro del proyecto, en caso de disponer de tiempo y capital para seguir con él, son las siguientes:

- Desarrollo de más mini juegos
- Desarrollo de una interfaz web amigable para el terapeuta con más opciones configurables.
- Desarrollo de un mini juego muy elaborado para Smartphone que siga cumpliendo la necesidad básica de calcular la distancia recorrida.
- Mejorar el diseño gráfico de todo el juego.
- Mejorar el entorno virtual de juego.
- Hacer toda la estética del juego en tres dimensiones (actualmente en mayor parte, pero no todo).

6. Índice de figuras

Figura 1 – Pantalla de resultados Pygpslog	10
Figura 2 – Vista de navegación Google Maps Navigation	11
Figura 3 – Interfaz de navegación Copilot.....	12
Figura 4 – Diferentes Pantallas de Runstar.....	13
Figura 5 – Pantalla de inicio de Cardio Trainer	14
Figura 6 – Pestaña de perfil y pestaña de recomendaciones de Foursquare	15
Figura 7 – Vista del mapa de SCVNGR	16
Figura 8 – Vista del recorrido realizado y pantalla de información de MotionX GPS	17
Figura 9 – Información proporcionada por Mobile Tracker	18
Figura 10 – Mapa de ruta y pestañas de opciones de Waze.....	19
Figura 11 – Imagen del periférico Wii Vitality Sensor	21
Figura 12 – Pulsímetro, pantalla de juego y usuario del EA Sports Active 2.....	22
Figura 13 – Componentes de un motor de juego	30
Figura 14 – Grafo de escena	31
Figura 15 – Grafo de escena con BV	31
Figura 16 – Capturas de pantalla de Gears Of War 2	36
Figura 17 – Tres pantallas de Bioshock	36
Figura 16 – Escenas varias del juego Far Cry 2.....	39
Figura 18 – Capturas de pantalla del juego Crysis	39
Figura 19 – Imágenes de Assassin's Creed.....	41
Figura 20 – Capturas del videojuego Prince of Persia.....	41
Figura 21 – Imágenes de Red Faction: Guerrilla.....	43
Figura 22 – Escenas del juego Red Faction: Armageddon.....	43
Figura 23 – Capturas de pantalla de H-Craft Championship	45
Figura 24 – Varias pantallas de Bloodgiver	45
Figura 25 – Varias vistas de Bolzplatz2006.....	48
Figura 26 – Capturas de pantalla de The Age of Racing.....	48
Figura 27 – Imágenes de Urban Galaxy.....	50
Figura 28 – Varias pantallas de Mythruna	50
Figura 29 – Distintas pantallas de Disney's Pirates of the Caribbean Online.....	52

Índice de figuras - Servidor: Base de datos

Figura 30 – Pantallas de Angels Fall First: the Second Antarean War	52
Figura 31 – Usuario con la camiseta sensorial y reverso de la camiseta.....	57
Figura 32 – Jugadores de Kinect en acción	59
Figura 33 – Ventas de WiiMote, Playstation Move y Kinect.....	60
Figura 34 – Usuario jugando con interacción totalmente natural	61
Figura 35 – Periférico Kinect.....	62
Figura 36 – Tracking de un jugador realizado por Kinect	62
Figura 37 – Distancia mínima operacional para Kinect	62
Figura 38 – Transformación en escala de grises.....	63
Figura 39 – Transformación de la escena al mapa de profundidad de Kinect	63
Figura 40 – Funcionamiento del periférico Kinect.....	64
Figura 41 – Diagrama tecnológico de la arquitectura de Kinect.....	65
Figura 42 – Hardware común en los Smartphones	66
Figura 43 – Interacción de un SO	66
Figura 44 – Cuotas globales de mercado de SmartPhones por marca de SO.....	68
Figura 45 – Progresión de cuotas de mercado de SmartPhones por SO en EEUU.....	69
Figura 46 – Cuotas globales de mercado de SmartPhones por SO	69
Figura 47 – Plataformas móviles más utilizadas por los desarrolladores.....	70
Figura 48 – Aplicaciones disponibles para los distintos SO de Smartphone en EEUU ...	71
Figura 49 – Terminales android de gran rango de precios	72
Figura 50 – Razones más consideradas por los desarrolladores al elegir un SO movil..	72
Figura 51 – Porcentaje de distribución de las versiones de Android	74
Figura 52 – Diagrama de Arquitectura Android	77
Figura 53 – Árbol de nodos view y viewgroups	79
Figura 54 – Árbol de LayoutParams de un árbol de nodos, view y viewgroups.	80
Figura 55 – Logo del entorno de desarrollo y la SDK	81
Figura 56 – Estadísticas de los servidores web más utilizados.	84
Figura 57 – Componentes que forman Wamp.....	88
Figura 58 – Sistemas Operativos y tecnologías XAMP	89
Figura 59 – Diagrama de Arquitectura Drappen's World	92
Figura 60 – Diagrama de transición de estados general	94
Figura 61 – Pantalla de bienvenida con el logo de la empresa.....	95
Figura 62 – Pantalla de bienvenida con el logo del serious game.....	95

Índice de figuras - Servidor: Base de datos

Figura 63 – Pantalla prototipo para el login	95
Figura 64 – Prototipo de la pantalla del menú	96
Figura 65 – Prototipo de una de las escenas de un minijuego.	96
Figura 66 – Prototipo para las Instrucciones de un minijuego en la pantalla de info.....	99
Figura 67 – Prototipo para las instrucciones de calibración de Kinect.	99
Figura 68 – Prototipo de una de las escenas de un minijuego.	100
Figura 69 – Prototipo de la pantalla de mni ranking posterior a un mini juego.....	100
Figura 70 – Prototipo de la pantalla de ranking posterior a la finalización de un nivel ..	101
Figura 71 – Prototipo para las Instrucciones de un minijuego en la pantalla de info..	Error!
Bookmark not defined.	
Figura 72 – Prototipo para las Instrucciones de un minijuego en la pantalla de info..	Error!
Bookmark not defined.	
Figura 73 – Diagrama de módulos de software general	103
Figura 74 – Diagrama de clases del módulo Gestor_Texturas	105
Figura 75 – Diagrama de clases del módulo Conexión_Web	106
Figura 76 – Diagrama de clases del módulo Gestor_Objetos.....	108
Figura 77 – Diagrama de clases del módulo Camiseta_Sensorial.....	109
Figura 78 – Comunicación Camiseta Sensorial con Irrlicht	110
Figura 80 – Imagen de un posible recorrido GPS.....	113
Figura 81 – Diagrama de conexión Aplicacion PC y BBDD	115
Figura 82 – Diagrama de conexión Android y BBDD.....	116
Figura 83 – Pantalla de bienvenida con el logo de la empresa.....	118
Figura 84 – Pantalla de bienvenida con el logo del serious game.	118
Figura 85 – Pantalla prototipo para el login	119
Figura 86 – Prototipo de la pantalla del menú	119
Figura 87 – Prototipo de diseño de una de las escenas de un minijuego.	120
Figura 88 – Prototipo para las Instrucciones de un minijuego en la pantalla de info.....	120
Figura 89 – Prototipo para las Instrucciones de un minijuego en la pantalla de info.....	121
Figura 90 – Prototipo para las Instrucciones de un minijuego en la pantalla de info.....	121
Figura 91 – Prototipo para las instrucciones de calibración de Kinect.	122
Figura 92 – Prototipo de diseño de una de las escenas de un minijuego.	122
Figura 93 – Prototipo funcionando en irrlicht, pantalla de mini juego.....	123
Figura 94 – Prototipo funcionando en irrlicht, pantalla de mini juego.....	123

Figura 95 – Prototipo de la pantalla de mni ranking posterior a un mini juego.....	124
Figura 96 – Prototipo de la pantalla de ranking posterior a la finalización de un nivel ..	124
Figura 97 – Aplicación del Smartphone ejecutándose.....	125

Bibliografía

- 1 pygpslog - Navegador e interprete de señales GPS gratuito para moviles Symbian. [Online]. 2009 [Consultado el 27 Enero 2011]. Disponible en: <http://aplicaciones-symbian.com/2009/01/20/pygpslog-navegador-e-interprete-de-senales-gps-gratuito-para-moviles-symbian/>.
- 2 Google. Google Maps Navigation. [Online]. [Consultado el 30 Mar 2011]. Disponible en: http://www.google.es/intl/es_ALL/mobile/navigation/#utm_campaign=es_es&utm_source=gh17smm&utm_medium=ha&utm_term=mapas%20GPS&dc=gh17smm.
- 3 Aplicaciones GPS para Android. [Online]. [Consultado el 10 Gener 2011]. Disponible en: <http://www.celularis.com/software/aplicaciones-gps-android.php>.
- 4 Runstar.se, aplicación Android para hacer ejercicio. [Online]. [Consultado el 02 Febrero 2011]. Disponible en: <http://www.webadictos.com.mx/2011/01/05/runstar-aplicacion-android-para-hacer-ejercicio/>.
- 5 Runstar. [Online]. [Consultado el 30 Marzo 2011]. Disponible en: <http://runstar.se/>.
- 6 Cardio trainer, un programa que nos ayuda a ejercitarnos con android. [Online]. 2010 [Consultado el 04 Marzo 2011]. Disponible en: <http://www.vitonica.com/wellness/cardio-trainer-un-programa-que-nos-ayuda-a-ejercitarnos-con-android>.
- 7 Acerca de Foursquare. [Online]. [Consultado el 07 Enero 2011]. Disponible en: <http://es.foursquare.com/about>.
- 8 SCVNGR: Android y iPhone app de geolocalización innovadora. [Online]. 2010 [Consultado el 03 Febrero 2011]. Disponible en: <http://www.celularis.com/software/scvngr-android-iphone-app.php>.

Bibliografía - Servidor: Base de datos

- 9 MotionX GPS, un GPS imprescindible para deportistas. [Online]. 2009 [Consultado el 30 Marzo 2011]. Disponible en: <http://www.applesfera.com/aplicaciones-moviles/motionx-gps-un-gps-imprescindible-para-deportistas>.
- 10 MobileTracker para Blackberry. [Online]. 2010 [Consultado el 30 Marzo 2011]. Disponible en: <http://zonablackberry.com.ve/forum/aplicaciones-originales-y-de-terceros/57290-mobiletracker-para-blackberry-v1-5-1-por-skylab-mobilesystems.html>.
- 11 Waze GPS & Tráfico - Social, Diversión! [Online]. 2011 [Consultado el 30 Marzo 2011]. Disponible en: <http://itunes.apple.com/es/app/id323229106?mt=8>.
- 12 El Wii Vitality Sensor podría usarse en un juego llamado Wii Relax. [Online]. 2010 [Consultado el 20 Enero 2011]. Disponible en: <http://news.softpedia.com/es/El-Wii-Vitality-Sensor-podria-usarse-en-un-juego-llamado-Wii-Relax-160339.html>.
- 13 Wii Vitality Sensor, cómo funciona y para qué servirá. [Online]. 2009 [Consultado el 30 Enero 2011]. Disponible en: <http://www.xataka.com/consolas-y-videojuegos/wii-vitality-sensor-como-funciona-y-para-que-servira>.
- 14 Wii Vitality Sensor sigue vivo. [Online]. 2010 [Consultado el 30 Enero 2011]. Disponible en: <http://dswii.es/29408/wii-vitality-sensor-sigue-vivo/>.
- 15 Brugué XC. EA Sports Active 2. [Online]. 2010 [Consultado el 29 Enero 2011]. Disponible en: http://www.meristation.com/v3/des_analisis.php?pic=360&id=cw4ce2d760a2357&id=cw4c16892e9a5b0&idp=&tipo=art&c=1&pos=0.
- 16 Motor de juego. [Online]. [Consultado el 25 Marzo 2011]. Disponible en: http://es.wikipedia.org/wiki/Motor_de_videojuego.
- 17 Ramírez FAM. Herramientas de Programación Gráfica para. Universidad Autónoma de Yucatán, Facultad de Matemáticas.
- 18 García T. Top 10: Los motores gráficos más importantes. [Online]. 2009 [Consultado el 1 Febrer 2011]. Disponible en: <http://www.neoteo.com/top-10-los-motores-graficos-mas-importantes>.
- 19 Unreal Engine 3. [Online]. 2006 [Consultado el 25 Marzo 2011]. Disponible en: <http://www.moddb.com/engines/unreal-engine-3>.
- 20 Unreal Engine. [Online]. [Consultado el 05 Febrero 2011]. Disponible en: http://es.wikipedia.org/wiki/UNREAL_ENGINE_3#Unreal_Engine_3.

Bibliografía - Servidor: Base de datos

- 21 CRYTEK. CryENGINE 2. [Online]. 2009 [Consultado el 9 f febrero]. Disponible en:
<http://crytek.com/cryengine/cryengine2/overview>.
- 22 Anvil (motor de videojuegos). [Online]. [Consultado el 17 Abril 2011]. Disponible en:
http://es.wikipedia.org/wiki/Anvil_%28motor_de_videojuegos%29.
- 23 Zepeda JO. Anvil Engine motor grafico. [Online]. [Consultado el 19 Marzo 2011].
Disponible en: <http://javierorlandozepeda.wordpress.com/anvil-engine-motor-grafico/>.
- 24 Motor 1.7.1 de Irrlicht. [Online]. [Consultado el 20 Abril 2011]. Disponible en:
<http://mscerts.programming4.us/es/691882.aspx>.
- 25 Características Ogre. [Online]. 2008 [Consultado el 1 Marzo 2011]. Disponible en:
<https://sites.google.com/site/ogreesp/caracteristicas>.
- 26 JMonkey Engine. [Online]. [Consultado el 2 Marzo 2011]. Disponible en:
http://es.wikipedia.org/wiki/JMonkey_Engine.
- 27 Grupo LosersJuegos. LosersJuegos. [Online]. [Consultado el 07 Abril 2011].
Disponible en: http://www.losersjuegos.com.ar/traduccion/que_es_panda3d.
- 28 Panda3D Features. [Online]. 2009 [Consultado el 05 Abril 2011]. Disponible en:
<http://www.panda3d.org/features.php>.
- 29 C++. [Online]. [Consultado el 12 Junio 2011]. Disponible en:
<http://es.wikipedia.org/wiki/C%2B%2B>.
- 30 Introducción C++. [Online]. [Consultado el 10 Mayo 2011]. Disponible en:
<https://belenus.unirioja.es/~crequia/introduccion.html>.
- 31 Visual Studio. [Online]. [Consultado el 13 Junio 2011]. Disponible en:
http://es.wikipedia.org/wiki/Visual_Studio.
- 32 Jaime Guixeres, Irene Zaragoza, Mariano Alcañiz, Javier Saiz. TIPS-shirt: Intelligent Textile Sensor for Obese Children E-Therapy. Valencia.: InterUniversity Research Institute for Bioengineering and Human Centered Technology I3BH y Universidad Politécnica de Valencia; 2011.
- 33 Kinect. [Online]. [Consultado el 1 Junio 2011]. Disponible en:
<http://es.wikipedia.org/wiki/Kinect>.
- 34 Move vs. Kinect. [Online]. 2010 [Consultado el 14 Mayo 2011]. Disponible en:
<http://www.elmundo.es/blogs/elmundo/el-gadgetoblog/2010/06/18/move-vs->

Bibliografía - Servidor: Base de datos

[kinect.html](#).

- 35 Vilchez A. Kinect vende 4 millones de unidades, gana a Wii y PlayStation Move. [Online]. 2011 [Consultado el 14 Junio 2011]. Disponible en: <http://www.configurarequipo.com/actualidad-informatica/3120/kinect-vende-4-millones-de-unidades-gana-a-wii-y-playstation-move>.
- 36 Walton B. Kinect Sales Reach 4 Million Units. [Online]. 2006 [Consultado el 20 Mayo 2011]. Disponible en: <http://www.vgchartz.com/article/83079/kinect-sales-reach-4-million-units-worldwide/>.
- 37 Rotella M. Kinect, la nueva videoconsola desarrollada por Microsoft. [Online]. [Consultado el 14 Mayo 2011]. Disponible en: <http://tecnogroup11.blogspot.com/2011/02/kinect-la-nueva-videoconsola.html>.
- 38 Carmody T. Funcionamiento del sensor de movimiento Kinect. [Online]. 2010 [Consultado el 20 Febrer 2011]. Disponible en: <http://www.ideasgeek.net/2010/11/10/funcionamiento-del-sensor-de-movimiento-en-kinect/>.
- 39 ¿Qué es Kinect? [Online]. [Consultado el 29 Mayo 2011]. Disponible en: <http://www.todokinect.com/que-es-kinect>.
- 40 Smartphone. [Online]. [Consultado el 10 Junio 2011]. Disponible en: <http://es.wikipedia.org/wiki/Smartphone>.
- 41 Android. [Online]. [Consultado el 10 Marzo 2011]. Disponible en: <http://es.wikipedia.org/wiki/Android>.
- 42 Android-SPA Team & PHPbb. Qué es Android. [Online]. [Consultado el 1 Enero 2011]. Disponible en: <http://www.android-spa.com/infoAndroid.php?tag=queEsAndroidParaUnSimpleUsuario>.
- 43 Android Market. [Online]. [Consultado el 12 Junio 2011]. Disponible en: http://es.wikipedia.org/wiki/Android_Market.
- 44 Mobility S. El mercado de las aplicaciones móviles. [Online]. 2011 [Consultado el 28 Mayo 2011]. Disponible en: http://www.slideshare.net/slashmobility/el-mercado-de-las-aplicaciones-moviles-7034222?from=ss_embed.
- 45 Android Developers Community. Resources - Platform Versions. [Online]. 2011 [Consultado el 28 Mayo 2011]. Disponible en: <http://developer.android.com/resources/dashboard/platform-versions.html>.

Bibliografía - Servidor: Base de datos

- 46 Android: An Open Handset Alliance Project. [Online]. [Consultado el 20 Junio 2011]. Disponible en: <http://esistemasoperativos.mforos.com/1834161/10189495-android-an-open-handset-alliance-project/>.
- 47 Qué es Android (técnico). [Online]. [Consultado el 20 Juny 2011]. Disponible en: <http://www.android-spa.com/infoAndroid.php?tag=queEsAndroid>.
- 48 Eclipse. [Online]. [Consultado el 12 Junio 2011]. Disponible en: http://es.wikipedia.org/wiki/Eclipse_%28software%29#Caracter.C3.ADsticas.
- 49 Catdroid. Herramientas de Catdroid. [Online]. 2011 [Consultado el 03 Febrero 2011]. Disponible en: <http://catdroid.org/blog/category/herramientas-catdroid/>.
- 50 Servidor Web. [Online]. [Consultado el 13 Junio 2011]. Disponible en: http://es.wikipedia.org/wiki/Servidor_web.
- 51 PINO JLL. Servidores web más usados. [Online]. 2010 [Consultado el 28 Mayo 2011]. Disponible en: <http://lopezpino.es/2010/07/30/servidores-web-mas-usados/>.
- 52 Grupo Desconectate. Los servidores apache ya acaparan un 60 por ciento de la web. [Online]. 2011 [Consultado el 15 Mayo 2011]. Disponible en: <http://www.desconectate.info/208/Los%20servidores%20apache%20ya%20acaparan%20un%2056%20por%20ciento%20de%20la%20web.html>.
- 53 Apache Software Foundation. Visión general de las nuevas funcionalidades de Apache 2.0. [Online]. [Consultado el 09 May 2011]. Disponible en: http://www.apache2.es/2.0.58/new_features_2_0.html.
- 54 MySQL. [Online]. [Consultado el 13 Junio 2011]. Disponible en: <http://es.wikipedia.org/wiki/MySQL>.
- 55 Pecos D. PostGreSQL vs. MySQL. [Online]. [Consultado el 6 Junio 2011]. Disponible en: http://danielpecos.com/docs/mysql_postgres/index.html.
- 56 phpMyAdmin. [Online]. [Consultado el 13 Junio 2011]. Disponible en: <http://es.wikipedia.org/wiki/PhpMyAdmin>.
- 57 SourceForge. Project of the Month, December 2002. [Online]. 2002 [Consultado el 10 Junio 2011]. Disponible en: <http://sourceforge.net/potm/potm-2002-12.php>.
- 58 WAMP. [Online]. [Consultado el 10 Junio 2011]. Disponible en: <http://es.wikipedia.org/wiki/WAMP>.

Bibliografía - Servidor: Base de datos

59 XAMPP. [Online]. [Consultado el 13 Junio 2011]. Disponible en:
<http://es.wikipedia.org/wiki/XAMPP>.

60 about libcurl. [Online]. [Consultado el 1 Marzo 2011]. Disponible en: <http://libcurl.org/>.

61 iPhone Developer Labs. Posts Tagged 'iOS vs Android'. [Online]. 2011 [Consultado el 28 Mayo 2011]. Disponible en: <http://www.iphonedevlperlabs.com/tag/ios-vs-android/>.