

ANTEPROYECTO DEL TRABAJO DE FIN DE GRADO

INFORMACIÓN GENERAL

Alumno/a	Francisco Morillas Jiménez			
Titulación:	Grado en Ingeniería del Software			
Tutor/es:	José Carlos Canal Velasco Alejandro Pérez Vereda			
Título	Procesamiento de eventos distribuidos para aplicaciones de computación social móvil			
Subtítulo <i>(solo si en grupo)</i>				
Título en inglés	Distributed Event Processing for Mobile Social Computing Applications			
Subtítulo en inglés <i>(solo si en grupo)</i>				
Trabajo en grupo:	<input checked="" type="checkbox"/> Sí	<input type="checkbox"/>	<input checked="" type="checkbox"/> No	<input type="checkbox"/>
Otros integrantes del grupo:				

INTRODUCCIÓN

Contextualización del problema a resolver. Describir claramente de dónde surge la necesidad de este TFG y el dominio de aplicación. En caso de que el TFG se base en trabajos previos, debe aclararse cuáles son las aportaciones del TFG.

El desarrollo de los *smartphones* en los últimos años ha desencadenado una auténtica revolución en nuestra forma de vida. Existen diversos motivos por los que la computación móvil ha adquirido tanta importancia. La alta disponibilidad que ofrecen estos dispositivos, que podemos utilizar en cualquier momento y lugar, y el importante crecimiento en sus capacidades de cómputo junto con el desarrollo de nuevos modelos de tecnología de redes de telecomunicaciones con mayores velocidades de descarga, como el 5G en la actualidad, son algunos de ellos. Todo esto ha propiciado que el mercado del software se haya concentrado principalmente en las aplicaciones móviles.

Esa revolución tecnológica ha supuesto un auge de las aplicaciones sociales y colaborativas. Estas aplicaciones hacen uso de las capacidades técnicas de los dispositivos para procesar cierta información. Sin embargo, en la gran mayoría de los casos, las aplicaciones envían los datos que son generados a un servidor para su procesamiento, con el objetivo de prestar sus servicios. A pesar de los grandes avances en la capacidad de cómputo, memoria y almacenamiento de los *smartphones*, continúa siendo imperante el tradicional modelo cliente-servidor. Una popularización de una alternativa a dicho modelo permitiría reducir la dependencia en la actualidad de muchos desarrolladores de la contratación de servidores dedicados.

Con este trabajo, se pretende defender el uso de una arquitectura descentralizada *peer to peer* para aplicaciones que tengan como principal característica la interacción entre usuarios, de forma que los datos que se vayan recopilando en el *smartphone* sean procesados en ellos. Para ello, el proyecto a desarrollar utiliza la arquitectura de *Digital Avatars* [1], la cual consiste en un framework que permite mantener un perfil virtual de un usuario determinado, con datos sobre él mismo que se actualizan a través de un sistema de procesamiento de eventos. Estos eventos pueden ser disparados por un cambio detectado por los sensores del dispositivo. La principal utilidad del framework reside en la capacidad que tiene cada avatar digital de interactuar con otros, y regular su respuesta ante eventos que procedan del propio dispositivo o de otros avatares digitales mediante reglas de comportamiento. De esta manera, cada avatar digital es capaz de reaccionar de forma distinta ante un evento que proceda de fuera del dispositivo, permitiendo crear una experiencia más adaptada a las preferencias de cada usuario.

Para ilustrar las posibilidades que ofrece esta tecnología, se pretende desarrollar una aplicación móvil distribuida que consiste en un juego llamado *Catch the Thief*. En él, los usuarios pueden tener dos roles distintos: los policías deberán arrestar a los ladrones y los ladrones tendrán el objetivo de robar tantos tesoros como puedan. Este juego hará uso de la tecnología de *Digital Avatars* para la compartición de, entre otros datos, la localización de los usuarios.

OBJETIVOS

Descripción detallada de en qué consistirá el TFG. En caso de que el objeto principal del TFG sea el desarrollo de software, además de los objetivos generales deben describirse sus funcionalidades a alto nivel.

El objetivo principal del trabajo a realizar consiste en el desarrollo de una aplicación móvil para dispositivos Android, la cual hará uso de la arquitectura *Digital Avatars*. Este software será un importante mecanismo de divulgación del modelo, demostrando que sus capacidades técnicas ofrecen una interacción óptima entre dispositivos, así como la posibilidad de una gestión de almacenamiento de información de los usuarios descentralizada.

La aplicación móvil consiste en un juego que hará uso de los datos de localización de cada dispositivo. En él, los usuarios podrán crear una partida o unirse a otra que esté en espera. El juego deberá llevarse a cabo en un espacio de forma que los jugadores no se encuentren a una distancia en la que no puedan interactuar entre ellos caminando.

Cuando los usuarios se encuentren en una partida, tendrán asignados desde el principio un rol de forma aleatoria o elegido por el propio usuario siempre que exista una división equitativa entre equipos. Los roles son dos: ladrón o policía. El usuario que sea ladrón tendrá el objetivo de buscar tesoros con el objetivo de “robarlos”. Los tesoros consisten en dispositivos *beacon* [2] que estarán identificados, que emiten una señal unívoca de Bluetooth para notificar su posición a los *smartphones* de los usuarios. Los *beacons* podrán ser físicos o virtuales, para facilitar el uso de la aplicación. Un ladrón podrá robar un tesoro si se encuentra cerca de uno y no hay un policía en una posición próxima a la suya. Por el contrario, un policía tendrá el objetivo de arrestar junto a sus compañeros a todos los ladrones. Será capaz de arrestar a un ladrón si su posición es cercana a la suya y hay otro policía cerca. El juego terminará cuando todos los tesoros sean robados, con victoria de los ladrones, o bien cuando se haya arrestado a todos los ladrones, con victoria de los policías.

La aplicación almacenará en cada avatar digital datos relacionados con el historial del usuario. Esta información permitirá que el usuario obtenga una retroalimentación de su desempeño en el juego. Algunos datos que estarán vinculados a su perfil incluyen el número de partidas ganadas al jugar con cada rol, entre otros. A su vez, los usuarios podrán agregar a otros usuarios a su lista de amigos por medio del envío de solicitudes de amistad, de manera que puedan saber si su amigo se encuentra en línea para poder jugar.

Todas las situaciones del juego serán implementadas mediante el uso de eventos que serán creados periódicamente según los datos que generen los sensores de los dispositivos. Estos eventos se transmitirán entre dispositivos generando a su vez otros eventos. El procesamiento de los eventos se realizará con el motor *Siddhi* [3], que permite procesar flujos de datos en tiempo real. El mecanismo de comunicación entre dispositivos consistirá en el envío de notificaciones *push*, esto no requerirá el uso de ningún servidor.

ENTREGABLES

Listado de resultados que generará el TFG (aplicaciones, estudios, manuales, etc.)

Aplicación móvil para Android

Guía de Instalación

Manual de Usuario

Memoria del TFG

MÉTODOS Y FASES DE TRABAJO**METODOLOGÍA:**

Descripción de la metodología empleada en el desarrollo del TFG. Especificar cómo se va a desarrollar. Concretar si se trata de alguna metodología existente y, en caso contrario, describir y justificar adecuadamente los métodos que se aplicarán.

Se empleará una metodología de desarrollo ágil [4] para el proyecto de software. De esta forma, el proceso de desarrollo será iterativo, en el que por cada iteración se realizará una reunión con los tutores para que puedan observar el progreso realizado en el desarrollo mediante los requisitos que se vayan implementando, dando su opinión. Así no solo habrá una comunicación más fluida entre las partes, la capacidad de respuesta a posibles cambios que aparezcan durante el desarrollo también será mayor.

FASES DE TRABAJO:

Enumeración y breve descripción de las fases de trabajo en las que consistirá el TFG.

1. Estudio:

Se realizará una tarea de investigación y aprendizaje de las tecnologías a utilizar.

2. Análisis de requisitos:

Esta fase consistirá en la especificación de los requisitos funcionales y no funcionales del software.

3. Diseño:

Esta fase está dividida en 5 subfases:

- Modelado de la arquitectura
- Creación del modelo del dominio
- Elaboración de bocetos de interfaz de usuario
- Modelado de la base de datos
- Especificación de los casos de uso

4. Implementación

En esta fase se elaborará el código fuente, se realizará la integración del framework de *Digital Avatars* en la aplicación, se creará la base de datos y se pondrá en marcha el sistema de envío de notificaciones *push* para la comunicación.

5. Verificación y pruebas

Se realizarán las pruebas pertinentes para asegurar el correcto funcionamiento del software desarrollado.

6. Documentación

Se dividirá en 3 subfases

- Elaboración de la guía de instalación
- Elaboración del manual de usuario
- Redacción de la memoria del TFG

TEMPORIZACIÓN:

La siguiente tabla deberá contener una fila por cada una de las fases enumeradas en la sección anterior. En caso de tratarse de un trabajo en grupo, se añadirá una columna HORAS por cada miembro del equipo. Debe especificarse claramente el número de horas dedicado por cada alumno/a y la suma de horas individual deberá ser también de 296.



FASE	HORAS
	<i>Francisco Morillas Jiménez</i>
1. Estudio	13
2. Análisis de requisitos	7
3. Diseño	43
a. Modelado de la arquitectura	4
b. Creación del modelo del dominio	8
c. Elaboración de bocetos de interfaz de usuario	7
d. Modelado de la base de datos	4
e. Especificación de los casos de uso	20
4. Implementación	120
5. Verificación y pruebas	22
6. Documentación	91
a. Elaboración de la guía de instalación	5
b. Elaboración del manual de usuario	20
c. Redacción de la memoria del TFG	66
	296

ENTORNO TECNOLÓGICO**TECNOLOGÍAS EMPLEADAS:**

Enumeración de las tecnologías utilizadas (lenguajes de programación, frameworks, sistemas gestores de bases de datos, etc.) en el desarrollo del TFG.

Java como lenguaje de programación

Base de Datos no relacional (documentos JSON): *Couchbase Lite* [5]

Siddhi como motor de procesamiento de eventos (CEP) [3]

Firebase Cloud Messaging [6] para el envío de notificaciones *push*

RECURSOS SOFTWARE Y HARDWARE:

Listado de dispositivos (placas de desarrollo, microcontroladores, procesadores, sensores, robots, etc.) o software (IDE, editores, etc.) empleados en el desarrollo del TFG.

IDE: *Android Studio*

MagicDraw para el modelado del software

Uso de *smartphone* con Bluetooth y GPS

Ordenador personal para el desarrollo del software y la redacción de la documentación

Beacons [2]

REFERENCIAS

Listado de referencias (libros, páginas web, etc.)

[1] Pérez-Vereda, A., Canal, C., & Pimentel, E. (2020). A Formal Programming Framework for Digital Avatars. *Software Engineering and Formal Methods*, 236–251. https://doi.org/10.1007/978-3-030-57506-9_18

[2] Akpinar, E. (2021, 14 julio). Bluetooth beacons: Everything you need to know. Pointr. Recuperado 1 de junio de 2022, de <https://www.pointr.tech/blog/beacons-everything-you-need-to-know>

[3] Siddhi. Siddhi. Recuperado 1 de junio de 2022, de <https://siddhi.io/>

[4] Brush, K., & Silverthorne, V. (2022, 10 enero). Desarrollo de software ágil o Agile. *ComputerWeekly.es*. Recuperado 1 de junio de 2022, de <https://www.computerweekly.com/es/definicion/Desarrollo-de-software-agil-o-Agile>

[5] Developer Portal | Couchbase. (s. f.). Couchbase. Recuperado 1 de junio de 2022, de <https://developer.couchbase.com/>

[6] Firebase Cloud Messaging | Firebase Documentation. (s. f.). Firebase. Recuperado 1 de junio de 2022, de <https://firebase.google.com/docs/cloud-messaging>

Málaga, 3 de junio de 2022

Firma tutor/tutora:

Firma cotutor/a:

Firma tutor/a coordinador/a: