



Patricia Santana Cruz

Escuela de Ingeniería Informática
Universidad de Las Palmas de Gran Canaria

Escuela de Ingeniería Informática. Universidad de Las Palmas de Gran Canaria.

Proyecto fin de carrera

Título: I-Found

Alumna: Patricia Santana Cruz

Fecha: Mayo de 2014

Tutor:

José Miguel Santos Espino

Profesor del Departamento de Informática y Sistemas

Área de conocimiento: Ciencias de la Computación e Inteligencia Artificial

Universidad de Las Palmas de Gran Canaria

Escuela de Ingeniería Informática. Universidad de Las Palmas de Gran Canaria.

Agradecimientos

Agradezco en primer lugar a mi tutor José Miguel Santos Espino, quien en todo momento me apoyó a lo largo del camino, me aconsejó, motivó y consintió mis locuras esporádicas. Gracias por creer en mí.

A Ana Rodríguez Camacho por su asesoramiento y apoyo no solo en lo profesional, sino también en lo personal. Llegaste y pusiste color, donde antes todo estaba en escala de grises.

También quiero agradecer a todos aquellos usuarios que formaron parte del proyecto a lo largo de las sucesivas pruebas, que fueron varias y algunas más largas que otras. Muchas gracias a todos por los buenos momentos y por enseñarme a hacer las cosas mejor.

En último lugar, pero no por ello menos importante, quiero agradecer con un gracias muy grande, a mi familia, pareja y amigos, especialmente a mi madre, quien gracias a su apoyo incondicional desde ya ni recuerdo cuándo, soy la persona que soy hoy, y he conseguido poder redactar estas palabras al fin.

Muchas gracias a todos, por todo.

Escuela de Ingeniería Informática. Universidad de Las Palmas de Gran Canaria.

Índice general

1. <i>I-Found</i>: recuperando lo perdido	7
1.1. Estudio del dominio del problema	10
1.1.1. Las oficinas de objetos perdidos en la actualidad	10
1.1.2. Legislación aplicable	11
1.1.3. Estado del arte	13
1.2. Organización de la memoria	14
1.2.1. El documento	14
1.2.2. El CD adjunto	15
2. Metodología de desarrollo	17
3. Recursos y tecnologías	19
3.1. Recursos	19
3.1.1. Recursos humanos: un proyecto colaborativo	19
3.1.2. Recursos software: un proyecto libre	21
3.1.3. Recursos hardware	23
3.2. Tecnologías	24
3.2.1. El sistema operativo <i>Android</i>	24
3.2.2. Arquitectura de <i>Android</i>	25
3.2.3. Componentes de una aplicación <i>Android</i>	26

3.2.4.	Ciclo de vida de una aplicación y una <i>Activity</i>	27
3.2.5.	Android Support Library: v7 appcompat	30
3.2.6.	Google Cloud Messaging (<i>GCM</i>)	31
4.	Planificación del trabajo y estimación del coste	33
5.	Análisis	35
5.1.	Especificación de requisitos	35
5.2.	Actores	38
5.3.	Casos de uso y diagramas	39
6.	Diseño	53
6.1.	Diseño de la interfaz de usuario	53
6.1.1.	Prototipo interactivo	53
6.1.2.	Evaluación del prototipo	56
6.1.3.	Diseño final de la interfaz de usuario	56
6.1.4.	Las pruebas	59
6.2.	Diseño de la arquitectura de sistemas	60
6.2.1.	El servidor	60
6.2.2.	El cliente	61
6.3.	Diseño de la base de datos	62
6.3.1.	El servidor	62
6.3.2.	El cliente	63
7.	Resultado y manual de usuario	65
7.1.	Inicio de la aplicación	65
7.2.	Inicio de sesión, registro y recuperación de contraseña	66
7.3.	Pantalla principal	70

7.4. Navegación	71
7.5. Publicar	71
7.5.1. ¿Qué perdiste?	71
7.5.2. ¿Qué encontraste?	74
7.5.3. ¿Qué te robaron?	74
7.6. Almacén	74
7.6.1. Contactar con el dueño de una publicación	74
7.6.2. Buscar un <i>OPA</i>	76
7.7. Mensajes	76
7.7.1. Conversaciones	78
7.7.2. Notificaciones del Sr. Found	79
7.7.3. Eliminar conversaciones o notificaciones	79
7.8. Mis Cosas	80
7.8.1. Eliminar una publicación propia	80
7.8.2. Editar una publicación propia	80
7.9. Menú de configuración	82
7.9.1. Recepción de notificaciones	82
7.9.2. Eliminar cuenta	83
7.9.3. Cerrar sesión	84
7.9.4. Cambiar contraseña	84
8. Pruebas y mantenimiento	85
8.1. Pruebas de usabilidad	85
8.2. Pruebas de daltonismo	85
8.3. El mantenimiento: la comunidad	88
9. Modelo o ideas de negocio	89

10.Trabajo futuro	91
11.Conclusiones	93
Lista de acrónimos	95
Lista de figuras	96
Lista de tablas	99
Apéndices	103
A. Entrevista 'Lista de los deseos'	105
B. Entrevista sobre el prototipo	107
C. Entrevista sobre el producto final	111
D. Formulario de solicitud de información sobre objetos perdidos del Ayuntamiento de Las Palmas	114
Bibliografía y referencias	116

Escuela de Ingeniería Informática. Universidad de Las Palmas de Gran Canaria.

Capítulo 1

I-Found: recuperando lo perdido

El presente proyecto final de carrera surge con la idea de cubrir una necesidad social, a un servicio no existente en muchos contextos. Sus objetivos principales son: el estudio de la plataforma *Android* para dispositivos móviles y el desarrollo de una aplicación funcional, experimental, libre y gratuita, que ofrezca la posibilidad de poner en contacto directo, sin intermediarios, a personas que han perdido o les ha sido robado un objeto, persona u animal, con personas que los hayan encontrado y viceversa, dentro de cualquier ámbito posible. *I-Found*, se comportaría por lo tanto, como una oficina **virtual** de objetos perdidos, con la salvedad, de que no solo 'alojará' objetos, sino también personas y animales. Se hace referencia de ahora en adelante, a este conjunto de **objeto, persona o animal, como OPA**, a lo largo de todo este documento.

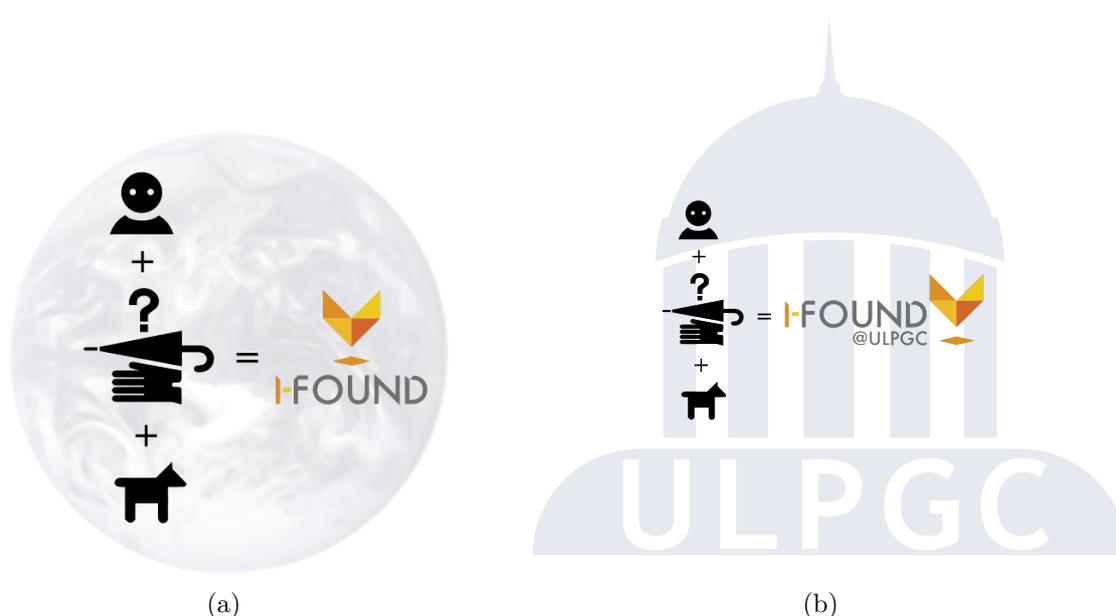


Figura 1.1: Diagrama conceptual de I-Found (a) y I-Found@ULPGC (b).

Partiendo de la base de que lo que se trata aquí, es un proyecto final de carrera que no debería

superar las novecientas horas de trabajo, se hablaría de un proceso demasiado ambicioso si se quisiera llegar con él a todos los ámbitos posibles, tal y como se explicó en el párrafo anterior, por lo que se ha decidido acotarlo al ámbito de la Universidad de Las Palmas de Gran Canaria (*ULPGC*), creando así la app *I-Found@ULPGC*, leído como *I-Found at ULPGC* (*I-Found en la ULPGC* en su versión española) y mostrando de esta manera además, que *I-Found* es un proyecto global, adaptable y personalizable a diferentes contextos de los que se hablarán en otras secciones más adelante en este documento.

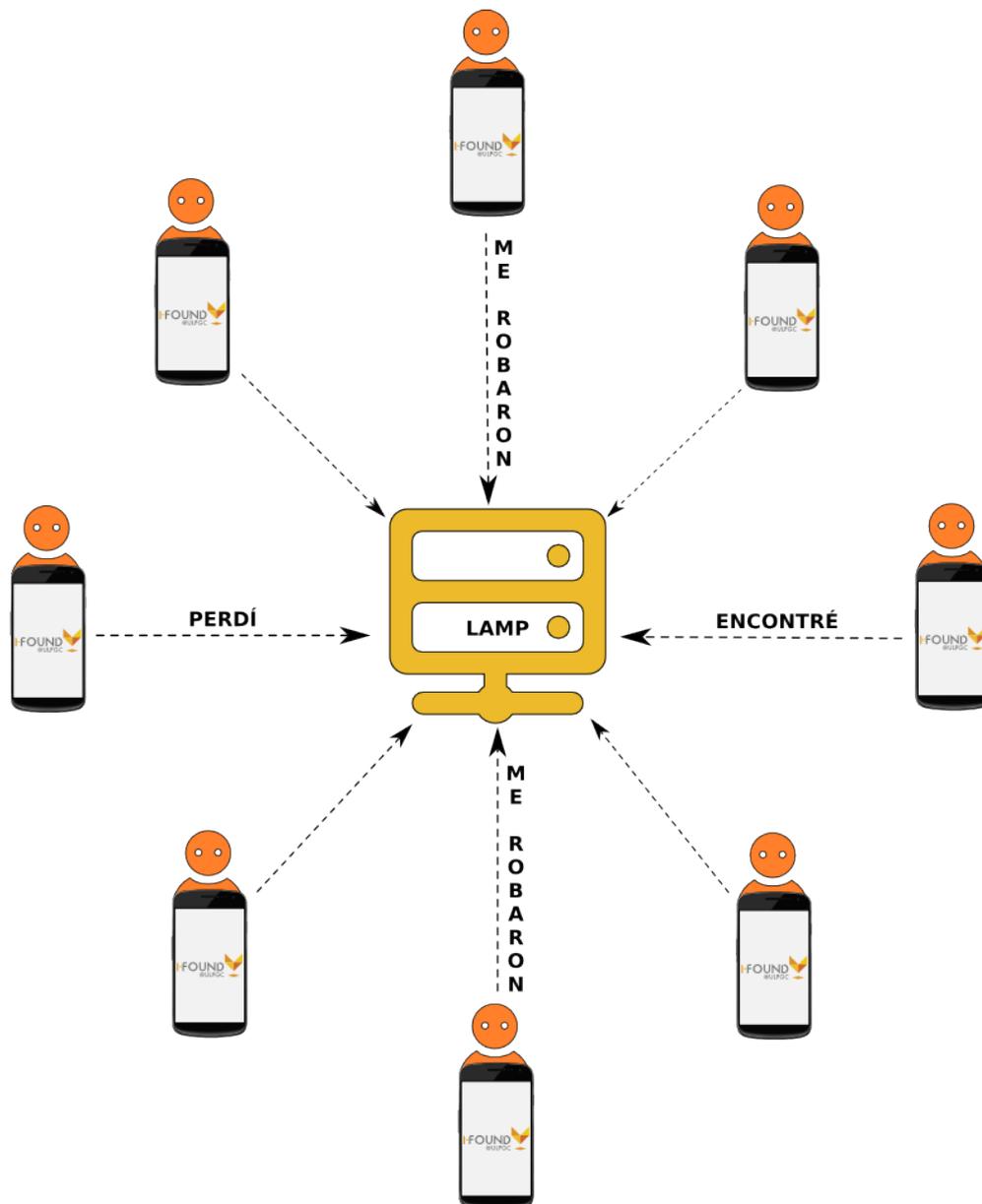


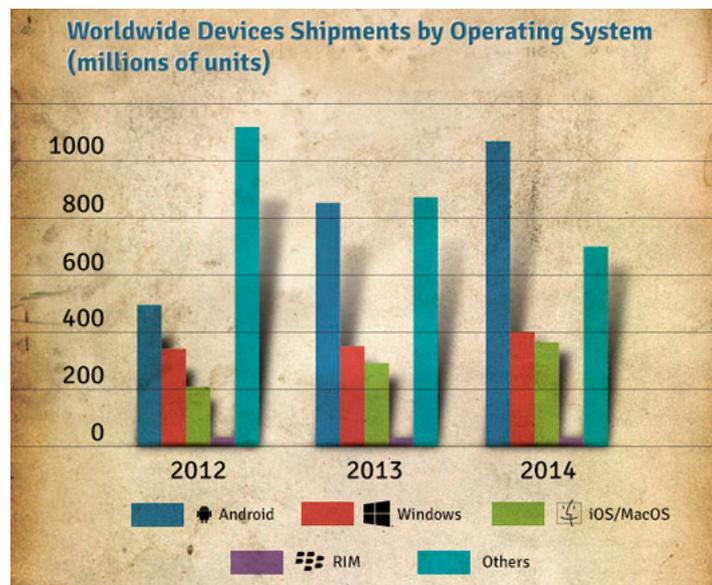
Figura 1.2: Diagrama de funcionamiento de I-Found@ULPGC.

Se justifica también aquí la inclinación por el desarrollo de una aplicación móvil. Esto se debe a que el número de usuarios que poseen un smartphone¹ en la actualidad es cada vez

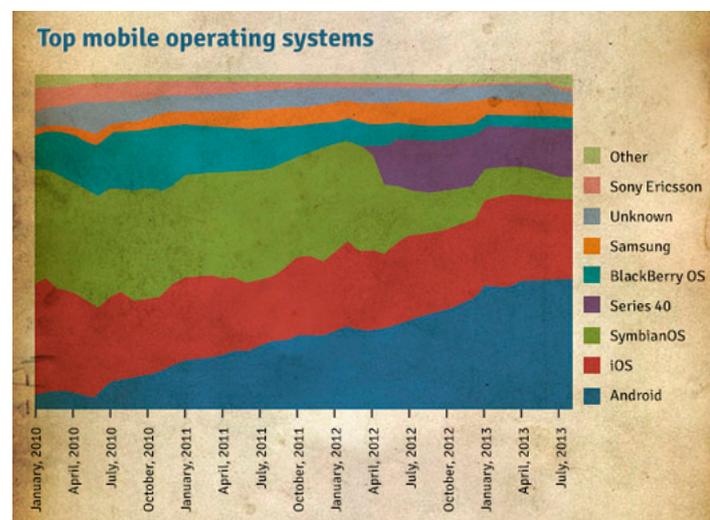
¹Término usado para identificar a un teléfono móvil con una mayor capacidad de almacenar datos y realizar

mayor, llegando a un cincuenta y seis por ciento de la cuota de mercado, comparándolo con otros dispositivos telefónicos.

Además, se ha elegido la tecnología *Android* para el proyecto, debido a su carácter de libre distribución y al crecimiento que ha sufrido en un breve espacio de tiempo, convirtiéndose en el líder del mercado de los sistemas operativos móviles, tal y como se puede apreciar en las estadísticas recogidas por *Super Monitoring* de la figura 1.3.



(a)



(b)

Figura 1.3: Estadísticas de *Super Monitoring* [29].

Se podía haber elegido cualquier otro *framework*² para el desarrollo de una aplicación web actividades semejantes a una minicomputadora y conectividad que un teléfono móvil convencional [48].

²Estructura conceptual y tecnológica, normalmente con artefactos o módulos de software concretos, que puede

multiplataforma, como por ejemplo *PhoneGap* [41], sin embargo, ha sido expreso deseo de la alumna y uno de los objetivos del proyecto, el aprendizaje de la tecnología *Android*. Por otro lado, aunque este tipo de *frameworks* son un gran avance y apoyo para realizar de forma sencilla y rápida aplicaciones móviles, tienen como contra que eliminan la posibilidad de acceder a gran parte del hardware del dispositivo. Además, para aplicaciones de una cierta complejidad como de la que se habla en este proyecto, podría afectar de forma muy negativa a su rendimiento, que es otro de los grandes problemas de este tipo de tecnologías.

1.1. Estudio del dominio del problema

El propósito de este capítulo es situar el proyecto dentro de un contexto en el mercado de las tecnologías móviles y proveer al lector de la información necesaria para entender las peculiaridades del mismo.

1.1.1. Las oficinas de objetos perdidos en la actualidad

Cada día aparecen en el mundo miles de objetos extraviados, cuyos dueños no son encontrados. Lo que suele suceder en la mayoría de los casos, cuando una persona encuentra un objeto extraviado, es que lo ignora, o lo hace de su propiedad. Pocas veces se da el caso en el que la persona lo devuelve a su dueño, o lo lleva a alguna oficina de objetos perdidos, por lo que se ha decidido estudiar el porqué de este comportamiento y se ha hecho un pequeño análisis de la situación.

Dejando a un margen los aspectos de educación y cultura, y reuniendo información de algunas de las comisarías de policía nacional y local de la isla, además de haber preguntado a varios ciudadanos, se han descubierto una serie de factores que provocarían el anterior comportamiento descrito:

- Los organismos pertinentes con oficinas de objetos perdidos no disponen de un procedimiento o protocolo estándar. Es más, en varios de los casos, se comprobó que los agentes o personas encargadas no sabían exactamente cómo proceder ante la situación.
- Los organismos pertinentes muchas veces no disponen de almacén o logística en general para poder guardar ni catalogar los objetos, por lo que muchas veces se pierden, o son introducidos en una caja. En algunos casos, si no aparece el dueño, son dados a la beneficencia y en otros casos, subastados.
- En la mayoría de los casos, los usuarios no saben cuáles son los organismos o lugares a los que acudir en el caso de encontrar un objeto. Todos nombraron el ayuntamiento, la policía nacional o la policía local, pero ninguno estaba seguro de la ubicación, ni tampoco del procedimiento a seguir.

servir de base para la organización y desarrollo de software. Puede incluir soporte de programas, bibliotecas, y un lenguaje interpretado, entre otras herramientas, para así ayudar a desarrollar y unir los diferentes componentes de un proyecto [49].

- El interés real de toda esta problemática nace en la persona que pierde el objeto y no tanto en la que lo encuentra, pero traducido a la realidad, es la persona que lo encuentra, la que se convierte en la parte activa de la situación, asumiendo con ello toda la responsabilidad, lo que se traduce en: consumo de tiempo, consumo de gasolina y consumo de dinero, puesto que los lugares destinados a recoger los objetos perdidos, suelen estar en lugares muy céntricos en la ciudad, poblados de zonas azules o verdes.
- Hay pocas oficinas de objetos perdidos. En concreto, en la ciudad de Las Palmas de Gran Canaria, hay solamente una oficialmente reconocida y se encuentra en el centro de la ciudad, a gran distancia de los hogares de muchos de los usuarios.
- No existe una base de datos pública para la consulta ciudadana con la información centralizada, de tal forma que si un usuario que pierde un objeto, quiere encontrarlo, debe recorrer todas las oficinas oficiales de objetos perdidos y los organismos con 'oficinas no oficiales' también.
- Varios usuarios tienen miedo a que se les acuse de robo.
- Varios usuarios confirman no confiar en los organismos para este tipo de gestiones, afirmando creer que podrían apropiarse de lo que ellos devuelven.
- Varios usuarios afirman tener 'pereza' o vivir lejos de los puntos de entrega de objetos perdidos, por lo que optan finalmente por dejarlos donde los encuentran.

En conclusión, hay muchos factores por los que se produce este comportamiento, y este proyecto pretende solucionar algunos de ellos.

1.1.2. Legislación aplicable

El hallazgo, depósito y restitución de los objetos perdidos viene regulado en el Código Civil español, en sus artículos 615 y 616. En los mismos se establece lo siguiente:

- **Artículo 615**

El que encontrare una cosa mueble, que no sea tesoro, debe restituirla a su anterior poseedor. Si éste no fuere conocido, deberá consignarla inmediatamente en poder del Alcalde del pueblo donde se hubiese verificado el hallazgo. El Alcalde hará publicar éste, en la forma acostumbrada, dos domingos consecutivos. Si la cosa mueble no pudiese conservarse sin deterioro o sin hacer gastos que disminuyan notablemente su valor, se venderá en pública subasta luego que hubiesen pasado ocho días desde el segundo anuncio sin haberse presentado el dueño, y se depositará su precio. Pasados dos años, a contar desde el día de la segunda publicación, sin haberse presentado el dueño, se adjudicará la cosa encontrada o su valor al que la hubiese hallado. Tanto éste como el propietario estarán obligados, cada cual en su caso, a satisfacer los gastos.

- **Artículo 616**

Si se presentare a tiempo el propietario, estará obligado a abonar, a título de premio, al que hubiese hecho el hallazgo, la décima parte de la suma o del precio de la cosa encontrada.

Cuando el valor del hallazgo excediese de 2.000 pesetas el premio se reducirá a la vigésima parte en cuanto al exceso.

En la ciudad de Las Palmas de Gran Canaria, este servicio lo realiza la oficina de Objetos Perdidos ubicada en las Oficinas Centrales de la Policía Local, que dispone de los siguientes servicios:

1. Facebook [31]: Aquí se anuncian algunos de los objetos que llegan a la comisaría, incluyendo fotos de los mismos. Muy pocas personas saben de su existencia.
2. Página web oficial [32]: A través de esta página, los ciudadanos pueden tramitar una solicitud de información sobre objetos perdidos. Cabría pensar que se trata de un sencillo formulario online, pero se trata de un .pdf, tal y como el que se muestra en el 'Apéndice D', que el ciudadano tendrá que descargar, rellenar y luego volver a enviar por e-mail, a la dirección oficial de la comisaría, entregar en persona, o en alguna oficina de 'Atención al ciudadano'.

Se comprueba pues, que los servicios ofrecidos hasta el momento son semi-telemáticos. Este proyecto pretende hacer este tipo de gestiones, independientemente del contexto, totalmente telemáticas, atendiendo además a lo recogido en la legislación.

1.1.3. Estado del arte

Como parte del estudio del problema, se ha elaborado un análisis de las diferentes aplicaciones ya existentes con funcionalidades similares a las que se abarcan en este proyecto, con el fin de poder aprender de su experiencia de usuario y además, poder identificar sus carencias. De dicho análisis, se han obtenido los resultados que se muestran en la tabla 1.4, que a su vez, se han comparado con las funcionalidades que ofrece *I-Found@ULPGC*.

	Interfaz amigable	Sistema de mensajería integrado	Subir imágenes (galería/cámara)	Sistema de contacto por alias	Idioma (español)	Localización dada	Notificaciones
Lost & Found, Crime Watch, etc							
Sniffer - Lost & Found board							
분실물 찾기							
Lost n Found							
I-Found@ULPGC							

Figura 1.4: Comparativa entre *I-Found@ULPGC* y apps con funciones similares.

Como se puede comprobar en la tabla, son varias las aplicaciones existentes en el mercado cuya misión u objetivo es el mismo que el de este proyecto. Como se puede observar, ninguna engloba todas las funcionalidades mostradas.

1.2. Organización de la memoria

1.2.1. El documento

Este apartado está destinado a ofrecer una breve explicación de lo que podrá encontrarse en cada uno de los capítulos y anexos que componen este documento:

- **Metodología de desarrollo.** En este capítulo, se especifican las técnicas utilizadas para el desarrollo del proyecto desde la perspectiva del desarrollo de software.
- **Recursos y tecnologías.** En esta sección, se introducen los diferentes recursos, humanos, hardware y software que han sido necesarios para la realización de este proyecto, así como las diversas tecnologías que se han utilizado.
- **Planificación del trabajo y estimación del coste.** Aquí se detalla la planificación de todo el proceso y se presenta una estimación aproximada del coste del conjunto.
- **Análisis.** Estudio en profundidad de la funcionalidad deseada, expresada en términos de captura de requisitos y casos de uso.
- **Diseño.** En este capítulo se presenta un estudio en profundidad de la arquitectura del sistema en su conjunto, las bases de datos y las interfaces de usuario.
- **Resultado y manual de usuario.** Exposición de los resultados obtenidos, explicados en gran detalle para poder comenzar a utilizar el producto.
- **Pruebas y mantenimiento.** Aquí se explican con detalle todas las pruebas a las que ha sido sometido el proyecto.
- **Modelo o ideas de negocio.** En este capítulo se explican diferentes enfoques o líneas de trabajo, con los que se podría sacar provecho económico de este proyecto.
- **Trabajo futuro.** Aquí se presentan diferentes ideas para enriquecer y mejorar el proyecto, que podrían ser objetivo de próximas versiones.
- **Conclusiones.** Se dan a conocer las conclusiones finales extraídas de todo el proceso de desarrollo del proyecto.
- **Apéndices.** Se incluye en este apartado información adicional que mejora la comprensión del conjunto.
- **Bibliografía y referencias.** En este capítulo pueden consultarse todas las referencias utilizadas a lo largo del desarrollo del proyecto, así como la bibliografía empleada para su ejecución.

1.2.2. El CD adjunto

El CD que se adjunta a este documento incluye el código fuente desarrollado, junto con todo el trabajo relacionado que ha sido necesario realizar para completar el proyecto, además del trabajo de diseño gráfico derivado de la ejecución del mismo.

La estructura de las carpetas es la siguiente:

1. directorio *PSC/*: trabajo completo sobre este proyecto, realizado por Patricia Santana Cruz.
 - subdirectorio *fuentes/*: código fuente de *I-Found@ULPGC*.
 - subdirectorio *prototipo/*: prototipo interactivo completo de *I-Found@ULPGC*.
 - subdirectorio *prototipo/videos/*: vídeos de las pruebas de usuario sobre el prototipo.
 - subdirectorio *licencia/*: licencia para *I-Found@ULPGC*.
 - subdirectorio *iconos/*: iconos de la app y sus licencias.
 - subdirectorio *memoria/*: este documento en formato .pdf.
2. directorio *ARC/*: trabajo gráfico derivado de este proyecto, realizado por Ana Rodríguez Camacho.

Capítulo 2

Metodología de desarrollo

La metodología usada en el transcurso de este proyecto ha sido el **Proceso Unificado Racional**, *Rational Unified Process* en inglés, habitualmente resumido como RUP, este es un proceso de desarrollo de software que junto con el **Unified Modeling Language UML**, constituye la metodología estándar más utilizada para el análisis, la implementación y la documentación de sistemas orientados a objetos. Como modelo de ingeniería del software [50] se ha utilizado la técnica del prototipado, que se describe a continuación.

La técnica del prototipado se caracteriza por:

- Ser común a muchas disciplinas de ingeniería.
- Facilitar la creación del modelo del software a construir.
- Poder adoptar tres formas posibles:
 1. Prototipo que describa la interacción hombre-máquina (en papel o PC).
 2. Prototipo que implemente un subconjunto de funciones requeridas.
 3. Programa existente que ejecute parte o toda la función deseada pero que tenga otras características que deban ser mejoradas.

Las fases de la construcción de los prototipos son las siguientes:

1. **RRR**: Recolección y refinamiento de requisitos;
2. **DR**: Diseño rápido;
3. **CP**: Construcción del prototipo;
4. **EPC**: Evaluación del prototipo por el cliente;
5. **RP**: Refinamiento del prototipo;
6. **PI**: Producto de ingeniería.

En el capítulo **5. Análisis** se verá la ejecución de este proceso con detalle, que se ha producido de dos iteraciones.

Capítulo 3

Recursos y tecnologías

En este capítulo se detallan todos los recursos tanto humanos como de tipo software y hardware, además de las diversas tecnologías que han sido necesarias para completar el desarrollo de este proyecto.

3.1. Recursos

3.1.1. Recursos humanos: un proyecto colaborativo

Como ya se ha mencionado en secciones anteriores, *I-Found@ULPGC*, es una aplicación de código abierto y gratuita, es decir, todos los usuarios de la misma tendrán libertad para ejecutar, copiar, distribuir, estudiar e incluso contribuir modificando el código de la misma. Es este, uno de los motivos por los que este proyecto es un potencial proyecto colaborativo, que podrá generar tras él una comunidad de usuarios que lo soporte y alimente de diversas maneras: contribuyendo con nuevo código, avisando de posibles *bugs*¹, proponiendo nuevas funcionalidades, traducándolo a diferentes idiomas, etc. Un proyecto colaborativo, une los esfuerzos e ideas de diversas personas apasionadas, creando productos fuertes, altamente competitivos y mucho más ricos.

Otro motivo por el que se puede decir que *I-Found@ULPGC* es colaborativo, es la presencia de Ana Rodríguez Camacho como colaboradora en el diseño gráfico del proyecto. Ana es estudiante de Técnico Superior en Artes Plásticas y Diseño en Gráfica Publicitaria en la Escuela de Arte y Superior de Diseño Gran Canaria (*EASD*) y como proyecto final de carrera, decidió colaborar en *I-Found@ULPGC*, incorporándose tras la fase de prototipado y pruebas de usuario del mismo, dando orientación para mejorarlo, además de aportar la tipografía y esquema de colores que se usaría a lo largo del ciclo de trabajo. Su contribución, para ser más precisos, se ha centrado principalmente en el **asesoramiento de diseño** basándose en el prototipo y la creación de la **identidad corporativa** y **plan de comunicación visual** de la aplicación.

¹Es un error o fallo en un programa o sistema de software que desencadena un resultado indeseado [51].

El objetivo buscado en su trabajo ha sido la unificación de criterios de actuación y difusión de la información visual, mediante la generación de una imagen que contribuyera a la clara diferenciación respecto a los competidores de *I-Found@ULPGC* y que la dotara de personalidad propia, teniendo en cuenta, además, la interacción del usuario, ya fuera directamente a través de la app o mediante otras vías de comunicación (marketing y publicidad, redes sociales, web, stands y otras instalaciones efímeras, etc.).

El **asesoramiento de diseño** basándose en el prototipo, se ha realizado siempre en conjunto con la desarrolladora, por un lado para facilitar el proceso de desarrollo y por otro para aunar la experiencia de ambos en usabilidad² con el fin de contribuir a un mejor resultado.

Con el desarrollo de la **identidad corporativa** se ha buscado la manifestación física de la marca, centrándose en la identidad visual mediante un conjunto de signos que traducen gráficamente la esencia corporativa, y que se han recogido a través de un documento denominado "Manual de Identidad". Éste contiene, entre otros aspectos: logotipo, imago tipo, logosímbolo corporativo, esquemas constructivos y de reproducción de los mismos, escalas de reducción y aplicación a diferentes dispositivos (digitales e impresos), usos correctos e incorrectos de la imagen gráfica, cromatismo corporativo (versiones a color, escala de grises, versiones positiva y negativa y fondos corporativos), espacios de reserva y utilización de manchetas, tipografía principal y secundaria y elementos de estilo.

Con el **plan de comunicación visual** se ha trazado una guía de actuación cuyo objetivo ha sido lograr desarrollar una estrategia de publicitación y posicionamiento de la marca en el mercado. Esto se conseguiría a través de una campaña de implantación en varias fases y que en la primera etapa se centraría en la *ULPGC*.

Gracias a esta rica colaboración, *I-Found@ULPGC* se convierte por lo tanto, en uno de los primeros proyectos de esta facultad, en crear un producto resultado de la unión de esfuerzos de dos entidades educativas diferentes: la Escuela de Arte y Superior de Diseño Gran Canaria y la Escuela de Ingeniería Informática de la *ULPGC*, creando un producto final de mayor calidad, consecuencia del trabajo de dos estudiantes que han puesto en práctica todos sus conocimientos.

Todo el trabajo gráfico en detalle, puede verse completo en el directorio *ARC/* del CD adjunto a este documento.

²Se refiere a la facilidad con que las personas pueden utilizar una herramienta particular o cualquier otro objeto fabricado por humanos con el fin de alcanzar un objetivo concreto. La usabilidad también puede referirse al estudio de los principios que hay tras la eficacia percibida de un objeto [47].

3.1.2. Recursos software: un proyecto libre

Todas las herramientas utilizadas para el desarrollo de este proyecto, son de software libre³ a excepción de una de ellas (*SCR Screen Recorder Pro*), por lo que solamente ha sido necesaria la compra de esta licencia.

1. Para el desarrollo:

- ***Java Runtime Environment (JRE)***
Conjunto de utilidades que permite la ejecución de programas *Java*.
- ***Eclipse***
Entorno de desarrollo integrado (*IDE*), de código abierto y multiplataforma. Mayoritariamente se utiliza para desarrollar lo que se conoce como aplicaciones de cliente enriquecido y se ha utilizado para este proyecto durante todo el proceso de desarrollo y diseño final.
- ***Android Development Tools (ADT)***
Plug-in⁴ diseñado para desarrollar aplicaciones móviles basadas en *Android* usando el entorno de desarrollo Eclipse. Entre sus características, destacan las siguientes:
 - Permite la descarga y actualización de las últimas versiones de las librerías de desarrollo para *Android*.
 - Facilita la depuración de código gracias a su herramienta de Log.
 - permite la simulación de las aplicaciones desarrolladas, gracias a su *AVD Manager (Android Virtual Device Manager)*.
- ***API Key***
Necesaria para el uso del servicio *GCM (Google Cloud Messaging)* de Google, que permite el envío de *notificaciones push*⁵.
- ***phpMyAdmin***
Herramienta escrita en *PHP* con la intención de manejar la administración de MySQL a través de páginas web, utilizando Internet, facilitando la gestión de la base de datos, motivo por el cual, se ha utilizado en este proyecto.
- ***Git***
Software para el control de versiones.

2. Para el análisis y diseño:

- ***Pencil***
Herramienta para la creación de prototipos interactivos de la interfaz gráfica de usuario.

³En inglés "free software", aunque esta denominación a veces se confunde con "gratis" por la ambigüedad del término "free.^{en} el idioma inglés, por lo que también se usa "libre software", es la denominación del software que respeta la libertad de todos los usuarios que adquirieron el producto y, por tanto, una vez obtenido el mismo puede ser usado, copiado, estudiado, modificado, y redistribuido libremente de varias formas [46].

⁴Aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica. Es ejecutada por la aplicación principal e interactúan por medio de la API [53].

⁵Tipo de comunicación en la que es el servidor el que inicia la petición al cliente cuando tiene una información o notificación nueva, permitiendo un importante ahorro de recursos y tiempo respecto a la tecnología convencional pull.

- ***Kazam***
Herramienta para la grabación de los vídeos hechos a los usuarios durante las pruebas del prototipo.
- ***Dia***
Aplicación de propósito general para la creación de diagramas, desarrollada como parte del proyecto *GNOME*⁶. Se ha utilizado para la creación de diagramas de bases de datos.
- ***Umbrello***
Herramienta libre para crear y editar diagramas UML, que ayuda en el proceso del desarrollo de software. Se ha utilizado en este proyecto para la creación de los diagramas de caso de uso y clases.
- ***Inkscape***
Editor de gráficos en formato vectoriales *SVG*, gratuito, libre y multiplataforma, que se ha utilizado para la creación o edición de iconos y otros elementos gráficos del proyecto.
- ***Android Asset Studio***
Aplicación online, para la creación y edición de iconos e imágenes de tipo *Nine-patch*⁷.
- ***Coblis* — *Color Blindness Simulator***
Herramienta para la simulación de los problemas visuales asociados con el color. Fue usada para hacer pruebas al proyecto y comprobar que estos grupos, también pueden utilizar la aplicación y verla sin problemas.

3. Para la documentación:

- ***Texmaker***
Editor gratuito distribuido bajo la licencia *GPL* para escribir documentos de texto, multiplataforma, que integra muchas herramientas necesarias para desarrollar documentos con \LaTeX ⁸, en una sola aplicación. Ha sido utilizado como herramienta principal para la redacción de esta memoria.
- ***LibreOffice Writer***
Procesador de texto perteneciente a la suite ofimática libre *LibreOffice*. Es de código abierto y fue desarrollada por The Document Foundation. Se creó como bifurcación de OpenOffice.org en 2010. Ha servido de apoyo para algunas de las partes de este documento.

4. Para la presentación:

- ***LibreOffice Impress***
Gestor de presentaciones también perteneciente a la suite ofimática *LibreOffice*. Ha servido para la creación de la presentación de este proyecto.

⁶Es un entorno de escritorio e infraestructura de desarrollo para sistemas operativos *GNU/Linux*, *Unix* y derivados *Unix* como, *BSD* o *Solaris*; compuesto enteramente de software libre [38].

⁷Son archivos especiales que *Android* reconoce y que le permite cambiar su tamaño (alto o ancho) para ajustarse a la resolución de la pantalla del dispositivo, sin distorsionarla.

⁸Sistema de composición de textos, orientado especialmente a la creación de libros, documentos científicos y técnicos [54].

- ***SCR Screen Recorder Pro***

Herramienta de screencast⁹ para dispositivos móviles con sistema operativo *Android*. Se ha utilizado para grabar la interacción del usuario con la aplicación, con el objetivo de ser mostrada en la demo. Es la única aplicación propietaria que se ha utilizado en este proyecto y aún habiendo otras opciones libres o gratuitas en el mercado, se ha decidido utilizar esta, por su resultado de notable mejor calidad. Existe también una versión gratuita, sin embargo, esta solamente permite hacer vídeos muy breves que además incluyen una marca de agua en el resultado final y no permiten ver con claridad algunas funcionalidades de la aplicación. Su coste es de tan solo 4.50 euros.

- ***OpenShot Video Editor***

Editor de video no lineal, que ha sido utilizado para la edición de los vídeos de la demo.

3.1.3. Recursos hardware

Todos los recursos hardware necesarios, han sido suministrados por la alumna del proyecto, por lo que no ha supuesto ningún tipo de gasto extra para la *ULPGC*.

1. Servidor con sistema operativo de la familia *GNU/Linux* instalado con infraestructura *LAMP*¹⁰, con conexión a Internet y que tenga la siguiente configuración software:
 - Servidor Web *Apache*, versión 2.0 o superior.
 - Intérprete *PHP* versión 5.3 o superior.
 - Sistema de gestión de base de datos *Mysql* versión 5.5 o superior.
 - Simple Mail Transfer Protocol (*SMTP*) configurado.
2. Dos dispositivos móviles con sistema operativo *Android* en su versión 2.2 (API 8, Froyo) o superior, con capacidad de activación de opciones para el desarrollador y conexión a Internet. Se ha decidido elegir como versión mínima la 2.2 de la *API*, para poder abarcar la mayor cuota de mercado posible, atendiendo a la figura 3.1, correspondiente a un gráfico proporcionado por *Google Developers*.

⁹Es una grabación digital de la salida por pantalla del ordenador, smartphone u otro dispositivo, a veces conteniendo narración de audio [52].

¹⁰Sistema de infraestructura de internet que usa las siguientes herramientas: *Linux*, el sistema operativo (en algunos casos también se refiere a LDAP); *Apache*, el servidor web; *emphMySQL/MariaDB*, el gestor de bases de datos; *Perl*, *PHP*, o *Python*, los lenguajes de programación.

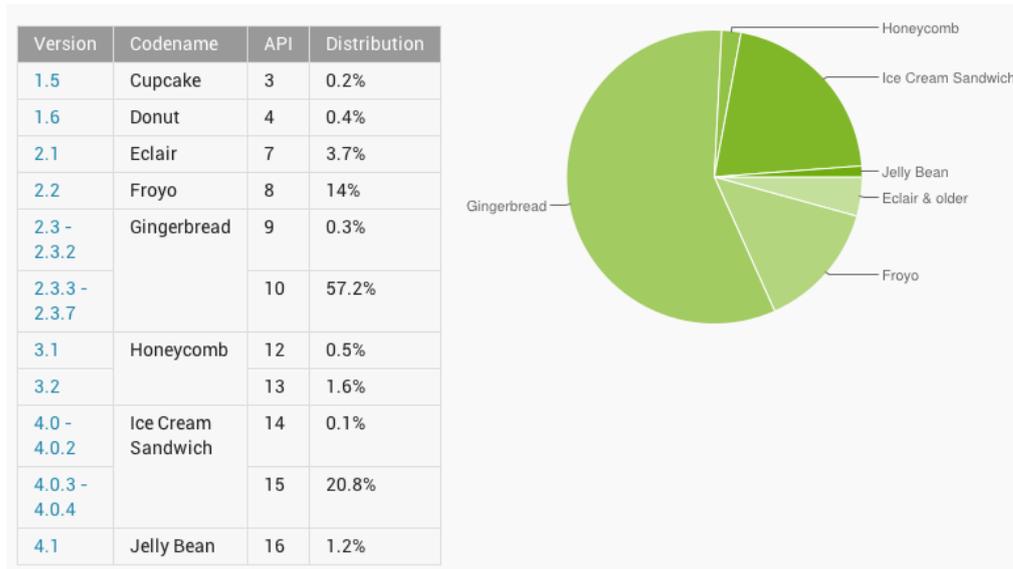


Figura 3.1: Cuota de mercado 2013 de las versiones de Android.

3.2. Tecnologías

Esta sección ofrece al lector una visión global del sistema operativo *Android* y algunas de las tecnologías asociadas a él a la hora de afrontar un proyecto de este tipo.

3.2.1. El sistema operativo *Android*

Es un sistema operativo basado en el kernel de *GNU/Linux* diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes o tabletas, inicialmente desarrollado por *Android Inc.*, a la cual *Google* respaldó económicamente y más tarde compró. El sistema está escrito en *C* y *C++* y se apoya en *Java* para la interfaz gráfica, lenguaje escogido por *Google* para escribir el conjunto de librerías y componentes que formarán el ecosistema de desarrollo de aplicaciones para esta tecnología.

3.2.2. Arquitectura de *Android*

La figura 3.2 muestra la arquitectura de *Android*. Como se puede ver, está formada por cuatro capas. Una de las características más importantes es que todas las capas están basadas en software libre.

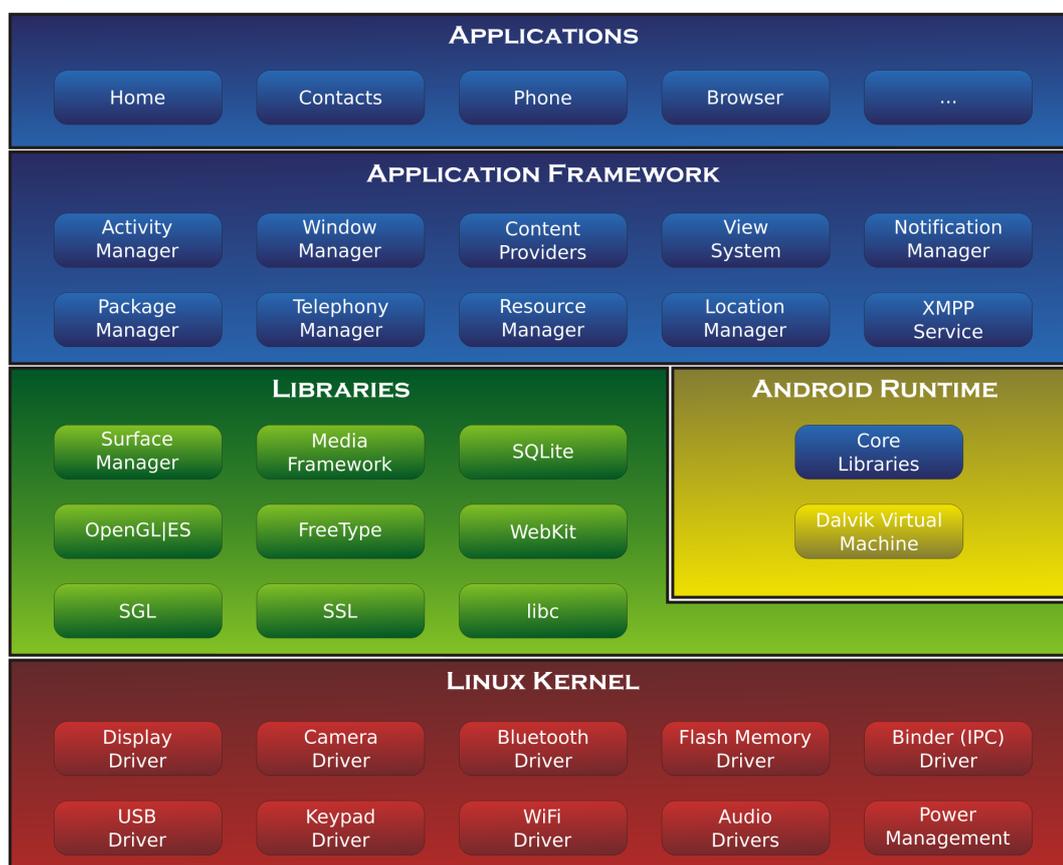


Figura 3.2: Arquitectura de *Android*. Fuente: Wikipedia

Dada la gran complejidad de librerías y ramificaciones disponibles en el lenguaje *Java* y a las limitaciones de los dispositivos móviles donde ha de ejecutarse *Android* (poca memoria y procesador limitado) no fue posible utilizar una máquina virtual *Java* estándar para el *Runtime de Android*, sino que *Google* creó una nueva: *Dalvik*, cuyas características principales, pueden verse a continuación:

- Ejecuta ficheros *Dalvik* ejecutables (.dex) (formato optimizado para ahorrar memoria).
- Está basada en registros.
- Cada aplicación corre en su propio proceso *Linux* con su propia instancia de la máquina virtual *Dalvik*.
- Delega al kernel de *Linux* algunas funciones como *threading*¹¹ y el manejo de memoria a bajo nivel.

¹¹Técnica que hace uso de hilos de ejecución o subprocesos. La creación de un nuevo hilo es una característica

En circunstancias normales, el código *Java* se transformaría durante la compilación a bytecode de *Java*, sin embargo, *Dalvik* no es capaz de reconocer este bytecode, por lo que es necesaria la transformación del mismo al *Dalvik bytecode*, cuyo nombre es *Dex*, *Dalvik executable*.

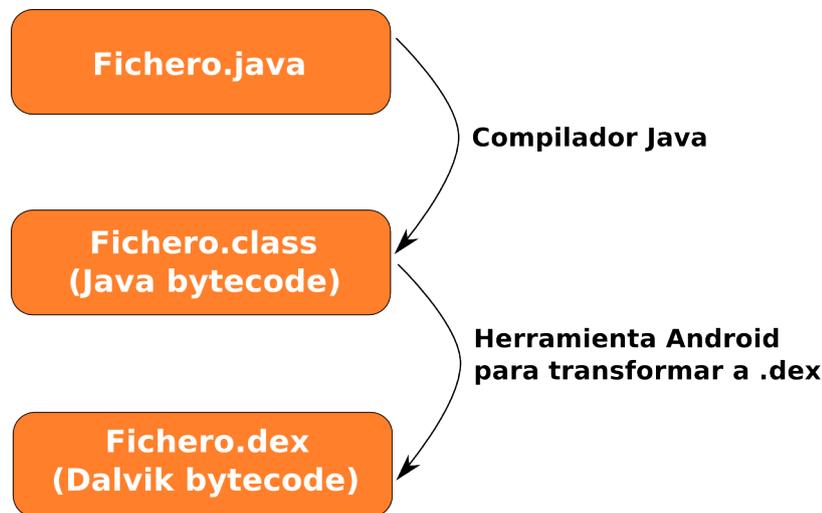


Figura 3.3: Proceso de transformación de Java bytecode a Dalvik bytecode.

3.2.3. Componentes de una aplicación *Android*

En este apartado se habla sobre los componentes que pueden encontrarse en cualquier aplicación *Android*. Estos pueden ser combinados libremente (sin ser obligatorio el uso de todos ellos), previa declaración de los mismos en el fichero *AndroidManifest.xml*, donde también se definen los permisos de la aplicaciones.

1. **View**

Las vistas son los elementos que componen la interfaz de usuario de una aplicación: un botón, una entrada de texto, etc. Todas descienden de la clase *View* y pueden ser definidas utilizando código *Java*, pero muy comunmente se definirán utilizando ficheros *XML*.

2. **Layout**

Es un conjunto de *Views* agrupadas de una determinada forma y también pueden ser definidos en código o en ficheros *XML*.

3. **Activity**

Representan el componente principal de la interfaz gráfica de una aplicación *Android*. Se puede pensar en una actividad como el elemento análogo a una ventana o pantalla en cualquier otro lenguaje visual. Su función principal es la creación de la interfaz de usuario.

que permite a una aplicación realizar varias tareas a la vez (concurrentemente). Los distintos hilos de ejecución comparten una serie de recursos tales como el espacio de memoria, los archivos abiertos, situación de autenticación, etc. Esta técnica permite simplificar el diseño de una aplicación que debe llevar a cabo distintas funciones simultáneamente.

La mayoría de aplicaciones están compuestas de varias actividades independientes entre sí, pero que trabajan unidas para conseguir un objetivo común. Toda actividad pertenece a una clase descendiente de *Activity*.

4. **Service**

Son componentes sin interfaz gráfica que se ejecutan en segundo plano. En concepto, son exactamente iguales a los servicios o demonios presentes en cualquier otro sistema operativo. Pueden realizar cualquier tipo de acciones, por ejemplo actualizar datos, lanzar notificaciones no intrusivas, o incluso mostrar elementos visuales, si se necesita en algún momento la interacción con el usuario para obtener una confirmación. De igual manera que las actividades, se implementan mediante la clase con su mismo nombre.

5. **Intent**

Representa la voluntad de realizar alguna acción. Se utiliza cada vez que se quiere:

- Lanzar una *Activity*.
- Lanzar un *Service*.
- Lanzar un anuncio de tipo *Broadcast*.
- Comunicar con un *Service*.

Los componentes lanzados pueden ser internos o externos a la aplicación. También son usados para el intercambio de información entre los componentes.

6. **Broadcast receiver**

Es un componente destinado a detectar y reaccionar, ante determinados mensajes o eventos globales generados por el sistema. También es posible, mediante el uso de los *Intents*, generar mensajes broadcast en la propia aplicación, que estén dirigidos a cualquier *Broadcast receiver* que se encuentre activo. Este componente no tiene una interfaz gráfica asociada, pero puede utilizar la *API Notification Manager* para avisar de manera no intrusiva al usuario del evento que se ha producido, a través de la barra de notificaciones presente en el sistema. Al igual que los componentes anteriores, los componentes *Broadcast receiver* se implementan a través de la clase con su mismo nombre.

7. **Content provider**

Es el mecanismo que se ha definido en *Android* para compartir datos entre las aplicaciones. Mediante estos componentes, es posible compartir determinados datos de la aplicación, sin mostrar detalles sobre su almacenamiento interno, su estructura, o su implementación. De la misma forma, la aplicación podrá acceder a los datos de otra, a través de los *Content provider* que se hayan definido en ella.

3.2.4. **Ciclo de vida de una aplicación y una *Activity***

Una aplicación *Android* se ejecuta en su propio proceso *Linux*. Este proceso es creado, cuando parte del código necesita ser ejecutado, y continuará vivo hasta que ya no sea requerido y el sistema reclame su memoria para asignársela a otra aplicación.

El tiempo de vida de un proceso no es controlado directamente por la aplicación, sino que es el sistema quien lo determina, basado en el conocimiento que tiene de las partes de la aplicación, la importancia de la misma y cuánta memoria hay disponible. Por lo tanto, el usuario no tiene conocimiento sobre estas áreas, ya que según la naturaleza de *Android*, son tareas del sistema operativo.

En el caso de las actividades, *Android* las maneja como una pila. Cuando se crea una nueva actividad, esta se coloca en lo más alto de la pila y relega a la actividad anterior a permanecer justo debajo de ella en la propia pila. Una actividad en *Android* puede estar en uno de estos cuatro estados:

Etapa	Descripción
Activa (Running)	La actividad está en la parte más alta de la pila, lo que quiere decir que es visible y tiene el foco.
Visible (Paused)	La actividad es visible, pero no tiene el foco. Se alcanza este estado cuando pasa a activa otra actividad con alguna parte transparente o que no ocupa toda la pantalla. Cuando una actividad está tapada por completo, pasa a estar parada.
Parada (Stopped)	Cuando la actividad no es visible. Se recomienda guardar el estado de la interfaz de usuario, preferencias, etc...
Destruída (Destroyed)	Cuando la actividad termina al invocarse el método <i>finish()</i> , o es acabada por el sistema <i>Android</i> . Es eliminada de la pila de actividades.

Tabla 3.1: Estados posibles de una *Activity*.

Cada vez que una actividad cambia de estado se van a producir eventos que podrán ser capturados por ciertos métodos de la actividad:

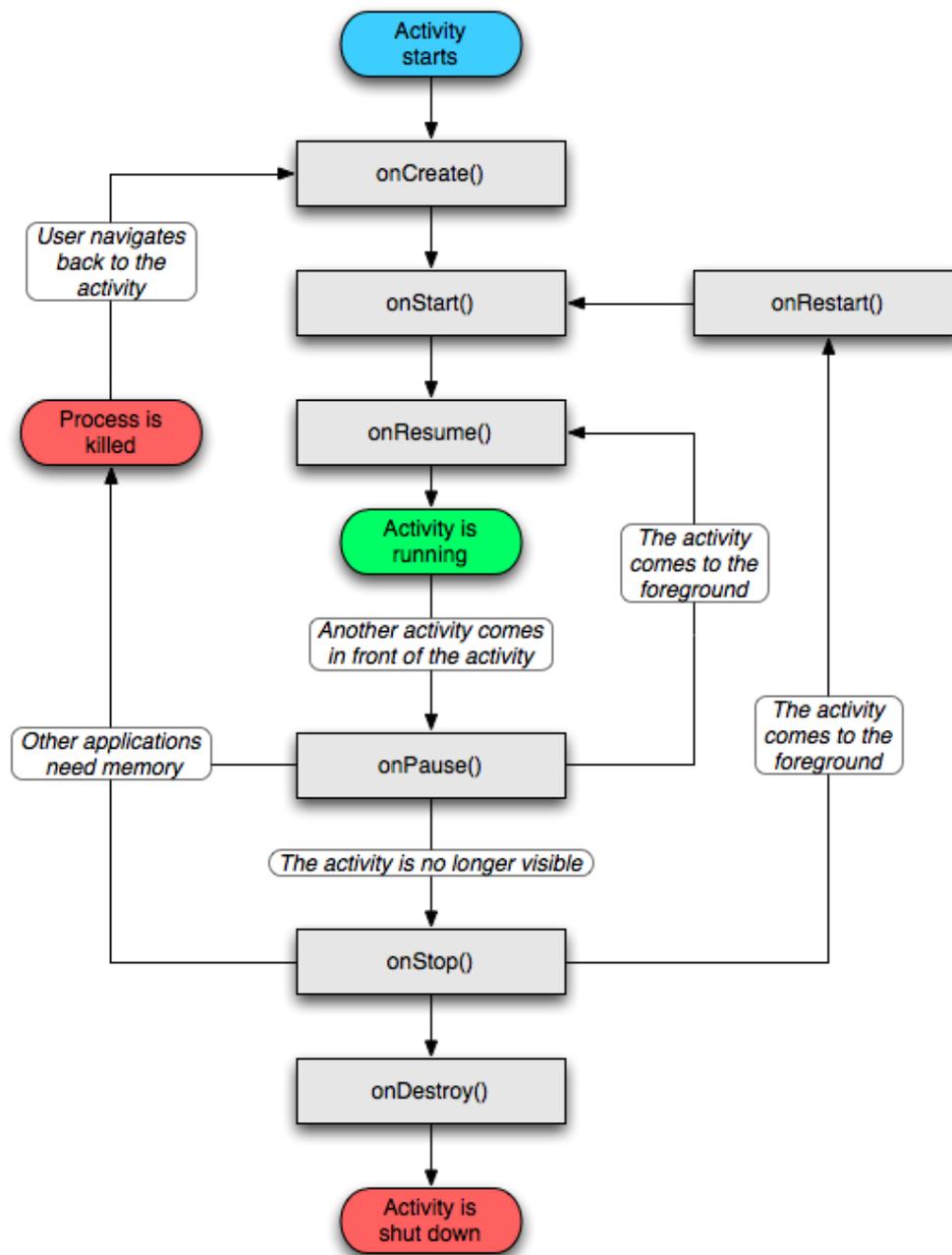


Figura 3.4: Ciclo de vida de una Activity en Android.

En la siguiente tabla, se explica cada uno de los métodos:

Evento	Descripción
<i>onCreate(Bundle)</i>	Se llama en la creación de la actividad y se utiliza para realizar todo tipo de inicializaciones, tales como la creación de la interfaz de usuario o la inicialización de estructuras de datos. Puede recibir información de estado de instancia (en una instancia de la clase <i>Bundle</i>), por si se reanuda desde una actividad que ha sido destruida y vuelta a crear.
<i>onStart()</i>	Indica que la actividad está a punto de ser mostrada al usuario.
<i>onResume()</i>	Se llama cuando la actividad va a comenzar a interactuar con el usuario.
<i>onPause()</i>	Indica que la actividad está a punto de ser lanzada a segundo plano, normalmente porque otra aplicación es lanzada. Es el lugar adecuado para detener animaciones, música o almacenar los datos que estaban en edición.
<i>onStop()</i>	La actividad ya no va a ser visible para el usuario. Si hay muy poca memoria, es posible que la actividad se destruya sin llamar a este método.
<i>onRestart()</i>	Indica que la actividad va a volver a ser representada, después de haber pasado por <i>onStop()</i> .
<i>onDestroy()</i>	Se llama antes de que la actividad sea totalmente destruida, por ejemplo, cuando el usuario pulsa el botón «volver» o cuando se llama al método <i>finish()</i> . Si hay muy poca memoria, es posible que la actividad se destruya sin llamar a este método.

Tabla 3.2: Eventos lanzados por las actividades al cambiar de estado.

Conocer esto bien es de vital importancia para poder crear una aplicación totalmente funcional y que responda adecuadamente según las necesidades del desarrollador y el cliente.

3.2.5. Android Support Library: v7 appcompat

Un punto clave que ha marcado muchas diferencias a la hora de desarrollar este proyecto ha sido la versión mínima de *Android* escogida, siendo esta la 2.2, correspondiente a la *API* 8, comunmente conocida como *Froyo*.

Hasta hace no mucho tiempo, si se quería desarrollar una aplicación unificada que fuera visualmente igual en todas las versiones de *Android*, no había más opción que usar librerías de terceros, como es por ejemplo el caso de *ActionBar Sherlock* para poder usar el *ActionBar*¹² en

¹²Es la barra de título y herramientas que aparece en la parte superior de muchas aplicaciones actuales.

las versiones más antiguas. Sin embargo, esto ha cambiado recientemente, ya que *Google* liberó el paquete *appcompat* que permite, para seguir con el mismo ejemplo, utilizar el *ActionBar* nativo en todas las versiones de *Android* a partir de la 2.2.

Aunque esta acción, ha conseguido facilitar la labor del desarrollador, sigue siendo un aspecto poco trivial. En el proyecto, para solventar problemas de incompatibilidades varias entre versiones, se ha necesitado hacer uso de esta librería, aumentando notablemente la complejidad del proyecto, con vistas y elementos pertenecientes a la librería de soporte y al resto de librerías.

3.2.6. Google Cloud Messaging (*GCM*)

Para poder entender el servicio *GCM*, primero es necesario entender lo que son las **notificaciones push**. Este tipo de notificaciones, no son más que mensajes que una aplicación servidora envía a una aplicación cliente indicándole que tiene algún tipo de información nueva disponible. Lo que distingue a este tipo de notificaciones, de las convencionales, es que el servidor inicia la comunicación, no espera a que el cliente pregunte si hay algo nuevo. La ventaja es la inmediatez: la información llega al cliente (y por tanto al usuario) en cuanto está disponible en el servidor.

GCM, es un servicio gratuito (previo registro en *Google API Console* [36]) que permite que los desarrolladores envíen datos de los servidores a sus aplicaciones de *Android*. Esto se consigue introduciendo un nuevo actor en el proceso, un servidor de mensajería *push* o *cloud to device*, que se situaría entre el servidor de la aplicación y la aplicación móvil. Este servidor intermedio se encargará de recibir las notificaciones enviadas desde el servidor de la aplicación y hacerlas llegar a las aplicaciones móviles instaladas en los dispositivos correspondientes. Para ello, deberá conocer la existencia de ambas aplicaciones, lo que se consigue mediante un “protocolo” bien definido de registros y autorizaciones entre los distintos actores que participan en el proceso que se muestra en la figura 3.5.

Normalmente muestra un icono, el título de la actividad en la que se encuentra el usuario, una serie de botones de acción, y un menú desplegable donde se incluyen más acciones que no tienen espacio para mostrarse como botón o simplemente no se quieren mostrar como tal.

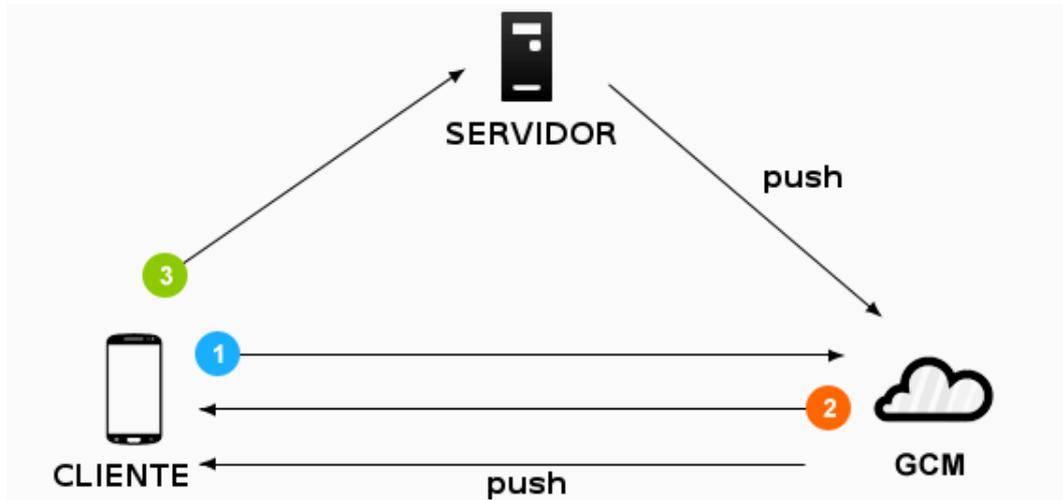


Figura 3.5: Arquitectura de Google Cloud Messaging. Fuente: Android Hive.

1. La aplicación cliente se registra en el servidor *GCM* como cliente capaz de recibir mensajes desde dicho servicio.
2. Si el registro se termina correctamente, el servidor *GCM* enviará al cliente *Android* un código de registro o *Registration ID*, que deberá conservar para las futuras comunicaciones.
3. La aplicación cliente envía el *Registration ID* a el servidor de la aplicación, para que sea guardado.
4. Push servidor-*GCM*: El servidor de la aplicación ya puede enviar un mensaje para que sea recibido por el dispositivo. Para ello, envía primero el mensaje al servidor *GCM*, donde es puesto en cola.
5. Push *GCM*-cliente: El servidor *GCM* envía el mensaje que le ha enviado previamente el servidor de la aplicación, al dispositivo cliente cuando se encuentre accesible.

Este servicio ha sido utilizado en *I-Found@ULPGC* para permitir la comunicación directa entre los diferentes usuarios y para el envío de notificaciones varias.

Capítulo 4

Planificación del trabajo y estimación del coste

En todo proyecto es de vital importancia saber cuantificar un tiempo estimado para el desarrollo total lo más acertado posible. De esta manera, no solamente el cliente se verá satisfecho, sino que el desarrollador o equipo verá cumplidos los plazos establecidos, alcanzando sus objetivos y retroalimentando la motivación del mismo. Una vez obtenida la planificación, se podrá además calcular un presupuesto estimado para el proyecto, pudiendo informar así previamente al potencial cliente y planificar con mayor exactitud la repartición del presupuesto en los diferentes recursos.

En el caso de *I-Found@ULPGC*, se puede comprobar en la gráfica de la figura 4.1, que al tratarse de un proyecto de una cierta complejidad, compuesto por dos subaplicaciones, una cliente y otra servidora, una gran parte del tiempo de trabajo se ha dedicado al desarrollo.

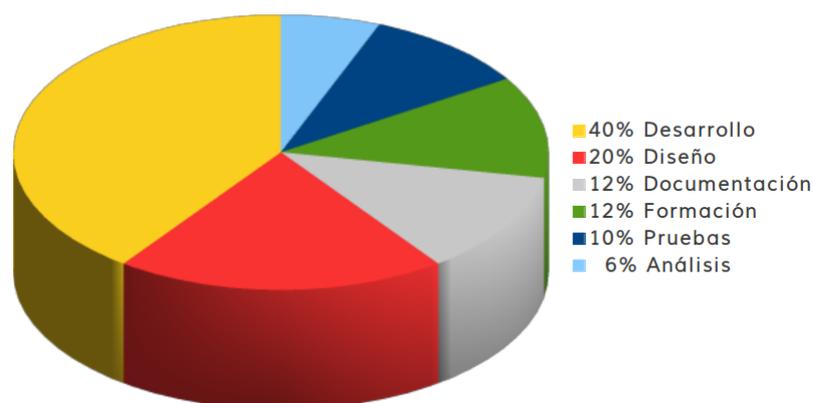


Figura 4.1: Gráfica de la planificación del trabajo.

Además, también puede apreciarse claramente que se ha invertido una gran cantidad de tiempo al análisis y diseño de la misma, puesto que era de prioridad centrar los esfuerzos en el usuario, ya que se quería desarrollar un producto por y para el mismo.

El resto del tiempo se ha dedicado a las pruebas de funcionamiento y con usuarios, la documentación y la formación en *Android*, *Java*, *PHP*, comunicación cliente-servidor, *Mysql* y otras herramientas y tecnologías necesarias derivadas del proceso de aprendizaje de las anteriores.

Partiendo de esta gráfica, se han hecho las estimaciones de coste mostradas en la tabla 4.1.

Licencia	Coste
<i>SCR Screen Recorder Pro</i>	4.50 €
Etapa	Coste
40 % Desarrollo	7.200 €
20 % Diseño	3.600 €
12 % Documentación	2.160 €
12 % Formación	0 €
10 % Pruebas	1.800 €
6 % Análisis	1.080 €
TOTAL	15.844,50 €

Tabla 4.1: Estimación del coste del proyecto.

Se han estimado un total de 15.840 euros (20 por hora), para remunerar las horas trabajadas en este proyecto, partiendo del hecho de que la alumna aún tiene poca experiencia como programadora y gestora de proyectos y que por lo tanto no se han tenido en cuenta las horas de formación. A este coste, se añaden otros 4.50 euros por licencias, un insignificante precio a pagar que además surge por una cuestión muy periférica al proyecto. En total, 15.844,50 euros.

Por otro lado, y como ya se ha mencionado en capítulos anteriores, no se añaden costes de hardware, puesto que la alumna ha facilitado todos los materiales necesarios. Tampoco se incurre en otros costes por software, dado que el resto utilizado es libre y por lo tanto no ha sido necesaria la compra de ningún otro tipo de licencia.

El material gráfico utilizado a lo largo de todo el proyecto también es de naturaleza libre y coste cero.

Capítulo 5

Análisis

Todo proyecto debe comenzar con un correcto estudio previo a la implementación, que defina claramente qué acciones podrán ejecutarse, quién las podrá realizar y cómo será el flujo de funcionamiento de las mismas, sin necesidad de entrar en los detalles propios de la implementación.

Así pues, en este apartado se muestran una serie de pasos previos a la implementación del producto, que ayudarán a comprender, elaborar y mantener correctamente el mismo, posteriormente.

5.1. Especificación de requisitos

Dado que la idea de software que se plantea en este proyecto, no fue propuesta por un cliente real, sino por la alumna, fue necesario salir a la calle a buscar a potenciales usuarios clientes, con el fin de crear un producto con un diseño y funcionalidades, lo más centradas en las necesidades reales de estos, suprimiendo así, la dificultad que supone ser desarrollador y cliente del propio proyecto, evitando de esta manera perder perspectiva del mismo y presuponer detalles que no se deberían.

Para conseguir este objetivo, se decidió seguir una serie de pasos que implicaron:

1. Identificar las aplicaciones con funcionalidades similares a las que se propone en este proyecto, que son aquellas mostradas en la figura 1.4 del apartado 'Estado del arte' de este documento.
2. Identificar las funcionalidades ofrecidas por las mismas.
3. Identificar las funcionalidades consideradas carentes.

4. En base a los pasos anteriores, elaborar una entrevista a modo de *juego de rol*¹ ('Apéndice A'), para un grupo de usuarios elegido, con el fin de detectar los requisitos de los mismos.
5. Estudiar los resultados de la entrevista y elaborar un listado con las funcionalidades a implementar, elaborando así la especificación de requisitos.
6. Elaborar un prototipo interactivo que simulara las anteriores funcionalidades.
7. Evaluar el prototipo interactivo con los usuarios, para comprobar si cumplían los requisitos recogidos en el paso cuatro.

Resueltos los tres primeros pasos, en el apartado 'Estado del arte' de este documento, se creó una entrevista y se entrevistaron, a diez usuarios, hombres y mujeres, que potencialmente podrían usar la aplicación, procedentes todos de diversos sectores profesionales y de edades diferentes, ya que la universidad se compone de un grupo de usuarios de edades muy variables, eso si, partiendo de los diecisiete años. Los datos de interés de dichos entrevistados, pueden verse en la tabla 5.1, refiriéndose la H a hombre y la M a mujer.

Sexo	Edad	Sector profesional
H	59	Administrativo
H	27	Salud
H	26	Técnico
H	17	Estudiante
M	46	Salud
M	41	Idiomas
M	35	Deporte
M	33	Deporte
M	27	IT
M	29	Gastronomía

Tabla 5.1: Relación de usuarios entrevistados.

¹Juego en el que, tal como indica su nombre, uno o más jugadores desempeñan un determinado rol, papel o personalidad. Cuando una persona hace el papel de X significa que está interpretando un papel que normalmente no hace [45].

Como resultado de las entrevistas, se extrajo la siguiente lista de funcionalidades deseadas por los usuarios. En la parte derecha, podemos ver el número de usuarios que hizo referencia a la correspondiente funcionalidad de la izquierda.

Funcionalidad deseada	Nº usuarios
Quiero poder publicar un <i>OPA</i> que haya perdido.	10
Quiero poder publicar un <i>OPA</i> que haya encontrado.	10
Quiero poder publicar un <i>OPA</i> que me hayan robado.	10
Quiero poder buscar si mi <i>OPA</i> ya ha sido encontrado y usar algún filtro que lo facilite.	10
Quiero poder adjuntar una foto desde mi cámara o galería a la publicación.	10
Quiero poder decir la ubicación donde encontré o perdí el <i>OPA</i> .	10
Quiero poder publicar si he perdido a mi mascota.	10
Quiero recibir notificaciones si alguien publica algo parecido a lo que yo haya perdido o me hayan robado.	10
Quiero poder configurar la recepción de mis notificaciones.	10
Me gustaría poder dar una descripción de la publicación.	10
Me gustaría que hubiera un apartado de configuración que tuviera todo centralizado.	10
Me gustaría poder editar mis publicaciones.	10
Me gustaría poder poner la fecha de cuándo perdí o encontré mi <i>OPA</i>	10
Me gustaría que fuera una app bonita y sencilla de usar.	9
Me gustaría poder ver un listado de todas mis publicaciones.	9
No quiero dar mis datos reales de nombre, dirección, teléfono, etc...	8
Quiero poder ofrecer una recompensa si alguien encuentra algo que haya perdido.	8
Quiero poder eliminar mi cuenta.	8
Quiero recibir notificaciones si un usuario me escribe un mensaje.	8
Me gustaría poder eliminar mis conversaciones con otras personas.	8
Quisiera saber de un vistazo, qué mensajes he leído ya y cuales no.	8
Quiero poder hacer mis publicaciones muy rápidamente.	7
Quiero poder comunicarme con otros usuarios directamente en la app, sin tener que usar el e-mail.	7
Me gustaría poder eliminar mis publicaciones.	7
Me gustaría poder marcar las notificaciones leídas.	7
Quiero poder cerrar sesión.	5
Me gustaría diferenciar fácilmente el tipo de publicación que estoy viendo (encontrado, robado o perdido)	4
Me gustaría que la app, me mostrara el sitio oficial (policía, ayuntamiento, oficina de objetos perdidos...) más cercano a donde estoy actualmente ubicado.	2
Me gustaría poder contactar con varios usuarios a la vez.	1

Tabla 5.2: Listado de funcionalidades deseadas por los usuarios.

En base a este complejo estudio y tomando como referencia la experiencia de otras aplicaciones ya existentes en el mercado con anterioridad, se seleccionaron las funcionalidades primarias de la aplicación, a las que se unieron algunas otras propuestas por la alumna, tal y como se puede ver en el siguiente listado:

1. Publicar algo: la pérdida, robo o el haber encontrado un *OPA*.
2. Buscar algo: un *OPA*, perdido, encontrado o robado.
3. Gestionar mensajes: ver , eliminar y escribir nuevos.
4. Gestionar publicaciones: ver, editar y eliminar.
5. Modificar configuración: notificaciones y perfil de usuario.

Además, como requisitos no funcionales, se propusieron los siguientes:

1. La aplicación debe ser adaptable a futuras modificaciones o ampliaciones.
2. La aplicación debe ser compatible para las versiones de *Android* desde la 2.2 en adelante.
3. La aplicación debe ser sencilla y usable.
4. La aplicación debe poder instalarse en la tarjeta SD del dispositivo.

El resto de funcionalidades que no se han podido incluir en esta versión de *I-Found@ULPGC*, pueden encontrarse en el capítulo **10. Trabajo futuro** de este documento.

5.2. Actores

Para poder cumplir con los deseos de permanencia en el anonimato de los potenciales clientes, es necesario ser un usuario registrado y haber iniciado sesión para poder acceder a las distintas funcionalidades de *I-Found@ULPGC*, pero además, cualquier usuario que no cumpla con estas condiciones, tiene que poder tener acceso a los formularios de registro e inicio de sesión. Por otro lado, también se ha comentado en capítulos anteriores, que la app, consiste de dos partes que se comunican, ya que se trata de un producto cliente-servidor. En resumidas cuentas, se pueden definir a los actores como se ve en la tabla 5.3.

Actor	Objetivo
Dispositivo	Hacer las peticiones al servidor que pida el cliente a través de la interfaz.
Servidor	Responder a las peticiones de los dispositivos.
Usuario autenticado	Acceder a las funcionalidades principales de la aplicación.
Usuario	Acceder a la parte pública de la aplicación.

Tabla 5.3: Relación de actores y objetivos.

Partiendo de la tabla 5.3, se ha elaborado el diagrama correspondiente a la figura 5.1 en el que se identifica la relación entre los actores.

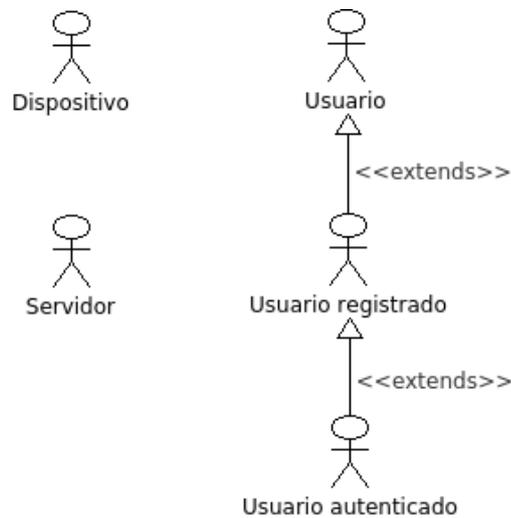


Figura 5.1: Diagrama de actores.

5.3. Casos de uso y diagramas

Se han categorizado los casos de uso en tres ámbitos:

1. **Explotación de contenido:** se refiere a las interacciones de los usuarios con el sistema para obtener resultados e información.
2. **Generación de contenido:** se refiere a las acciones realizadas por determinados usuarios para mejorar o aumentar el contenido de la información disponible en la aplicación.
3. **Sistemas:** entidades necesarias para el correcto funcionamiento del sistema pero no relacionados con la explotación o la generación de contenido.

Los casos de uso van asociados a diagramas de casos de uso, para su mejor comprensión.

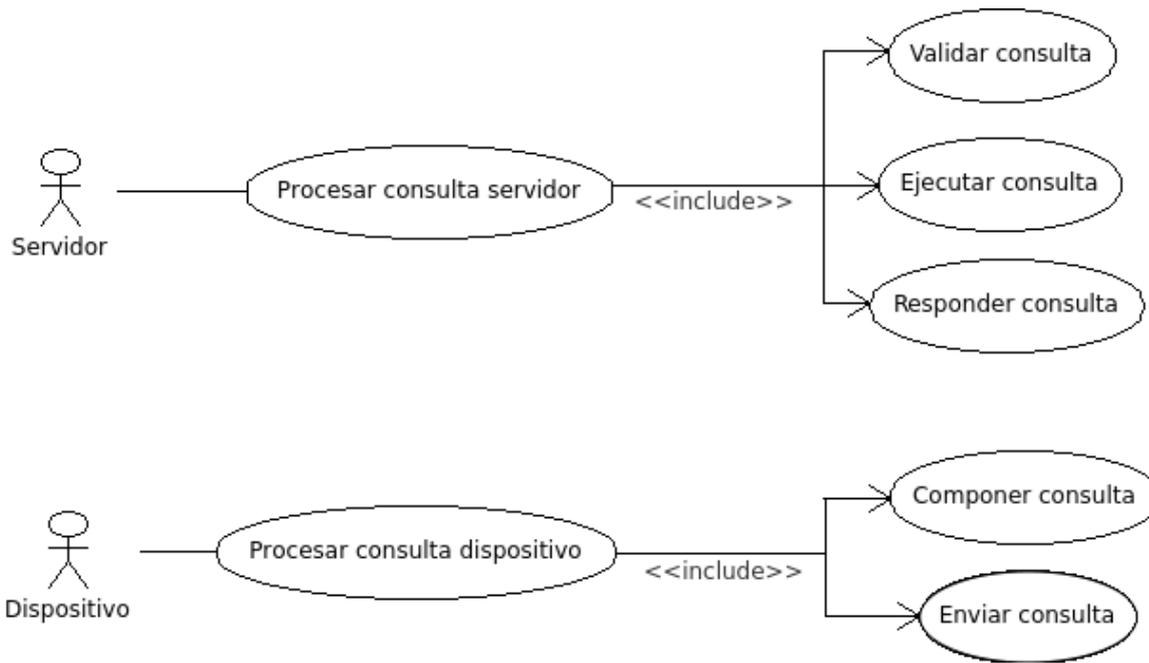


Figura 5.2: Diagrama de comunicación Servidor - Dispositivo.

ID: IFU-01	Validar consulta
Ámbito	Comunicación Servidor – Dispositivo.
Nivel	Sistemas
Descripción	Se comprueba que una consulta o petición al Servidor, esté contemplada y bien formada.
Actor principal	Servidor
Individuos e intereses	Servidor: Responder a la solicitud generada por el Dispositivo.
Precondición	Un Dispositivo ha enviado una consulta y el Servidor la ha recibido correctamente.
Disparador	Un Dispositivo compone y envía una consulta al Servidor.
Flujo normal	<ol style="list-style-type: none"> 1. Un Dispositivo compone y envía una consulta al Servidor. 2. El Servidor recibe correctamente la consulta. 3. El Servidor comprueba que la consulta está contemplada y está bien formada.
Referenciado por	Procesar consulta Servidor.
Diagrama	Diagrama 5.2.

ID: IFU-02	Ejecutar consulta
Ámbito	Comunicación Servidor – Dispositivo.
Nivel	Sistemas
Descripción	Se llevan a cabo las acciones necesarias para atender la consulta enviada por un Dispositivo.
Actor principal	Servidor
Individuos e intereses	Servidor: Generar la información requerida por un Dispositivo.
Precondición	Un Dispositivo ha enviado una consulta, el Servidor la ha recibido correctamente y la ha validado.
Disparador	El Servidor valida una consulta y posteriormente la ejecuta.
Flujo normal	<ol style="list-style-type: none"> 1. Un Dispositivo compone y envía una consulta al Servidor. 2. El Servidor recibe y valida la consulta. 3. El Servidor ejecuta las acciones necesarias para generar la respuesta a una consulta. formada.
Referenciado por	Procesar consulta Servidor.
Diagrama	Diagrama 5.2.

ID: IFU-03	Responder consulta
Ámbito	Comunicación Servidor – Dispositivo.
Nivel	Sistemas
Descripción	Se genera y envía una respuesta válida a una consulta enviada por un Dispositivo.
Actor principal	Servidor
Individuos e intereses	Servidor: Generar y enviar una respuesta ante una consulta enviada por un Dispositivo.
Precondición	La ejecución de la consulta recibida ha terminado correctamente.
Disparador	El Servidor ha ejecutado correctamente una consulta, tras lo cual compone la respuesta y la envía.
Flujo normal	<ol style="list-style-type: none"> 1. El Servidor recibe una consulta y la valida. 2. El Servidor ejecuta correctamente la consulta. 3. El Servidor compone la respuesta y la envía al Dispositivo. 4. El Dispositivo recibe la respuesta.
Referenciado por	Procesar consulta Servidor.
Diagrama	Diagrama 5.2.

ID: IFU-04	Componer consulta
Ámbito	Comunicación Servidor – Dispositivo.
Nivel	Sistemas
Descripción	Se compone una consulta bien formada y contemplada por el Servidor.
Actor principal	Dispositivo
Individuos e intereses	Dispositivo: Generar y enviar una consulta bien formada al Servidor.
Precondición	El Dispositivo necesita obtener datos del Servidor para dar respuesta a un evento.
Disparador	El Dispositivo captura un evento y antes de responder a ese evento, necesita información que debe solicitar al Servidor.
Flujo normal	<ol style="list-style-type: none"> 1. El Dispositivo captura un evento. 2. El Dispositivo compone una consulta válida y bien formada.
Referenciado por	Procesar consulta Dispositivo.
Diagrama	Diagrama 5.2.

ID: IFU-05	Enviar consulta
Ámbito	Comunicación Servidor – Dispositivo.
Nivel	Sistemas
Descripción	Se envía una consulta válida y bien formada desde el Dispositivo a el Servidor.
Actor principal	Dispositivo
Individuos e intereses	Dispositivo: Enviar una consulta al Servidor.
Precondición	<ul style="list-style-type: none"> • El Dispositivo tiene conexión a Internet. • El Dispositivo conoce los protocolos necesarios para comunicarse con el Servidor.
Disparador	El Dispositivo ha generado correctamente una consulta.
Flujo normal	<ol style="list-style-type: none"> 1. El Dispositivo genera una consulta. 2. El Dispositivo se conecta a la red y envía la consulta para que sea recibida por el Servidor.
Referenciado por	Procesar consulta Dispositivo.
Diagrama	Diagrama 5.2.

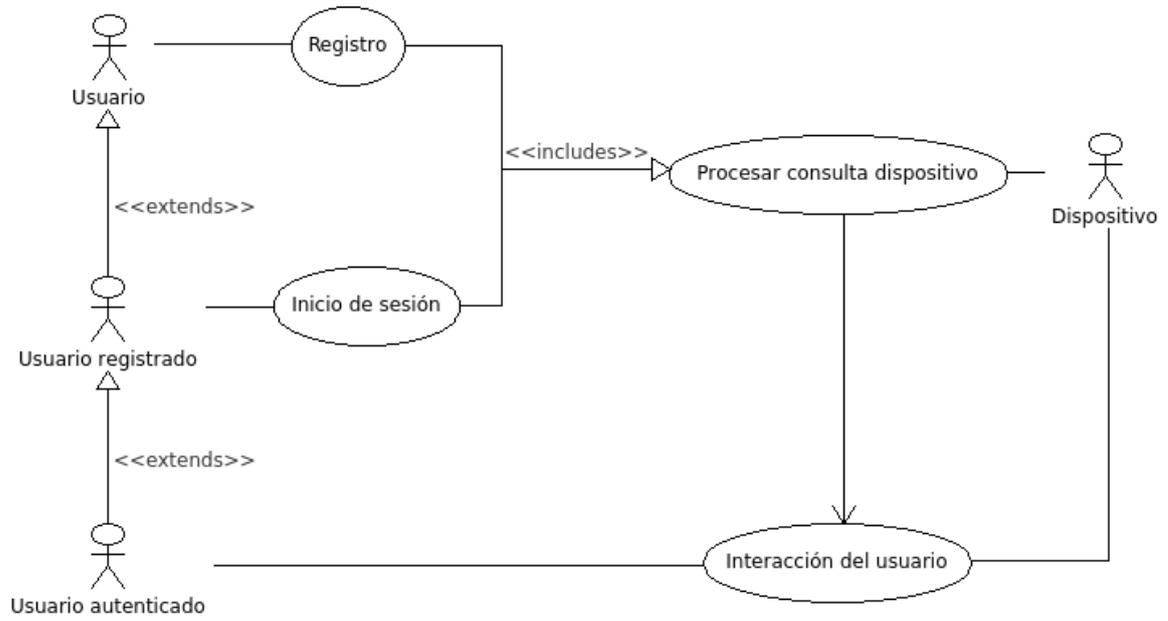


Figura 5.3: Diagrama de registro e inicio de sesión de usuario.

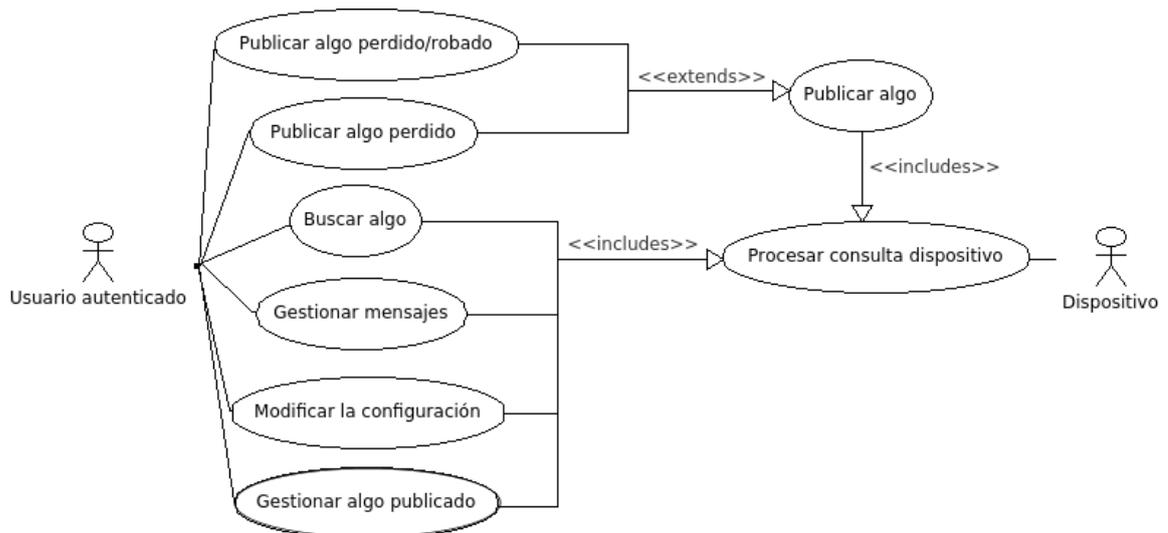


Figura 5.4: Diagrama de interacción del Usuario autenticado.

ID: IFU-06	Registro
Ámbito	Registro e inicio de sesión de Usuario.
Nivel	Sistemas
Descripción	El Usuario se registra e inicia sesión automáticamente en la aplicación.
Actor principal	Usuario
Individuos e intereses	<ul style="list-style-type: none"> ● Dispositivo: Recoger la información de registro (nombre de usuario, contraseña por duplicado e e-mail por duplicado) y enviar la información al Servidor para que la registre y permita al Usuario iniciar sesión. ● Servidor: Recibir la solicitud de registro, iniciar sesión y procesar la solicitud. ● Usuario: Informar al Dispositivo de su nombre de usuario, e-mail, contraseña por duplicado e e-mail por duplicado también, para poder registrarse e iniciar sesión en la aplicación.
Precondición	<ul style="list-style-type: none"> ● El Dispositivo tiene conexión a Internet. ● El Dispositivo conoce los protocolos necesarios para comunicarse con el Servidor. ● El Usuario no está registrado.
Disparador	El Usuario desea hacer uso de la aplicación y para ello, se requiere su registro en la misma.
Flujo normal	<ol style="list-style-type: none"> 1. El Usuario no registrado accede a la aplicación. 2. El Dispositivo muestra una interfaz para recoger la información relativa al registro del usuario: nombre de usuario, contraseña por duplicado e e-mail por duplicado. 3. El Dispositivo compone la consulta correspondiente al registro e inicio de sesión de usuario con la información recogida. 4. El Servidor recibe la consulta, la valida y responde al Dispositivo. 5. El Dispositivo inicia la sesión del Usuario registrado y le permite la interacción con el resto de funcionalidades de la aplicación.
Referenciado por	–
Diagrama	Diagrama 5.3.

ID: IFU-07	Inicio de sesión
Ámbito	Registro e inicio de sesión de usuario.
Nivel	Sistemas
Descripción	El Usuario registrado tras facilitar sus credenciales, inicia sesión en la aplicación.
Actor principal	Usuario registrado
Individuos e intereses	<ul style="list-style-type: none"> • Dispositivo: Recoger la información de autenticación (nombre de usuario y contraseña) y enviar una solicitud de inicio de sesión al Servidor. • Servidor: Recibir la solicitud de inicio de sesión y procesarla. • Usuario registrado: Informar al Dispositivo de su nombre de usuario y contraseña para acceder a las funcionalidades de la aplicación.
Precondición	<ul style="list-style-type: none"> • El Dispositivo tiene conexión a Internet. • El Dispositivo conoce los protocolos necesarios para comunicarse con el Servidor. • El Usuario registrado no está autenticado ya a través del Dispositivo en el Servidor.
Disparador	El Usuario registrado requiere iniciar sesión antes de poder usar cualquier funcionalidad de la aplicación que no sean registro o autenticación.
Flujo normal	<ol style="list-style-type: none"> 1. El Usuario registrado accede a la aplicación. 2. El Dispositivo muestra una interfaz para recoger la información relativa a la autenticación del usuario: nombre de usuario y contraseña. 3. El Dispositivo compone la consulta correspondiente al inicio de sesión con la información recogida. 4. El Servidor recibe la consulta, la valida y responde al Dispositivo. 5. El Dispositivo inicia la sesión del Usuario registrado y le permite la interacción con el resto de funcionalidades de la aplicación.
Referenciado por	–
Diagrama	Diagrama 5.3.

ID: IFU-08	Publicar algo perdido o robado
Ámbito	Interacción del usuario.
Nivel	Generación de contenido.
Descripción	El Usuario autenticado publica un <i>OPA</i> que ha perdido o le ha sido robado, a través de la aplicación.
Actor principal	Usuario autenticado.
Individuos e intereses	<ul style="list-style-type: none"> ● Dispositivo: Mostrar la interfaz al usuario para obtener la información necesaria relacionada con el <i>OPA</i> perdido o robado, además de gestionar y procesar la consulta. ● Servidor: Recibir la consulta y procesarla, almacenando toda la información cedida por el usuario. ● Usuario autenticado: Informar al Dispositivo de los datos necesarios para poder publicar el <i>OPA</i> que ha perdido o le ha sido robado.
Precondición	<ul style="list-style-type: none"> ● El Dispositivo tiene conexión a Internet. ● El Dispositivo conoce los protocolos necesarios para comunicarse con el Servidor.
Disparador	El Usuario autenticado ha perdido o le ha sido robado un <i>OPA</i> y desea publicarlo.
Flujo normal	<ol style="list-style-type: none"> 1. El Usuario autenticado accede a la aplicación. 2. El Dispositivo muestra una interfaz para recoger la información relativa al <i>OPA</i> perdido o robado del Usuario autenticado: categoría (obligatorio), descripción (opcional), foto (opcional), lugar en el que se perdió (obligatorio), fecha en la que pasó (opcional), recompensa ofrecida (opcional), confirmar si el <i>OPA</i> es robado o no. 3. El Dispositivo compone la consulta correspondiente con la información recogida. 4. El Servidor recibe la consulta, la valida y responde al Dispositivo. 5. El Dispositivo muestra una respuesta al Usuario autenticado para mostrar que la publicación se ha realizado correctamente.
Referenciado por	–
Diagrama	Diagrama 5.4.

ID: IFU-09	Publicar algo encontrado
Ámbito	Interacción del usuario.
Nivel	Generación de contenido.
Descripción	El Usuario autenticado publica un <i>OPA</i> que ha encontrado, a través de la aplicación.
Actor principal	Usuario autenticado.
Individuos e intereses	<ul style="list-style-type: none"> • Dispositivo: Mostrar la interfaz al Usuario autenticado, para obtener la información necesaria relacionada con el <i>OPA</i> encontrado, además de gestionar y procesar la consulta. • Servidor: Recibir la consulta y procesarla, almacenando toda la información cedida por el usuario. • Usuario autenticado: Informar al Dispositivo de los datos necesarios para poder publicar el <i>OPA</i> que el usuario ha encontrado.
Precondición	<ul style="list-style-type: none"> • El Dispositivo tiene conexión a Internet. • El Dispositivo conoce los protocolos necesarios para comunicarse con el Servidor.
Disparador	El Usuario autenticado ha encontrado un <i>OPA</i> y desea publicarlo.
Flujo normal	<ol style="list-style-type: none"> 1. El Usuario autenticado accede a la aplicación. 2. El Dispositivo muestra una interfaz para recoger la información relativa al <i>OPA</i> encontrado del Usuario autenticado: categoría (obligatorio), descripción (opcional), foto (opcional), lugar en el que se perdió (obligatorio) y fecha en la que pasó (opcional) 3. El Dispositivo compone la consulta correspondiente con la información recogida. 4. El Servidor recibe la consulta, la valida y responde al Dispositivo. 5. El Dispositivo muestra una respuesta al Usuario autenticado para mostrar que la publicación se ha realizado correctamente.
Referenciado por	–
Diagrama	Diagrama 5.4.

ID: IFU-10	Buscar algo
Ámbito	Interacción del usuario.
Nivel	Generación de contenido.
Descripción	El Usuario autenticado busca un <i>OPA</i> entre todas las publicaciones de los diferentes <i>OPA</i> existentes.
Actor principal	Usuario autenticado.
Individuos e intereses	<ul style="list-style-type: none"> ● Dispositivo: Gestionar la interacción del usuario y procesar la consulta. ● Servidor: Recibir la consulta y la procesarla. ● Usuario autenticado: Interactuar con el Dispositivo buscando el <i>OPA</i> que desea encontrar.
Precondición	<ul style="list-style-type: none"> ● El Dispositivo tiene conexión a Internet. ● El Dispositivo conoce los protocolos necesarios para comunicarse con el Servidor.
Disparador	El Usuario autenticado ha perdido, encontrado, o le ha sido robado un <i>OPA</i> .

Flujo normal	<ol style="list-style-type: none">1. El Usuario autenticado accede a la aplicación.2. El Dispositivo muestra una interfaz con un listado de todas las publicaciones disponibles, por orden de llegada y en los que se puede ver de forma resumida, la información que se introdujo en el momento de su publicación, además del elemento destacado en el caso de que sea nuevo e identificado según sea un <i>OPA</i> robado, encontrado o perdido.3. Si el Usuario autenticado lo desea, el Dispositivo muestra una nueva interfaz para cada publicación del listado para poder ver toda la información de la misma en detalle y que además permite contactar con el usuario que lo ha publicado.4. Si el Usuario autenticado lo desea, el Dispositivo muestra una nueva interfaz para permitir el filtrado de publicaciones según: categoría del <i>OPA</i>, lugar en el que se encontró, perdió o robó, fecha en la que ocurrió y tipo de <i>OPA</i> (perdido, encontrado o robado).5. El Dispositivo compone la consulta correspondiente con la información recogida.6. El Servidor recibe la consulta, la valida y responde al Dispositivo.7. El Dispositivo muestra una respuesta al Usuario autenticado para mostrar lo que desea.
Referenciado por	–
Diagrama	Diagrama 5.4.

ID: IFU-11	Gestionar mensajes
Ámbito	Interacción del usuario.
Nivel	Explotación de contenido.
Descripción	El Usuario autenticado responde, envía o elimina un mensaje.
Actor principal	Usuario autenticado.
Individuos e intereses	<ul style="list-style-type: none"> ● Dispositivo: Gestionar la interacción del usuario y procesar la consulta. ● Servidor: Recibir la consulta y procesarla. ● Usuario autenticado: Informar al dispositivo de los datos necesarios para poder enviar o eliminar un mensaje.
Precondición	<ul style="list-style-type: none"> ● El Dispositivo tiene conexión a Internet. ● El Dispositivo conoce los protocolos necesarios para comunicarse con el Servidor.
Disparador	Algún usuario se ha puesto en contacto con el Usuario autenticado , o el Usuario autenticado necesita ponerse en contacto con algún usuario o eliminar algún mensaje.
Flujo normal	<ol style="list-style-type: none"> 1. El Usuario autenticado accede a la aplicación. 2. El Dispositivo muestra una interfaz para que el Usuario autenticado pueda seleccionar el mensaje a responder o eliminar, o el usuario al que enviarle el nuevo mensaje. 3. El Dispositivo compone la consulta correspondiente con la información recogida. 4. El Servidor recibe la consulta, la valida y responde al Dispositivo. 5. El Dispositivo muestra una respuesta al Usuario autenticado para mostrar que su mensaje ha sido enviado correctamente.
Referenciado por	–
Diagrama	Diagrama 5.4.

ID: IFU-12	Modificar configuración
Ámbito	Interacción del usuario.
Nivel	Explotación de contenido.
Descripción	El Usuario autenticado modifica la configuración de la aplicación.
Actor principal	Usuario autenticado .
Individuos e intereses	<ul style="list-style-type: none"> • Dispositivo: Gestionar la interacción del Usuario autenticado y procesar la consulta. • Servidor: Recibir la consulta y la procesarla. • Usuario autenticado: Informar al Dispositivo de las modificaciones necesarias en la configuración.
Precondición	<ul style="list-style-type: none"> • El Dispositivo tiene conexión a Internet. • El Dispositivo conoce los protocolos necesarios para comunicarse con el Servidor.
Disparador	El Usuario autenticado desea modificar la configuración de la aplicación.
Flujo normal	<ol style="list-style-type: none"> 1. El Usuario autenticado accede a la aplicación. 2. El Dispositivo muestra una interfaz a través del menú contextual, para que el Usuario autenticado pueda modificar la configuración de la aplicación, con respecto a la recepción de notificaciones y al perfil del usuario. 3. El Dispositivo compone la consulta correspondiente con la información recogida. 4. El Servidor recibe la consulta, la valida y responde al Dispositivo. 5. El Dispositivo muestra una respuesta al Usuario autenticado para mostrar que sus cambios han sido guardados.
Referenciado por	–
Diagrama	Diagrama 5.4.

ID: IFU-13	Gestionar algo publicado
Ámbito	Interacción del usuario.
Nivel	Generación de contenido.
Descripción	El Usuario autenticado edita o elimina una publicación.
Actor principal	Usuario autenticado .
Individuos e intereses	<ul style="list-style-type: none"> ● Dispositivo: Gestionar la interacción del Usuario autenticado y procesar la consulta. ● Servidor: Recibir la consulta y procesarla. ● Usuario autenticado: Informar al Dispositivo de las modificaciones en la publicación.
Precondición	<ul style="list-style-type: none"> ● El Dispositivo tiene conexión a Internet. ● El Dispositivo conoce los protocolos necesarios para comunicarse con el Servidor.
Disparador	El Usuario autenticado ha cometido un error en alguna publicación y desea enmendarlo, o desea eliminarla para siempre.
Flujo normal	<ol style="list-style-type: none"> 1. El Usuario autenticado accede a la aplicación. 2. El Dispositivo muestra una interfaz para que el Usuario autenticado pueda modificar la publicación, permitiéndole eliminarla o editarla. 3. El Dispositivo compone la consulta correspondiente con la información recogida. 4. El Servidor recibe la consulta, la valida y responde al Dispositivo. 5. El Dispositivo muestra una respuesta al Usuario autenticado para mostrar que sus cambios han sido guardados.
Referenciado por	–
Diagrama	Diagrama 5.4.

Capítulo 6

Diseño

A lo largo de este capítulo se mostrarán con detalle las distintas fases por las que pasó este proceso a lo largo del proyecto, pasando por el diseño de la interfaz humana, el de las bases de datos del servidor y local a la aplicación, y de la arquitectura del sistema en su totalidad.

6.1. Diseño de la interfaz de usuario

En base al listado de casos de uso del capítulo 5. **Análisis**, producto de las entrevistas realizadas a los potenciales usuarios, se creó un prototipo de la interfaz del usuario. Este prototipo, permite refinar los requerimientos de la aplicación o identificar casos de uso aún no descubiertos, además de definir la interfaz de usuario y servir de prueba de concepto de la implementación, antes de integrar todas las ideas sobre la misma.

6.1.1. Prototipo interactivo

Para la elaboración del prototipo se usó una aplicación denominada *Pencil* [28], libre y gratuita, que permite diseñar *mockups*¹ con poco esfuerzo, además de la creación de un prototipo interactivo mediante el uso de enlaces.

Se pueden ver a continuación algunas imágenes correspondientes al prototipo interactivo. La versión completa del mismo se encuentra en la carpeta *PSC/prototipo/* del CD adjunto a este documento escrito. Para poder comenzar a utilizarlo, basta simplemente con abrir el archivo *index.html* en cualquier navegador disponible.

¹Es un modelo a escala o tamaño real de un diseño o un dispositivo, utilizado para la demostración, evaluación del diseño, promoción, y para otros fines. Es un prototipo, si proporciona al menos una parte de la funcionalidad de un sistema y permite pruebas del diseño.



Figura 6.1: Pantallas del prototipo de: inicio de sesión y registro (a), principal (b) y de publicación de pérdida de un OPA (c).



Figura 6.2: Pantallas de: publicación de haber encontrado un OPA (a), mensajes y notificaciones del Sr. Found (b) y almacén (c).

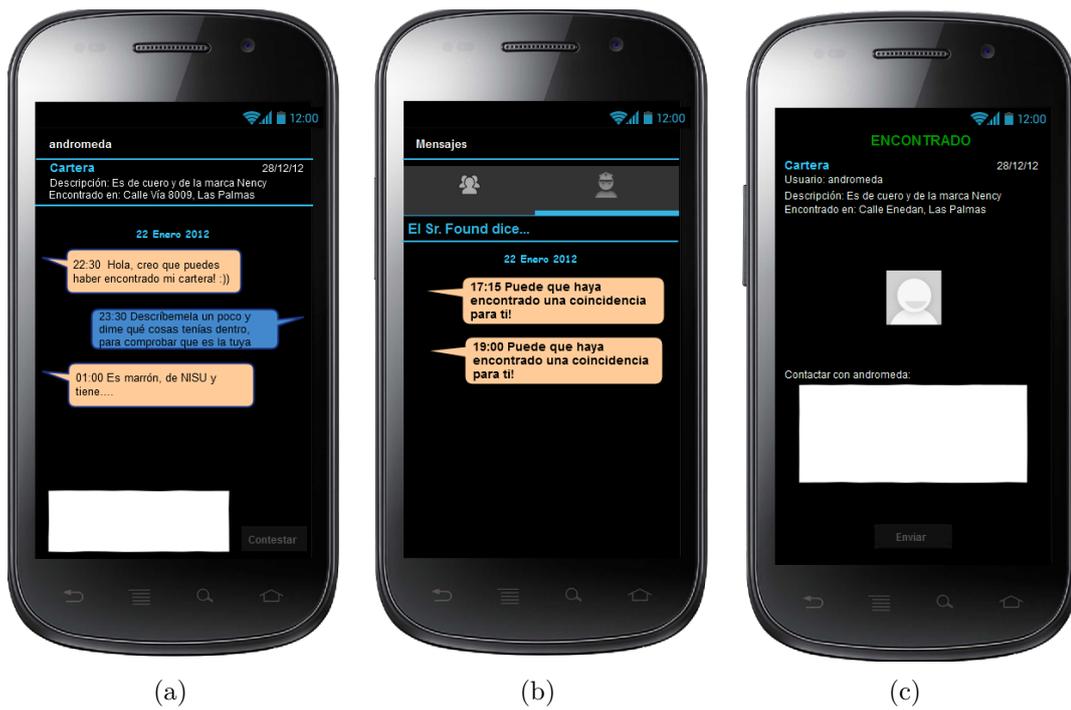


Figura 6.3: Pantallas de: detalle de un chat (a), notificaciones del Sr. Found (b) y de detalle de un OPA (c).

6.1.2. Evaluación del prototipo

Una vez creado el prototipo, se sometió a pruebas con los potenciales usuarios clientes, algunos de los cuales aceptaron y dieron su permiso para ser grabados durante el proceso, utilizando para ello la entrevista anexada al proyecto en el 'Apéndice B'. Los vídeos fueron de gran utilidad para comprobar los comportamientos de los usuarios. Pueden ser encontrados en el CD adjunto a este documento, en el directorio *PSC/prototipo/videos/*.

Tras validar el prototipo y comprobar con satisfacción que cumplía en gran medida con las expectativas de todos los usuarios, se realizaron pequeños ajustes que fueron reevaluados mediante un segundo prototipo, y comenzó la fase de **Diseño final de la interfaz de usuario**.

6.1.3. Diseño final de la interfaz de usuario

En base a las evaluaciones del prototipo, la colaboradora del proyecto, de diseño gráfico, creó un esquema de diseño para el proyecto, dió orientación para mejorar el prototipo y además aportó la tipografía y esquema de colores que se usarían a lo largo del mismo. El esquema aportado puede verse en la imagen 6.4.

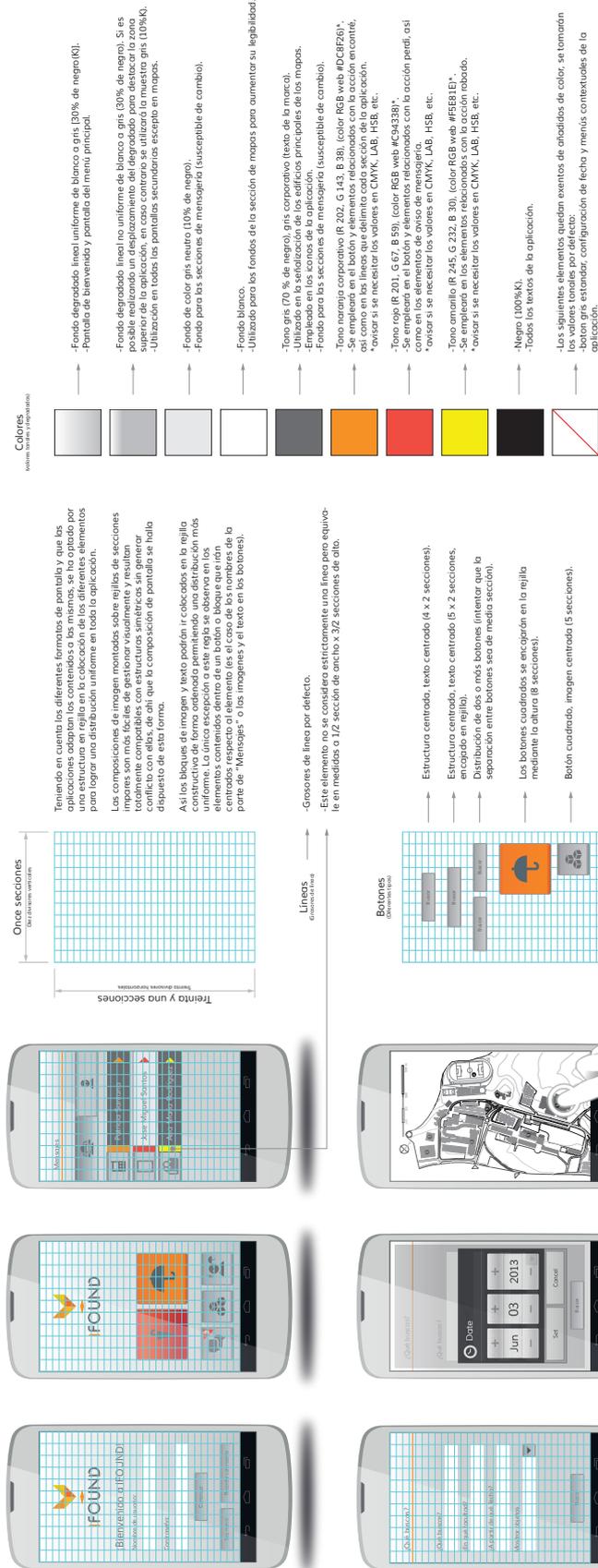


Figura 6.4: Esquema de diseño. Fuente: Ana Rodríguez Camacho.

Atendiendo al esquema propuesto por Ana, la colaboradora de diseño gráfico, y los resultados de las evaluaciones del prototipo, se creó el diseño de la interfaz de usuario, para la primera versión de *I-Found@ULPGC*. El resultado completo en imágenes puede verse en el capítulo **7. Resultado y manual de usuario** de este documento. La aplicación completa puede encontrarse en el directorio *PSC/fuente/* del CD.

El *Sr. Found*: humanizando el software

Hay una idea que se ha venido repitiendo a la largo de este documento y es la de hacer el acceso a *I-Found@ULPGC* lo más sencillo, rápido y eficaz posible. Como apoyo a esta idea que se persigue se decidió introducir un elemento al proyecto. Un elemento nuevo, del que no disponen tampoco sus competidores: un rol que humanizara el software. Aparece pues de esta manera el *Sr. Found*, un agente virtual que se encargará, bajo los ojos de los usuarios, de organizar el almacén de la aplicación, siendo él mismo el que envíe las notificaciones a los usuarios cuando se encuentre cualquier coincidencia con un *OPA* que un usuario haya perdido o le haya sido robado.

Además, si un usuario no recuerda su contraseña de acceso, puede hacer uso de la funcionalidad «olvidé la contraseña» para recuperarla, enviándosele así desde el sistema, un e-mail con una contraseña temporal, para que el usuario pueda volver a acceder a *I-Found@ULPGC* para cambiarla por otra deseada. El e-mail anteriormente mencionado, tendrá también como remitente al Sr. Found, el agente virtual de la aplicación.

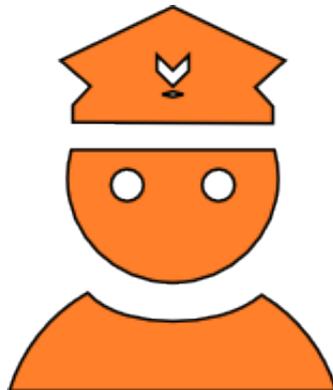


Figura 6.5: El Sr. Found.

La idea tras la humanización del software no es nueva, y hay estudios que avalan su importancia, como el realizado por varios profesores de la Universidad de Murcia (UMU), la Universidad Carlos III de Madrid y la Universidad Complutense de Madrid y cuyo título es '*Using the Affect Grid to Measure Emotions in Software Requirements Engineering*' [40].

Los iconos

Todos los iconos utilizados en *I-Found@ULPGC* son libres y gratuitos. Algunos de ellos han sido recopilados de almacenes libres de imágenes y usados en su forma original, mientras que otros han sido adaptados para su uso en el proyecto, o creados totalmente nuevos. Todos los archivos originales, sus modificaciones y licencias concretas, pueden ser encontrados en el CD adjunto a este documento, en el directorio *PSC/iconos/*.

6.1.4. Las pruebas

Para poder comprobar la satisfacción de los potenciales usuarios con respecto a la primera versión del producto final, se realizó una pequeña entrevista ('Apéndice C'), con la que para sorpresa de la alumna, se obtuvieron muy gratos resultados. Los usuarios se mostraron muy contentos especialmente con la simplicidad y rapidez del manejo de *I-Found@ULPGC*. El diseño les pareció agradable y además, vieron cubiertas todas sus expectativas.

6.2. Diseño de la arquitectura de sistemas

I-Found@ULPGC, sigue una arquitectura cliente-servidor, tal y como la mostrada en la figura 6.6. Esto quiere decir que el trabajo de la misma está compuesto por dos subsistemas, es decir, dos programas independientes que trabajan conjuntamente para conseguir una finalidad: el servidor, que centraliza el servicio, y el cliente, que se encarga de la interacción con el usuario. Además, en este caso, y tal y como se explica en el capítulo 3. **Recursos y tecnologías** de este documento, se ha hecho uso de los servicios del servidor de *GCM* para poder implementar la funcionalidad de notificaciones push para los clientes.

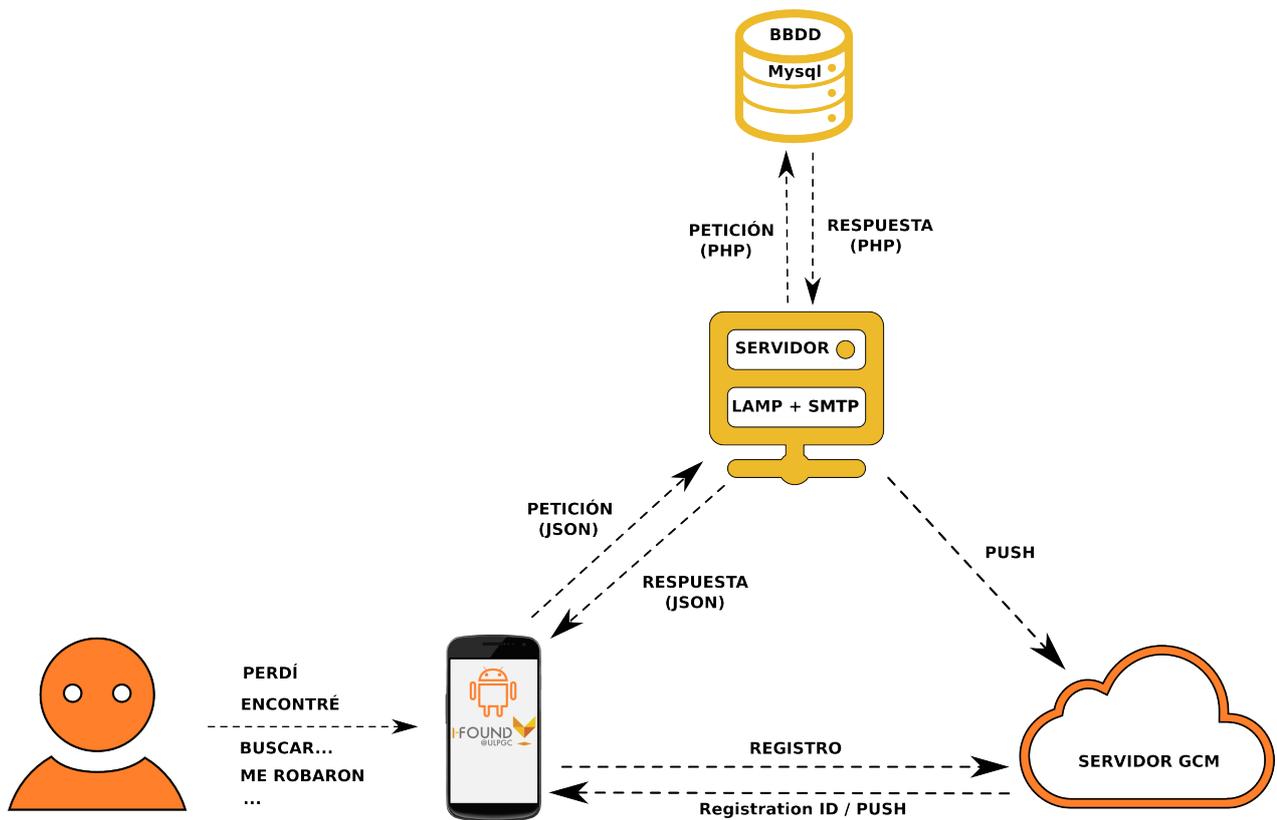


Figura 6.6: Arquitectura de sistemas de *I-Found@ULPGC*.

6.2.1. El servidor

El subsistema Servidor es responsable de ofrecer y gestionar una interfaz que proporcione a sus clientes, los dispositivos, un medio para acceder a la información. En este caso, se ha decidido utilizar una infraestructura de tipo *LAMP* con *GNU/Linux*, *Apache*, *MySql* y *PHP*.

Este subsistema también aloja la base de datos que contiene toda la información de *I-Found@ULPGC* y sus usuarios, a la que se accede mediante peticiones en lenguaje *MySql* a través de *PHP*.

Para la comunicación entre los dispositivos y el servidor, se ha elegido el protocolo *JSON*². El uso de este protocolo implica que ambos extremos intercambiarán mensajes escritos en *JSON*, usando *HTTP (Hypertext Transfer Protocol)* como protocolo de transporte. Se ha elegido este protocolo de comunicación, debido a su gran sencillez de manejo y lectura y a la existencia de implementaciones tanto para *Android*, como para otras plataformas, objetivo de trabajo futuro para este proyecto.

Además, en este subsistema se ha activado un protocolo para la transferencia simple de correo electrónico (*SMTP*), para poder así enviar correos a los usuarios, bajo petición de los mismos, en el caso de solicitar recordar su contraseña de acceso a la aplicación.

6.2.2. El cliente

El subsistema *Cliente*, es el responsable de identificar y procesar las consultas de los usuarios, solicitando al subsistema *Servidor* la información que sea necesaria para completar cada petición.

El lenguaje de programación utilizado para este subsistema ha sido *Java* y el entorno en el que se ha desarrollado, ha sido *Eclipse*, previa instalación del plug-in *ADT (Android Development Tools)*, para el mismo entorno, que permite el desarrollo de aplicaciones *Android* usando este *framework*. Además, este plug-in dispone de una funcionalidad de emulación, a través de los *Android Virtual Device (AVD)*, por medio de la cual se ha podido probar la aplicación para diferentes arquitecturas y versiones de *Android*.

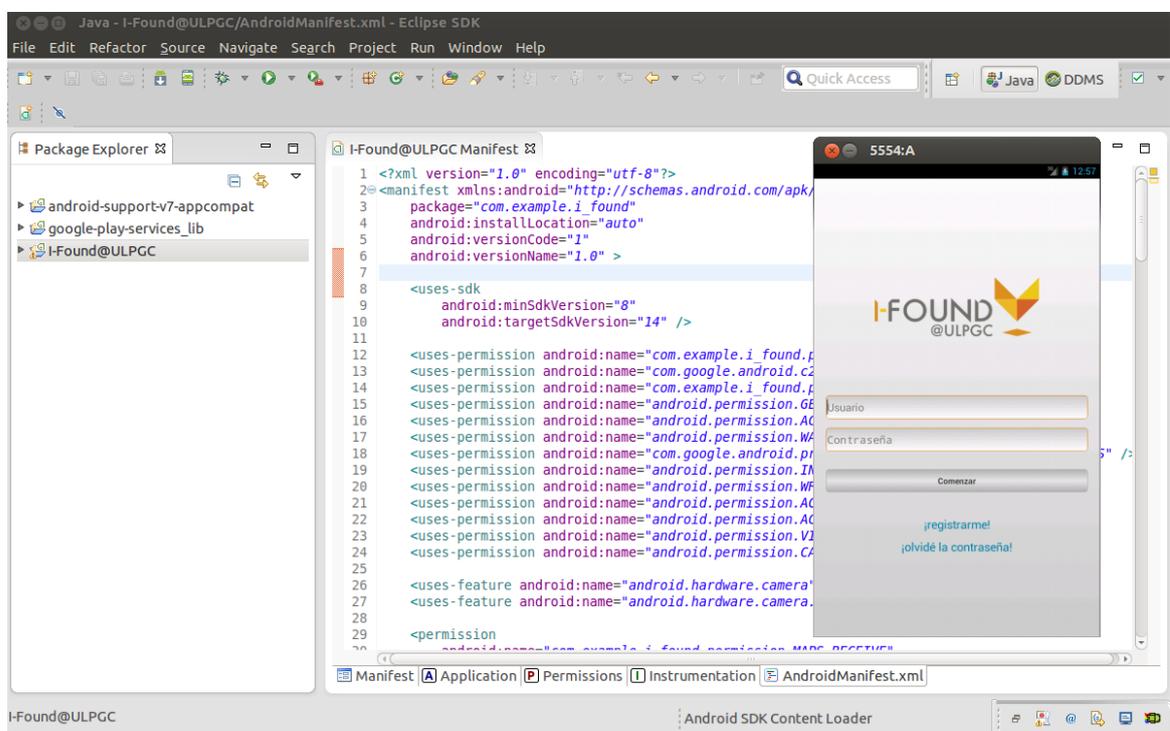


Figura 6.7: Eclipse y AVD en funcionamiento.

²Es un formato ligero para el intercambio de datos, subconjunto de la notación literal de objetos de *JavaScript* que no requiere el uso de *XML*.

6.3. Diseño de la base de datos

6.3.1. El servidor

Como sistema de gestión de base de datos, se decide usar *MySQL*³, principalmente debido a su amplia documentación y a su compatibilidad con el lenguaje seleccionado para desarrollar el Servidor, *PHP*.

En la figura 6.8, se puede ver el diagrama entidad-relación configurado para este proyecto.

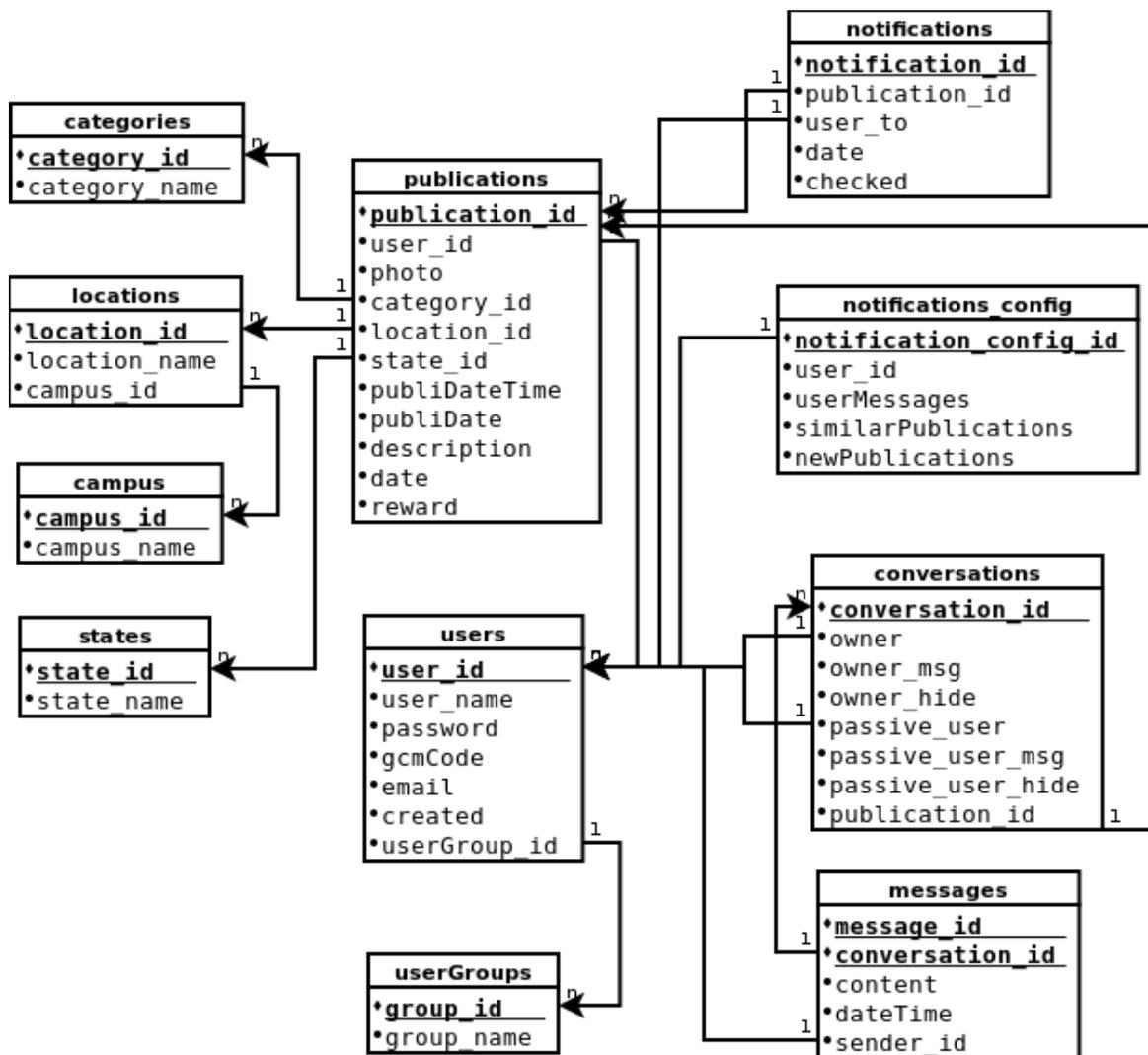


Figura 6.8: Diagrama entidad-relación de la base de datos.

³Sistema de gestión de bases de datos de tipo relacional. Se ofrece bajo licencia GNU GPL.

6.3.2. El cliente

Para poder alojar información necesaria de manera local en el cliente *Android* ha sido necesaria, por un lado, la creación de una base de datos en *SQLite*, para guardar detalles del usuario, el código *GCM*, configuración y sesión entre otros, y por otro lado, la implementación de la clase *PreferenceActivity*, asociada a un archivo *XML*, para almacenar las preferencias del usuario logueado, en cuanto a la aplicación.

SQLite

SQLite es un motor de bases de datos muy popular, debido principalmente a su pequeño tamaño, que no necesita servidor, precisa de poca configuración, es transaccional y de código libre.

Además, *Android* incorpora todas las herramientas necesarias para la creación y gestión de bases de datos *SQLite*, y entre ellas, una *API* para llevar a cabo de manera sencilla todas las tareas necesarias. En este sistema, la forma en la que se crea, actualiza, y conecta con una base de datos *SQLite* es a través de la clase auxiliar *SQLiteOpenHelper*.

En este caso, solo se ha necesitado el apoyo de dos tablas para la base de datos local. Se puede ver su configuración en la figura 6.9.

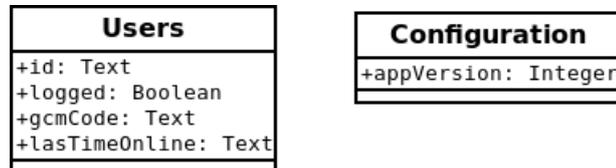


Figura 6.9: Diagrama de la base de datos *SQLite* en el cliente.

PreferenceActivity

Es un tipo de actividad. Su uso facilita enormemente el trabajo del programador para la gestión y creación de la pantalla de preferencias o configuración de un proyecto. Para ello, el programador solamente necesita implementar esta clase y pasarle un archivo *XML* con el diseño deseado. Esta actividad se encargará de forma casi autónoma de guardar todas las preferencias elegidas por el usuario de forma automática en un fichero creado por ella misma también.

Capítulo 7

Resultado y manual de usuario

A lo largo de este capítulo, el lector podrá ver en imágenes el resultado completo y final del desarrollo de *I-Found@ULPGC*. Las imágenes van acompañadas de textos, que de forma muy detallada explican el uso de la aplicación, de tal forma que el usuario pueda comenzar a utilizarla cuando desee.

7.1. Inicio de la aplicación

Con el dispositivo encendido, acceder al panel de aplicaciones y pulsar y soltar sobre el icono de la nueva aplicación instalada, *I-Found@ULPGC*.

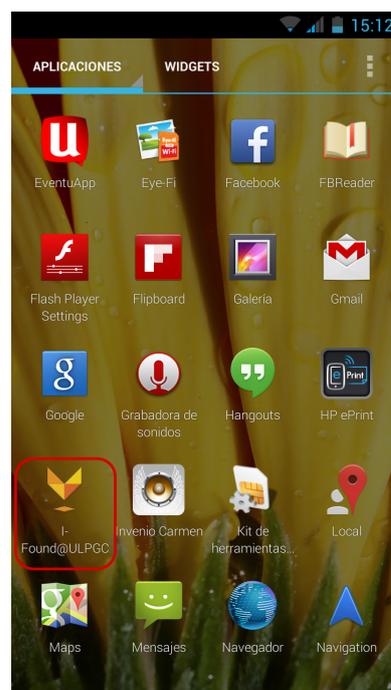


Figura 7.1: Pantalla de aplicaciones.

7.2. Inicio de sesión, registro y recuperación de contraseña

La primera pantalla mostrada nada más abrir la aplicación, permite al usuario las siguientes acciones:

- **Iniciar sesión**, en el caso de que el usuario ya esté registrado en la aplicación. En esta pantalla, el usuario deberá ingresar su nombre de usuario y contraseña. En el caso de que el nombre de usuario no exista o la contraseña sea incorrecta, se mostrará un error al usuario.

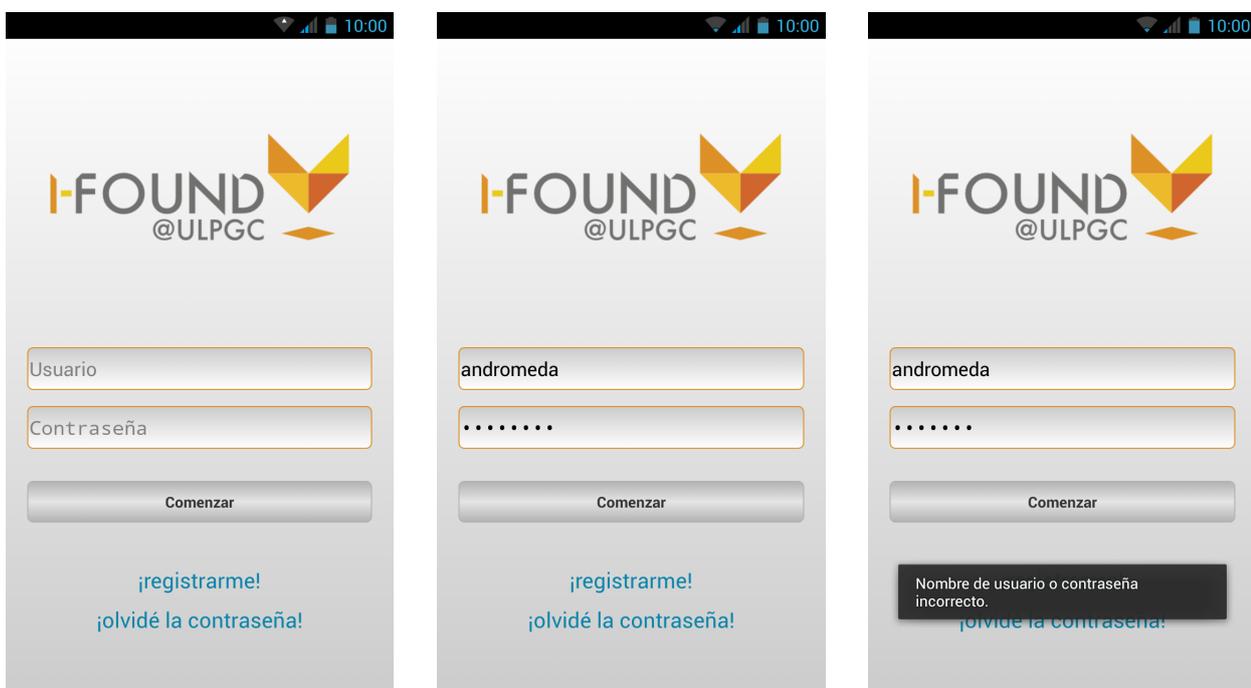


Figura 7.2: Pantallas de inicio de sesión.

- **Registrarse**, en el caso de que el usuario no esté registrado en la aplicación. En esta pantalla, el usuario deberá introducir su potencial nombre de usuario, la contraseña, y el e-mail, estas dos últimos por duplicado, para validar la correcta escritura de ambos. Si las contraseñas o los e-mails no coinciden, o estos últimos no son válidos, se mostrará un error al usuario. Además, en este paso se presenta el *Sr. Found* al usuario, en la última pantalla tras el registro.

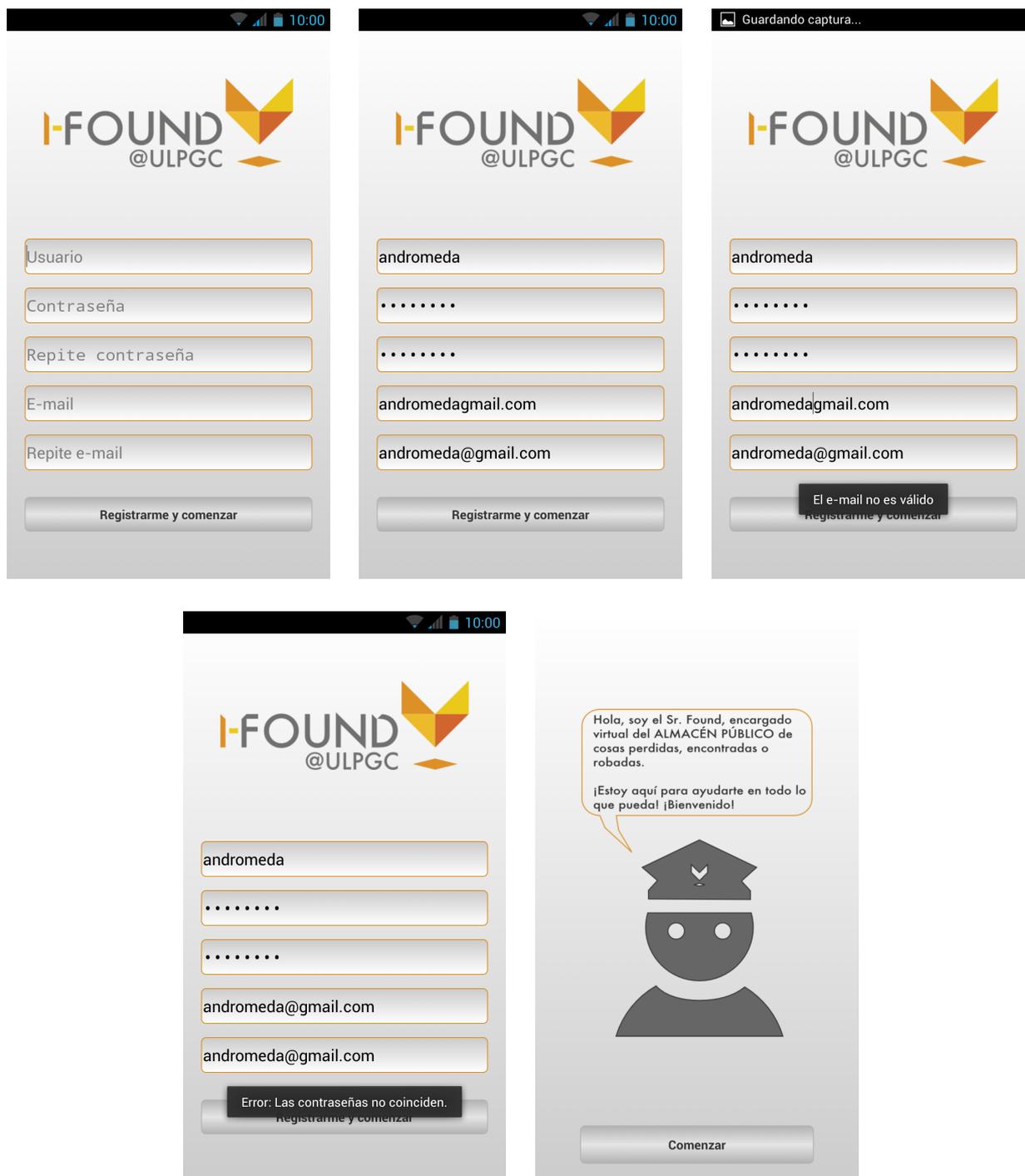


Figura 7.3: Pantallas de registro de usuario.

- **Recuperar su contraseña**, en el caso de que el usuario esté registrado en la aplicación y haya olvidado su clave de acceso. Para ello, el usuario deberá introducir su nombre de usuario y pulsar sobre el enlace «¡olvidé la contraseña!». Si el nombre de usuario es correcto y existe actualmente en la base de datos de *I-Found@ULPGC*, el sistema creará una clave de acceso temporal, que será enviada por e-mail (el introducido durante el registro) al usuario de tal forma que pueda volver a acceder a la aplicación, y desde

ahí cambiar su contraseña por otra que desee. Esto último podrá hacerlo a través de la pantalla de «Configuración», a la que se puede acceder desde los menús de navegación contextuales disponibles en todas las pantallas de la aplicación. A dichos menús, el usuario podrá acceder mediante el botón «menú» del dispositivo, o pulsando y soltando sobre el icono de *I-Found@ULPGC* situado en la parte superior izquierda de cada pantalla. En el caso de que el nombre de usuario no existiera en la base de datos, o el e-mail con la contraseña temporal no pudiera enviarse, se mostrará al usuario un mensaje de error. En caso contrario, un mensaje de aviso de que se le ha enviado el e-mail anteriormente mencionado.

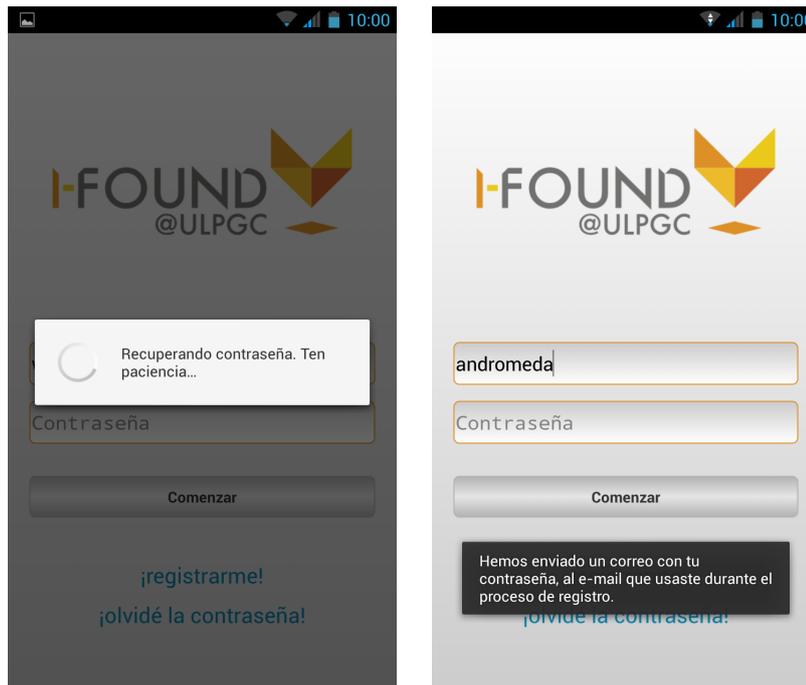
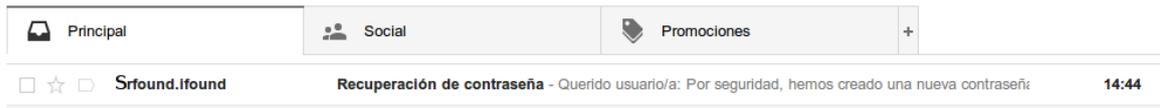


Figura 7.4: Pantallas de recuperación de contraseña.

Tanto el contenido del e-mail, como la vista del mismo en la bandeja de entrada del usuario al recibirlo (esto es dependiente de la aplicación gestora de e-mails que se use), pueden verse en las figuras 7.5(a)(b).

7.2. Inicio de sesión, registro y recuperación de contraseña



(a)



(b)

Figura 7.5: Recepción de e-mail de recuperación de contraseña.

7.3. Pantalla principal

Esta es la pantalla principal de la aplicación y desde la cual se puede acceder a toda su funcionalidad.

El usuario podrá visualizar esta pantalla tras iniciar la aplicación si no hubiera cerrado sesión la última vez que la utilizó, o accediendo a ella a través de cualquiera de los menús contextuales de navegación disponibles en cualquiera de las demás pantallas.



Figura 7.6: Pantalla principal de I-Found@ULPGC.

7.4. Navegación

Para poder acceder a cualquier funcionalidad de la aplicación, a partir de cualquier pantalla, el usuario puede usar los menús contextuales de navegación. Para acceder a ellos, debe pulsar sobre el botón físico de «menú» del dispositivo, o sobre el icono de *I-Found@ULPGC* situado en la parte superior izquierda de todas las pantallas.

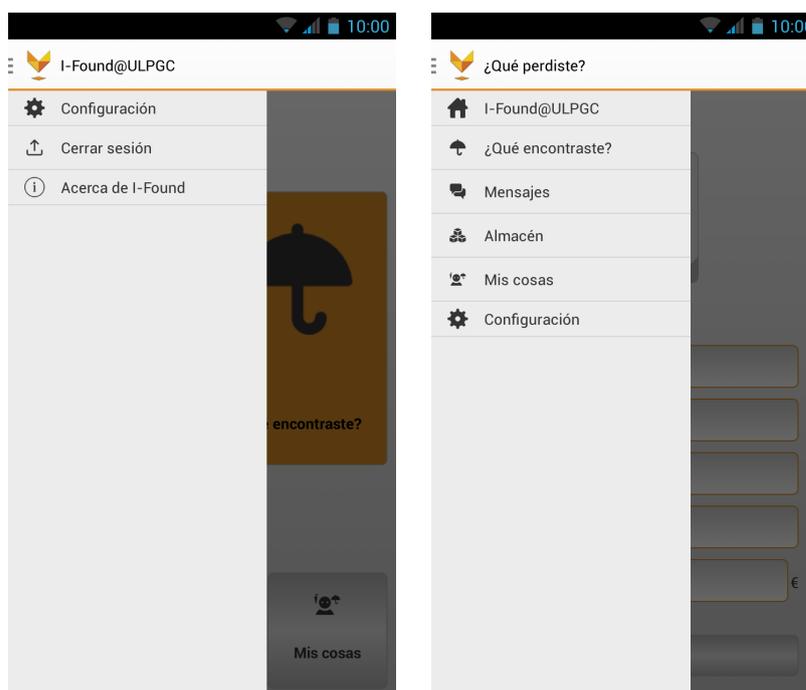


Figura 7.7: Menús contextuales de navegación en diferentes pantallas.

7.5. Publicar

Una publicación es todo aquel anuncio sobre cualquier *OPA* que haya sido encontrado, perdido o robado y que un usuario registrado de la aplicación quiera hacer público a través de *I-Found@ULPGC*.

Para poder publicar, el usuario puede usar los botones «¿Qué perdiste?», para un *OPA* perdido o robado, y «¿Qué encontraste?», para un *OPA* encontrado, ambos situados en la pantalla principal de la aplicación.

7.5.1. ¿Qué perdiste?

Pulsando sobre el botón «¿Qué perdiste?» situado en la pantalla principal, el usuario podrá acceder a esta funcionalidad, y publicar la pérdida o robo de un *OPA*. Para ello, el usuario

dispone de los siguientes campos:

1. **Adjuntar una foto a la publicación.** Esto puede hacerlo a través del icono de cámara fotográfica, cuadrado, de la parte superior de la pantalla. Por medio de él, podrá hacer una foto nueva, o elegir una ya disponible de su galería. Si por el contrario, el usuario no adjuntara ninguna imagen nueva de la cámara o de su galería, el cuadro mostrará una imagen que represente la categoría elegida en el siguiente punto por el usuario. Este campo es opcional y puede verse en detalle, en la figura 7.8(c).
2. **Elegir una categoría,** a través del listado mostrado al pulsar sobre el cuadro de texto situado a la derecha de «¿Qué perdiste?». Actualmente, las categorías disponibles son: animal, cartera, persona, transporte, bolso, llaves, libros, reproductor, smartphone, ropa y otro. Este campo es obligatorio, y así se indica con el carácter '*' mostrado en el interior del cuadro de texto anteriormente mencionado. Puede verse con detalle en la figura 7.8(d).
3. **Añadir una descripción,** de un máximo de 140 caracteres, sobre el *OPA*. Este campo es opcional.
4. **Elegir una ubicación,** a través del listado mostrado al pulsar sobre el cuadro de texto situado a la derecha de «¿Dónde fue?». Actualmente, las localizaciones disponibles, son todos los campus y edificios oficiales de la *ULPGC*. Este campo es obligatorio, y así se indica con el carácter '*' mostrado en el interior del cuadro de texto anteriormente mencionado. Puede verse con detalle en las figuras 7.8(e)(f).
5. **Elegir la fecha,** en la que sucedió la pérdida o robo. Este campo es opcional.
6. **Ofrecer una recompensa,** a la persona que encuentre el *OPA*. Este campo, por supuesto, es opcional.
7. **Declarar si el *OPA* fue robado.** En este caso, el usuario no tiene más que pulsar sobre el checkbox destinado a tal finalidad. Este campo también es opcional para el usuario.

Una vez rellenados los campos, al menos los obligatorios, el usuario podrá hacer pública su publicación, pulsando sobre el botón «Publicar». Seguidamente, el usuario será redirigido a la pantalla del 'Almacén', en la que podrá ver su publicación con el estado de **¡Nuevo!**, comprobando así que la publicación se ha hecho de manera correcta.

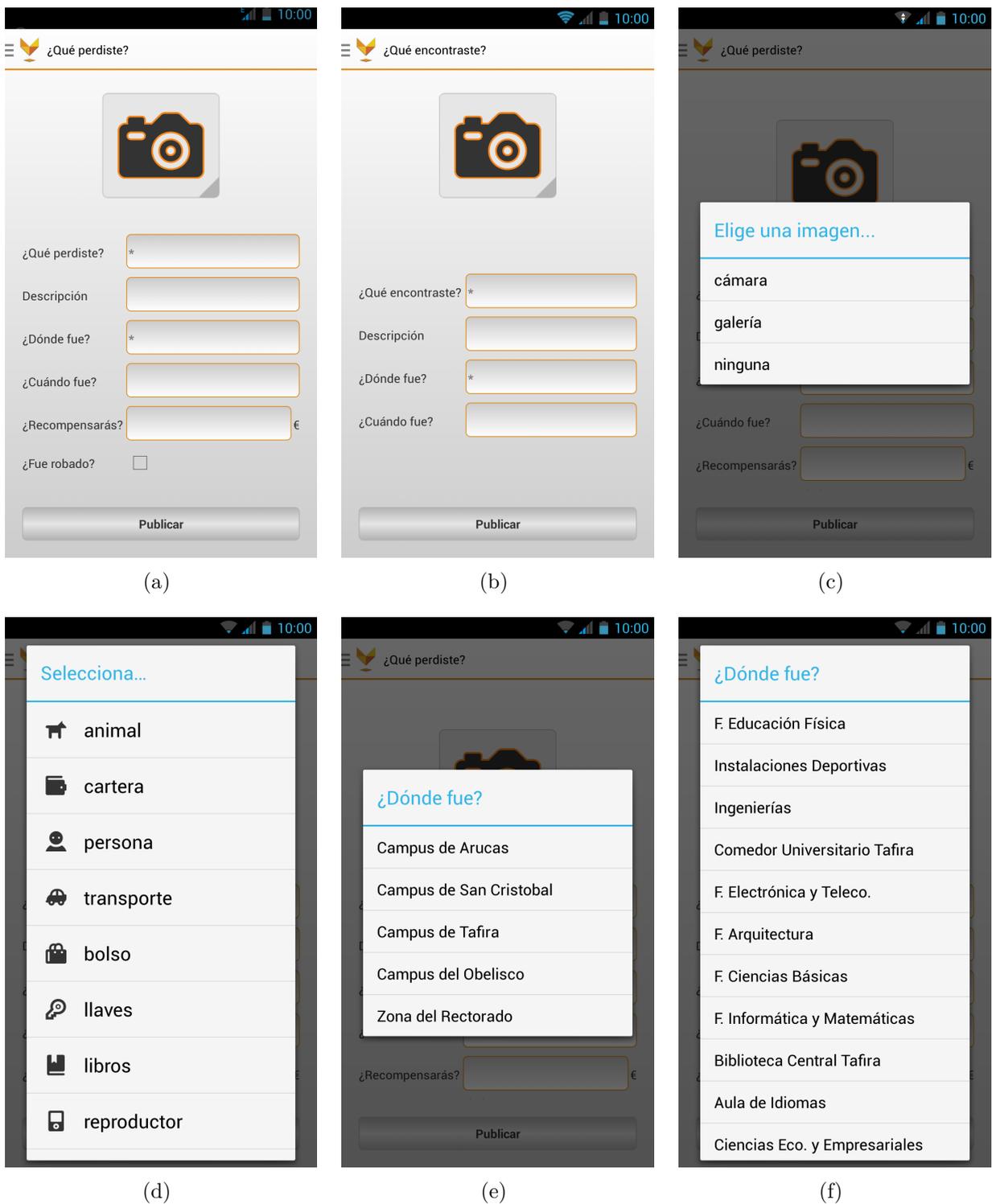


Figura 7.8: Pantallas «¿Qué perdiste?» (a), «¿Qué encontraste?» (b), y menús comunes a ambas funcionalidades (c), (d), (e), (f).

7.5.2. ¿Qué encontraste?

Pulsando sobre el botón «¿Qué encontraste?» situado en la pantalla principal, el usuario podrá acceder a esta funcionalidad, y crear una publicación sobre un *OPA* encontrado. Como se puede comprobar en la figura 7.8(b), la pantalla es muy similar a la utilizada para «¿Qué perdiste?» y las funcionalidades de los diferentes campos y botones, son idénticas.

7.5.3. ¿Qué te robaron?

El usuario puede publicar un anuncio de un *OPA* robado, usando la pantalla «¿Qué perdiste?» y seleccionando el checkbox destinado a tal fin, tal y como se describió en ese apartado.

7.6. Almacén

Este es el lugar en el que el usuario podrá ver absolutamente todas las publicaciones de todos los usuarios registrados en *I-Found@ULPGC*, incluyendo las suyas propias. Para entenderlo mejor, se puede decir que el almacén representa de manera virtual, el almacén de una oficina real de objetos perdidos, como la que se puede encontrar por ejemplo en un aeropuerto, o en una comisaría de policía. Es el lugar donde están todos los *OPA* de todos los usuarios.

Al entrar en el «Almacén», el usuario verá todas las publicaciones que sean nuevas desde la última vez que entró, marcadas con la palabra **¡Nuevo!**. Además, para facilitar visualmente la búsqueda rápida en el «Almacén», *I-Found@ULPGC* establece un código de colores para poder identificar si una publicación es sobre un *OPA* perdido (rojo), robado (amarillo) o encontrado (naranja).

7.6.1. Contactar con el dueño de una publicación

El usuario podrá ver todos los detalles de una publicación, pulsando sobre ella en el «Almacén», tal y como se muestra en las figuras 7.9(b)(c). Además, de esta misma manera, podrá ver si una publicación es suya, o de otro usuario, en cuyo caso podría ponerse en contacto con él, pulsando sobre el botón «Contactar con *nombre de usuario*». Si el usuario decide hacer esto, se abrirá una nueva pantalla, en la que podrá comenzar una conversación a modo de chat, tal y como se muestra en la imagen 7.12(a).

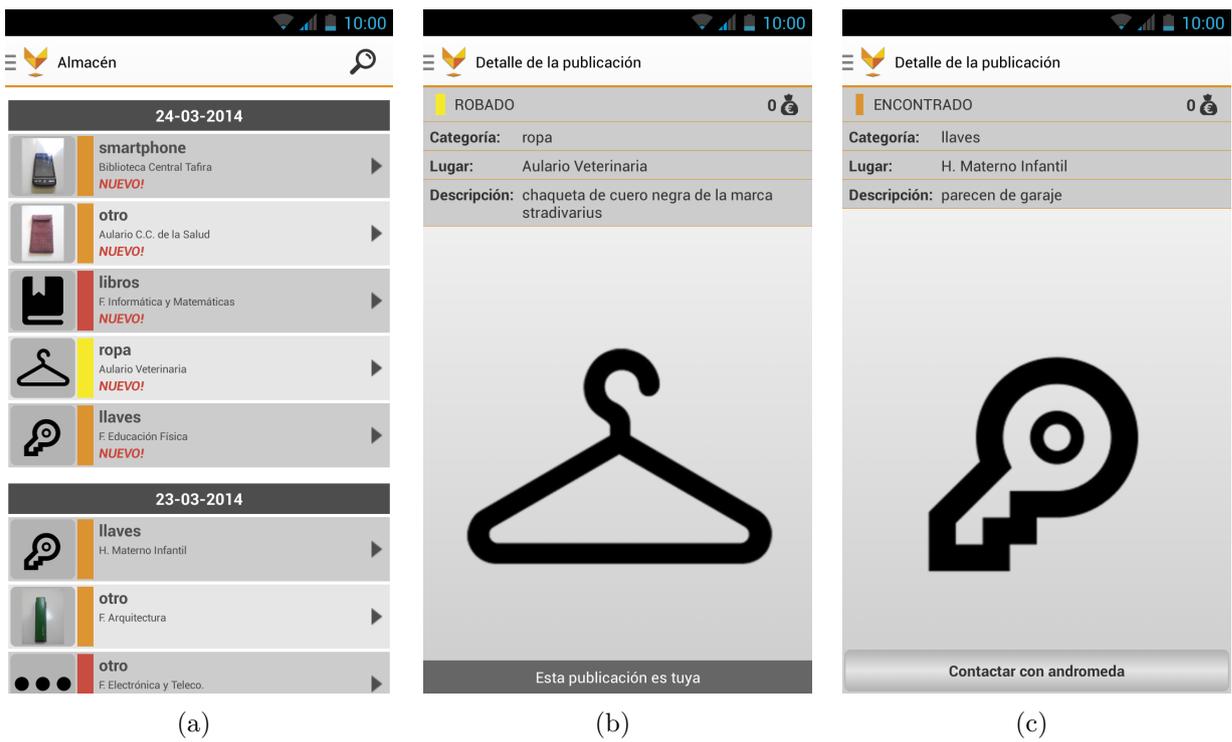


Figura 7.9: Pantallas de la vista principal del almacén (a) y de detalles de una publicación (b), (c).

7.6.2. Buscar un OPA

Como es probable que el almacén guarde muchos OPA, se ha dotado al mismo de una funcionalidad de búsqueda, para permitir al usuario filtrar publicaciones del «Almacén» según sus necesidades. A esta funcionalidad se accede desde la pantalla del «Almacén» pulsando sobre el icono , o usando el menú contextual de navegación de dicha pantalla y pulsando sobre «¿Qué buscas?».

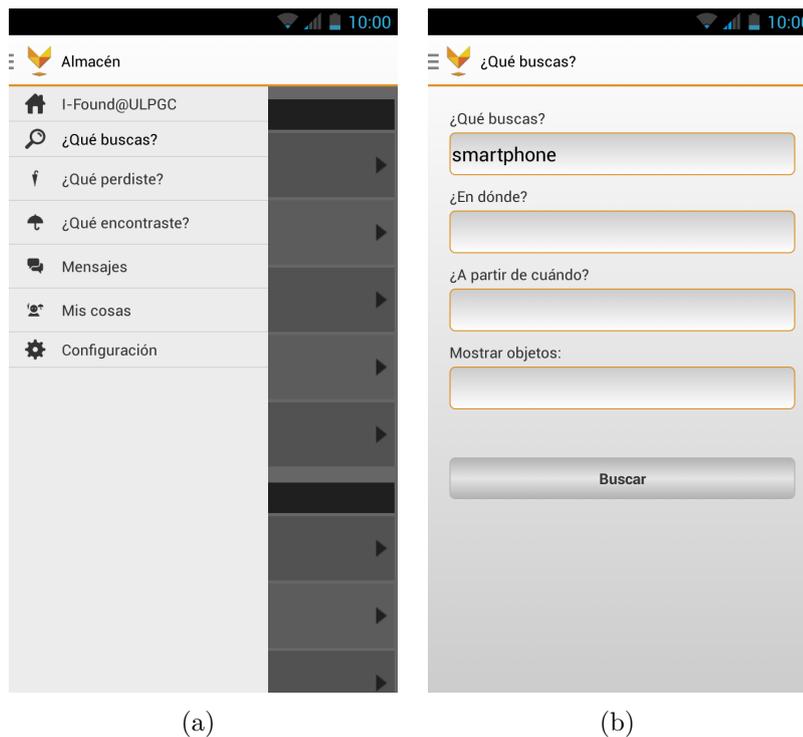


Figura 7.10: Pantallas para la búsqueda en el «Almacén».

7.7. Mensajes

Pulsando sobre este botón en la pantalla principal, el usuario puede acceder a la pantalla de todos sus mensajes, que a su vez estará dividida en dos subpantallas a las que se pueden acceder usando las dos pestañas superiores dedicadas para ello y diferenciadas con los iconos: , para las conversaciones (figura 7.11(b)) y , para las notificaciones del Sr. Found (figura 7.11(c)).

El icono de este botón, permanecerá gris si no hay conversaciones ni notificaciones nuevas, y cambiará a rojo en caso contrario. Lo mismo sucederá para los iconos de las pestañas de las subpantallas anteriormente mencionadas.

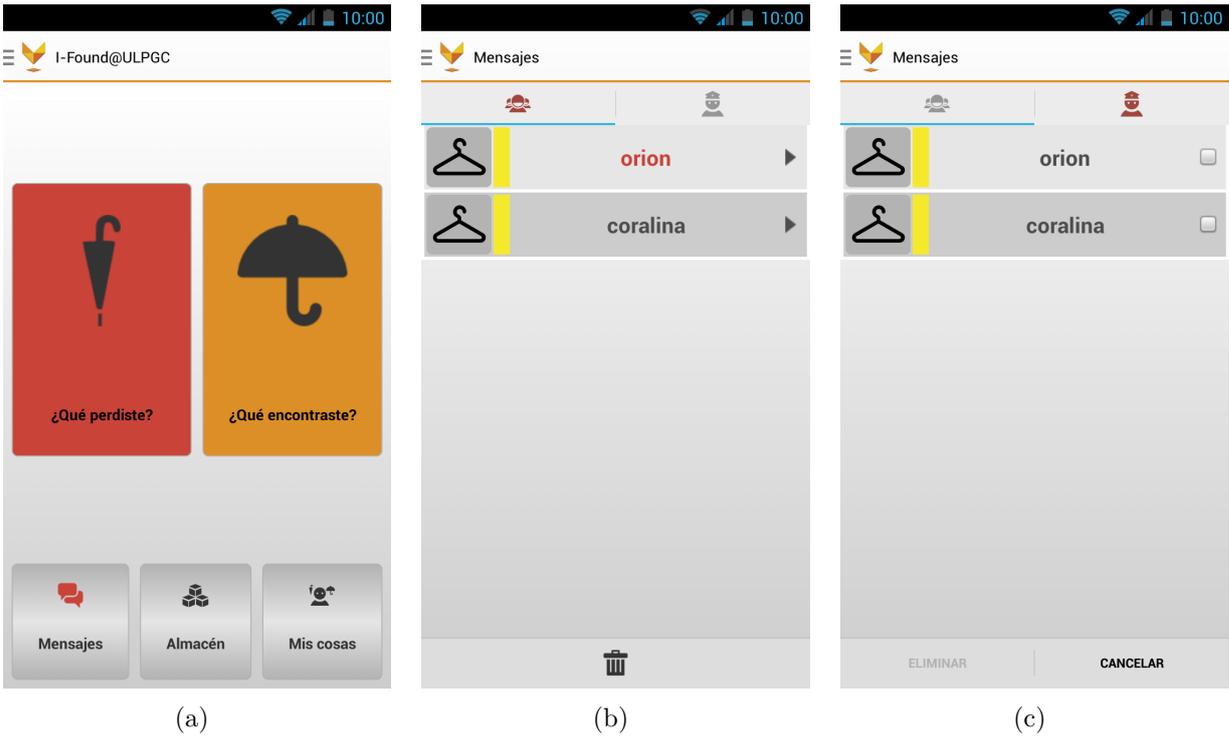


Figura 7.11: Pantallas de conversaciones y notificaciones del Sr. Found.

7.7.1. Conversaciones

Si el usuario que accede al «Almacén», tras ver los detalles de una publicación, decide iniciar una conversación con el dueño de esa publicación, se crea una nueva entrada en esta subpantalla, con el nombre del usuario con el que se ha comenzado dicho diálogo, tal y como se muestra en la figura 7.11(b). Lo mismo sucede si es el otro usuario quien inicia la conversación, de tal forma, que por medio de esta subpantalla, el usuario puede hacer un seguimiento de todas sus conversaciones con otros usuarios, además de poder continuar con ellas. Para acceder individualmente a cada una, el usuario deberá simplemente pulsar sobre la misma. Se abrirá entonces, una pantalla tipo chat, como la que se explicó en el apartado de «Almacén», en la que el usuario podrá ver al principio, un resumen detallado sobre la publicación de la que se está hablando, y seguidamente la conversación (figura 7.12(a)).

El nombre del otro usuario con el que el usuario mantiene la conversación, permanecerá gris si no hay mensajes nuevos para la conversación, y cambiará a rojo en caso contrario.

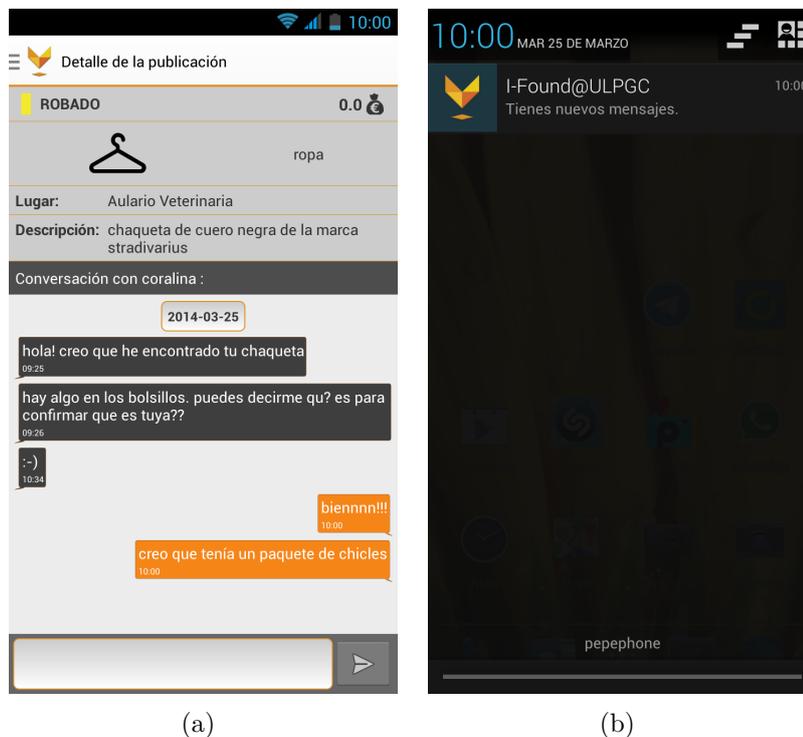


Figura 7.12: Pantallas de conversaciones.

7.7.2. Notificaciones del Sr. Found

El *Sr. Found*, el agente virtual de la aplicación y encargado del «Almacén» de los diferentes *OPA* perdidos, robados y encontrados, avisará al usuario por medio de notificaciones, cada vez que entren en el «Almacén» coincidencias de un *OPA* encontrado que pueda interesarle, porque puedan ser parecidos a algún *OPA* que ellos han perdido o les ha sido robado.

Para acceder individualmente a cada notificación, el usuario deberá simplemente pulsar sobre la misma. Se abrirá entonces la pantalla con los detalles de los datos de la publicación que podría interesarle, dándole la oportunidad de comenzar una conversación con el otro usuario que la publicó.

Los títulos de las notificaciones no vistas por el usuario, permanecerán en rojo. Una vez vistas, cambiarán a gris, tal y como se muestra en las figuras 7.13(a)(b).

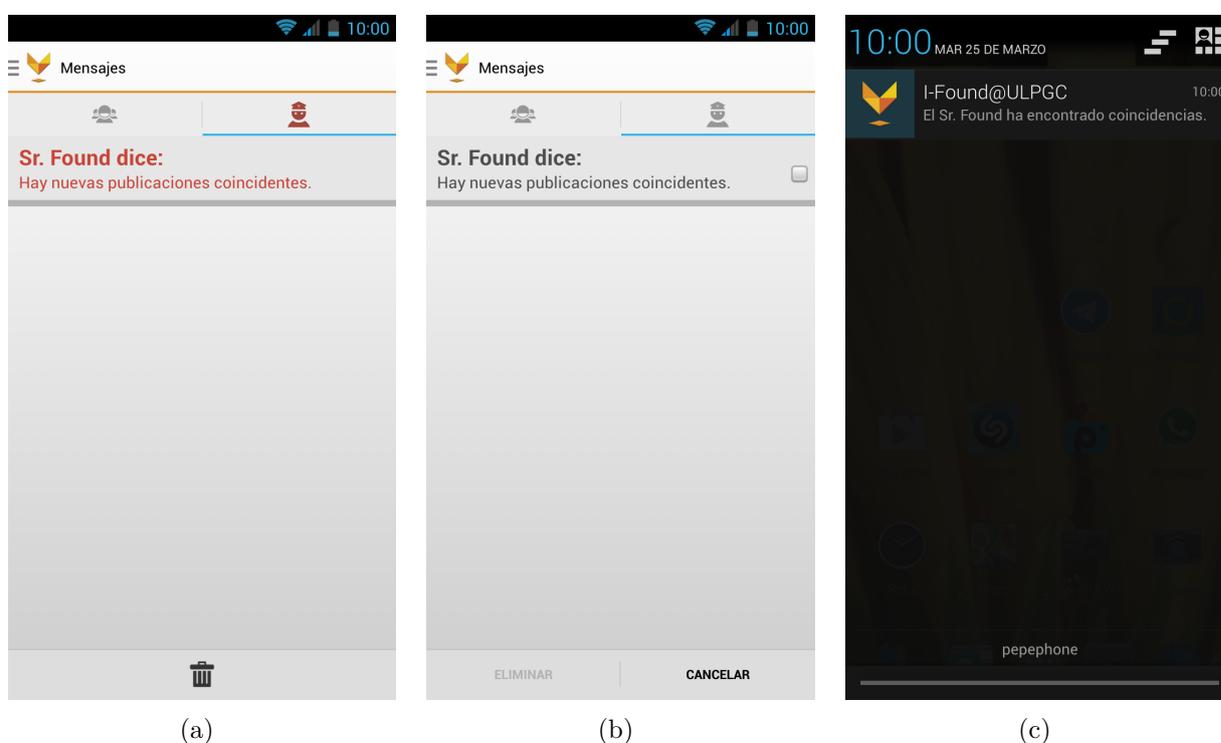


Figura 7.13: Pantallas de notificaciones del Sr. Found.

7.7.3. Eliminar conversaciones o notificaciones

Para eliminar una conversación o una notificación, el usuario deberá pulsar sobre el icono inferior . Cada elemento del listado de conversaciones o notificaciones, mostrará a su derecha entonces un checkbox, sobre el que el usuario podrá pulsar para seleccionar los elementos que desee eliminar. Una vez seleccionados dichos elementos, el usuario podrá pulsar sobre «Eliminar» para eliminarlos definitivamente, o sobre «Cancelar», para revocar dicha acción.

7.8. Mis Cosas

Aquí, el usuario podrá encontrar todas sus publicaciones. Esta pantalla, sirve al usuario para mantener un control de todas las publicaciones que ha hecho, además de para poder editar las mismas, o eliminarlas para siempre.

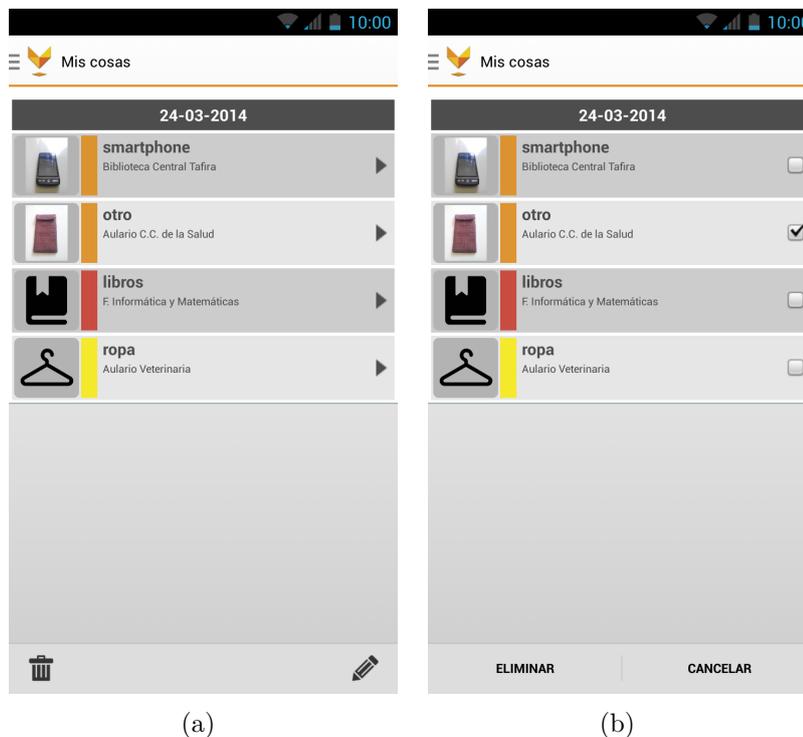


Figura 7.14: Pantallas 'Mis Cosas' (a) y eliminar publicaciones propias (b).

7.8.1. Eliminar una publicación propia

El usuario solamente podrá eliminar publicaciones propias. Para poder hacer esto, deberá pulsar sobre el icono  situado en la parte inferior izquierda de la pantalla. Una vez hecho esto, cada elemento del listado de publicaciones, mostrará a su derecha un checkbox, sobre el que el usuario podrá pulsar para seleccionar las publicaciones que desea eliminar. Una vez seleccionadas, el usuario podrá pulsar sobre «Eliminar» para eliminarlas definitivamente, o sobre «Cancelar», para revocar dicha acción. Esta acción se puede ver con detalle en la figura 7.14(b).

7.8.2. Editar una publicación propia

El usuario solamente podrá eliminar publicaciones propias. Para poder hacer esto, deberá pulsar sobre el icono  situado en la parte inferior derecha de la pantalla. Una vez hecho esto,

cada elemento del listado de publicaciones, mostrará a su derecha un checkbox, sobre el que el usuario podrá pulsar para seleccionar la publicación que desea editar. Una vez seleccionada, el usuario podrá pulsar sobre «Editar» para editarla, o sobre «Cancelar», para revocar dicha acción (figura 7.15(a)).

Si finalmente el usuario decide editar la publicación, se cargará la pantalla «¿Qué perdiste?» o «¿Qué encontraste?», con los datos de la actual publicación, para que el usuario pueda editar lo que necesite (figura 7.15(b)). Una vez hechos los cambios, el usuario deberá pulsar sobre el botón «Actualizar» y la publicación se actualizará en el «Almacén».

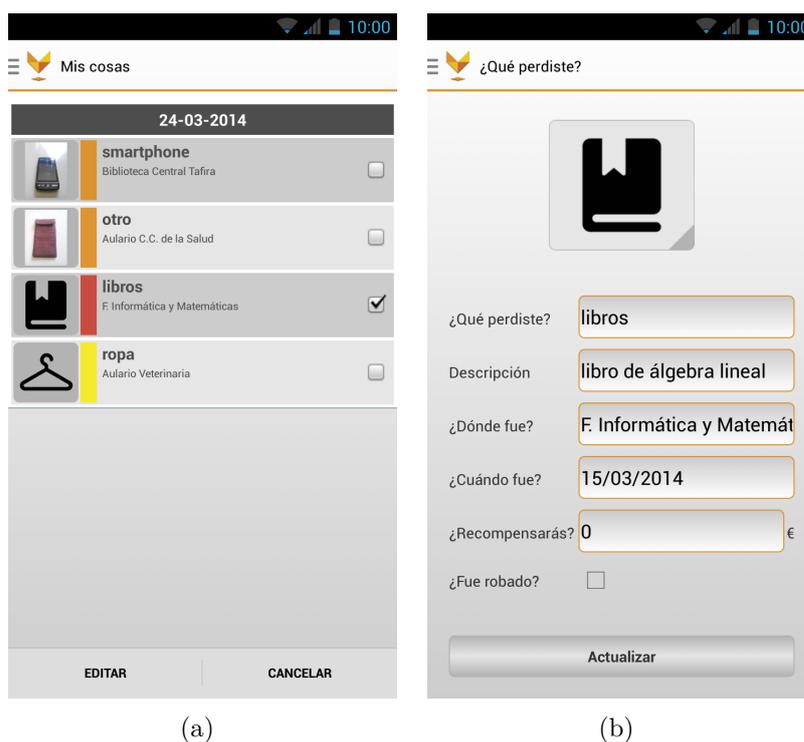


Figura 7.15: Pantallas de edición de publicaciones propias.

7.9. Menú de configuración

El usuario puede acceder al menú de configuración de la aplicación a través de los menús de navegación contextuales presentes en todas las pantallas. Desde ella, el usuario podrá configurar la recepción de notificaciones, eliminar definitivamente su cuenta de *I-Found@ULPGC*, cerrar la sesión y cambiar de contraseña.

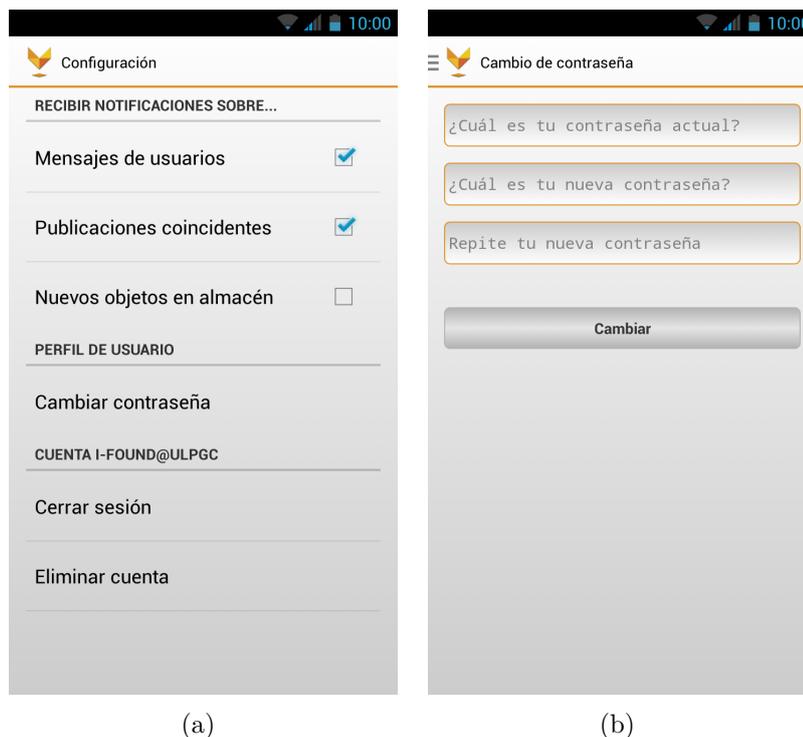


Figura 7.16: Pantalla principal de configuración (a) y detalle de cambio de contraseña (b).

7.9.1. Recepción de notificaciones

Se entiende por notificaciones, todos los mensajes de *I-Found@ULPGC*, que el usuario puede ver en la barra de notificaciones, situada en la parte superior de la pantalla del teléfono.

Actualmente, *I-Found@ULPGC* dispone de tres tipos de notificaciones:

- **Mensajes de usuarios.** El usuario recibe este tipo de notificaciones, cada vez que otro usuario se comunica con él (figura 7.12(b)). Si el usuario pulsa sobre la notificación, se mostrará la pantalla «Mensajes» automáticamente. Este tipo de notificaciones está activado por defecto en *I-Found@ULPGC*. El usuario puede desactivarla pulsando sobre el checkbox situado a su derecha.
- **Publicaciones coincidentes.** Suponiendo que el usuario se llama Sergio. Sergio recibiría este tipo de notificaciones (figura 7.13(c)) cuando sucedieran los siguientes hechos:

1. Sergio publica haber **perdido** una cartera en el Edificio de Telecomunicaciones.
2. Ana publica haber **encontrado** una cartera en el Edificio de Telecomunicaciones.
3. Sergio recibe una notificación de tipo 'Publicaciones coincidentes', ya que Ana ha publicado haber encontrado algo muy similar, sino igual, a lo que Sergio había perdido, y *I-Found@ULPGC* entiende, que esto podría interesarle mucho a Sergio, así que lo avisa.

Este tipo de notificaciones viene activado por defecto en *I-Found@ULPGC*, pero el usuario puede desactivarlo si lo desea en todo momento, deseleccionando el checkbox situado a su derecha, tal y como ocurría en el tipo de notificaciones anterior.

- **Nuevas publicaciones.** El usuario recibe este tipo de notificaciones, cada vez que otro usuario publica una nueva publicación en el «Almacén». Están desactivadas por defecto.

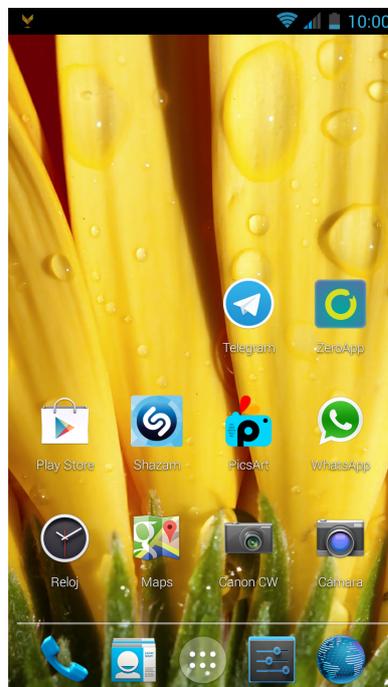


Figura 7.17: Detalle de llegada de notificaciones I-Found@ULPGC.

7.9.2. Eliminar cuenta

Por medio de esta opción, el usuario puede eliminar su cuenta de *I-Found@ULPGC* para siempre, y con ello, todas sus publicaciones y conversaciones. Todos los usuarios que mantengan conversaciones con este usuario, serán notificados del suceso.

7.9.3. Cerrar sesión

Con esta opción, el usuario cierra su sesión en *I-Found@ULPGC*, por lo tanto, y de forma automática, también dejará de recibir notificaciones de cualquier tipo.

7.9.4. Cambiar contraseña

Aquí, el usuario puede cambiar su contraseña, tal y como se muestra en la figura 7.16(b).

Capítulo 8

Pruebas y mantenimiento

En este capítulo, se describen las diferentes pruebas por las que ha pasado el proyecto durante su desarrollo y se muestran algunos de los resultados obtenidos de ellas. Además, también se desarrollan con más profundidad, las ideas de mantenimiento tras él y el valor de la comunidad de software libre que las haría posible.

8.1. Pruebas de usabilidad

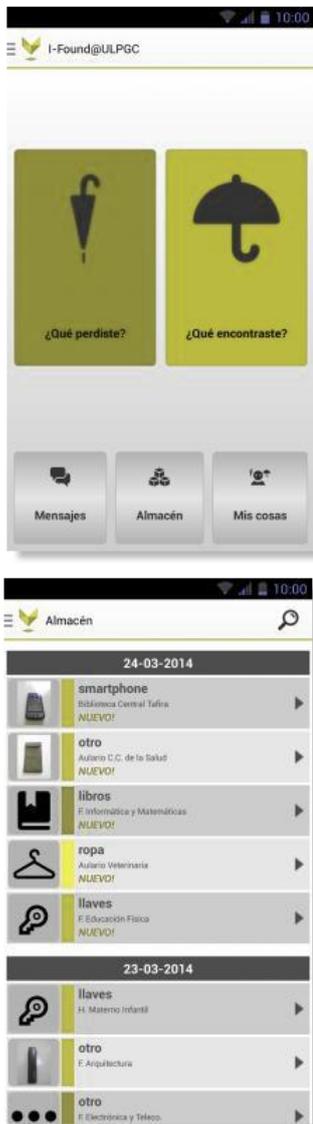
Se entiende como usabilidad, la facilidad con que las personas pueden utilizar una herramienta particular o cualquier otro objeto fabricado por humanos con el fin de alcanzar un objetivo concreto. *I-Found@ULPGC*, ha seguido un proceso de desarrollo tal, que ha nacido de las usuarios, por y para ellos. En los inicios fue a ellos a quienes se les interrogó por sus necesidades específicas, y de qué forma esperarían encontrarlas visualmente en una pantalla. Además, también a ellos se les facilitaron los prototipos, que pudieron juzgar y evaluar. Y por último, han sido ellos también quienes han evaluado la aplicación final desarrollada. Las pruebas de usabilidad, por lo tanto, han sido el centro de la ejecución de este proyecto en todo momento.

8.2. Pruebas de daltonismo

Como ya ha quedado bien reflejado en diferentes capítulos a lo largo de este documento, *I-Found@ULPGC* se ha centrado en toda la medida de lo posible en el usuario, y es por este motivo por el que se decidió tener en cuenta y escoger con cuidado, los colores elegidos para el diseño. Esto se ha hecho, debido a que existen gran cantidad de personas que tienen problemas de percepción del color, al menos 1 de cada 12 hombres y 1 de cada 200 mujeres de ascendencia europea (se puede encontrar en todos los grupos raciales, pero son más comunes entre los caucásicos).

Para probar la eficiencia del proyecto en este aspecto, se ha usado la herramienta libre

Coblis — *Color Blindness Simulator* [42]. Esta herramienta, es un simulador que permite ver, cómo personas con diferentes tipos de daltonismo (protanopía, deuteranopía y tritanopía, entre otros), verían la aplicación. Se pueden ver algunas simulaciones en las figuras 8.1 y 8.2.



- Normal Color Vision
- Red-Blind/Protanopia
- Green-Blind/Deuteranopia
- Blue-Blind/Tritanopia
- Red-Weak/Protanomaly
- Green-Weak/Deuteranomaly
- Blue-Weak/Tritanomaly
- Monochromacy/Achromatopsia
- Blue Cone Monochromacy

- Normal Color Vision
- Red-Blind/Protanopia
- Green-Blind/Deuteranopia
- Blue-Blind/Tritanopia
- Red-Weak/Protanomaly
- Green-Weak/Deuteranomaly
- Blue-Weak/Tritanomaly
- Monochromacy/Achromatopsia
- Blue Cone Monochromacy

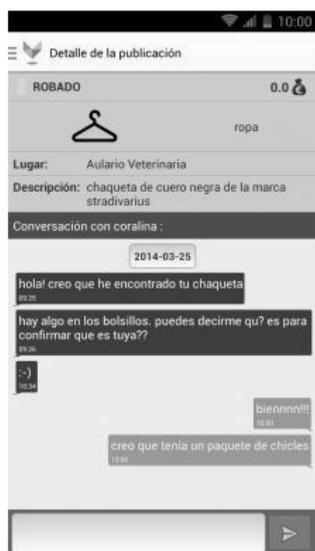
Figura 8.1: Visión de I-Found@ULPGC de una persona con protanopía.



- Normal Color Vision
- Red-Blind/Protanopia
- Green-Blind/Deuteranopia
- Blue-Blind/Tritanopia
- Red-Weak/Protanomaly
- Green-Weak/Deuteranomaly
- Blue-Weak/Tritanomaly
- Monochromacy/Achromatopsia
- Blue Cone Monochromacy



- Normal Color Vision
- Red-Blind/Protanopia
- Green-Blind/Deuteranopia
- Blue-Blind/Tritanopia
- Red-Weak/Protanomaly
- Green-Weak/Deuteranomaly
- Blue-Weak/Tritanomaly
- Monochromacy/Achromatopsia
- Blue Cone Monochromacy



- Normal Color Vision
- Red-Blind/Protanopia
- Green-Blind/Deuteranopia
- Blue-Blind/Tritanopia
- Red-Weak/Protanomaly
- Green-Weak/Deuteranomaly
- Blue-Weak/Tritanomaly
- Monochromacy/Achromatopsia
- Blue Cone Monochromacy

Figura 8.2: Visión de I-Found@ULPGC de una persona con acromatopsia.

8.3. El mantenimiento: la comunidad

I-Found@ULPGC es un proyecto libre, por lo que su código puede ser visto, modificado y redistribuido por quien lo desee, sin necesidad de tener que pagar ningún tipo de licencia. La idea tras el proyecto es crear una comunidad de usuarios y desarrolladores que lo sustente y alimente de diferentes maneras posibles. En el caso de los usuarios, proporcionándoles herramientas que les permitan contribuir avisando de posibles fallos del software, o aportando ideas sobre nuevas funcionalidades posibles. En el caso de los desarrolladores, dándoles más herramientas, para que puedan ayudar tal y como lo harían el resto de usuarios, y además, para que puedan contribuir mejorando el proyecto en términos de código, diseño de la arquitectura, añadiendo nuevas funcionalidades, etc.

Este tipo de comunidad es ideal para dar soporte y mantenimiento a proyectos de esta naturaleza y es lo que se pretende conseguir crear.

Capítulo 9

Modelo o ideas de negocio

Es común el pensamiento de que es de gran dificultad recuperar la inversión económica puesta en un proyecto de software libre, si no imposible. Sin embargo, y tal y como se explica en el documento "*Modelos de negocio con software libre*" [43] de Irene Fernández Monsalve de la UOC (Universitat Oberta de Catalunya), hay diversas soluciones que explican de manera detallada de qué forma puede coexistir un modelo de negocio lucrativo con un proyecto basado en software libre, de código abierto y gratuito. A continuación, se expone un resumen del documento con las diferentes posibles soluciones:

1. Especialistas/verticales, centradas principalmente en el producto de software libre y que eventualmente pueden adoptar modelos mixtos de doble licencia (libre y propietaria), de accesorios propietarios (plug-ins o adición de funcionalidades privadas), de venta distribuida del producto, o bien modelos de comercialización de servicios sobre el producto, como el software como servicio.
2. Servicios asociados, como los desarrollos a medida, la selección, instalación, personalización, integración, certificación técnica, formación, soporte y mantenimiento de productos; que eventualmente se pueden orientar a la distribución de plataformas, a la integración a gran escala o al servicio de pequeñas empresas y microempresas.
3. Mercados auxiliares de hardware, que con el software libre complementan una orientación principal de negocio a la venta de productos físicos o bien directamente al negocio de los contenidos accesibles desde un determinado hardware.
4. Otros mercados auxiliares, como las herramientas de colaboración, las certificaciones legales, la venta de libros o el merchandising.

Partiendo de las ideas expuestas en este documento y otras propias de la alumna, se han sugerido diferentes modelos o ideas de negocio, con los que poder impulsar *I-Found@ULPGC* al mercado y obtener ingresos con los que recuperar la inversión inicial y generar beneficios en el futuro.

En este caso además, se ha pensado en el uso de dos tipos de licencias para el producto: una licencia propietaria tradicional, y una licencia libre restrictiva. De esta manera, si alguien quiere generar un trabajo derivado, y redistribuirlo sin el código, puede hacerlo, pero deberá pagar una licencia. De lo contrario, todos los trabajos derivados deben redistribuirse con el código libre.

Se dividen así las diferentes ideas en dos grupos bien definidos:

1. Modelo basado en la publicidad y las comisiones.

- Hacer uso de banners y publicidad en la propia aplicación a través de *iAd*, *AdMob*, *Millennial Media*, *Place Play* o similares.
- Cobrar un tanto por ciento del valor de una recompensa, para usuarios que hayan perdido o les haya sido robado un *OPA* y hayan hecho uso de esta funcionalidad.
- Ofrecer un servicio a través del cual los usuarios puedan hacer donaciones voluntarias a través de la página web de *I-Found@ULPGC*, incluso premiando con algún detalle a aquellos usuarios que donen más cantidad que una cifra determinada. Por ejemplo, se podría decir, que todo usuario que done más de 20 euros tendrá como regalo una camiseta del *Sr. Found*.
- Recaudar dinero para nuevas funcionalidades, de tal forma que si varios usuarios quieren una funcionalidad determinada, se ponga en un listado a la misma con el valor económico que supondría llevarla a cabo. Los usuarios podrán entonces aportar sus dotaciones económicas para que se cree esa nueva funcionalidad, y una vez conseguido el dinero, podría implementarse. Un ejemplo podría ser el uso de mapas personalizables que sustituyan a los listados de localizaciones.
- En el caso de uso de mapas, cobrar a los restaurantes, bancos o diferentes comercios de la zona, un alquiler por insertar su logo, en el lugar del mapa en el que están ubicados, para ofrecerles publicidad.

2. Modelo basado en las licencias y servicios ofrecidos a organismos oficiales y otras instituciones.

- Integrar el proyecto en otros contextos, organismos oficiales o instituciones. Un ejemplo sería la policía o autoridad pertinente, para permitir la denuncia directa de objetos perdidos de forma telemática por parte de todos los ciudadanos. Si esta integración no permitiera la distribución libre del código resultante, lo que es muy probable, se cobraría por una licencia.
- Cobro por servicios ITCS (installation/training/support/consulting).
- Cobro por servicios de personalización y adaptación del diseño.
- Cobro por adición de nuevas funcionalidades.

Capítulo 10

Trabajo futuro

Como trabajo futuro, se proponen diversos puntos que aumentarían definitivamente la funcionalidad de *I-Found@ULPGC* y la dotarían de un mayor atractivo:

1. Portarlo a otras plataformas, como por ejemplo iOS, Windows Phone, Blackberry...
2. Traducirlo a otros idiomas.
3. Implantarlo en otras universidades del territorio español.
4. Adaptarlo a una versión “internacional”, de tal forma que no solamente se utilice en un entorno universitario, sino que también se haga en cualquier otro entorno.
5. Permitir la localización de ubicaciones por mapas, en lugar de usar listados seleccionables.
6. Permitir bugtracking directo desde la app, es decir, que cualquier usuario pueda avisar desde la propia app, de un comportamiento incorrecto o erróneo de la misma.
7. Desarrollar un cliente web.
8. Desarrollar una página web para el proyecto. La idea tras esta página, es que contenga al cliente web, el bugtracker, demos explicativas de *I-Found@ULPGC*, y que sea el punto de encuentro para todos los posibles usuarios y colaboradores del proyecto. Es decir, el propio repositorio y la app de *I-Found@ULPGC* se encontrarían disponibles en esta página, además de en Google Play Store. También esta misma página, podría incluir el sistema de donaciones voluntarias, mencionado en el capítulo **9. Modelo o ideas de negocio**.
9. Permitir compartir las publicaciones a través de e-mail, y las distintas redes o aplicaciones sociales (FB, Twitter, Whatsapp, Telegram, Line, etc...).
10. Dar soporte para una funcionalidad 'Nearby' que notifique a los usuarios de publicaciones de algún *OPA* a una distancia determinada desde su ubicación.

Capítulo 11

Conclusiones

Es posible afirmar con orgullo, que se han conseguido cumplir los objetivos fijados para este proyecto final de carrera: se ha desarrollado una aplicación libre para sistemas operativos *Android*, estudiando para ello la arquitectura y tecnologías asociadas para poder llevar a cabo este proceso satisfactoriamente. Con ella, todos los usuarios que pierdan o les sea robado un *OPA*, podrán ponerse en contacto directo con personas que los encuentren y viceversa.

Además, todo el proceso de desarrollo se ha centrado en el usuario final, potencial cliente de la aplicación, descubriendo así que este tipo de diseño es muy costoso en tiempo, debido a la realización de los diferentes prototipos antes de la versión final, sin embargo, ha sido un trabajo muy gratificante y ha permitido obtener unos resultados muy óptimos y con pocos errores, además de una aplicación usable testeada y compatible para personas que presentan dificultad en la visión, debido a algún tipo de daltonismo.

Otra adversidad que se ha encontrado ha sido la dificultad de adaptación del software para el abanico tan variado de versiones del que dispone *Android* y al que se ha querido llegar. Esto ha incurrido en tiempos mayores de desarrollo, para partes en las que se había estimado menos, y en el uso de librerías varias que en un principio no se pensó que se necesitarían.

Por otro lado, la experiencia a partir de la unión de esfuerzos con otra entidad educativa ha sido muy satisfactoria, obteniendo como resultado un producto más robusto y profesional que pueda llegar mejor al mercado, gracias a la unión de esfuerzos que se ha dado. La experiencia sin embargo, podría haberse aprovechado mejor, si se hubieran conocido las partes en fases anteriores del proyecto. Por ello, sería una gran idea tener un lugar, a modo de bolsa de proyectos, online, en el que poder poner en contacto a alumnos de las diferentes áreas que puedan necesitarse, por ejemplo, informática para el desarrollo, la escuela de arte para el diseño y la escuela de audiovisuales para las demos finales.

En definitiva, el proceso ha sido largo y muy complejo en algunas ocasiones, pero la experiencia ha sido completamente satisfactoria y el resultado final digno de ser mostrado y usado.

Lista de acrónimos

A

- ADT** Android Development Tools.
- API** Application Programming Interface.
- AVD** Android Virtual Device.

E

- EASD** Escuela de Arte y Superior de Diseño Gran Canaria.

G

- GCM** Google Cloud Messaging.
- GPL** General Public License.

H

- HTTP** Hypertext Transfer Protocol.

J

- JRE** Java Runtime Environment.
- JSON** JavaScript Object Notation.

L

- LAMP** Linux, Apache, Mysql and PHP.

O

OPA Objeto, persona o animal.

P

PHP PHP Hypertext Pre-processor (inicialmente PHP Tools, o, Personal Home Page Tools).

S

SMTP Simple Mail Transfer Protocol.

SVG Scalable Vector Graphics.

U

ULPGC Universidad de Las Palmas de Gran Canaria.

UOC Universitat Oberta de Catalunya.

X

XML eXtensible Markup Language.

Índice de figuras

1.1.	Diagrama conceptual de I-Found (a) y I-Found@ULPGC (b).	7
1.2.	Diagrama de funcionamiento de I-Found@ULPGC.	8
1.3.	Estadísticas de Super Monitoring [29].	9
1.4.	Comparativa entre I-Found@ULPGC y apps con funciones similares.	13
3.1.	Cuota de mercado 2013 de las versiones de Android.	24
3.2.	Arquitectura de Android. Fuente: Wikipedia	25
3.3.	Proceso de transformación de Java bytecode a Dalvik bytecode.	26
3.4.	Ciclo de vida de una Activity en Android.	29
3.5.	Arquitectura de Google Cloud Messaging. Fuente: Android Hive.	32
4.1.	Gráfica de la planificación del trabajo.	33
5.1.	Diagrama de actores.	39
5.2.	Diagrama de comunicación Servidor - Dispositivo.	40
5.3.	Diagrama de registro e inicio de sesión de usuario.	43
5.4.	Diagrama de interacción del Usuario autenticado.	43
6.1.	Pantallas del prototipo de: inicio de sesión y registro (a), principal (b) y de publicación de pérdida de un OPA (c).	54
6.2.	Pantallas de: publicación de haber encontrado un OPA (a), mensajes y notificaciones del Sr. Found (b) y almacén (c).	54

6.3. Pantallas de: detalle de un chat (a), notificaciones del Sr. Found (b) y de detalle de un OPA (c).	55
6.4. Esquema de diseño. Fuente: Ana Rodríguez Camacho.	57
6.5. El Sr. Found.	58
6.6. Arquitectura de sistemas de I-Found@ULPGC.	60
6.7. Eclipse y AVD en funcionamiento.	61
6.8. Diagrama entidad-relación de la base de datos.	62
6.9. Diagrama de la base de datos SQLite en el cliente.	63
7.1. Pantalla de aplicaciones.	65
7.2. Pantallas de inicio de sesión.	66
7.3. Pantallas de registro de usuario.	67
7.4. Pantallas de recuperación de contraseña.	68
7.5. Recepción de e-mail de recuperación de contraseña.	69
7.6. Pantalla principal de I-Found@ULPGC.	70
7.7. Menús contextuales de navegación en diferentes pantallas.	71
7.8. Pantallas «¿Qué perdiste?» (a), «¿Qué encontraste?» (b), y menús comunes a ambas funcionalidades (c), (d), (e), (f).	73
7.9. Pantallas de la vista principal del almacén (a) y de detalles de una publicación (b), (c).	75
7.10. Pantallas para la búsqueda en el «Almacén».	76
7.11. Pantallas de conversaciones y notificaciones del Sr. Found.	77
7.12. Pantallas de conversaciones.	78
7.13. Pantallas de notificaciones del Sr. Found.	79
7.14. Pantallas 'Mis Cosas' (a) y eliminar publicaciones propias (b).	80
7.15. Pantallas de edición de publicaciones propias.	81
7.16. Pantalla principal de configuración (a) y detalle de cambio de contraseña (b). . .	82

7.17. Detalle de llegada de notificaciones I-Found@ULPGC.	83
8.1. Visión de I-Found@ULPGC de una persona con protanopía.	86
8.2. Visión de I-Found@ULPGC de una persona con acromatopsia.	87

Índice de tablas

3.1. Estados posibles de una Activity.	28
3.2. Eventos lanzados por las actividades al cambiar de estado.	30
4.1. Estimación del coste del proyecto.	34
5.1. Relación de usuarios entrevistados.	36
5.2. Listado de funcionalidades deseadas por los usuarios.	37
5.3. Relación de actores y objetivos.	38

Apéndices

Apéndice A

Entrevista 'Lista de los deseos'

Esta entrevista se ha realizado con el fin de poder crear un software centrado en el usuario y separar la visión del desarrollador de la del cliente, para así obtener un producto más completo. Para llevarla a cabo, se han descrito las diferentes situaciones o historias en las que un usuario podría necesitar hacer uso de la aplicación, sin contextualizarla aún en la *ULPGC*, para que fuera lo más genérica posible. Estas son las pruebas:

- **Intro:** Explicación sobre la funcionalidad de la aplicación y creación del escenario base. Dejar al usuario usar su imaginación para saber qué esperan nada más iniciar la aplicación por primera vez. Además se les da papel y lápiz para dibujar en el caso de que les ayude.
- **Escenario A:** Vives en Barcelona, es una ciudad enorme, con varios centros. Tienes la suerte de vivir en un piso que solo queda a 15 minutos caminando desde tu casa. Te gusta cuidarte y cuidar el medio que te rodea, por lo que siempre que puedes, vas caminando. Hoy, de camino al trabajo encuentras unas llaves sobre la acera. Deben de habersele perdido a alguien. Quieres devolverle las llaves a la persona que las ha perdido, pero no tendrás tiempo de ir a la comisaría hasta dentro de unos días y además, te queda bastante lejos de casa y no dispones de medio de transporte propio. ¿Qué esperarías que te ofreciera la aplicación para ayudarte?.
- **Escenario B:** Hoy, y como cada día, has dejado salir solo al parque que hay justo en frente de tu casa, a tu mascota “Calcetines”, un Bulldog francés un poco travieso. Hoy, “Calcetines” no ha vuelto, y tras varias horas de búsqueda por el barrio, no lo has encontrado. Vas a imprimir carteles con su foto y avisar a veterinarios y todo lo que esté a tu alcance, pero quieres que el mayor número de personas posible, se entere de lo sucedido cuanto antes, para que también te ayuden a encontrarlo. ¿Qué esperarías que te ofreciera la aplicación para ayudarte?.
- **Escenario C:** Hoy, has invitado a almorzar a algunos invitados a casa. Te das cuenta de que no te queda pan para hacer tu prometido y delicioso pantumaca, así que coges la bici y te acercas un momento al super más cercano que tienes. Al finalizar tu compra e ir a buscar la bici, ya no se encuentra en el lugar en el que la has dejado. Quieres poner al máximo número de personas en conocimiento del supuesto robo. ¿Qué esperarías que te ofreciera la aplicación para ayudarte?.

- **Escenario D:** “Calcetines” sigue sin haber aparecido. Quieres comprobar si alguien lo ha encontrado. ¿Qué esperarías que te ofreciera la aplicación para ayudarte?.
- **Escenario E:** Descubres que alguien ha encontrado un objeto que perdiste. Quieres ponerte en contacto con ese usuario. ¿Qué esperarías que te ofreciera la aplicación para ayudarte?.
- **Resumen:** Dime tu lista de deseos sobre la aplicación. ¿Cómo te gustaría que fuera y qué te gustaría que te proporcionara?.

Apéndice B

Entrevista sobre el prototipo

Esta entrevista se ha realizado con el fin de testear y validar el prototipo interactivo creado como resultado de las primeras entrevistas de usuario que se hicieron en este proyecto, las cuales tenían la finalidad de averiguar qué necesidades reales tenían los potenciales usuarios de *I-Found@ULPGC*.

Introducción:

Como probablemente recuerdes de nuestra primera entrevista, la aplicación que estoy desarrollando, *I-Found@ULPGC*, tiene la intención de ayudar a poner en contacto directo, a personas que pierden o les son robados objetos, animales o personas, con personas que los encuentran y viceversa.

Lo que vas a probar hoy, no es la aplicación de verdad, sino un prototipo hecho a través de imágenes que enlazan a otras imágenes, por lo que a veces puede que no haga exactamente lo que esperas que haga. Por este motivo, es muy importante que pienses en alto continuamente, para yo poder discernir si se trata de un mal funcionamiento del prototipo o de algo en lo que no he pensado. Esto es MUY importante, necesito que digas cualquier tipo de pensamiento que pase por tu cabeza en voz alta. Da igual que sea bueno o malo, ten en cuenta, que esto lo hago para probar un producto que aún no está hecho y que estoy trabajando para hacer lo mejor posible, por lo que necesito la máxima sinceridad posible, por eso recuerda, piensa siempre en voz alta por favor.

Recuerda también, que no pasa nada si no sabes hacer alguna actividad, la intención de la prueba no es probarte a ti ni tus capacidades, sino al prototipo. Si no entiendes como realizar alguna actividad, el fallo estará en el prototipo, y no en ti.

Si no te importa y estás de acuerdo, me gustaría grabar la pantalla y tu voz mientras haces la prueba. Tu cara, o ninguna otra parte de tu cuerpo sería grabada. Estos vídeos pasarían a formar parte del material para mi proyecto y por lo tanto de la universidad. Vuelvo a repetirte, que tú no saldrías en los vídeos.

Estaré observándote en todo momento y tomando notas sobre cosas que yo crea que puedan mejorar mi prototipo, no sobre ti, pero aunque esté aquí, tu debes imaginar que no estoy, tienes que imaginar que estas tú con tu móvil, “sol@ ante el peligro”.

Por favor, otra cosa importante que voy a necesitar, es que silencies tu móvil y contestes a las siguientes preguntas para que conste que estas de acuerdo o no con lo anteriormente mencionado:

	Sí	No
¿Estás de acuerdo con todo lo explicado en este documento?		
¿Me dejarías grabar tu voz mientras haces la prueba y añadir el resultado como material para mi proyecto?		
¿Me dejarías grabar lo que haces en la pantalla mientras realizas la prueba y añadir el resultado como material para mi proyecto?		

Firma y fecha:

- **Actividad 1**

Acabas de bajarte la aplicación de la “Play Store” y ya la tienes instalada en tu móvil. Decides abrirla por primera vez y comenzar a utilizarla. Yo te avisaré cuando necesite que pases.

- **Actividad 2**

Ahora mismo te encuentras en la pantalla inicial de la aplicación, ¿podrías explicar en alto, qué crees que ves en ella, sin llegar a interactuar aún?.

- **Actividad 3**

Hoy, de camino al trabajo encuentras un libro de psicología caído sobre la acera. Debe de habersele caído a alguien y quieres devolvérselo. ¿Podrías solucionar esto a través de I-Found@ULPGC?. Yo te avisaré cuando necesite que pases.

- **Actividad 4**

Hoy al llegar a casa y después de un largo día de compras, te has dado cuenta de que has perdido tu cartera. Confías en que alguien pueda haberla encontrado ¿Puedes solucionar esto a través de I-Found@ULPGC?. Yo te avisaré cuando necesite que pases.

- **Actividad 5**

Hoy, has invitado a almorzar algunos invitados a casa. Te das cuenta que no te queda pan para hacer tu prometido y delicioso hummus, así que coges la bici y te acercas un momento al super más cercano que tienes. Al finalizar tu compra e ir a buscar la bici, ya no se encuentra en el lugar en el que la has dejado. Quieres poner al máximo número de personas en conocimiento del supuesto robo. ¿Puedes contarme en voz alta cómo lo harías a través de I-Found@ULPGC por favor?.

- **Actividad 6**

Parece que el usuario imke se ha puesto en contacto contigo porque cree que puede que hayas encontrado su libro de psicología. ¿Podrías contestarle?.

- **Actividad 7**

Parece que han entrado objetos nuevos al almacén, y entre ellos hay una cartera. Si es la tuya, ¿podrías ponerte en contacto con el usuario que la encontró?.

- **Actividad 8**

¿Podrías filtrar la búsqueda de tal forma que sólo busques todos aquellos objetos que sean “Cartera” y que se encuentren en Las Palmas?.

- **Actividad 9**

¿Podrías configurar I-Found@ULPGC para que te lleguen notificaciones de cualquier objeto nuevo que entre en el almacén?.

- **Actividad 10**

¿Podrías explicarme para qué crees que sirve cada una de las opciones en el menú de configuración?.

- **Actividad 11**

Desde la pantalla actual en la que estás, ¿podrías por favor volver a la pantalla de “Perdí”?.

- **Actividad 12**

¿Podrías mostrarme en I-Found@ULPGC, dónde aparecen todos tus objetos perdidos, robados o encontrados?.

- **Actividad 13**

Te has equivocado en la descripción de la cartera que perdiste y quieres cambiarlo. ¿Podrías hacerlo?.

- **Actividad 14**

Quieres eliminar la cartera de entre “Mis Cosas”. ¿Podrías hacerlo?.

- **Actividad 15**

Quieres dar de baja tu cuenta en I-Found@ULPGC, eliminarla para siempre. ¿Podrías mostrarme cómo harías esto?.

- **Actividad 16**

¿Podrías contarme en voz alta, quién crees que es y cuál es la misión del Sr. Found por favor?.

- **Actividad 17**

Entra en la pantalla “Almacén” y dime por favor, ¿qué crees que es el almacén? ¿qué son los objetos en verde y qué son los objetos en naranja amarronado?.

Apéndice C

Entrevista sobre el producto final

Esta entrevista se ha realizado con el fin de testear y validar la primera versión del producto final.

Introducción:

Como probablemente recuerdes de nuestras entrevistas anteriores, la aplicación que estoy desarrollando, *I-Found@ULPGC*, tiene la intención de ayudar a poner en contacto directo, a personas que pierden o les son robados objetos, animales o personas, con personas que los encuentran y viceversa. Se puede decir, que se trata de una oficina virtual de objetos perdidos, robados y encontrados.

Lo que vas a probar hoy, es un prototipo avanzado de la aplicación real, a falta de algunas mejoras funcionales. La intención de la entrevista de hoy, es comprobar si el producto desarrollado es usable e intuitivo, por lo tanto, es muy importante que pienses en alto continuamente. Esto es realmente MUY importante. Necesito que digas cualquier tipo de pensamiento que pase por tu cabeza en voz alta. Da igual que sea bueno o malo, ten en cuenta, que todo esto lo hago con la finalidad de probar un producto que aún no está acabado y en el que estoy trabajando para hacerlo lo mejor posible, por lo que necesito la máxima sinceridad posible. Por eso recuerda, piensa siempre en voz alta por favor.

Recuerda también, que no pasa nada si no sabes hacer alguna actividad, la intención de la prueba no es probarte a ti ni a tus capacidades, sino al producto. Si no entiendes como realizar alguna actividad, el fallo estará en el producto, y no en ti.

Estaré observándote en todo momento y tomando notas sobre cosas que yo crea que puedan mejorar mi prototipo, no sobre ti, pero aunque esté aquí, tu debes imaginar que no estoy, tienes que imaginar que estas tú con tu smartphone, “sola/o ante el peligro”.

Por favor, otra cosa importante que voy a necesitar, es que silencies tu móvil y contestes a la siguiente pregunta para que conste que estas de acuerdo o no con lo anteriormente mencionado:

	Sí	No
¿Estás de acuerdo con todo lo explicado en este documento?	<input type="checkbox"/>	<input type="checkbox"/>

Firma y fecha:

- **Actividad 1**

Hoy, después de tomarte un café para empezar el día en la cafetería de tu facultad, te encuentras unos apuntes. Deben de haberse olvidado a alguien. ¿Podrías publicarlo en *I-Found@ULPGC* ¡Seguramente su dueña/o esté desesperada/o por encontrarlos!. Yo te avisaré cuando necesite que pares.

- **Actividad 2**

Vaya... Te has dado cuenta de que se te ha olvidado poner el color del interior de la carpeta de los apuntes en la descripción de la publicación. ¿Podrías modificar tu publicación y añadirlo?.

- **Actividad 3**

Parece que el usuario andromeda se ha puesto en contacto contigo porque cree que puede que hayas encontrado sus apuntes. ¿Podrías contestarle?.

- **Actividad 4**

Desde la pantalla actual en la que estás, ¿podrías por favor volver a la pantalla de “Almacén”?.

- **Actividad 5**

Por favor, haz una búsqueda de todos los animales perdidos que existen en el “Almacén”.

- **Actividad 6**

¿Puedes configurar *I-Found@ULPGC* para que no recibas notificaciones de ningún tipo?.

- **Actividad 7**

¡Qué mala suerte! Has perdido las llaves de tu Corsa, corre a ver si tienes suerte y las encuentras en *I-Found@ULPGC*. Si no es así, publícalo a ver si hay suerte.

- **Actividad 8**

Parece que el *Sr. Found* tiene noticias para ti. ¿Puedes decirme de qué se trata?.

- **Actividad 9**

Quieres dar de baja tu cuenta en *I-Found@ULPGC*, eliminarla para siempre.

- **Actividad 10**

¿Podrías darme tu opinión más sincera sobre la app?. Me gustaría saber tanto lo que te gusta, como lo que no te gusta tanto.

Apéndice D

Formulario de solicitud de información sobre objetos perdidos del Ayuntamiento de Las Palmas

	SOLICITUD INFORMACIÓN OBJETOS PERDIDOS	ESPACIO RESERVADO PARA EL REGISTRO GENERAL
	32 - 08	

DATOS DEL RECLAMANTE

NOMBRE Y APELLIDOS	D.N.I.	TELÉFONO
DOMICILIO	TÉRMINO MUNICIPAL	

DESCRIPCIÓN DEL OBJETO PERDIDO

DATOS APROXIMADOS DEL LUGAR DE EXTRAVÍO

FECHA DE EXTRAVÍO	LUGAR	HORA
/ /		/

Esta solicitud puede usted entregarla bien personalmente en las dependencias de la Policía Local sitas en la Calle Eufemiano Fuentes Cabrera nº 3, por correo electrónico a la dirección: polrec@laspalmasgc.es, o en cualquier oficina de Atención al Ciudadano.

Las Palmas de Gran Canaria, ____de_____de 20__

Fdo. _____

En cumplimiento de lo establecido en la Ley Orgánica 15/99 de Protección de Datos de Carácter Personal, le informamos que sus datos personales quedarán incorporados y serán tratados en los ficheros del Ayuntamiento de Las Palmas de Gran Canaria. Asimismo le informamos de la posibilidad de que ejerza los derechos de acceso, rectificación, cancelación y oposición en la siguiente dirección: c/ León y Castillo, 270, 35005. Las Palmas de Gran Canaria. En caso de que en este documento deban incluirse datos de carácter personal referentes a personas físicas distintas de quien lo presente, deberá con carácter previo a su inclusión, informarles de los extremos contenidos en el párrafo anterior.

EXCMO. SR. ALCALDE-PRESIDENTE DEL AYUNTAMIENTO DE LAS PALMAS DE GRAN CANARIA
JEFATURA DE LA POLICÍA LOCAL - C/ EUFEMIANO FUENTES CABRERA, Nº 3 TLF.- 928.44.64.10

Bibliografía y referencias

- [1] Jesús Tomás Gironés (2012). El gran libro de Android.
- [2] W. Frank Ableson, Robi Sen y Chris King (2011). Android, guía para desarrolladores.
- [3] Christian W. Dawson y Gregorio Martín (2000). El Proyecto Fin de Carrera en Ingeniería Informática. 10 Edición en español, Madrid.
- [4] Bernardo Cascales Salinas, Pascual Lucas Saorín, José Manuel Mira Ros, Antonio José Pallarés Ruiz y Salvador Sánchez-Pedreño Guillón (2003). El libro de Latex. Madrid.
- [5] Android Developers Official Page, <http://developer.android.com/index.html>
- [6] Curso Programación Android http://www.sgoliver.net/blog/?page_id=2935
- [7] Java Documentation, <http://tinyurl.com/8yj8sqq>
- [8] Wikipedia Official Page, <http://es.wikipedia.org/wiki/Wikipedia:Portada>
- [9] MySQL official page, <http://www.mysql.com/>
- [10] PHP official page, <http://www.php.net/>
- [11] PHP Tutorial, http://www.w3schools.com/php/php_mysql_intro.asp
- [12] Comparativa entre Android e iOS, <http://tinyurl.com/o2ly9e5>
- [13] Instalación LAMP en Ubuntu 12.04, <http://tinyurl.com/oq97cdh>
- [14] Eclipse official page, <https://www.eclipse.org/>
- [15] Installation of ADT Plugin on Eclipse, <http://tinyurl.com/7lgr6xh>
- [16] Nosinmiubuntu: Android+Ubuntu, <http://www.nosinmiubuntu.com/>
- [17] JSON Tutorial, <http://www.w3schools.com/json/>
- [18] Actionbars vs. Menus, <http://tinyurl.com/po34ue8>
- [19] Android Asset Studio official page, <http://tinyurl.com/7rf5s4o>
- [20] StackOverflow official page, <http://stackoverflow.com/>

- [21] How do mobile apps make money, <http://tinyurl.com/pbmepzp>
- [22] Miniejercicios con Latex, <http://minisconlatex.blogspot.com.es/>
- [23] Página oficial de AdMob, <http://www.google.es/ads/admob/>
- [24] Página oficial de iAd, <http://advertising.apple.com/es/>
- [25] Millennial Media official page, <http://www.millennialmedia.com/>
- [26] Place Play official page, <http://www.placeplay.com/>
- [27] Página oficial de Apalabrados, <http://www.apalabrados.org/>
- [28] Página oficial de Pencil Project, <http://pencil.evolus.vn/>
- [29] Gráfica comparativa mercado smartphones, <http://tinyurl.com/k5skdjf>
- [30] Fuente para iconos libres, <https://www.iconfinder.com/>
- [31] Facebook oficina objetos perdidos LP, <https://www.facebook.com/objetosperdidos2.0>
- [32] Página oficina objetos perdidos LP, <http://tinyurl.com/nusvlup>
- [33] Código Civil, <http://tinyurl.com/ncgdeew>
- [34] Gartner Group: Tendencias telefonía, <http://www.gartner.com/newsroom/id/2408515>
- [35] Servicio GCM de Google, <http://developer.android.com/google/gcm/index.html>
- [36] Página oficial de Google API Console, <https://code.google.com/apis/console/>
- [37] Escuela de Arte y Superior de Diseño Gran Canaria, <http://www.eagrancanaria.org/site/>
- [38] Proyecto GNOME, <http://www.gnome.org/>
- [39] Página oficial de Openclipart, <http://www.openclipart.org>
- [40] Using the Affect Grid to Measure Emotions in Software Requirements Engineering, <http://tinyurl.com/nbdffd2>
- [41] Página oficial de PhoneGap, <http://phonegap.com/>
- [42] Coblis - Color Blindness Simulator, <http://tinyurl.com/mr99ojq>
- [43] Modelos de negocio con software libre, <http://tinyurl.com/o7eye3t>
- [44] Proyecto OpenShot Video Editor, <http://www.openshot.org/>
- [45] Wikipedia: Juego de rol, http://es.wikipedia.org/wiki/Juego_de_rol
- [46] Wikipedia: Software libre, http://es.wikipedia.org/wiki/Software_libre
- [47] Wikipedia: Usabilidad, <http://es.wikipedia.org/wiki/Usabilidad>

- [48] Wikipedia: Smartphone, <http://tinyurl.com/q5crxmw>
- [49] Wikipedia: Framework, <http://es.wikipedia.org/wiki/Framework>
- [50] Wikipedia: Ingeniería de software, <http://tinyurl.com/qdmo7yv>
- [51] Wikipedia: Bug, http://es.wikipedia.org/wiki/Error_de_software
- [52] Wikipedia: Screencast, <http://es.wikipedia.org/wiki/Screencast>
- [53] Wikipedia: Plug-in, <http://tinyurl.com/qcnmkgb>
- [54] Wikipedia: L^AT_EX, <http://es.wikipedia.org/wiki/LaTeX>

Impresión digital y encuadernación
realizada el 30 de abril de 2014, por
Arte y Cultura Siglo XXI.

