



Universitat Oberta
de Catalunya

Aplicación para determinar la talla de zapatos

– *Máster Oficial en Software Libre* –

Especialidad Desarrollo de Aplicaciones de Software Libre

Autor:	Javier Fínez de Dios
Consultor:	Gregorio Robles
Tutor Externo:	Roger Estruch
Fecha:	08/01/2013

LICENCIA DE PUBLICACIÓN DE ESTE DOCUMENTO



Esta obra está sujeta a la licencia Attribution-ShareAlike 3.0 Spain de Creative Commons. Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by-sa/3.0/es/> o envíe una carta a Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

RESUMEN

Grandes Zapatos es una empresa dedicada exclusivamente a satisfacer las necesidades de aquellas personas que usan zapatos de tallas grandes y que llevan años sin poder encontrar aquello que buscan.

A través www.grandeszapatos.com, se pretende la venta de zapatos de tallas grandes de marcas conocidas, tanto para hombre (47 o más) como para mujer (41 o más), basando su estrategia en: la elección de las marcas, la atención al cliente y la comunicación en redes sociales.

Uno de los elementos claves para que no se compren zapatos por Internet es que la gente no se las pueda probar.

En este proyecto se aborda el estudio y desarrollo de una aplicación basada en software libre (Android) que permite romper dicha barrera de compra, aconsejando al usuario su talla para un determinado fabricante de zapatos.

Índice

CAPÍTULO I INTRODUCCIÓN.....	1
I.1 OBJETIVOS.....	1
I.2 ESTADO DEL ARTE.....	2
I.3 ESTRUCTURA DE LA MEMORIA DEL PROYECTO.....	3
CAPÍTULO II ESTUDIO DE VIABILIDAD.....	4
II.1 ESTABLECIMIENTO DEL ALCANCE DEL PROYECTO.....	4
II.2 ESTUDIO DE LA SITUACIÓN ACTUAL.....	4
II.3 FUNDAMENTOS TEÓRICOS TALLAS DE ZAPATOS.....	5
II.4 DEFINICIÓN DE LOS REQUISITOS DEL SISTEMA.....	6
II.5 EVALUACIÓN DE ALTERNATIVAS DE SOLUCIÓN.....	7
II.6 JUSTIFICACIÓN DEL PROYECTO Y SELECCIÓN DE LA SOLUCIÓN.....	9
CAPÍTULO III PLAN DEL PROYECTO.....	11
CAPÍTULO IV ANÁLISIS DEL SISTEMA.....	13
IV.1 DIAGRAMA DEL SISTEMA.....	13
IV.2 ALCANCE DEL SISTEMA	13
IV.3 ENTORNO TECNOLÓGICO.....	14
IV.4 HERRAMIENTAS	15
IV.5 ESTÁNDARES Y NORMAS DEL SISTEMA.....	15
IV.6 USUARIOS PARTICIPANTES Y FINALES DEL SISTEMA.....	16
IV.7 ESTABLECIMIENTO DE REQUISITOS.....	16
IV.7.1 Obtención de requisitos.....	16
IV.7.2 Especificación de casos de uso.....	18
CAPÍTULO V DISEÑO DEL SISTEMA.....	21
V.1 ESPECIFICACIONES DE ESTÁNDARES Y NORMAS DE DISEÑO Y CONSTRUCCIÓN.....	21
V.2 DISEÑO MODULAR DE LA APLICACIÓN.....	21
V.3 DISEÑO DE LA INTERFAZ DE USUARIO Y NAVEGACIÓN DE LA APLICACIÓN	23
V.4 DISEÑO DEL MODELO DE DATOS	29
CAPÍTULO VI IMPLEMENTACIÓN DEL SISTEMA	30
VI.1 ENTORNO DE DESARROLLO.....	30
VI.2 ESTRUCTURA DE LA APLICACIÓN Y MANIFIESTO.....	30
VI.3 ACTIVIDAD PRINCIPAL	32
CAPÍTULO VII PRUEBAS E IMPLANTACIÓN.....	33
CAPÍTULO VIII CONCLUSIONES.....	35
VIII.1 LOGROS ALCANZADOS.....	35
VIII.2 CONOCIMIENTO ADQUIRIDO	35
VIII.3 TRABAJO FUTURO.....	35

Figuras

Fig. 1: Imagen iFootGauge.....	2
Fig. 2: Plan del Proyecto.....	12
Fig. 3: Arquitectura del Sistema.....	13
Fig. 4: Caso de uso de la obtención de tallas.....	19
Fig. 5: Caso de uso de la actualización local del catálogo.....	20
Fig. 6: Diseño Modular de la Aplicación Android.....	22
Fig. 7: Recursos de Diseño en la aplicación Android.....	23
Fig. 8: Diagrama de Navegación por la aplicación. (p1).....	25
Fig. 9: Diagrama de Navegación por la aplicación. (p2).....	26
Fig. 10: Diagrama de Clases de la aplicación.....	27
Fig. 11: Diseño de la Base de Datos del Catálogo para la Aplicación Android.....	29
Fig. 12: Proceso Creación y Publicación Aplicación Android.....	33

Tablas

Tabla 1: Ejemplo equivalencias GZ.....	5
Tabla 2: Distribución de las diferentes versiones de Android.....	14
Tabla 3: Catálogo de Requisitos del Sistema.....	17
Tabla 4: Caso de uso Obtener Talla del Usuario.....	19
Tabla 5: Caso de uso Actualización catálogo en sistema local.....	20
Tabla 6: Ficheros específicos aplicaciones Android.....	30

Apéndices

Apéndice I: Nociones básicas de Fotogrametría de objetos cercanos.....	36
Referencias Bibliográficas.....	37

Abreviaturas

ADT	A ndroid D evelopment T ools
ASL	The A pache S oftware L icense
GPL	G eneral P ublic L icense
GZ	Talla unificada G randes Z apatos
HTML	H yper T ext M arkup L anguage
IDE	I ntegrated D evelopment E nvironment
JDK	J ava D evelopment K it
MVC	Patrón M odel V iew C ontroler (Modelo Vista Controlador)
OHA	O pen H andset A lliance
PHP	H ypertext P re P rocessor
REST	R epresentational S tate T ransfer
SDK	S oftware D evelopment K it
SQL	S tructured Q uery L anguage
SSL	S ecure S ocket L ayer
UI	U ser I nterface
UML	U nified M odeling L anguage
UNE	U na N orma E spañola

Capítulo I Introducción

I.1 Objetivos

El proyecto a presentar al finalizar el primer semestre del curso 2012-2013, como Proyecto de Fin de Máster para el *Máster en Software Libre* impartido por la UOC, se trata de una aplicación para determinar la talla de zapatos, para la empresa **GrandesZapatos.com**.

Para ayudar a sobrepasar la barrera que supone el hecho de no poder probar el calzado cuando se compra por Internet, se ha decidido ofrecer una aplicación que oriente al usuario a través de tecnología fotográfica a conocer su talla y ofrecerle, según el modelo de zapato a elegir y el fabricante cuál es el que le conviene.

En resumen, se trata de un proyecto de aplicación a los procesos de marketing y comercio electrónico, de la tecnología de *realidad aumentada* [1]: que ayuda a enriquecer nuestra percepción de la realidad, con una nueva lente gracias a la cual la información del mundo real se complementa con la del digital.

Los principales objetivos del proyecto son :

- Crear una aplicación informática la cual, mediante una fotografía del pie, recomiende al usuario qué talla gasta de una determinada marca y modelo.
- Dicha aplicación debe estar disponible para terminales móviles con tecnología **Android** [2]
- Se podrá descargar desde la tienda virtual **GrandesZapatos.com**, basada en **Prestashop** [3].
- Utilizar recursos y metodologías propias del *Software Libre*, de tal forma que el software desarrollado y el conocimiento generado sea de utilidad para la comunidad. Se tendrán en cuenta las posibles sinergias con la comunidad del *Software Libre*.

I.2 Estado del Arte

En el momento de inicio del proyecto, se ha investigado la existencia en el mercado de alguna tecnología similar a lo que nos estábamos planteando.

Se ha visto la existencia de la aplicación **iFootGauge** [4], una aplicación para teléfonos **iPhone**, que también permite medirse el pie con la cámara del móvil. En la siguiente imagen se puede ver su funcionamiento:

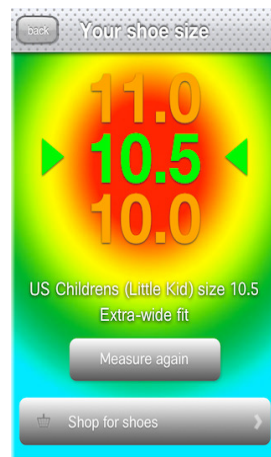


Fig. 1: Imagen iFootGauge

Pero dicha aplicación presenta serios inconvenientes :

1. No es software libre
2. Tiene un coste
3. Aparentemente no funciona muy bien
4. Sirve unicamente para Estados Unidos

Nuestro planteamiento es realizar una aplicación software libre, sin coste para el usuario, funcional y que sea útil para el mercado nacional.

I.3 Estructura de la memoria del Proyecto

En este apartado explicaré brevemente los apartados de los que se compone este documento.

- **Portada**
- **Licencia del Documento**
- **Resumen**
- **Indice de Contenido , Indice de Tablas y Figuras**
- **Abreviaturas utilizadas**
- **Introducción**, donde se presenta el problema planteado y el estado del arte de la tecnología con respecto al mismo.
- **Estudio de viabilidad**, donde se profundiza teóricamente en el problema, se evalúan la situación y las alternativas y se justifica la elección de la más adecuada.
- **Plan del Proyecto**, donde se planifica el trabajo a llevar a cabo dentro de los límites temporales que marcan los tiempos académicos.
- **Análisis del Sistema**, donde se analizan el enunciado de alto nivel del problema para traducirlo en unos requisitos técnicos a cumplir.
- **Diseño del Sistema**, donde partiendo de los requisitos definidos en el análisis se define la aplicación a construir : módulos de la aplicación, interfaz de usuario, modelo de datos...
- **Implementación del Sistema**, donde se construye la aplicación utilizando estándares y tecnologías software libre.
- **Pruebas e Implantación del Sistema**, proceso de pruebas y aceptación por los usuarios
- **Conclusiones**, objetivos logrados y no logrados, trabajo futuro...
- **Anexos**
- **Referencias Bibliográficas**

Capítulo II Estudio de Viabilidad

Se detalla el problema a resolver (requisitos) y cual es la situación del partida, se especifican las posibles soluciones y teniendo en cuenta diferentes restricciones se decide la alternativa a implementar exponiendo los motivos.

II.1 Establecimiento del Alcance del Proyecto

Las necesidades transmitidas son las de tener una aplicación que permita obtener de forma aproximada la talla de zapatos de un cliente de la tienda virtual **GrandesZapatos.com**, además de recomendarle en función de ella, el fabricante y el modelo al que opte.

Se ha propuesto realizar una aplicación para terminales móviles Android que en base a una fotografía del pie calcule la talla del calzado y nos recomiende en función del artículo (marca y modelo) a comprar, que talla del mismo elegir.

Durante la sesión de trabajo previa al Estudio de Viabilidad, surgió como opción secundaria, la creación de una aplicación que, basándose en anteriores compras en la tienda virtual, pueda recomendar al usuario su talla a elegir en otro artículo de la tienda.

II.2 Estudio de la Situación Actual

La empresa **GrandesZapatos.com**, está especializada en la venta virtual de zapatos de tallas grandes.

El principal inconveniente que tiene una tienda de zapatos virtual para vender, es que el cliente no puede probarse los zapatos, y por lo tanto la elección de la talla se convierte en un factor clave para la compra por Internet. Por lo que este proyecto está encaminado a eliminar dicha barrera de compra.

GrandesZapatos.com cuenta con un portal virtual que ya está en producción. Dicho portal está basado en **PrestaShop**. Esta basado en la versión **1.4.5.1**

En la actualidad, **GrandesZapatos.com** cuenta con probadores que prueban los distintos modelos de zapatos disponibles en la tienda y estandarizan un modelo de tallas **GZ**, para facilitar al usuario la elección del artículo adecuado, evitando las variaciones en cuanto a talla que pudieran surgir por fabricante, país de fabricación...

En la tienda virtual, para cada modelo de zapatos que consulta, el cliente puede ver una tabla **HTML** con la conversión de tallas similar a la siguiente :

GZ	EUR	USA	UK	CM
47	47 1/3	12,5	12	30,5
47,5	48	13	12,5	31
48	48 2/3	13,5	13	31,5
48,5	49 1/3	14	13,5	32
49	50	14,5	14	32,5
49,5	50 2/3	15	14,5	33
50	51 1/3	16	15	33,5
51	52 2/3	17	16	34,5
52	53 1/3	18	17	35,5

Tabla 1: Ejemplo equivalencias GZ

El problema es que con este sistema, aunque es válido como referencia, se obliga al usuario a mirar los datos en la página web. La idea sería obtener un modelo de datos que basándose en la información de estas tablas y de la medida de pie en cm del cliente, se le recomiende el nº que le quedaría bien de un determinado zapato.

II.3 Fundamentos Teóricos Tallas de Zapatos

Para conocer la talla del usuario, cuando las marcas usan tallas europeas, es relativamente sencillo conociendo la medida del pie en cm, puesto que está regulado por la norma **UNE 59850:1998 [5]**, y tan solo hay que multiplicar la medida del pie en cm por 1,5. Dicho de otra forma, la diferencia entre dos nº de talla europeos es de 6,66 mm. En todo caso hay que considerar que el pie se estira al caminar, debido a lo cual es conveniente que la medida del zapato sea de aproximadamente un cm superior a la medida del pie (medida del pie en actividad).

Muchas veces no se transforma bien la talla original del fabricante (cuando no se utiliza en origen el sistema europeo) a la del comprador. La dificultad crece porque la diferencia entre cada número no es igual en los distintos sistemas. Así, por ejemplo, en las tallas norteamericanas hay 1/3 de pulgada (8,47 mm) de diferencia entre cada talla completa, mientras que en las europeas la diferencia es de 2/3 de centímetro (6,66 mm). Además, los norteamericanos utilizan diferentes tallas para hombres y para mujeres, lo cual complica las cosas cuando se trata de calzado *unisex*, pues hay que indicar la talla de mujeres y la de hombres.

Para acabar de complicarlo, hasta hace poco en España se utilizaba un sistema de medida diferente al de otros países europeos (se medía la huella del pie, mientras que en los otros países se medía su proyección). Por eso, desde hace unos años hemos tenido que sumar una talla a la que usábamos con anterioridad.

Las marcas aplican sus propios criterios de cálculo y redondeos, por eso nos encontramos con que el calzado de unos fabricantes nos queda más grande que el de otros. En cualquier caso, es importante medir el pie para escoger mejor.

Medir un pie (su proyección, que es la forma de calcular la talla europea) es muy parecido a medir a una persona.

Es importante medir ambos pies y elegir el tamaño mayor (muy pocas personas tienen los pies idénticos). En cualquier caso, para que la medida sea correcta hay que estar de pie y con el peso del cuerpo repartido entre ambos pies.

Luego añadimos 1 cm para conseguir la medida del pie en actividad y ya sólo se trata de aplicar la multiplicación por 1,5. La talla que hay que elegir es la del tamaño inmediato superior.

II.4 Definición de los requisitos del sistema

En este apartado se definirán, clasificarán y priorizarán los requisitos identificados en función de las principales necesidades u objetivos planteadas por el cliente :

- Crear una aplicación informática la cual, mediante una fotografía del pie, diga al usuario qué talla gasta de una determinada marca.
- Dicha aplicación debe estar disponible para terminales móviles con tecnología **Android**, así como poder descargarse desde la tienda virtual **GrandesZapatos.com**, que funciona sobre **Prestashop**.

Clasificación de los requisitos

A continuación se clasificarán los requisitos detectados, en distintos tipos, asignándoles prioridades.

Técnicos

1. Aplicación disponible para terminales **Android**, siempre y cuando estos tengan capacidades fotográficas. [100]
2. Se ha de poder descargar desde la tienda virtual **Prestashop**, con la que opera **GrandesZapatos.com** [100]
3. Ha de estar basada en software libre y estándares abiertos [100]

Operativos

1. Calcular los cm en función de una fotografía del pie [100]
2. Comunicación con **Prestashop** para obtener datos de los productos a recomendar al usuario. [100]
3. Módulo de ayuda desde la tienda virtual para recomendar a un cliente otro modelo partiendo de la talla de otro que ya hubiera comprado [20]
4. Que los datos para la consulta de la aplicación se actualicen o bien automáticamente en Android o bien sean fáciles para un usuario no informático [20]

Legales

1. El código del Proyecto se deberá proporcionar bajo una licencia libre. [100]
2. La documentación del proyecto se deberá proporcionar bajo una licencia libre. [100]
3. Las herramientas para desarrollo, documentación y gestión deberán ser libres. [100]

Económicos

1. El coste económico del desarrollo para el cliente será de 0, puesto que se realizará bajo un convenio de colaboración con la **UOC**, para la realización de un Trabajo Final de Máster. [100]
2. El coste tanto en herramientas de desarrollo como en licencias de explotación ha de ser inexistente. [100]

II.5 Evaluación de Alternativas de Solución

Por la parte que afecta al sistema *e-commerce* no existe una alternativa real : tenemos que ceñirnos a las tecnologías que provee el entorno tecnológico de **Prestashop**.

Se debe profundizar en el análisis de como incorporara **Prestashop** los modelos **HTML** de la talla **GZ** (están codificados en el atributo talla, que se manipula directamente desde el *back-office* de la tienda virtual) y en como automatizar las consultas para cuando tengamos disponibles sistemas externos que nos pidan información de artículos en función de una medida en cm, fabricantes y modelos de calzado.

Desde el punto de vista del desarrollo de la aplicación, tenemos también acotadas las alternativas, ha de ser móvil y estar basada en tecnología Android. Para responder a dichos requisitos solamente quedan las siguientes vías :

1. Solución basada parcial o totalmente en un producto existente en el mercado. Se descarta contratación porque no se cumplirían requisitos económicos. No se ha detectado en el mercado de aplicaciones para Android ninguna solución concreta al problema planteado.
2. Desarrollo a medida.

El riesgo de encontrar una solución existente e intentar adaptarla para conseguir los objetivos del proyecto estaría en no encontrar el soporte necesario para la integración.

Por tanto, la solución final consistirá en un desarrollo a medida de las especificaciones dadas por el cliente, basado en la utilización de *Frameworks* y librerías de desarrollo *Open Source*.

El mayor riesgo que puede suponer esta solución, es que la aplicación no sea capaz de calcular la longitud del pie con la precisión deseada, o que se encuentre algún grave problema técnico en el desarrollo del mismo creando retrasos en la entrega de la solución.

Para paliar estos riesgos, se afianzarán todos los avances realizados durante las fases de ejecución del proyecto.

No se realiza un análisis **Coste/Beneficio**, porque no hay realmente alternativas que comparar. La alternativa elegida es a coste cero, cumpliendo todos los requisitos especificados y se justifica por los motivos indicados en el siguiente apartado.

Se ha visto la posibilidad de utilizar las librerías *Open Source* de visión artificial, que se pueden integrar con Android: **OpenCV [6]**.

OpenCV es una librería de visión por computador open source, escrita en C y C++, que incluye más de 500 funciones útiles en muchas áreas de la visión artificial: inspección de la producción, imagen médica, calibración de cámaras, visión estéreo, robótica...

Se han estudiado a fondo dichas librerías, y se ha desestimado su uso por que no aporta nada específico para la medida de imágenes.

Se ha recogido información exhaustiva de técnicas de **Fotogrametría [7]**, para evaluar posibles algoritmos a aplicar para calcular el tamaño de un objeto dentro de una imagen.

Después del análisis de las técnicas de Fotogrametría, se desestima su uso tanto por requerir equipos específicos (cámaras especiales), y el uso de varias tomas para obtener visión estereoscópica y además de amplios conocimientos en la materia.

II.6 Justificación del Proyecto y selección de la solución.

La justificación de la elaboración del presente proyecto se fundamenta en los siguientes aspectos:

- **Aptitud del alumno** : se ha de demostrar la capacidad del alumno para desarrollar proyectos en el ámbito del desarrollo mediante software libre, como consolidación de los conocimientos adquiridos en el Máster en Software Libre impartido por la UOC.
- **Necesidad** : Responde a una necesidad existente en el mercado, durante el estudio de viabilidad no se ha detectado ninguna herramienta software basada en software libre que resuelva el problema planteado de forma eficaz.
- **Económica**: Se satisfacen los requisitos económicos planteados por el cliente.
- **Tecnológica**: Se utilizan plenamente tecnologías *Software Libre* de gran auge en la actualidad: **Android** y **Prestashop**.

Por lo expuesto, la solución elegida consistiría en un desarrollo a medida basado en las siguientes tecnologías :

Android es un proyecto de *Google* en colaboración con la *Open Handset Alliance* [8]. Es un entorno de software creado para dispositivos móviles que incluye un sistema operativo basado en Linux, una completa interfaz de usuario, aplicaciones, bibliotecas de código...

Android se comercializa bajo dos licencias de código abierto diferentes. El núcleo de Linux se comercializa bajo la licencia **GPL** (*General Public License*), exigencia común para cualquiera que emplee el núcleo de este sistema operativo de código abierto. La plataforma Android, sin el núcleo, tiene una licencia **ASL** (*Apache Software License*). Aunque ambos modelos de licencia están orientados al código abierto, la principal diferencia es que la licencia de Apache se considera más proclive al uso comercial.

PrestaShop es una aplicación de *e-commerce* basada en Software libre. Ha ganado en su el premio *Open-Source Business Application* 2011, existen más de 100000 tiendas, más de 300000 miembros en su comunidad, está disponible en 41 idiomas y tiene una solida presencia mundial con tiendas en más de 150 países.

Para calcular la medida del pie del usuario en cm, se opta por diseñar un software que se base en comparaciones proporcionales a referencias a un objeto externo de dimensiones estándar (conocidas), lo cual será suficiente para las características del proyecto que estamos abordando.

Debido a la complejidad que podría tener obtener el resultado aceptable de cm en función de la imagen del pie (parámetros cámara, calibración...), se considera el plan alternativo que es que el usuario introduzca el tamaño en cm de la longitud de su pie.

Dicha alternativa se incorporará como modo de funcionamiento adicional de la aplicación que además tendrá la utilidad de hacer la aplicación funcional en dispositivos que no cuenten con cámara digital.

Capítulo III Plan del proyecto

Se consideran las siguientes fases, para la ejecución del proyecto :

- 1. Arranque del proyecto**
- 2. Estudio de Viabilidad y Análisis del Sistema**
- 3. Diseño del Sistema**
- 4. Desarrollo**
- 5. Implantación**

De forma paralela a estas fases se ejecutarán diversas tareas de seguimiento y control del proyecto. A continuación se muestra el diagrama de *Gantt*, con el plan estimado para el proyecto:

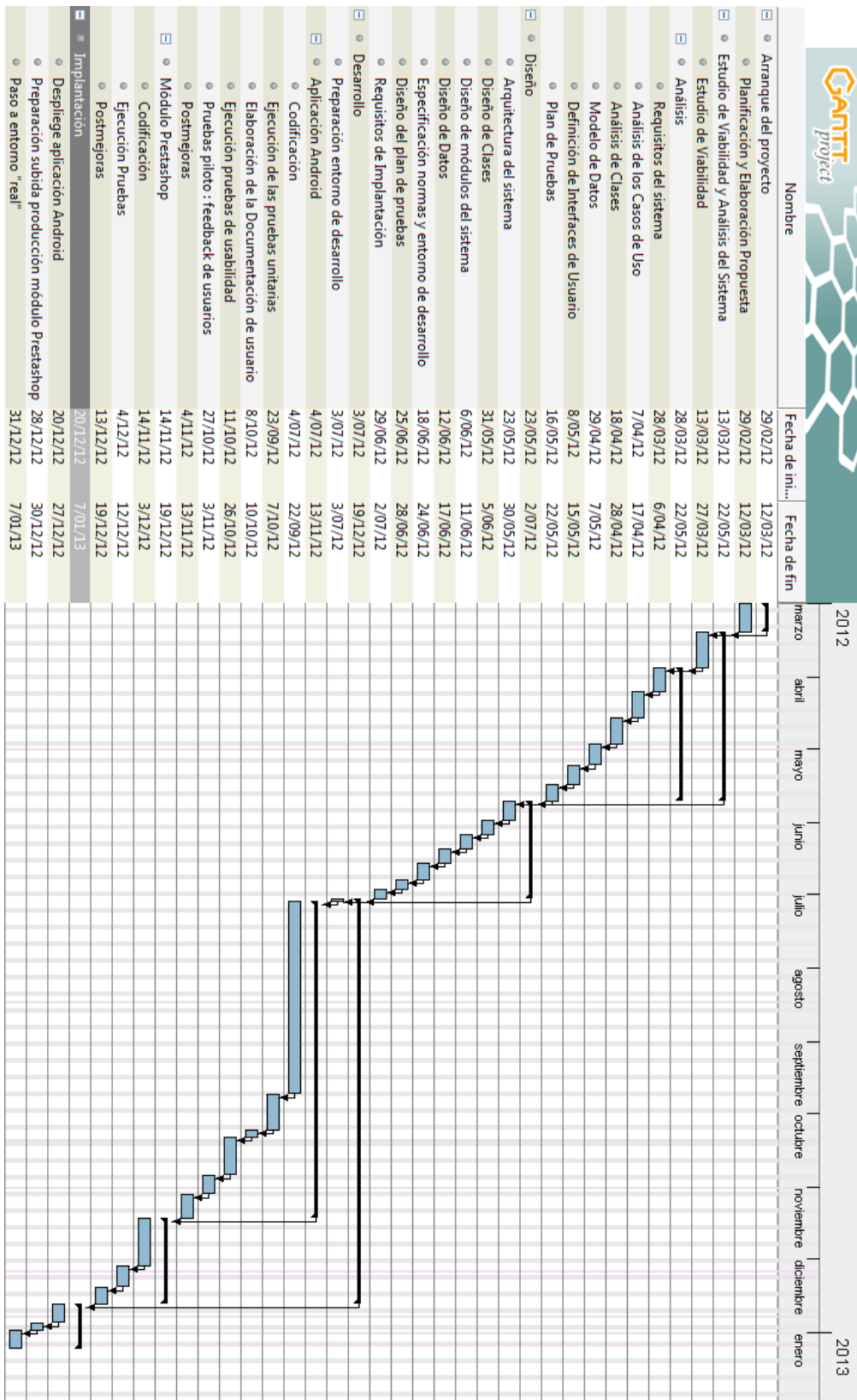


Fig. 2: Plan del Proyecto

Capítulo IV Análisis del sistema

En este apartado se especificarán con detalle los requisitos que deberá satisfacer el sistema, con el objetivo de poder realizar el diseño del mismo.

IV.1 Diagrama del sistema

Primero presentaremos, para aclarar ideas, un diagrama del sistema tal como lo hemos concebido. En la siguiente imagen se muestra:

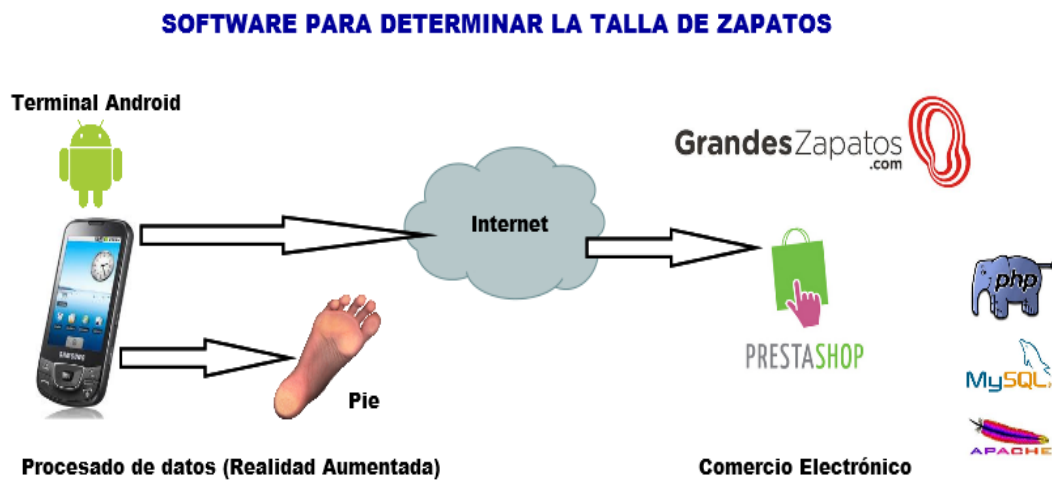


Fig. 3: Arquitectura del Sistema

IV.2 Alcance del sistema

El proyecto a construir es una aplicación informática, que permita obtener de forma aproximada la talla de zapatos de un cliente, mediante una fotografía del pie.

La aplicación deberá ser capaz de comunicarse con la tienda virtual **GrandesZapatos.com**, para ofrecer la recomendación de un artículo concreto de la misma, en función de la talla estimada del usuario y de la marca del artículo.

La aplicación tiene que estar disponible para terminales móviles **Android**

La aplicación debe poder descargarse desde la tienda virtual **GrandesZapatos.com**

En el modelo de datos del catálogo de la tienda virtual tenemos el atributo talla que corresponde al modelo de tallas unificado **GZ** donde se unifica la talla correcta. Nuestra aplicación tendrá que determinar que artículo interesa al usuario mediante la correspondencia entre la medida en cm y la talla **GZ**.

IV.3 Entorno tecnológico

Desde el *back-office* de la tienda **Prestashop** podemos obtener la siguiente información sobre el entorno de ejecución :

- Servidor Linux
- Servidor Web Apache/2.2.13 (Unix) mod_ssl/2.2.13 OpenSSL/0.9.8g
PHP/5.2.11 with Suhosin-Patch
- PHP 5.2.11
- MySQL 5.1.38-log
- PrestaShop 1.4.5.1

En cuanto a la plataforma móvil **Android** tenemos que saber que está diseñado para funcionar en diferentes tipos de sistemas y que coexisten a la vez dispositivos con diversas versiones del mismo sistema operativo.

Con cada versión del mismo se implementa un nivel de la **API**, por eso resulta conveniente tener en mente a la hora de desarrollar a que dispositivos “objetivo” nos queremos dirigir.

En la pagina de *Android Developers* [9] localizamos la estadística actualizada con los dispositivos que están funcionando en todo el mundo con una versión dada de la plataforma. En la siguiente tabla podemos ver el % de distribución de las diversas versiones:

Plataforma	Nombre	API	Distribución
Android 1.5	Cupcake	3	0,3%
Android 1.6	Donut	4	0,7%
Android 2.1	Eclair	7	6,0%
Android 2.2	Froyo	8	23,1%
2.3 – 2.3.2		9	0,5%
2.3.4 – 2.3.7	Gingerbread	10	63,2%
Android 3.0		11	0,1%
Android 3.1		12	1%
Android 3.2		13	2,2%
4.0 – 4.0.2	Ice Cream	14	0,5%
Android 4.0.3	Sandwich	15	2,4%

Tabla 2: Distribución de las diferentes versiones de Android

IV.4 Herramientas

Todas las herramientas utilizadas en el proyecto son *Software Libre*. Se necesitará usar las siguientes:

- **Eclipse [10]**: Es un producto de una comunidad open-source, cuyos proyectos están enfocados en crear una plataforma de desarrollo abierta compuesta por *Frameworks* extensibles y herramientas para para crear y gestionar software a lo largo de su ciclo de vida.
- **JDK [11]**: Software que provee las herramientas de desarrollo necesarias para desarrollar programas en lenguaje Java.
- **Android SDK**: Proporciona las herramientas y las bibliotecas necesarias para comenzar a desarrollar aplicaciones que se ejecutan en dispositivos Android.
- **Android Developer Tools (ADT) [12]**: es un plug-in para el IDE de Eclipse que está diseñado para proporcionar una entorno potente e integrado para la creación de aplicaciones Android. Amplia las capacidades de eclipse para poder configurar rápidamente nuevos proyectos Android, crear la interfaz de usuario, añadir los paquetes de la API de Android, depurar las aplicaciones utilizando el SDK e incluso poder exportar con o sin firma los archivos *.apk* con el fin de distribuir la aplicación.
- **LibreOffice**: es una suite de productividad personal de código abierto utilizada para generar la documentación del proyecto.
- **Dia** : software para la creación de diagramas, con distintos paquetes de formas conforme a diferentes necesidades.
- **GanttProject** : herramienta de escritorio multi-plataforma para la planificación y gestión de proyectos.
- **ArgoUML** : Es una herramienta de modelado UML, que incluye soporte para todos los diagramas del estándar UML 1.4

IV.5 Estándares y normas del sistema

Se contemplarán los siguientes estándares para el desarrollo del sistema :

- **Diseño** : *Android Design* [13], guía para el diseño de aplicaciones Android, con posibles descargas de iconos, estilos..
- **Codificación Android**: *Guía de estilo de codificación* [14], para contribuidores al proyecto de software libre Android.

- **Guía del diseñador de Prestashop [15]**
- **Guía del desarrollador de Prestashop [16]**

IV.6 Usuarios participantes y finales del sistema

El usuario participante tanto en la definición de requisitos como en la aceptación de la solución final es el responsable de la tienda **GrandesZapatos.com**.

Los usuarios finales son los clientes de tienda virtual **GrandesZapatos.com**, que se descarguen y ejecuten la aplicación para el cálculo de la talla en sus terminales Android.

IV.7 Establecimiento de Requisitos

IV.7.1 Obtención de requisitos

Se tratará de realizar un estudio más pormenorizado sobre los requisitos que el que fue hecho durante el estudio de viabilidad del sistema, a través de las conversaciones mantenidas con el responsable de la tienda **GrandesZapatos.com**

Se catalogarán y definirán los requisitos del sistema basándonos en las categorías FURPS+ (*Functionality, Usability, Reliability, Performance, Support,...*) definidas por Craig Larman (*UML y patrones*) [17].

En la siguiente tabla tenemos el catálogo de requisitos que hemos establecido:

Nº	Categoría	Prioridad	Estado	Descripción
1	Implementación	Alta	Aprobado	Disponibilidad para terminales Android
2	Empaquetamiento	Alta	Aprobado	Descarga disponible en la tienda virtual Prestashop
3	Legales	Alta	Aprobado	Basado en software libre y estándares abiertos
4	Funcionalidad	Alta	Aprobado	Obtener el tamaño en cm del pie en base a una fotografía
5	Funcionalidad	Alta	Aprobado	Comunicación con Prestashop para obtener datos de los productos a recomendar.
6	Funcionalidad	Baja	Propuesto	Módulo de ayuda desde la tienda on-line para recomendar a un cliente otro modelo partiendo de la talla GZ.
7	Legales	Alta	Aprobado	El código del proyecto se debe proporcionar bajo una licencia libre.
8	Legales	Alta	Aprobado	La documentación del proyecto se deberá proporcionar bajo una licencia libre.
9	Legales	Alta	Aprobado	Las herramientas de desarrollo, documentación y gestión deberán ser libres.
10	Económicos	Alta	Aprobado	El coste económico para el cliente ha de ser 0 €
11	Económicos	Alta	Aprobado	El coste de las herramientas de desarrollo y de las licencias de explotación deberá ser 0 €
12	Usabilidad	Media	Aprobado	La aplicación será sometida a test de <i>usabilidad</i>
13	Fiabilidad	Media	Aprobado	La aplicación deberá ser fiable.
14	Rendimiento	Media	Aprobado	La aplicación deberá tener un rendimiento aceptable
15	Interfaz	Baja	Propuesto	La aplicación deberá estar disponible en varios idiomas (inglés, castellano ...)
16	Soporte	Media	Propuesto	La aplicación deberá contar con un módulo de ayuda
17	Funcionalidad	Alta	Aprobado	La aplicación permitirá consultar a partir de la medida en cm (obtenida mediante fotografía basada en referencias o por <i>input</i> de usuario) que modelo le conviene de una marca determinada (de los disponibles en la tienda)

Tabla 3: Catálogo de Requisitos del Sistema

Debido a que tenemos que crear software para sistemas móviles, tenemos que tener en cuenta distintas restricciones que nos impone el *hardware*: ha de ser un código eficiente, gestionando la capacidad limitada de los sistemas sobre los que se va a ejecutar, el diseño de la interfaz ha de ser realizado para soportar distintos formatos de pantalla y hemos de minimizar el tráfico de datos en las conexiones a Internet.

Por otro lado, hemos de tener cierta consideración con el usuario final de la aplicación: la aplicación ha de tener un comportamiento predecible y correcto, presentar una interfaz de usuario (UI) atractiva e intuitiva, y responder de forma eficaz.

Resumiendo, se demanda que la aplicación sea diseñada para tener un buen desempeño, con capacidad de respuesta, segura, accesible y que el diseño sea uniforme para ofrecer una buena experiencia al usuario.

IV.7.2 Especificación de casos de uso

Partiendo de los requisitos especificados, vamos a exponer los principales casos de uso de sistema.

Comenzaremos por el caso de uso principal del sistema, en el que el usuario desea obtener que talla de una determinada marca y modelo de zapato le vendría bien.

Se puede ver en la siguiente tabla la definición de dicho caso de uso:

Caso de Uso	Obtener la talla del Usuario
Actores Primarios	Cliente; Cliente Potencial
Sistema	Aplicación Android Dispositivo móvil
Participantes	Cliente; Cliente Potencial
Nivel	Objetivo del Usuario
Condiciones previas	Dispositivo con aplicación Android instalada, Catálogo local actualizado.
Operaciones	
	1 Se detecta si el dispositivo móvil incorpora cámara digital. Si no solo se habilita la entrada manual de la medida.
	2 Seleccionar si realizar la fotografía o realizar el <i>input</i> de usuario de la medida en cm.
	3 Se de la opción de elegir al usuario un producto de un determinado fabricante.
	4 Se le recomienda la talla correcta.

Tabla 4: Caso de uso Obtener Talla del Usuario.

En la siguiente imagen podemos ver el diagrama UML del caso de uso de la obtención de la talla:

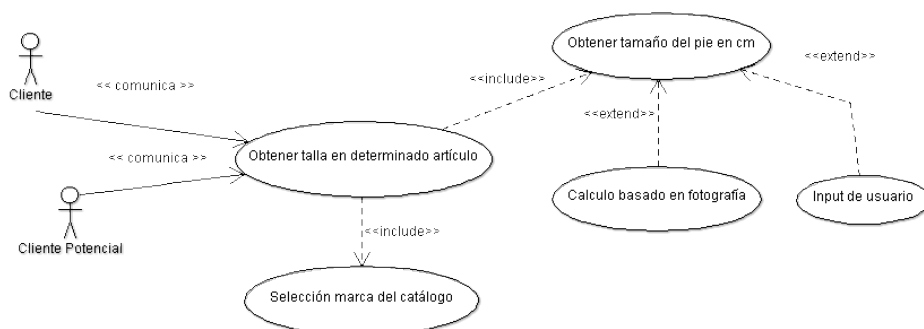


Fig. 4: Caso de uso de la obtención de tallas

Otro caso de uso considerado es la actualización del catálogo local al sistema móvil. Para ello la opción considerada es implementar un *WebService* en Prestashop donde se puedan consultar los datos, como se puede ver en la Fig.5 . En la siguiente tabla recogemos su definición:

Caso de Uso	Actualizar el catálogo local
Actores Primarios	Cliente; Cliente Potencial
Sistema	Aplicación Android Dispositivo móvil
Participantes	Cliente; Cliente Potencial; Grandeszapatos.com
Nivel	Objetivo del Usuario
Condiciones previas	Aplicación Android instalada, Conexión a Internet.
Operaciones	
	1 Conexión al Servicio Web de GrandesZapatos.com
	2 Descarga y almacenamiento local del catálogo de productos.

Tabla 5: Caso de uso Actualización catálogo en sistema local.

Esta funcionalidad debería servir tanto para minimizar el volumen de consultas a Internet desde el dispositivo móvil, como para implementar el modelo conceptual de correspondencia entre la medida del pie en cm y las diferentes formas de realizar las tallas en función de los fabricantes y productos disponibles, de tal forma que dicho modelo se pueda implementar de una forma paralela afectando lo más mínimo al funcionamiento de la tienda on-line.



Fig. 5: Caso de uso de la actualización local del catálogo

Capítulo V Diseño del Sistema

El resultado de esta fase es la obtención de los modelos y especificaciones que definen el sistema a construir a partir del análisis realizado en la fase anterior.

Se realizará por un lado el diseño modular de la aplicación y por otro el diseño de la interfaz de usuario y las opciones de navegación que tendrá el usuario a través de la misma.

Así mismo se diseñará el modelo de datos y se realizará un diagrama de clases del sistema.

V.1 Especificaciones de estándares y normas de diseño y construcción

Para el diseño de la interfaz de usuario Android seguiremos en la medida posible los principios desarrollados por el *Android User Experience Team*, pensando en cuidar al máximo la experiencia del usuario.

Podemos encontrar interesantes recomendaciones sobre **Patrones de Diseño** a utilizar en el diseño de una aplicación Android [18]

V.2 Diseño Modular de la Aplicación

Tras la realización de un estudio detallado sobre la actualización del catálogo de la aplicación Android desde la Web de la tienda, se desestimó la alternativa de utilizar el **WebService REST [19]** que ofrece **Prestashop**, puesto que no nos es posible agregar recursos nuevos, que es lo que nos interesaría, porque tenemos que definir el modelo de datos que contenga la medida en cm, la talla correspondiente de GrandesZapatos, y el modelo de calzado, su fabricante y la talla propia del fabricante.

Al no ser posible hacerlo de esta manera, se desestimó la alternativa optando en su lugar por implementar el catálogo en la aplicación Android de forma embebida (**SQLite [20]**), por lo tanto el caso de uso de Actualización del catálogo Android no estará disponible.

En el siguiente diagrama se especifican los módulos principales de los que se compondrá nuestra aplicación:

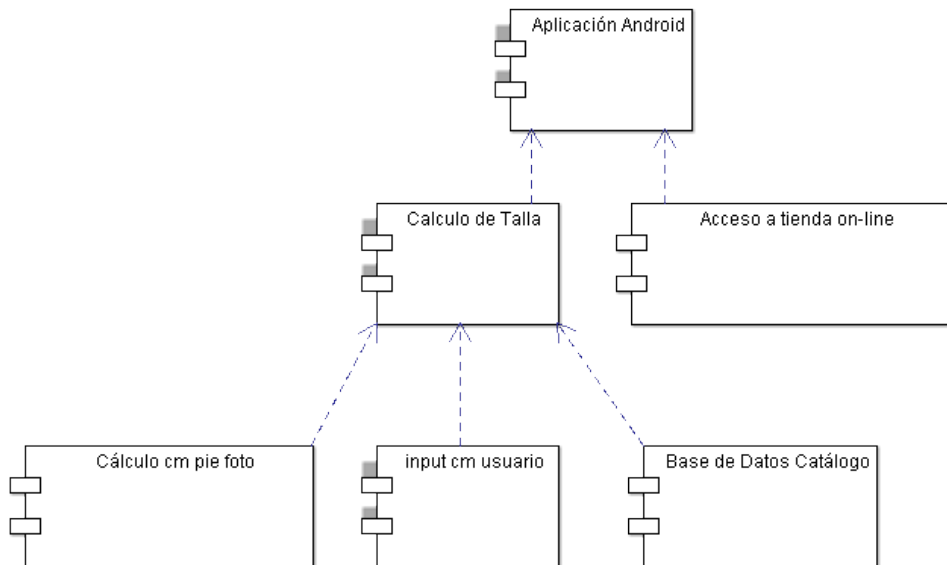


Fig. 6: Diseño Modular de la Aplicación Android

- **Aplicación Android:** Es la aplicación para recomendación de talla de calzado que vamos a construir.
- **Módulo de Cálculo de Talla:** Es el núcleo de la aplicación que se basará en otros módulos de cálculo auxiliares, a más bajo nivel, para darnos una recomendación del calzado del catálogo de la tienda que nos irá bien.
- **Acceso a la tienda on-line:** enlace de acceso que abrirá la web de la tienda on-line para que el cliente pueda comprar, consultar la tienda...
- **Calculo cm pie basado en fotografía:** Es la parte central de la aplicación, que en base a una fotografía obtenida con el móvil nos permitirá obtener los cm que mide el mismo.
- **Input usuario cm pie:** Sustituirá a la funcionalidad del módulo anterior, y será útil para móviles sin cámara o bien cuando el usuario ya conozca la medida de su pie en cm.
- **Base de datos catálogo:** base de datos **SQLite** donde se implementará el modelo de datos que asocia un tamaño en cm con los productos disponibles en la tienda por fabricante, modelo y tallas propias de la tienda grandes zapatos y del fabricante.

V.3 Diseño de la Interfaz de usuario y Navegación de la aplicación

Cuando abordamos el desarrollo de una aplicación Android, a parte de conocer el Android SDK (entorno de desarrollo), para la implementación, es importante tener claro la interfaz de usuario y su comportamiento.

Para diseñar la interfaz de usuario, tendremos que optar por **patrones de diseño**, donde podemos encontrar la formas de mostrar los datos, cómo ordenarlos, utilizar filtros típicos, como realizar la entrada de datos y el cambio entre vistas de la aplicación. Es decir los patrones de diseño son darán soluciones generales a problemas recurrentes.

Todo lo anterior, se implementa de forma bastante práctica mediante las herramientas de desarrollo de Android, a través de ficheros de **layout** y otros recursos, como se muestra en la Fig. 7:

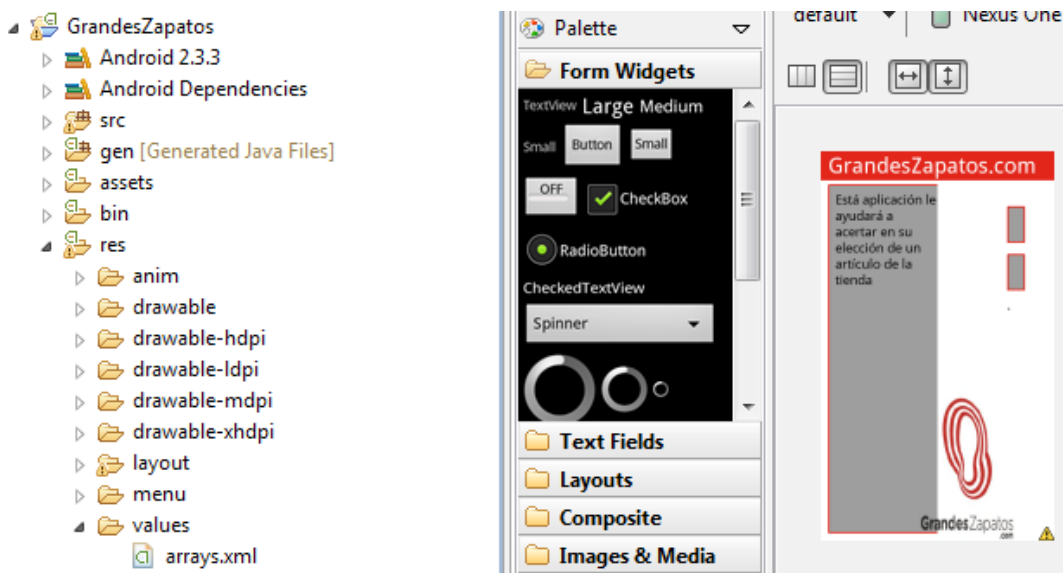


Fig. 7: Recursos de Diseño en la aplicación Android

La carpeta RES contiene los recursos y definiciones en XML (iconos, imágenes, cadenas de texto, estilos, temas, colores...)

Las definiciones XML:

- Alternativa a la programación JAVA para UI
- Facilitan el diseño de la UI
- Visualización más agradable y humana
- Mejoran las organización de los recursos del proyecto
- No proporcionan arquitectura MVC [21]
- Acceso desde clases Java R y Android.R

Los **Layout** Establecen la forma de posicionamiento de los elementos en la pantalla:

- Se definen mediante Java y XML
- En la Activity se establece el Layout a Utilizar.

Para el diseño de las pantallas una herramienta bastante útil es el programa **DroidDraw [22]**

A continuación se presenta el diseño seleccionado para nuestras pantallas (Fig. 8 y 9):

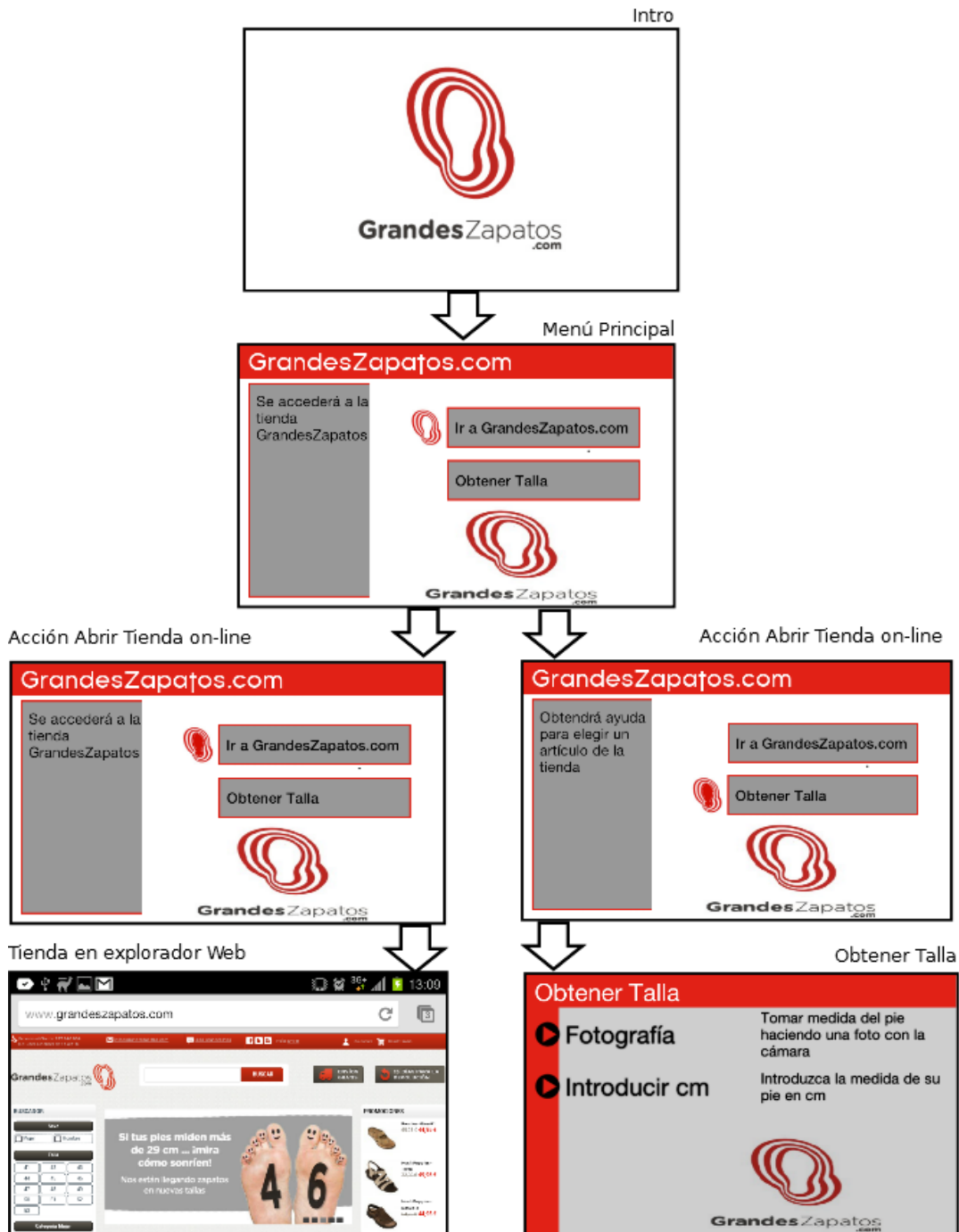


Fig. 8: Diagrama de Navegación por la aplicación. (p1)

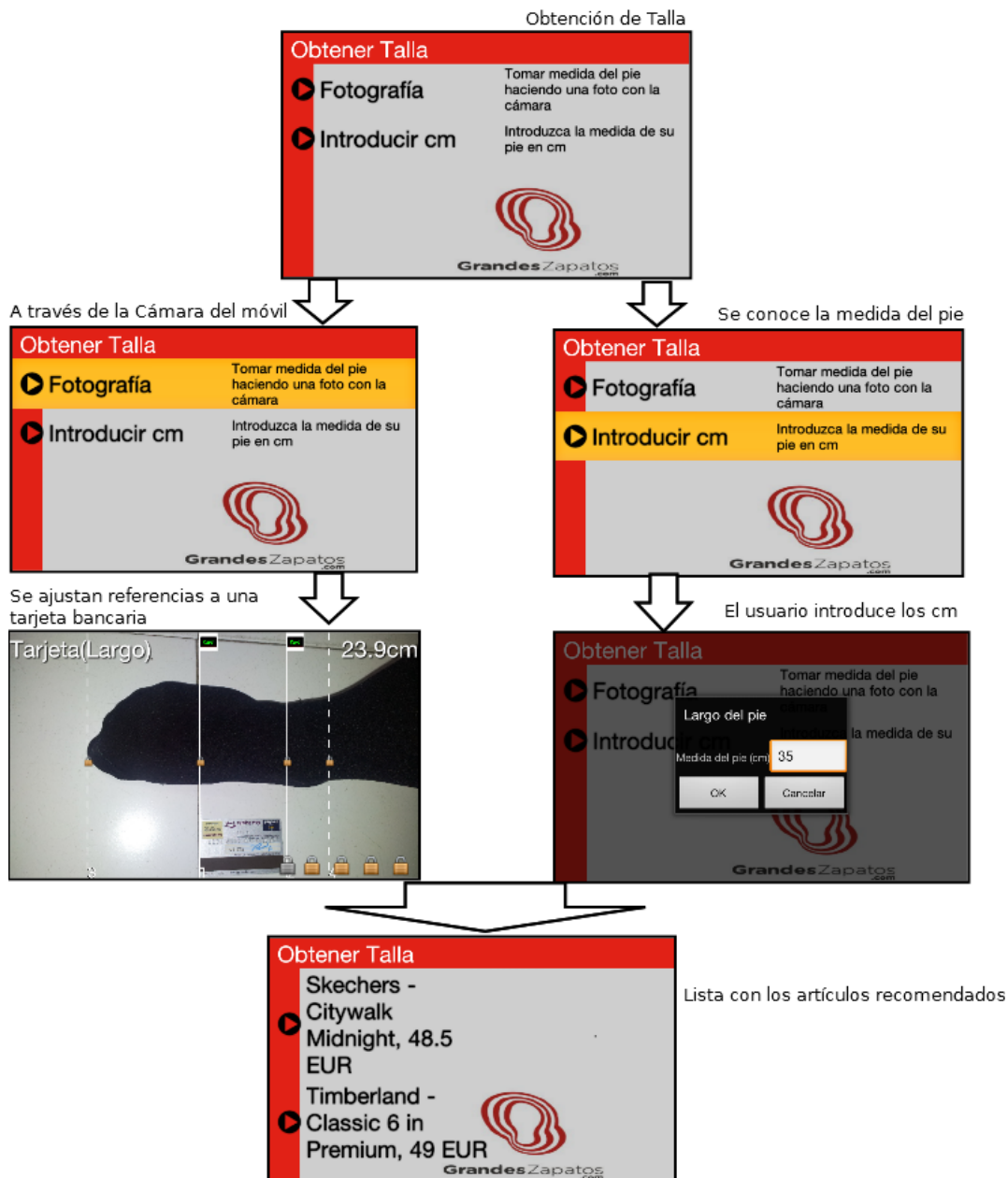


Fig. 9: Diagrama de Navegación por la aplicación. (p2)

* Hay que destacar que para poder medir el pie sobre la fotografía se ha tenido que incluir una referencia, en este caso tendríamos que fotografiarlo junto con una tarjeta de crédito en el sentido longitudinal de esta y en función de esto el usuario tendrá que mover los renglones para que obtener una medida muy aproximada de su pie.

A continuación se muestra el diagrama con las principales clases de la aplicación:

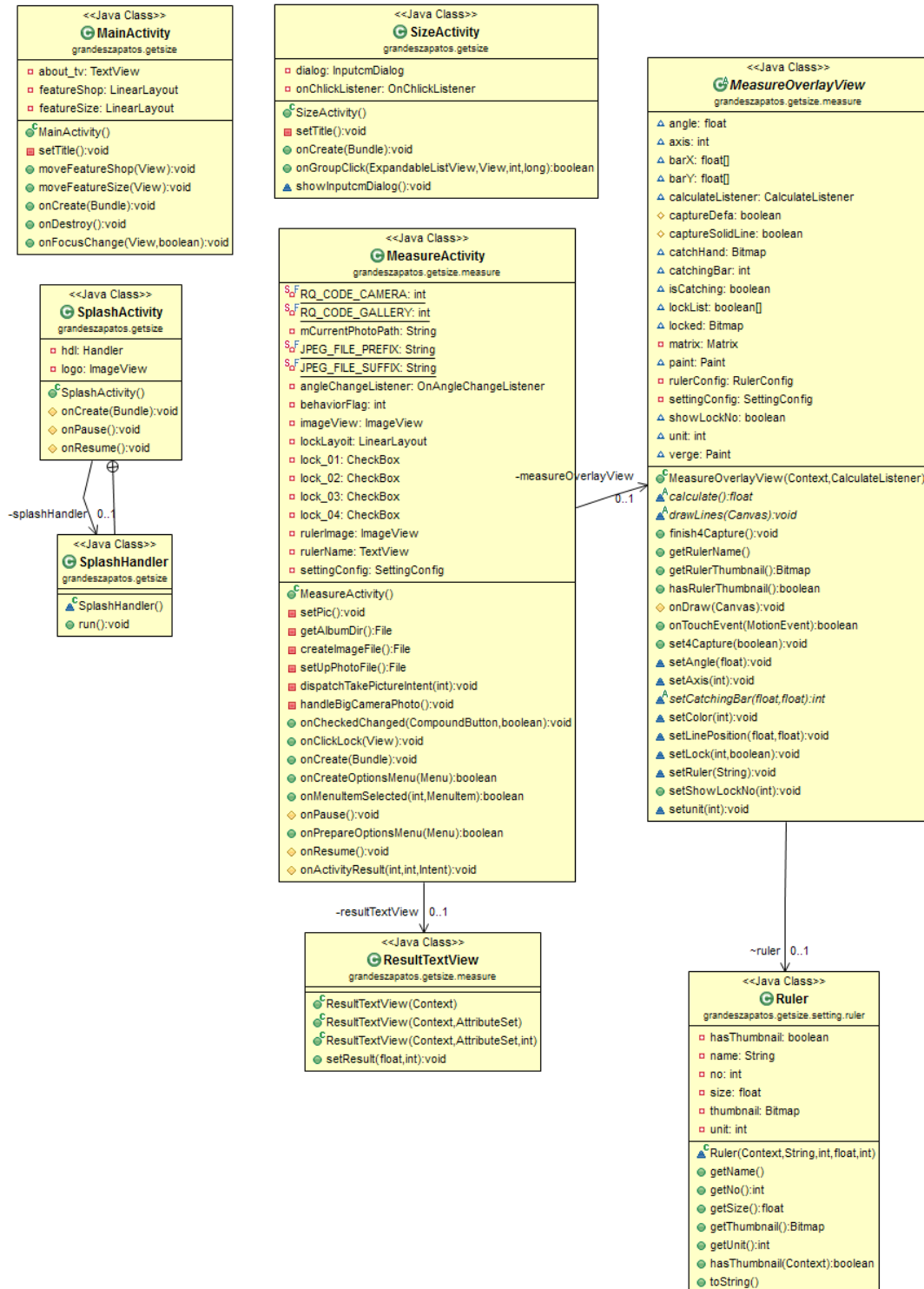


Fig. 10: Diagrama de Clases de la aplicación

A continuación se explica de forma resumida cada una de las principales clases de la aplicación, según el orden en que se han mostrado la navegación a través de distintas pantallas :

- **SplashHandler:** Intro de la aplicación.
- **MainActivity:** Pantalla principal de la aplicación donde se permite elegir entre acceder directamente a la tienda on-line (*moveFeatureShop*) o buscar una recomendación de talla (*moveFeatureSize*).
- **SizeActivity:** Clase donde se da a elegir entre obtener la talla a través de una fotografía del pie, o bien una ventana de diálogo con el usuario en el que este puede introducir la medida de su pie en cm. En el caso de que se opte por realizar la fotografía del pie (junto con la tarjeta de crédito), se llamará directamente (mediante un Intent) a la aplicación de la cámara propia del sistema operativo Android y una vez guardada nos aparecerán los renglones para marcar las referencias.
- **MeasureActivity:** Actividad con las funciones de trabajo con la cámara fotográfica del teléfono (SetPic, getAlbumDir, createImageFile, dispatchTakePictureIntent...)
- **MeasureOverLayView:** Define el manejo de las referencias de medida, su visualización en pantalla para que el usuario opere con ellas y el proceso de cálculo y manejo de los mismos (onTouchEvent, Calculate...)

V.4 Diseño del Modelo de datos

Para nuestra aplicación vamos a utilizar una base de datos por defecto, es decir, utilizaremos un fichero de base de datos externo creado con un editor de bases de datos SQLite (**SQLite Database Browser [23]**).

Para preparar la base de datos creamos un nuevo fichero llamado *catalogo* y ejecutamos las siguientes sentencias SQL :

1. CREATE TABLE "android_metadata" ("locale" TEXT DEFAULT 'es_ES')
2. INSERT INTO "android_metadata" VALUES ('es_ES')

A continuación creamos una tabla llamada catálogo con el diseño de nuestros datos (primary key, cm, TallaGZ, TallaFabricante, Modelo). Tenemos que tener como nombre de la clave primaria de las tablas de nuestra base de datos a “_id” para que Android pueda identificarlas. Se puede ver como quedaría en la imagen siguiente:

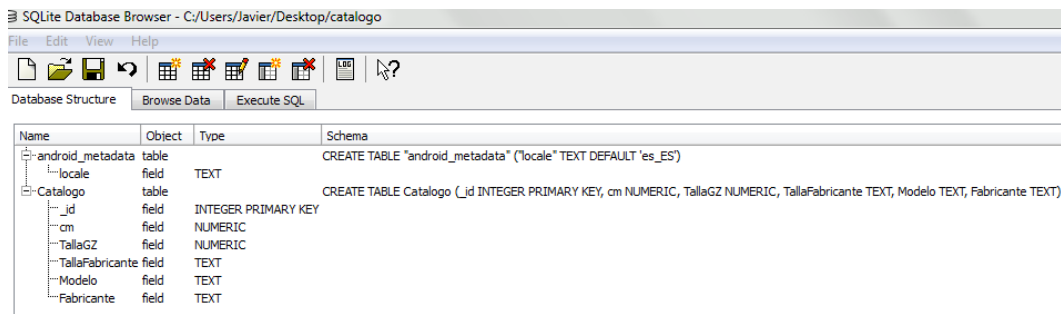


Fig. 11: Diseño de la Base de Datos del Catálogo para la Aplicación Android

La tabla del catálogo creado se puede alimentar de forma sencilla con la misma aplicación, utilizando la función para importar un fichero csv, donde tendremos todos los datos del catálogo de la tienda on-line según son requeridos por la aplicación.

Capítulo VI Implementación del Sistema

VI.1 Entorno de Desarrollo

Eclipse es el entorno de desarrollo utilizado en este proyecto para llevar a cabo la implementación de la aplicación Android. Eclipse es una comunidad de código abierto cuyos proyectos están enfocados en crear una plataforma de desarrollo de software libre.

El Proyecto Eclipse fue creado originalmente por **IBM** (*International Business Machines*) en Noviembre del 2001. Posteriormente fue albergado por la Fundación Eclipse, que tiene entre sus objetivos fomentar, tanto una comunidad de código abierto, como un ecosistema de productos y servicios complementarios.

Algunas de las principales ventajas, y por consiguiente, razones de la elección de Eclipse a la hora de implementar la aplicación para Android:

- Eclipse es uno de los más completos **IDEs** de Java disponibles en la actualidad. Es además muy fácil de usar, por lo que requiere poco tiempo de aprendizaje.
- El Plug-in **ADT** para Eclipse lanzado por la **OHA**, permite la creación y compilación de proyectos Android, y su ejecución y depuración mediante el emulador de Android.

VI.2 Estructura de la aplicación y manifiesto

En la siguiente tabla se muestran los principales componentes de una aplicación Android:

Directorio	Fichero	Propósito
/	AndroidManifest.xml	Información de la aplicación que será mostrada al sistema Android, necesitamos acceso a Internet y permisos para el uso de la cámara: <i>android.permission.WRITE_EXTERNAL_STORAGE</i> <i>android.permission.INTERNET</i> <i>android.permission.ACCESS_NETWORK_STATE</i>
/res/layout		Contiene la definición del interfaz gráfico
/res/values	Strings.xml	Contiene los valores de las cadenas que serán utilizados en la aplicación
/res/drawable		Contiene todas las imágenes a utilizar por el interfaz gráfico
/gen	R.java	Contiene los identificadores de los elementos contenidos en main.xml y string.xml. Es un fichero que no debe ser modificado por el usuario, se modifica automáticamente en función de los cambios en los ficheros de arriba.

Tabla 6: Ficheros específicos aplicaciones Android.

Una aplicación Android está definida por el contenido de su manifiesto. Cada aplicación Android declara todas sus actividades, puntos de entrada, interfaces de comunicación, permisos e *intents* a través del fichero **AndroidManifest.xml** :

```
<?xml version="1.0" encoding="utf-8"?>
<manifest android:versionCode="1"
    android:versionName="0.0.1"
    package="grandeszapatos.camerameasure"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <uses-permission android:name="android.permission.CAMERA" />
    <uses-feature android:name="android.hardware.camera"/>
    <uses-feature android:name="android.hardware.camera.autofocus"/>
    <uses-sdk android:minSdkVersion="8" />
    <supports-screens
        android:anyDensity="true"
        android:smallScreens="true"
        android:normalScreens="true"
        android:largeScreens="true" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <application
        android:theme="@style/CustomTheme.Black"
        android:label="@string/app_name"
        android:icon="@drawable/icono"
        android:debuggable="true">
        <activity
            android:theme="@android:style/Theme.Black.NoTitleBar.Fullscreen"
            android:name=".SpLashActivity"
            android:screenOrientation="Landscape">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:theme="@android:style/Theme.Black.NoTitleBar.Fullscreen"
            android:name=".MainActivity"
            android:clearTaskOnLaunch="true"
            android:screenOrientation="Landscape"
            android:configChanges="keyboardHidden|orientation">
        </activity>
        <activity
            android:theme="@android:style/Theme.Black.NoTitleBar.Fullscreen"
            android:name=".LengthActivity"
            android:clearTaskOnLaunch="true"
            android:screenOrientation="Landscape"
            android:configChanges="keyboardHidden|orientation" />
        </activity>
        <activity
            android:theme="@android:style/Theme.Black.NoTitleBar.Fullscreen"
            android:name=".measure.MeasureActivity"
            android:clearTaskOnLaunch="true"
            android:screenOrientation="Landscape"
            android:configChanges="keyboardHidden|orientation" />
        </activity>
        <activity
            android:theme="@android:style/Theme.Black.NoTitleBar.Fullscreen"
            android:name=".measure.InputcmActivity"
            android:clearTaskOnLaunch="true"
            android:screenOrientation="Landscape"
            android:configChanges="keyboardHidden|orientation" />
        </activity>
    </application>
</manifest>
```

En el manifiesto se puede ver que la aplicación define permisos para utilización de la Cámara del teléfono, para el grabar datos en memoria y para el Acceso a Internet. Se declaran los elementos constitutivos de la misma.

Una aplicación Android se puede componer de 4 bloques principales :

- **Activity** : Es el bloque básico de la aplicación. Una actividad usa una serie de funciones para interactuar con su entorno. Específicamente, una actividad debe sobrecargarse con el método *onCreate*. También es recomendable sobrecargar los métodos *onStop*, *onPause*, *onResume* y *onKeyDown*. Con estas funciones se consigue manejar de forma más precisa la aplicación.

- **Intent Receiver:** Un objeto “reactivo” lanzado para manejar una tarea específica. Se encuentra a la escucha de *intents* para poder manejarlas cuando son recibidas.
- **Service:** Es un proceso lanzado en segundo plano y por lo tanto que no tendrá ninguna interacción con el usuario.
- **Content Provider:** *Framework* para el manejo y almacenamiento de datos.

En el manifiesto vemos que nuestra aplicación se compone unicamente de varias **Activity**, que se corresponden con las pantallas mostradas en los diagramas de navegación.

VI.3 Actividad Principal

A continuación se muestra parte del listado de código de la Actividad principal o pantalla principal de la aplicación :

```
package grandeszapatos.camerameasure;
import grandeszapatos.camerameasure.setting.SuccessionUtil;
import grandeszapatos.camerameasure.R;
import android.widget.TextView;

public class MainActivity extends Activity implements OnFocusChangeListener {

    private TextView about_tv;
    private LinearLayout featureShop;
    private LinearLayout featureLength;

    private void setTitle() {
        .....
    }

    public void moveFeatureShop(View var1) {
        Intent browserIntent = new Intent(Intent.ACTION_VIEW, Uri.parse("http://www.grandeszapatos.com"));
        startActivity(browserIntent);
    }

    public void moveFeatureLength(View var1) {
        Intent LenghtActivity = new Intent();
        LenghtActivity.setClass(this, LengthActivity.class);
        LenghtActivity.putExtra("action_flags", 0);
        this.startActivity(LenghtActivity);
    }

    public void onCreate(Bundle var1) {
        super.onCreate(var1);
        this.setContentView(R.layout.dashboard);
        SuccessionUtil.succession(this.getApplicationContext());
        this.setTitle();
    }

    public void onDestroy() {
        super.onDestroy();
    }

    public void onFocusChange(View var1, boolean var2) {
        if(var2) {
            switch(var1.getId()) {
                case R.id.feature_shop_item:
                    this.about_tv.setText(R.string.about_shop);
                    break;
                case R.id.feature_length_item:
                    this.about_tv.setText(R.string.about_length);
                    break;
            }
        }
    }
}
```

Se corresponde con la interacción del usuario con la pantalla principal de la aplicación, dónde este tendrá dos opciones: acceder directamente a través del navegador web o bien obtener la talla que le venga bien del catálogo de la tienda.

Capítulo VII Pruebas e Implantación

En este capítulo se explica el proceso de depuración, pruebas y pasos necesarios para publicar una aplicación Android. El diagrama esquemático de dicho proceso se puede ver en la Fig.12:

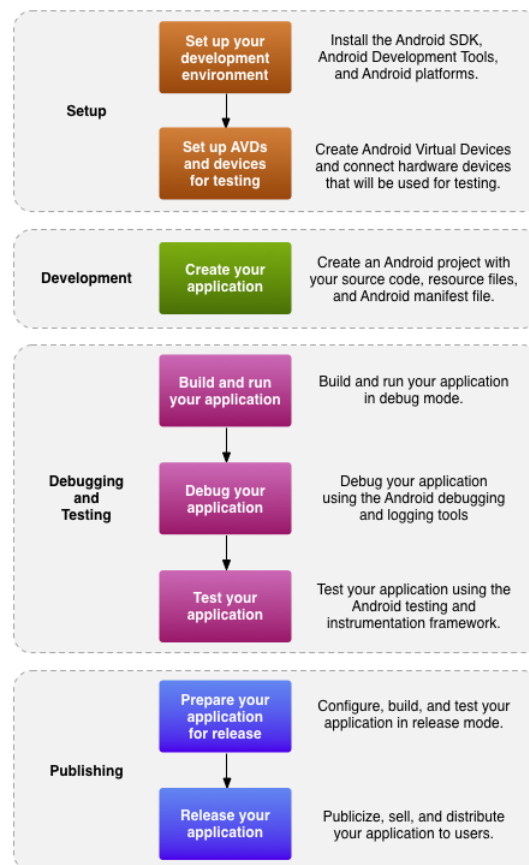


Fig. 12: Proceso Creación y Publicación Aplicación Android

Una vez codificada la aplicación seguimos los siguientes pasos:

1. Compilarla en modo depuración y ejecutar pruebas
2. Depuramos la aplicación con **Eclipse** y **LogCat** para ver las incidencias que ocurran.
3. Podríamos haber utilizado las potentes herramientas de testing que ofrece Android (JUnit, Monkeyrunner) o herramientas de automatización como **Robotium [24]**.
4. Configurar la aplicación para modo *release* y probarla en terminales reales.
5. Publicación de la aplicación (toda aplicación Android debe estar firmada digitalmente)

Capítulo VIII Conclusiones

Para finalizar la memoria de este proyecto, se reseñan las principales conclusiones obtenidas al acabar el mismo :

VIII.1 Logros Alcanzados

En este trabajo se ha desarrollado una aplicación móvil para dispositivos con tecnología Android, basada en software libre, que permite obtener de forma aproximada la talla de zapatos que usaría un usuario para determinado fabricante.

VIII.2 Conocimiento adquirido

En general la experiencia ha sido muy positiva. Personalmente he aprendido interesantes lecciones en los siguientes puntos:

- Profundización en el desarrollo **Android**, comprobando las facilidades que **Google** ofrece a los programadores, en cuanto a documentación y herramientas de desarrollo, y el potencial que tiene.
- Consulta de documentación, referencias y foros de desarrollo de software.
- Uso del IDE **Eclipse** para desarrollo y test de aplicaciones Android
- Conocimientos plataforma *e-commerce* Software Libre **Prestashop**, administración y creación de módulos.
- Elaboración y planificación de proyectos basados en Software Libre.

Considero que he tenido que utilizar demasiado tiempo en investigación del uso de librerías como **OpenCV**, implementación módulos Prestashop y configuración de la tienda virtual, así como en investigación sobre la forma de realizar las tallas del calzado, para lo cual también intenté sin éxito que me atendieran en el **INESCOP [25]**, todo lo cual se ha tenido que descartar del proyecto por diversas razones. Igualmente todo ello lo considero de provecho, pese a que dicho tiempo se podría haber invertido en mejoras en el proyecto.

VIII.3 Trabajo futuro

Como continuación a este proyecto se pueden proponer distintas ampliaciones:

- Que una vez obtenida la recomendación el usuario pueda realizar la compra en la tienda on-line
- Mejoras en el sistema de medida del pie del usuario

Apéndices

Apéndice I: Nociones básicas de Fotogrametría de objetos cercanos.

Fotogrametría según SAF (asociación americana de Fotogrametría) es el conjunto de ciencia, arte, y tecnología que obtiene informaciones fiables acerca de objetos físicos y su entorno mediante procesos de registro, medida e interpretación de imágenes fotográficas y datos obtenidos a partir de energía electromagnética radiante o de fenómenos magnéticos.

Restituidor

Instrumento capaz de obtener medidas precisas sobre imágenes fotográficas y que permite la formación del modelo tridimensional. Necesita un mecanismo de visión estereoscópica

Se utiliza para Aplicaciones no topográficas (Fotogrametría Terrestre): Cualquier cosa que se pueda fotografiar desde dos posiciones.

Por ejemplo para determinar Forma y dimensiones de objetos fijos

- Estructuras para obtener deformaciones, despieces constructivos y realizar restauraciones.
- Arquitectura y arqueología.
- Criminología

En todos los casos el método es el mismo:

- Toma de vistas, (al menos dos) con cámara especial.
- Determinación de los elementos externos de cada haz.
- Identificación de los rayos homólogos.
- Búsqueda de intersección de estos rayos.

El objetivo final siempre es el mismo; obtener coordenadas a partir de fotografías.

Referencias Bibliográficas

- [1] Realidad Aumentada [on-line] URL: http://es.wikipedia.org/wiki/Realidad_aumentada
- [2] Proyecto Software Libre Android [on-line] URL: <http://www.prestashop.com/es/>
- [3] Prestashop [on-line] URL: <http://source.android.com/>
- [4] Aplicación iFootGauge [on-line] URL: <https://itunes.apple.com/us/app/ifootgauge-whats-your-shoe/id412793207?mt=8>
- [5] Norma UNE 59850:1998 [on-line] URL: <http://www.aenor.es/aenor/normas/normas/fichanorma.asp?tipo=N&codigo=N0006514&PDF=Si#.UMzHrnePx68>
- [6] Librerías OpenCV [on-line] URL: <http://docs.opencv.org/>
- [7] Fotogrametría aérea, terrestre y de objeto cercano [on-line] URL: <http://www.rediris.es/list/info/fotogrametria.html>
- [8] Open Handset Alliance [on-line] URL: <http://www.openhandsetalliance.com/>
- [9] Plataformas Android [on-line] URL: <http://developer.android.com/resources/dashboard/platform-versions.html>
- [10] Proyecto Eclipse [on-line] URL: <http://www.eclipse.org/>
- [11] Kit de desarrollo Java [on-line] URL: <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- [12] Herramientas de desarrollo Android [on-line] URL: <http://developer.android.com/intl/es/tools/index.html>
- [13] Guía de diseño de aplicaciones Android [on-line]: <http://developer.android.com/intl/es/design/index.html>
- [14] Guía de estilo de codificación Android [on-line] URL: <http://source.android.com/source/code-style.html>
- [15] Guía del diseñador de Prestashop [on-line] URL: <http://doc.prestashop.com/pages/viewpage.action?pageId=3506387>

- [16] Guía del desarrollador de Prestashop [on-line] URL: <http://doc.prestashop.com/pages/view-page.action?pageId=3506384>

- [17] Craig Larman: UML y Patrones. Introducción al análisis y diseño orientado a objetos.

- [18] Patrones de diseño Android [on-line] URL: <http://www.androidpatterns.com/>

- [19] Web Services REST [on-line] URL:
http://en.wikipedia.org/wiki/Representational_state_transfer

- [20] Base de Datos SQLite [on-line] URL: <http://www.sqlite.org/>

- [21] Modelo Vista Controlador [on-line] URL: http://es.wikipedia.org/wiki/Modelo_Vista_Controlador

- [22] Editor interfaz gráfica para Android [on-line] URL: <http://www.droiddraw.org/>

- [23] Diseñador de bases de datos SQLite [on-line] URL: <http://sqlitebrowser.sourceforge.net/>

- [24] Test Framework Robotium [on-line] URL: <http://code.google.com/p/robotium/>

- [25] INESCOP (Instituto Tecnológico del Calzado y Conexas) [on-line] URL: <http://www.inescop.es/>