



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior de Linares

Trabajo Fin de Grado

APLICACIÓN WEB/MOVIL PARA LA GESTIÓN DE COMPRA-VENTA DE ARTÍCULOS DE SEGUNDA MANO

Alumno: Pablo Cueto Siles

Tutor: Ángel Aguilera García
Depto.: Informática

Septiembre, 2017

AGRADECIMIENTOS

La finalización de este trabajo fin de grado supone el punto final de cuatro años y la satisfacción de acabar con éxito los estudios universitarios en la Universidad de Jaén.

Para llegar hasta aquí ha sido necesario contar con personas que me han apoyado y ayudado constantemente y por esto y muchas cosas más, tengo que dar las gracias a mis padres, José Manuel y Amor, que mediante sus consejos me han enseñado el significado del sacrificio y el trabajo duro.

También agradecer la ayuda recibida por parte de mi tutor, Ángel, que me ha ayudado dándome consejos durante todo el trabajo, me ha facilitado información necesaria para poder realizarlo y ha estado disponible siempre que lo he necesitado.

Gracias a todos mis amigos que me han ayudado cuando lo necesitaba y mención especial para todos los compañeros que me han acompañado a lo largo de esta parte tan importante de mi vida como lo han sido Juan Carlos, David, Luis y Macarena.

Y, por supuesto, gracias a mi novia Estíbaliz por aguantarme todos estos años con su apoyo constante.

RESUMEN

El diseño y desarrollo de tiendas online ha crecido exponencialmente los últimos años buscando una sensación de compra en una tienda física. Las empresas deben adaptarse a las nuevas tecnologías que avanzan rápidamente, buscando las nuevas tendencias y gustos del consumidor a la hora de realizar compras por internet. Esta adaptación es muy importante ya que la importancia del desarrollo y diseño de sus tiendas online es de vital importancia para poder competir en el mercado.

El sistema de ventas ha cambiado. Antes se compraba todo en tiendas físicas, pero ahora la comodidad y las prestaciones que ofrecen las empresas que venden sus productos y servicios en tiendas online han hecho que este tipo de tiendas crezca.

En este trabajo se ha diseñado y desarrollado una aplicación web con software libre para la compra/venta de artículos nuevos y usados, dando la oportunidad a empresas y particulares poder comprar y vender artículos de todo tipo.

Dada la importancia de los dispositivos móviles y la competencia en el mercado con otras empresas que disponen de tiendas online, también se ha desarrollado en este trabajo una aplicación móvil para los dispositivos compatibles con el SO (sistema operativo) Android, ya que suponen más del sesenta por ciento de todos los dispositivos móviles vendidos mundialmente. Con esta aplicación se tendrán disponibles los mismos servicios que se proporcionan en la tienda online sin necesidad de tener que usar un navegador web.

ABSTRACT

The design and development of online stores has grown exponentially in recent years looking for a sense of shopping in a physical store. Companies must adapt to new technologies that are rapidly advancing, looking for new trends and consumer tastes when making purchases online. This adaptation is very important since the importance of the development and design of it online stores is of vital importance to be able to compete in the market.

The sales system has changed. Previously, everything was bought in physical stores, but now the comfort and benefits offered by companies that sell their products and services in online stores have made this type of store grows.

In this work a web application with free software for the purchase and sale of new and used articles has been designed and developed, giving the opportunity to companies and private persons to be able to buy and sell articles of all kinds.

Given the importance of mobile devices and competition in the market with other companies that have online stores, a mobile application for devices compatible with the Android operating system (OS) has also been developed in this work, because the sixty percent of all mobile devices sold worldwide. With this application will be available the same services that are provided in the online store without having to use a web browser.

ÍNDICE

Agradecimientos.....	1
Resumen.....	2
Abstract.....	3
Índice de ilustraciones	7
Índice de tablas	10
Índice de acrónimos	13
1 Introducción	14
1.1 Motivación.....	14
1.2 Objetivos.....	16
1.3 Fases de desarrollo.....	16
1.4 Medios empleados	17
2 Análisis del estado del arte.....	18
2.1 Páginas Web.....	18
2.1.1 Evolución del desarrollo de páginas Web.....	18
2.1.2 CMS para desarrollar páginas Web	20
2.1.2.1 WordPress.....	22
2.1.2.2 Adobe Dreamweaver	22
2.1.2.3 Joomla!.....	23
2.1.2.4 Drupal.....	24
2.1.3 Elección de una plataforma para el trabajo	25
2.2 Aplicaciones Android.....	26
2.2.1 Evolución de la telefonía móvil	26
2.2.2 Principales sistemas operativos para móviles	27
2.2.2.1 Android.....	29
2.2.2.2 iOS	30
2.2.2.3 Windows Phone.....	31
2.2.2.4 BlackBerry	32
2.2.3 Elección de una plataforma para el trabajo	32
2.3 Repaso a la historia de tiendas online	33

2.3.1	Información general	34
2.3.2	Privacidad y seguridad	35
2.3.3	Historia.....	35
3	Plataformas de trabajo: wordpress y android.....	37
3.1	WordPress	37
3.1.1	Arquitectura de WordPress	39
3.1.2	Métodos de instalación de WordPress	42
3.1.2.1	Instalación de WordPress en un Servidor LOCAL	42
3.1.2.2	Instalación de WordPress en un Servidor o Hosting	46
3.1.3	Base de datos de WordPress.....	47
3.1.3.1	Tablas	47
3.1.3.2	Diagrama Entidad-Relación de las tablas de WordPress.....	49
3.2	Android	50
3.2.1	Arquitectura de Android.....	51
3.2.1.1	Kernel de Linux.....	52
3.2.1.2	Runtime de Android (RTA).....	52
3.2.1.3	Librerías	53
3.2.1.4	Armazón de aplicaciones o entorno de aplicación	53
3.2.1.5	Aplicaciones	54
3.2.2	Entornos de desarrollo	54
3.2.3	Versiones de Android.....	55
3.2.4	Componentes de una aplicación Android	65
3.2.4.1	Vista (View)	66
3.2.4.2	Layout	66
3.2.4.3	Actividad (Activity)	66
3.2.4.4	Servicio (Service)	66
3.2.4.5	Intención (Intent).....	66
3.2.4.6	Receptor de anuncios (Broadcast receiver)	67
3.2.4.7	Proveedores de contenido (Content Provider)	67
3.2.5	Ciclo de vida de una aplicación Android.....	67
3.2.6	Android Studio	70
4	Análisis y Diseño.....	72
4.1	Análisis	72

4.1.1	Planteamiento del problema.....	72
4.1.2	Planteamiento de la solución	72
4.1.3	Requisitos y marco regulatorio	73
4.1.3.1	Requisitos.....	73
4.1.3.2	Marco regulatorio.....	74
4.1.4	Requisitos de usuario.....	74
4.1.4.1	Requisitos de usuario de capacidad	75
4.1.4.2	Requisitos de usuario restrictivos	79
4.1.5	Casos de uso	81
4.2	Diseño.....	93
4.2.1	Diseño de la arquitectura de la aplicación Web y Android	93
4.2.2	Diseño de la base de datos.....	98
4.2.3	Diseño de la interfaz	102
5	Desarrollo.....	109
5.1	Aplicación Web con WordPress (IdlaShop)	109
5.2	Aplicación Android con Android Studio.....	115
6	Planificación y presupuesto.....	129
6.1	Planificación.....	129
6.2	Presupuesto.....	131
7	Conclusiones y líneas futuras.....	133
7.1	Conclusiones	133
7.2	Líneas futuras	134
8	Anexos.....	136
8.1	Anexo 1: Como instalar un servidor local	136
8.2	Anexo 2:Legislación	138
8.3	Anexo 3: Pasos para crear un proyecto en Android Studio	144
8.4	Anexo 4: Firebase y como crear un proyecto	145
8.5	Anexo gráfico	147
9	Referencias.....	149

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Ejemplo de Web 1.0	18
Ilustración 2: Web 3.0	20
Ilustración 3: Uso CMS	25
Ilustración 4: Evolución de la telefonía móvil	27
Ilustración 5: Evolución conectividad móvil	27
Ilustración 6: Cuota de mercado SO móviles mediados 2017	28
Ilustración 7: Android	29
Ilustración 8: iOS	30
Ilustración 9: Windows Phone	31
Ilustración 10: BlackBerry	32
Ilustración 11: Nº de aplicaciones disponibles en tiendas móviles. Marzo 2017	33
Ilustración 12: Amazon en 1995	36
Ilustración 13: Icono de WordPress	39
Ilustración 14: Carpetas WordPress	39
Ilustración 15: Carpeta wp-content	40
Ilustración 16: Carpeta wp-content +	41
Ilustración 17: Carpeta wp-content ++	42
Ilustración 18: Acceso a la página del servidor local	43
Ilustración 19: PhpMyAdmin	44
Ilustración 20: Página principal de WordPress.....	45
Ilustración 21: Diagrama E-R de la base de datos de WordPress.....	49
Ilustración 22: T-mobile G1, primer dispositivo Android	51
Ilustración 23: Arquitectura de Android	52
Ilustración 24: Android 1.5 Cupcake	55
Ilustración 25: Android 1.6 Donut.....	56
Ilustración 26: Android 2.0/2.1 Éclair	56
Ilustración 27: Android 2.2 Froyo	57
Ilustración 28: Android 2.3 Gingerbread	58
Ilustración 29: Android 3.0 Honeycomb	58
Ilustración 30: Android 4.0 Ice Cream Sandwich	60
Ilustración 31: Android 4.1 Jelly Bean.....	60
Ilustración 32: Android 4.4 KitKat.....	61
Ilustración 33: Android 5.0 Lollipop.....	61
Ilustración 34: Android 6.0 Marshmallow	62
Ilustración 35: Android 7.0 Nougat.....	63

Ilustración 36: Android 8.0 Android O	63
Ilustración 37: Cuota de mercado versiones Android	65
Ilustración 38: Ciclo de vida de una actividad Android	69
Ilustración 39: Entorno usado	71
Ilustración 40: Entorno usado 2	71
Ilustración 41: Casos de uso	82
Ilustración 42: Casos de uso 2	90
Ilustración 43: Arquitectura aplicaciones	94
Ilustración 44: Conexión con el servidor	95
Ilustración 45: Servidor	99
Ilustración 46: Tablas base de datos 1	100
Ilustración 47: Tablas base de datos 2	100
Ilustración 48: Base de datos	101
Ilustración 49: Tema Zerif Lite	102
Ilustración 50: Página principal	103
Ilustración 52: Escritorio de vendedor	104
Ilustración 51: Página de Contacto	104
Ilustración 53: Página de nuevo producto	105
Ilustración 54: Página del carrito	106
Ilustración 55: Página de finalizar compra 1	106
Ilustración 56: Página de finalizar compra 2	107
Ilustración 57: Menú principal y páginas de inicio	107
Ilustración 58: Páginas Menú, Acceder y Contacto	108
Ilustración 59: Pasos del hosting	109
Ilustración 61: Añadir base de datos al hosting	110
Ilustración 60: Elección de WordPress en el hosting	110
Ilustración 62: Plugins de WordPress	111
Ilustración 63: Ajustes de WooCommerce	115
Ilustración 64: Carpetas proyecto Android Studio 1	116
Ilustración 65: Carpetas proyecto Android Studio 2	116
Ilustración 66: Vista de layout vertical	117
Ilustración 67: Ajustes de layout	118
Ilustración 68: Vista de layout horizontal	118
Ilustración 69: Incorporación de WebView	119
Ilustración 70: Datos de una notificación	128
Ilustración 71: Envío de una notificación	128
Ilustración 72: Comprobación de la notificación	128

Ilustración 73: Diagrama de Gantt	130
Ilustración 74: MAMP 1	137
Ilustración 75: MAMP 2	137
Ilustración 76: Proyecto Android Studio 1	144
Ilustración 77: Proyecto Android Studio 2	144
Ilustración 78: Proyecto Android Studio 4	144
Ilustración 79: Proyecto Android Studio 5	144

ÍNDICE DE TABLAS

Tabla 1: CMS más populares	21
Tabla 2: Ventajas e inconvenientes de WordPress.....	22
Tabla 3: Ventajas e inconvenientes de Joomla!	23
Tabla 4: Ventajas e inconvenientes de Drupal.....	24
Tabla 5: Configuración wp-config	45
Tabla 6: Tablas base de datos de WordPress	48
Tabla 7: Requerimientos de Android Studio.....	70
Tabla 8: Requisitos de usuario de capacidad 1	75
Tabla 9: Requisitos de usuario de capacidad 2	75
Tabla 10: Requisitos de usuario de capacidad 3	76
Tabla 11: Requisitos de usuario de capacidad 4	76
Tabla 12: Requisitos de usuario de capacidad 5	76
Tabla 13: Requisitos de usuario de capacidad 6	76
Tabla 14: Requisitos de usuario de capacidad 7	76
Tabla 15: Requisitos de usuario de capacidad 8	77
Tabla 16: Requisitos de usuario de capacidad 9	77
Tabla 17: Requisitos de usuario de capacidad 10	77
Tabla 18: Requisitos de usuario de capacidad 11	77
Tabla 19: Requisitos de usuario de capacidad 12	78
Tabla 20: Requisitos de usuario de capacidad 13	78
Tabla 21: Requisitos de usuario de capacidad 14	78
Tabla 22: Requisitos de usuario de capacidad 15	79
Tabla 23: Requisitos de usuario de capacidad 16	79
Tabla 24: Requisitos de usuario de capacidad 17	79
Tabla 25: Requisitos de usuario restrictivos 1.....	79
Tabla 26: Requisitos de usuario restrictivos 2.....	80
Tabla 27: Requisitos de usuario restrictivos 3.....	80
Tabla 28: Requisitos de usuario restrictivos 4.....	80
Tabla 29: Requisitos de usuario restrictivos 5.....	80
Tabla 30: Casos de uso 1.....	82
Tabla 31: Casos de uso 2.....	83
Tabla 32: Casos de uso 3.....	83
Tabla 33: Casos de uso 4.....	84
Tabla 34: Casos de uso 5.....	84
Tabla 35: Casos de uso 6.....	85

Tabla 36: Casos de uso 7	85
Tabla 37: Casos de uso 8.....	86
Tabla 38: Casos de uso 9.....	86
Tabla 39: Casos de uso 10.....	86
Tabla 40: Casos de uso 11.....	87
Tabla 41: Casos de uso 12.....	87
Tabla 42: Casos de uso 13.....	88
Tabla 43: Casos de uso 14.....	88
Tabla 44: Casos de uso 15.....	88
Tabla 45: Casos de uso 16.....	89
Tabla 46: Casos de uso 17.....	89
Tabla 47: Casos de uso 18.....	89
Tabla 48: Casos de uso 19.....	90
Tabla 49: Casos de uso 20.....	90
Tabla 50: Casos de uso 21.....	91
Tabla 51: Casos de uso 22.....	91
Tabla 52: Casos de uso 23.....	91
Tabla 53: Casos de uso 24.....	92
Tabla 54: Casos de uso 25.....	92
Tabla 55: Casos de uso 26.....	93
Tabla 56: Casos de uso 27.....	93
Tabla 57: Arquitectura página web 1	95
Tabla 58: Arquitectura página web 2	96
Tabla 59: Arquitectura página web 3	96
Tabla 60: Arquitectura página web 4	97
Tabla 61: Arquitectura página web 5	97
Tabla 62: Arquitectura página web 6	98
Tabla 63: Arquitectura página web 7	98
Tabla 64: Arquitectura página web 8	98
Tabla 65: Plugin WooCommerce	111
Tabla 66: Plugin Advenced CSS.....	111
Tabla 67: Plugin Asesor de Cookies.....	112
Tabla 68: Plugin Contact Form	112
Tabla 69: Plugin Pirate Forms	112
Tabla 70: Plugin Loco Translate	112
Tabla 71: Plugin Ofertas para WooCommerce	112
Tabla 72: Plugin Redsys WooCommerce	113

Tabla 73: Plugin Theme my login	113
Tabla 74: Plugin WooCommerce Vendors	113
Tabla 75: Plugin Woo Checkout Field Editor.....	113
Tabla 76: Plugin WooCommerce Dynamic Galery LITE	113
Tabla 77: Plugin WooCommerce PDF Invoices & Packing Slips.....	113
Tabla 78: Plugin WooCommerce Product FAQs.....	114
Tabla 79: Plugin WooCommerce Simple Auctions.....	114
Tabla 80: Plugin WooCommerce Products Filter	114
Tabla 81: Plugin YITH WooCommerce Social Login.....	114
Tabla 82: Plugin YITH WooCommerce Wishlist.....	114
Tabla 83: Plugin wp-pro-quiz	115
Tabla 84: Tareas del trabajo	129
Tabla 85: Presupuesto	132

ÍNDICE DE ACRÓNIMOS

ADT	Android Developer Tools
API	Applications Programming Interface (Interfaz de Programación de Aplicaciones)
CERN	Organización Europea de Investigación Nuclear
CIF	Código de Identificación Fiscal
CMS	Content Management System (Sistemas de Gestión de Contenidos)
CMNC	Comisión Nacional de los Mercados y la Competencia
EE.UU	Estados Unidos
FECEMD	Federación Española de Marketing Directo y Comercio Electrónico
FTP	File Transfer Protocol (Protocolo de Transferencia de Archivos)
IAE	Impuesto de Actividades Económicas
IDE	Integrated Development Enviroment (Entorno de desarrollo Integrado)
JRE	Java Runtime Enviroment (Entorno de Ejecución Java)
LAN	Local Area Network (Redes de Área Local)
LOCM	Ley de Ordenación del Comercio Minorista
LOPD	Ley Oficial de Protección de Datos
LSSI	Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico
MAMP	Mac, Apache, MySQL y PHP
MVC	Modelo-Vista-Controlador
NDK	Native Development Kit (Kit de Desarrollo Nativo)
NFC	Near Field Communication (Cenca del Campo de Comunicación)
PDA	Personal Digital Assistant (Asistente Personal Digital)
RGPD	Reglamento General de Protección de Datos
RTA	Runtime de Android
SDK	Software Development Kit (Kit de Desarrollo Software)
SEO	Search Engine Optimization (Optimización en motores de Búsqueda)
SO	Sistema Operativo
TFG	Trabajo Fin de Grado
VoIP	Voice Over IP (Voz Sobre IP)
VPN	Virtual Private Network (Red Privada Pública)
WPAN	Wireless Personal Area Network (Redes Inalámbricas de Área Local)
WWW	World Wide Web (Red Mundial Web)

1 INTRODUCCIÓN

En esta primera parte se va a dar a conocer una idea general de lo que el lector se va a encontrar a lo largo de este documento.

El presente documento recoge toda la información del desarrollo llevado a cabo para el Trabajo Final de Grado (TFG). La idea para realizar este tipo de trabajo surge por la necesidad de comprobar los conocimientos obtenidos durante la realización de los estudios universitarios. El resultado es una tienda, en la cual, cualquier usuario, puede registrarse para poder comprar y vender productos de todo tipo de otros usuarios también registrados. Para esta tienda se han desarrollado una aplicación Web y una aplicación Android con la idea de que los usuarios dispongan de los mismos servicios que proporcionan ambas aplicaciones.

El funcionamiento, tanto de la tienda online como de la aplicación Android, se basa fundamentalmente en una base de datos compartida, la cual almacena todos los datos de los usuarios que quieran disponer de los servicios ofrecidos. Para disponer de dichos servicios, cada usuario deberá registrarse con un nombre de usuario y un correo electrónico donde se enviará una solicitud para confirmar la identidad de dicho usuario. Una vez registrado, podrá completar su cuenta con sus datos personales que serán privados y los datos de su “tienda personal” que serán públicos. Estos datos se guardarán en la base de datos y harán que el usuario se postule como comprador y vendedor por defecto.

Para la realización de este trabajo ha sido muy importante la motivación y conocer desde el principio los objetivos que se querían alcanzar.

1.1 Motivación

La motivación principal es el aprendizaje del funcionamiento de aplicaciones web y Android ya que ofrecen muchas posibilidades de desarrollo, son de código abierto y permitirán obtener nuevos conocimientos, tanto técnicos, como del mercado, a la hora de competir con otras empresas del sector. Estos conocimientos podrán servir de gran utilidad en la carrera profesional del alumno dada la importancia que han conseguido los dispositivos con Android y la necesidad que tienen las empresas de ofrecer servicios en Internet.

Esto ha sido posible gracias a que los avances en la tecnología en estos últimos años ha aumentado muy por encima de lo esperado, haciendo que las empresas se actualicen para ofrecer sus servicios con la mejor calidad posible. Es por esto que, dispositivos como los teléfonos móviles no solo disponen de llamadas y mensajes, sino

que disponen de conexión a internet, GPS, cámara de fotos, Bluetooth y un largo etcétera, que los ha convertido prácticamente en ordenadores de bolsillo. A estos dispositivos que tienen estas funcionalidades se le conoce como Smartphones (teléfonos inteligentes) y suponen uno de los mayores mercados en el mundo ya que se estima que hay más de dos mil millones en todo el mundo.

Si a este hecho, le añadimos que los fabricantes de Smartphones y empresas de todos los sectores han decidido lanzar APIs (Applications Programming Interface o Interfaz de Programación de Aplicaciones) o lo que es igual, aplicaciones, para diferentes sistemas operativos (Android, iOS, Windows Phone, etc.), se ponen en manos de los usuarios todas las funcionalidades de estos terminales y servicios ofrecidos. De este modo, empresas y desarrolladores se aprovechan y consiguen un mayor número de ventas al contar con mejores aplicaciones y servicios que ofrecer.

Es de gran importancia destacar el aumento del uso de Tablets (tabletas) que comparten SO con los Smartphones, lo que permite a los desarrolladores de aplicaciones trabajar para un gran número de dispositivos.

Debido a la gran cantidad de dispositivos diferentes, es importante diferenciar las aplicaciones creadas para Smartphone y las creadas para ordenador. Aquellas desarrolladas para Smartphones poseen una menor potencia que las desarrolladas para ordenador debido a su tamaño, ya que supone poseer un procesador menos potente, una batería con menor capacidad y una pantalla más pequeña. Pero para las aplicaciones para Smartphones no es necesario contar con un navegador web para tener los mismos servicios que se ofrecen en internet.

El aumento del desarrollo y uso de aplicaciones web y móvil de tiendas de productos y servicios de todo tipo se debe a la comodidad que se le ofrece a los usuarios de no depender de una tienda física o de un horario de apertura en concreto. Empresas como Amazon, eBay, Wallapop, etc., que se encuentran dentro de este mercado, poseen tanto aplicación web como móvil y han servido de ejemplo para la realización de este trabajo.

Por ello mismo aparece la motivación principal, la cual es desarrollar una aplicación web y otra móvil totalmente funcionales pero con ámbito de desarrollo y no comercial, para que en un futuro puedan usarse como base para una empresa o ampliación de conocimientos.

A través de este trabajo habrá que hacer frente a problemas reales semejantes a los que se puede encontrar un ingeniero a lo largo de su carrera profesional. Además, permitirá aplicar conocimientos y habilidades obtenidas durante los años de formación universitaria.

1.2 Objetivos

Los objetivos que se quieren alcanzar con este TFG se han planificado teniendo en cuenta:

- **Limitaciones de tiempo.** El TFG es equivalente a una asignatura de 12 créditos ECTS, lo que supone que la aplicación y la documentación de la misma, debe realizarse en unas 300 horas de trabajo. Esto implica una duración de un cuatrimestre con una duración parcial de 20h/semana.
- **Capacidades del alumno.** Se trata de un entorno de desarrollo nuevo para el alumno, por lo que la realización del trabajo supone una complejidad adicional debida al aprendizaje de dicho entorno.
- **Requisitos de la aplicación.** Cuanto menos requisitos se necesiten, será más accesible, por lo que la penetración en el mercado será mayor.

El desarrollo desde el inicio de aplicaciones de este tipo aporta al alumno nuevos conocimientos y situaciones semejantes a las que se pueden encontrar en el mundo profesional. Por lo tanto, el objetivo principal de un TFG debe ser el de aprender y preparar al alumno para la vida laboral.

Centrándonos en las aplicaciones que se van a desarrollar, el objetivo es que cualquier usuario pueda comprar y vender de forma segura y accesible cualquier artículo ya sea nuevo o usado. Para ello será necesario el desarrollo de una aplicación web y otra móvil que ofrezcan los mismos servicios, autenticándose para ello, y guardar todos los datos recogidos en una base de datos compartida por ambas aplicaciones. Los usuarios registrados podrán comprar y vender todos aquellos artículos que estén almacenados en dicha base de datos a través de alguna de las pasarelas de pago que se ofrecen (Paypal, Redsys, etc.). Se quiere conseguir unas aplicaciones con un funcionamiento robusto y sin errores y que tengan un alto grado de simplicidad para facilitar su uso.

1.3 Fases de desarrollo

El siguiente trabajo está dividido en diferentes partes siguiendo un formato por el cual se introducirán las tecnologías abordadas en este TFG. Seguido de este punto, se introducirán las plataformas de desarrollo elegidas para las aplicaciones, donde se defenderá por qué han sido elegidas frente a otras.

Una vez que se haga la introducción, se procederá a describir el diseño y el desarrollo de la solución que se ha obtenido al realizar este TFG, razonando las elecciones y decisiones tomadas y exponiendo las dificultades encontradas a lo largo del mismo.

Para terminar se demostrará y describirá el funcionamiento de las aplicaciones en dispositivos reales y se finalizará con el aparatado de conclusiones y líneas futuras.

1.4 Medios empleados

Para llevar a cabo el desarrollo de este trabajo ha sido necesario el uso de los siguientes elementos:

- Por lo que a equipo se refiere, para la implementación ha sido necesario un portátil con todos los elementos software que se describirán a continuación y un dispositivo Android de cualquier tamaño de pantalla.
- SO: macOS Sierra.
- Herramientas de desarrollo: navegadores Safari y Google Chrome con WordPress, IDE (Integrated Development Enviroment o Entorno de desarrollo Integrado) Android Studio.
- Firebase de Google para notificaciones.
- MySQL Workbench para las bases de datos.
- Microsoft Office 2016 y Microsoft PowerPoint 2016.
- Un editor de texto.
- Diagrammix Lite para casos de uso.
- GanttProject 2.8.5 para el diagrama de Gantt.

2 ANÁLISIS DEL ESTADO DEL ARTE

Este punto está dividido en tres partes diferenciadas:

- **Páginas Web.** Se llevará a cabo un breve resumen sobre la evolución del desarrollo de páginas web, una comparativa entre los distintos CMS (Content Management System o Sistemas de Gestión de Contenidos) para desarrollarlas y la posterior elección de uno dada sus características.
- **Aplicaciones Android.** Se llevará a cabo un breve resumen sobre la evolución de la telefonía móvil, una comparativa de las distintas plataformas móviles existentes para desarrollar aplicaciones y la posterior elección de una dada sus características.
- **Repaso a la historia de tiendas online.**

2.1 Páginas Web

2.1.1 Evolución del desarrollo de páginas Web

El desarrollo de páginas web ha avanzado a pasos agigantados desde sus inicios, allá por la década de los '90. La WWW (World Wide Web) fue inventada por Tim Berners Lee, un informático de la Organización Europea de Investigación Nuclear (CERN), como herramienta de comunicación entre los científicos; y es aquí cuando se crea lo que conocemos como la web 1.0.

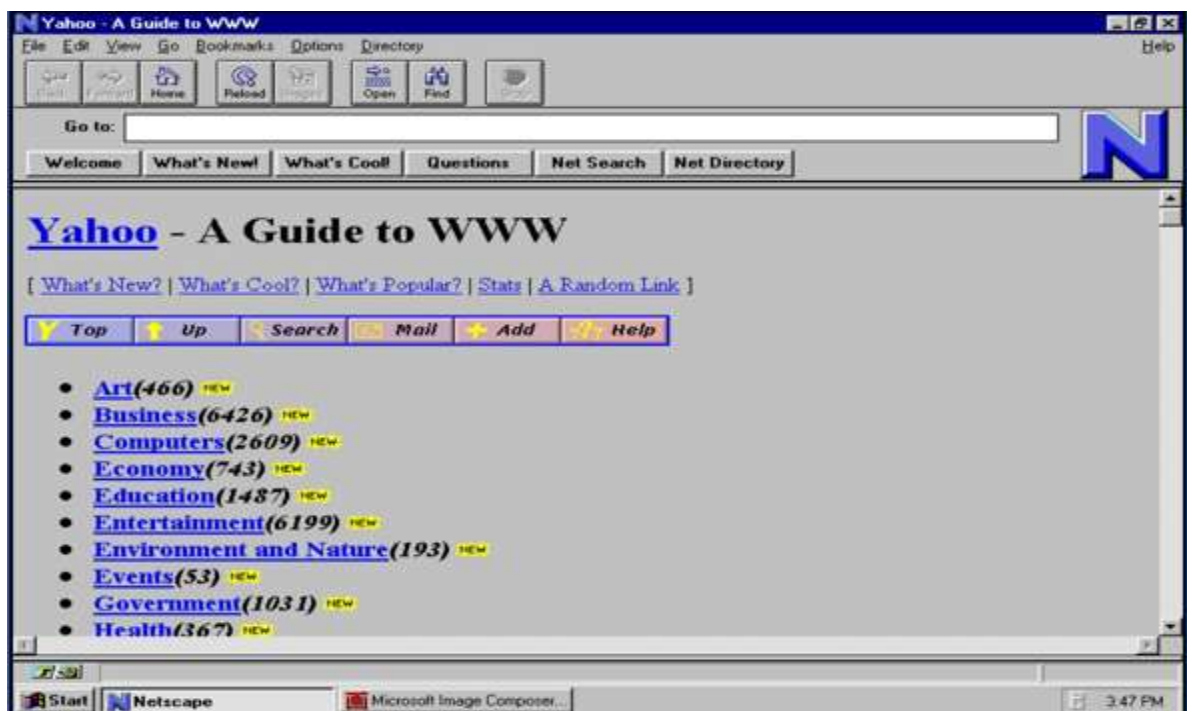


Ilustración 1: Ejemplo de Web 1.0

Esta versión nace como un sistema de hipertexto para compartir información en Internet, con la finalidad de publicar documentos. La web estaba administrada por el webmaster¹ que tenía todo el control ya que no había interacción con los usuarios, la web solo era de lectura. En 1990, WWW, empieza a tomar forma, es cada vez más visual, incorpora imágenes, diferentes formatos, colores, etc. Las grandes empresas empiezan a dilucidar la potencia de esta herramienta que te permite conectar con cualquier parte del mundo, encontrando en ella un sitio donde incorporar información corporativa, estar más cerca de sus clientes... en definitiva, empiezan las estrategias online.

Pero esta versión de web tiene carencias, los productores de contenidos no conocen la reacción de los lectores, es una página estática, no hay comunicación. Como medida surge la versión web 2.0.

En esta nueva versión los mandos son tomados por los usuarios. Surgen aplicaciones y páginas que utilizan la inteligencia colectiva. Consecuencia de ello, las páginas pueden ser personalizadas, los usuarios pueden cambiar e introducir datos y se convierte en una herramienta dinámica que permite el intercambio de información.

La información se transforma en comunicación gracias a la interacción y a la incorporación de comentarios de texto, videos, chats, etc. Nacen con ello los blogs, las redes sociales, las wikis, etc. Ejemplos: eBay, Facebook, Wikipedia, etc. Este cambio supuso una gran revolución, puesto que permitía devolver lo más valioso, la información.

Mientras que la web 2.0 significó la aparición de redes sociales, la versión 3.0 dará paso a las redes semánticas que pondrá su objetivo en la inteligencia artificial, un método para clasificar las páginas que no solo permitan a los usuarios encontrar la información sino comprenderla. El control pasa a manos de motores informáticos y procesadores de información. Se trata de procesadores que basándose en nuestros perfiles y actividad en redes sociales y búsquedas por internet descubren información y nos siguen el rastro a modo de fotos, videos, geolocalización... Sin duda, la web 3.0 es definida por el concepto “personalización”, ya que pretende devolver al usuario una información lo más afinada posible, filtrada en sus gustos y preferencias, evitando información que no sea de su interés.

Bibliografía [\[1\]](#).

¹Webmaster: también conocido como arquitecto web, desarrollador web, autor de sitio digital, administrador de sitio digital, y coordinador de sitio digital, es la persona responsable de mantenimiento o programación de un sitio web. Véase en: <https://es.wikipedia.org/wiki/Webmaster>



Ilustración 2: Web 3.0

2.1.2 CMS para desarrollar páginas Web

Atraer el mayor número posible de lectores a una página web y mantener su interés depende, principalmente, de un contenido atractivo. Los textos, las imágenes, los videos y los gráficos ofrecen un valor añadido al usuario. Pero este contenido, sin embargo, se ha de gestionar, publicar y comunicar, una tarea que, en función del tamaño de la página, puede ser de grandes dimensiones, e imposible de llevar a cabo si no es con un CMS adecuado.

Bibliografía [2].

A principios de los años 90, el concepto de CMS era desconocido. Algunas de sus funciones se realizaban con aplicaciones independientes: editores de texto y de imágenes, bases de datos y programación a medida.

Algunos ejemplos de empresas que empezaron a desarrollar estos CMS son: RedDot es una de las empresas pioneras que empezó el desarrollo de un CMS en el año 1994. No fue hasta finales del año siguiente cuando presentaron su CMS basado en una base de datos. Entre los CMS de código abierto, uno de los primeros fue Typo 3, que empezó su desarrollo en el año 1997. Dos años después, en 1999, IntraNet Solutions lanza Xpedio, el primer CMS de extremo a extremo.

Obviamente, la evolución de internet hacia portales con más contenido y la altísima participación de los usuarios a través de blogs y redes sociales, aceleró el nacimiento de productos más especializados y amigables, muy lejanos a los primeros, más hechos para el tráfico de Intranet o demasiado laboriosos para usar. En ese marco

podemos ya hablar de CMS más completos y globales, destacando nítidamente la primera versión oficial de WordPress lanzada en enero de 2004 y la primera versión en septiembre de 2005 de Joomla!.

Bibliografía [\[3\]](#).

En la búsqueda del CMS más acorde con los intereses de cada uno es común perder la perspectiva fácilmente. Hoy en día se encuentran en el mercado entre 250 y 300 CMS diferentes, que van desde simples sistemas modulares para blogs a gestores más completos para el ámbito profesional. Aquí, la cuestión central gira en torno a los requisitos que ha de cumplir un CMS para que pueda soportar eficazmente el proyecto previsto.

Cuando se trata de decantarse por una solución y no otra, el criterio de mayor peso debería ser cual es el ámbito de aplicación para el cual se ha desarrollado el software.

Muchos CMS están disponibles públicamente como soluciones open source gracias a las grandes comunidades que los sustentan. Se trata de soluciones de código abierto y reciben este nombre porque puede acceder al código del programa libremente, lo que permite que los usuarios puedan llevar a cabo modificaciones, mejoras y ampliaciones en el código y adaptar, así, el CMS a las necesidades personales.

Frente a estas, se encuentran las soluciones propietarias, desarrolladas por una determinada empresa y, por regla general, distribuidas comercialmente. Para utilizar un CMS propietario suele ser necesaria la adquisición de una licencia y el código fuente no está disponible libremente.

En cambio, el software de código abierto no requiere pago alguno por la licencia, lo que lo hace especialmente atractivo para las empresas.

Para entender mejor la diversidad existente actualmente se va a proceder a caracterizar los principales CMS, dando a conocer sus requisitos técnicos, ventajas e inconvenientes.

CMS	Número de páginas web	Disponibilidad
WordPress	18 millones aprox.	Código abierto
Adobe Dreamweaver	2,7 millones aprox.	De pago
Joomla!	2,5 millones aprox.	Código abierto
Drupal	772.000 aprox.	Código abierto

Tabla 1: CMS más populares

2.1.2.1 WordPress

WordPress es el CMS de código abierto más utilizado del mercado. Concebido originariamente como sistema de blogging, hoy la instalación básica cuenta con diversas extensiones que permiten convertir el software en un CMS en toda regla.

- **Requisitos técnicos:**

- Servidor Web: cualquier servidor que soporte PHP y MySQL/MariaDB (recomendado apache).
- Middleware²: PHP 7 o superior.
- Base de datos: MySQL 5.6 o superior o MariaDB 10.0 o superior.
- Hardware: los desarrolladores recomiendan un límite de memoria PHP de 32 MB.
- Otras recomendaciones: soporte HTTPS y el módulo mod_rewrite de Apache, con el que se pueden generar URL amigables³.

- **Ventajas e inconvenientes:**

Ventajas	Inconvenientes
√ Gran comunidad de soporte	X Las funciones de CMS requieren extensiones adicionales
√ Instalación y configuración muy simples	X Los plugins revelan a menudo fallos de seguridad
√ Interfaz muy intuitiva	X Estabilidad y rendimiento limitados con mucho tráfico
√ Las extensiones y los plugins se integran fácilmente	X Las frecuentes actualizaciones de seguridad complican la administración

Tabla 2: Ventajas e inconvenientes de WordPress

2.1.2.2 Adobe Dreamweaver

El editor HTML Dreamweaver es una alternativa sólida a los gestores de contenido libres. El software de diseño web, originalmente desarrollado por Macromedia, forma parte hoy de la Creative Cloud de Adobe Systems como producto propiedad de macOS y Windows.

Dreamweaver, herramienta de edición gráfica de páginas web y contenido móvil, combina para ello un editor WYSIWYG con un motor de codificación de gran rendimiento, el cual, contiene diversas funciones para los desarrolladores, entre ellas, el resaltado de

² Malware: software que se sitúa entre un SO y las aplicaciones que se ejecutan en él. Véase en: <https://es.wikipedia.org/wiki/Malware>

³ URL amigables: describen el contenido de la página web a la que llevan y son fáciles de entender, recordar y comunicar. Véase en: https://es.wikipedia.org/wiki/URL_semántica

sintaxis (syntax highlighting), el completamiento automático de código, la validación y la vista previa en vivo. Además de HTML5 y CSS, el software soporta estándares web como PHP, Java, JavaScript, ASP o XML, así como los preprocesadores CSS SASS, Less y SCSS. Naturalmente, el diseño responsivo también se cuenta entre sus facultades gracias al framework⁴ de CSS Bootstrap integrado.

Debido a la gran cantidad de funciones que contiene y alto precio que conlleva la adquisición de la licencia de uso, este completo software está pensado, más bien, para el sector profesional (diseñadores y agencias).

2.1.2.3 Joomla!

Joomla! constituye el tercer CMS más utilizado. Orientado por igual a expertos y amateurs, es mucho más exigente que WordPress en cuanto a las aplicaciones, pues, a diferencia de este, ya contiene de base muchas funciones de sistema de gestión de contenido. Un aspecto positivo para los menos versados en temas informáticos es, eso sí, la buena calidad de la documentación en la forma de manuales online elaborados por la comunidad, así como el activo foro de usuarios.

- **Requisitos técnicos:**

- Servidor web: Apache HTTP Server 2.0 (recomendado 2.4) con mod_mysql, mod_xml y mod_zlib; Nginx 1.8 o superior; MS IIS 7.
- Middleware: PHP 5.6 (recomendado 7.0 o superior).
- Bases de datos: MySQL 5.1 con soporte de InnoDB (recomendado 5.5.3 o superior); MS SQL Server 10.50.1600.1 (recomendado 10.50.1600.1 o superior), PostgreSQL 8.3.18 (9.1º superior), MariaDB 5.1 (o superior).

- **Ventajas e inconvenientes:**

Ventajas	Inconvenientes
√ Instalación y configuración muy simples	X Gestión de derechos con carencias
√ Gran archivo de diseños y extensiones	X Las extensiones más maduras son, en parte, de pago
√ Creación cómoda e intuitiva de plantillas	X Parte de las extensiones han de actualizarse manualmente

Tabla 3: Ventajas e inconvenientes de Joomla!

⁴ Framework: es una estructura conceptual y tecnológica de asistencia definida, normalmente, con artefactos o módulos concretos de software, que puede servir de base para la organización y desarrollo de software. Véase en <https://es.wikipedia.org/wiki/Framework>

2.1.2.4 Drupal

Drupal, de código abierto y modular, fue desarrollado originariamente como tablón de anuncios estudiantil, estando hoy disponible como un CMS open source totalmente válido. La instalación básica del software es extremadamente sencilla, aunque se puede ampliar con un gran número de módulos, y este se pone a disposición por parte de una activa comunidad que, aún hoy, sigue siendo su punto fuerte. El módulo básico ya está compuesto por un gran número de funciones para la web social como weblogs, foros y nubes de etiquetas⁵ (tag clouds).

- **Requisitos técnicos:**

- Servidor web: Apache, Nginx, MS IIS o cualquier servidor web que soporte PHP.
- Middleware: PHP 5.5.9 o superior.
- Bases de datos: MySQL 5.5.3, MariaDB 5.5.20, Percona Server 5.5.8 (o superior) con InnoDB como motor de almacenamiento primario; PostgreSQL 9.1.2 (o superior con la extensión de acceso a bases de datos PDO); SQLite 3.7.11 o superior.

- **Ventajas e inconvenientes:**

Ventajas	Inconvenientes
✓ Núcleo de software compacto	X Compleja configuración debido a la falta de compatibilidad con versiones anteriores
✓ Gran variedad de módulos y distribuciones predefinidas	X Aunque la versión básica es muy sencilla, requiere muchas instalaciones posteriores
✓ Soporta la gestión multidominio	X La instalación de módulos solo es posible mediante File Transfer Protocol (FTP)

Tabla 4: Ventajas e inconvenientes de Drupal

Bibliografía [2].

⁵ Nubes de etiquetas: representación visual de las palabras que conforman un texto, en donde el tamaño es mayor para palabras que aparecen con más frecuencia. Véase en: https://es.wikipedia.org/wiki/Nube_de_palabras

2.1.3 Elección de una plataforma para el trabajo

Para la elección final de uno de los CMS se ha tenido muy en cuenta el uso y el crecimiento en los últimos años. En el siguiente gráfico se puede observar el uso que se da a estos CMS.

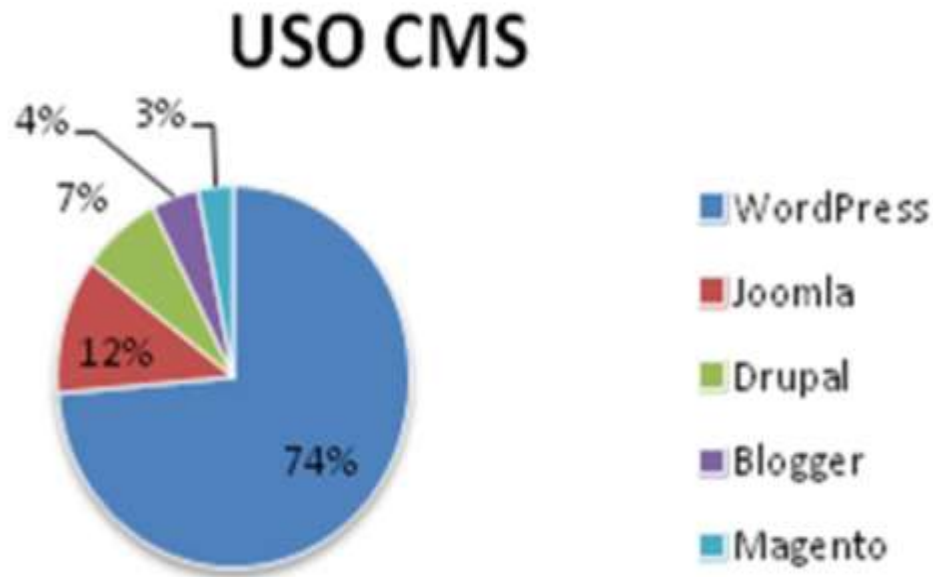


Ilustración 3: Uso CMS

Teniendo en cuenta la caracterización realizada en apartados anteriores, es fácil desde un inicio descartar uno de los cuatro CMS que no nos servirán para la realización con éxito del trabajo. En concreto se descarta Adobe Dreamweaver, ya que necesita de licencias de pago, está pensado para el sector profesional y lo que se busca en este trabajo es que se tenga el menor coste posible.

El siguiente CMS en descartarse sería Drupal, que aunque sea de código abierto, está muy por debajo de los otros dos restantes.

La decisión de usar WordPress o Joomla! queda en manos del usuario desarrollador. Se deberá elegir la que más se aproxime a sus preferencias y/o requisitos. En este caso se va a usar WordPress, ya que permite la instalación de extensiones que nos ayudarán a conseguir los objetivos del trabajo, una de ellas es WooCommerce.

En apartados siguientes, se procederá a realizar un amplio resumen de este CMS, así como las herramientas de trabajo utilizadas a lo largo del trabajo.

2.2 Aplicaciones Android

2.2.1 Evolución de la telefonía móvil

Los más atrevidos aseguran que los inicios de la telefonía móvil comenzaron con la Segunda Guerra Mundial. Motivado por la necesidad de comunicarse a largas distancias Motorola creó un equipo militar llamado Handie Talkie H12-16. Pero este sistema dista mucho de los teléfonos móviles de uso personal que comenzaron a comercializarse en la primera mitad de los años 80, de nuevo con Motorola como marca pionera.

Si echamos la vista atrás, los primeros móviles eran más gruesos y pesados que los actuales y contaban con una antena, aun así, es curioso ver cosas que se mantienen con el tiempo aunque con diferentes matices. En los primeros móviles, cambiabas la carcasa para que fuera único, sin embargo ahora, lo personalizas mediante una funda.

En los años 90, una vez los teléfonos móviles estaban más asentados entre nosotros, se dieron bastantes hitos importantes en su historia. Por ejemplo, en 1993 se comercializa el primer móvil con Personal Digital Assistant (PDA), aunque hay que señalar que pesaba medio kilo. Lo que sí se puede afirmar, es la fecha en la que vio la luz el primer móvil con envío de mensajes de texto y fue en el año 1995, el modelo que ofrecía esta función era el Nokia 2110. Pero aún quedan más sucesos importantes en los 90, en 1996 apareció el primer móvil con tapa, en este caso un Motorola StarTAC.

Ya en los 2000, apareció el método de envío de archivos mediante Bluetooth⁶. También apareció el primer móvil con pantalla color y cámara de fotos en el año 2002, fue una creación de la marca japonesa Sharp y el modelo el J-SH04.

En 2007 salió al mercado la primera generación de iPhone que revolucionó y puso patas arriba la industria telefónica. Tras él, llegaron más Smartphones, capaces de conectarse a Internet, recibir emails, descargar diferentes aplicaciones y no sólo hablar a través de llamadas y SMS sino también a través de servicios de mensajería instantánea como Whatsapp o Telegram.

Lo cierto es que los Smartphones no sólo han supuesto un gran cambio en la tecnología, sino también en nuestro día a día.

⁶ Bluetooth: especificación industrial para Redes Inalámbricas de Área Local (WPAN) que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia en la banda ISM de los 2,4 GHz. Véase en: <https://es.wikipedia.org/wiki/Bluetooth>



Ilustración 4: Evolución de la telefonía móvil

La tecnología mediante la que se conecta tu móvil a Internet está también en continua fase de cambios. Hace unos años disfrutabas una conectividad algo más limitada, sin embargo, hoy en día con el 4G llevas en tu Smartphone la mayor velocidad.

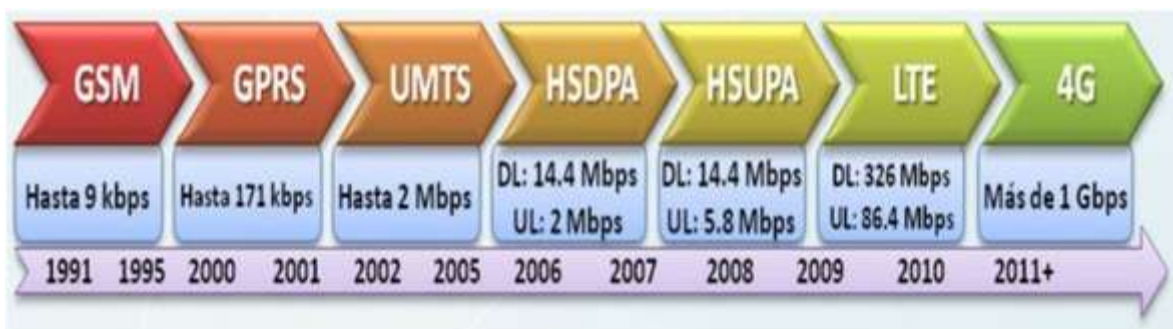


Ilustración 5: Evolución conectividad móvil

Aún queda mucho camino por recorrer en cuanto a telefonía, puedes escuchar diferentes teorías sobre adonde se dirige el futuro de los móviles, pantallas que proyectan hologramas, materiales transparentes, un sinfín de prototipos... sea como sea, lo que es seguro es que dentro de unos años al echar la vista atrás, tu actual Smartphone te parecerá una pieza de coleccionista.

Bibliografía [\[4\]](#).

2.2.2 Principales sistemas operativos para móviles

Los SO móviles son los encargados de administrar el hardware de los diferentes dispositivos móviles como Tablets, Smartphones, relojes... entre otros, teniendo en

cuenta las características que diferencian estos dispositivos móviles de los demás sistemas computacionales. Los SO móviles están enfocados en la movilidad, la conectividad inalámbrica y en la administración de forma optima del procesamiento, almacenamiento y consumo de la energía de la visualización, entre otros, que generan un consumo de energía bastante alto produciendo como resultado un uso inadecuado de la energía en el dispositivo.

Los SO móviles, en general, cuentan con unas capas específicas, sin embargo, en algunos SO esto es diferente de acuerdo a su funcionamiento.

- La capa Kernel o núcleo del SO es el encargado de administrar todos los elementos de hardware del dispositivo móvil.
- La capa Middleware o intermediador de aplicaciones del SO, son diferentes programas o módulos que permiten el uso de aplicaciones y librerías entre otras, para el funcionamiento del dispositivo móvil.
- La capa de administración de aplicaciones es la encargada de la ejecución, detención y finalización de las aplicaciones del SO.
- La capa de interfaz se encarga de administrar el uso que le da el usuario al dispositivo móvil ya sea de pantalla táctil o touch y los dispositivos con teclados QWERTY.

Según el servicio de estadísticas de NetMarketShare, la cuota de mercado de sistemas operativos móviles a nivel mundial a mediados de 2017 es la siguiente:

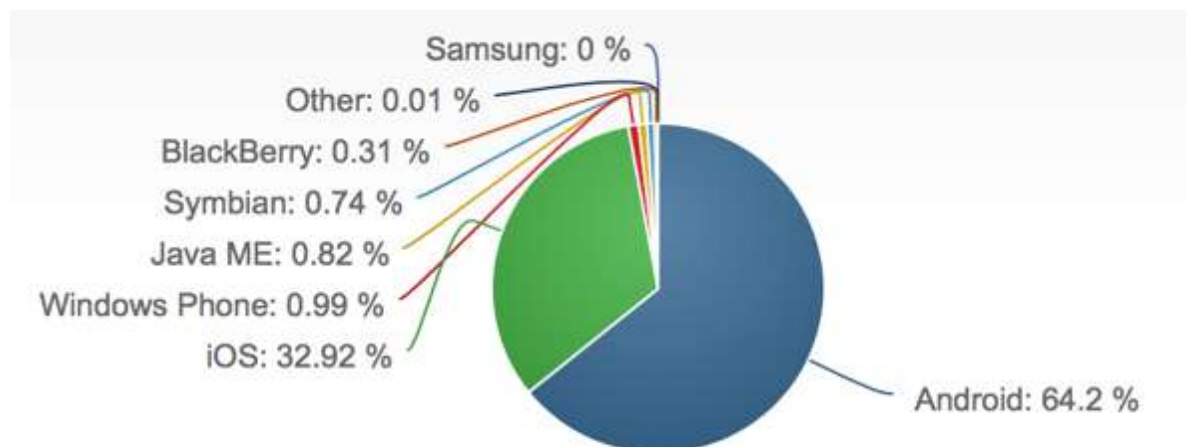


Ilustración 6: Cuota de mercado SO móviles mediados 2017

- Android: 64,2% (en países como España las diferencias son más significativas, donde Android tiene más del 90% de la cuota de mercado).
- iOS: 32,92%.
- Windows Phone: 0,99%.
- BlackBerry: 0,31%.

Cada SO cuenta con sus propias aplicaciones, aunque existen numerosas que no son compatibles con los demás. Esto genera competencia entre las distintas empresas que desarrollan y actualizan continuamente sus SO para conseguir que todos los usuarios encuentren lo que buscan.

Para entender mejor las principales diferencias existentes entre los diferentes SO móviles, se va a proceder a describir los principales, dando a conocer un poco de su historia, sus puntos fuertes y debilidades, tipos de aplicaciones y distintos dispositivos compatibles.

Bibliografía: [\[5\]](#) [\[6\]](#).

2.2.2.1 Android

Android es sin duda el líder del mercado móvil en SO. Está basado en Linux, disponiendo de un Kernel responsable de convertir el código escrito en Java de las aplicaciones, a código capaz de comprender dicho Kernel; utilizando una máquina virtual para ello. Su nombre se debe a su inventor, Andy Rubin que estaba diseñando este SO



Ilustración 7: Android

para ser usado en cámaras fotográficas profesionales. En 2003, Android Inc. creó el SO y lo vendió a Google en 2005 que lo modificó para utilizarlo en dispositivos móviles como los Smartphones y posteriormente en Tablets. Fue anunciado en el año 2007 y liberado en 2008; además de la creación de la Open Handset Alliance, compuesta por 78 compañías de hardware, software y telecomunicaciones dedicadas al desarrollo de estándares abiertos para teléfonos móviles, que ayudó mucho a Google a masificar el SO hasta el punto de llegar a ser usado por empresas como LG, Samsung, HTC entre otras.

- **Puntos fuertes:** está desarrollado de forma abierta, lo que lo convierte en una ventaja tanto para los desarrolladores de aplicaciones como para sus usuarios ya que cuenta con un gran número de dispositivos en el mercado. Se puede personalizar el dispositivo móvil al máximo y modificar funciones simplemente instalando una aplicación. Cuenta con navegador web compatible con Flash Player.
- **Debilidades:** la fragmentación es uno de los aspectos más negativos de Android. Al haber tanta diversidad de terminales con diferentes características, como el tamaño de pantalla, la velocidad del procesador, etc., implica que unos dispositivos sean compatibles con diferentes versiones del SO. Cuanto mejores sean las características del dispositivo, mayor rendimiento y funcionalidad tendrá y se podrán ejecutar las últimas versiones de las aplicaciones disponibles.
- **Aplicaciones:** las aplicaciones se pueden encontrar en la tienda Google Play. Cualquier desarrollador puede publicar su aplicación en esta tienda teniendo una

cuenta de desarrollador de Google Play. Al ser gratuita, cuenta con una gran cantidad de aplicaciones a un precio muy bajo o incluso gratuitas.

- **Dispositivos:** cuenta con una gran diversidad de dispositivos, ya sean Smartphones, Tablets, relojes, etc.

Algunas marcas son: Samsung, Sony, Huawei, etc.

2.2.2.2 iOS

iOS es el sistema operativo que da vida a dispositivos como el iPhone, iPad, iPod Touch o el Apple TV.



Ilustración 8: iOS

Anteriormente denominado iPhone OS, este SO fue creado por Apple originariamente para el iPhone, siendo después usado en el iPod Touch e iPad. Se trata de una versión reducida del sistema Mac OS X existente para ordenadores de la misma compañía. Fue presentado en 2007 implantado en el iPhone. Con este sistema operativo, Apple fue pionera con el concepto de teléfono inteligente tal y como lo conocemos hoy.

Según el servicio de estadísticas de [NetMarketShare](#) cuenta con un 23% de cuota de mercado a mediados de 2017.

- **Puntos fuertes:** ya que este SO solo está disponible para dispositivos de la empresa Apple, podemos destacar su buen diseño, funcionalidad, facilidad de uso y una variedad de aplicaciones y juegos, que lo convierte en un referente. Posee una de las mayores y mejores tiendas de aplicaciones y juegos del mundo, la App Store. Al ser desarrollado específicamente para unos dispositivos se consigue explotar al máximo sus capacidades. Se pueden compartir todo tipo de datos fácilmente entre dispositivos de la marca Apple.
- **Debilidades:** el sistema de Apple es más cerrado que el de sus competidores, siendo más restrictivo en las publicaciones y manteniendo un control mucho más exhaustivo de lo que se publica, lo que puede ser tanto una ventaja para algunos, como una desventajas para otros, no permitiendo ninguna modificación del sistema tal y como viene creado. Además, no se permite la instalación de aplicaciones que no hayan sido validadas y solo se pueden programar con Swift u Objective-C. Otro punto bastante negativo es el alto coste de los dispositivos de la marca ya que solo hay un fabricante.
- **Aplicaciones:** Apple fue la primera empresa en apostar por una tienda de aplicaciones, lo que la ha llevado, sin lugar a dudas, a que ésta sea la clave de su

éxito. Apple otorga el 85% de los ingresos al desarrollador de la aplicación y se queda con un 15%, intentando incentivar así a los desarrolladores.

- **Dispositivos:** este SO se encuentra disponible en los dispositivos de la marca Apple: iPhone, iPad e iPod Touch.

2.2.2.3 Windows Phone

Anteriormente llamado Windows Mobile, fue presentado en 2010 como Windows Phone. Es un SO móvil compacto desarrollado por Microsoft. Se basa en el núcleo del SO Windows CE y cuenta con un conjunto de aplicaciones básicas. Está diseñado para ser similar a las versiones de escritorio de Windows estéticamente y existe una gran oferta de software de terceros disponible para Windows Mobile, la cual se puede adquirir a través de la tienda en línea Microsoft Store.



Ilustración 9: Windows

Su cuota de mercado ha disminuido ya que contaba con un 2,6% en 2011 y a mediados de 2017 cuenta con un 0,99%.

- **Puntos fuertes:** el sistema gráfico es sencillo y agradable. Tiene un gran parecido gráfico al SO para ordenadores y permite una fácil integración con otros dispositivos con Windows. Con Windows Live ID se puede configurar y sincronizar fácilmente cuentas de correo Hotmail, Live o cualquier subdirección montada sobre los servidores de Windows Live.
- **Debilidades:** no se permite la configuración en el menú inicial con widgets⁷ o imágenes de fondo en la pantalla principal. Al igual que iOS, no es compatible con Flash y usa HTML5. Al contrario que Android, no permite la utilización de multitareas con aplicaciones de terceros, por lo tanto el usuario queda limitado. Usa Internet Explorer, un navegador que ha ido perdiendo seguidores por las prestaciones que ofrecen sus competidores.
- **Aplicaciones:** cuenta con una gran tienda, Microsoft Store, donde se pueden encontrar más de 600.000 aplicaciones aunque muchas menos que sus dos principales competidores. El principal software para desarrollar aplicaciones para Windows Phone es Windows App Studio.
- **Dispositivos:** para soportar este SO hay gran cantidad de dispositivos, tanto Smartphones y Tablets, y gran cantidad de empresas que lo tienen implantado, como Nokia o HTC entre otras.

⁷ Widget: es una pequeña aplicación que da fácil acceso a funciones frecuentemente usadas. Véase en: <https://es.wikipedia.org/wiki/Widget>

2.2.2.4 BlackBerry

BlackBerry es un SO móvil desarrollado por la empresa canadiense Research In Motion (RIM) que tuvo sus inicios en 1999. Aunque este sistema está orientado especialmente para un uso profesional (dar servicio a empresas y profesionales) ha tenido éxito entre los jóvenes debido a su servicio de mensajería instantánea.



Ilustración 10: BlackBerry

Su cuota de mercado está actualmente en un 0,31%, muy por debajo con respecto a años anteriores.

- **Puntos fuertes:** tiene una gran productividad en la empresa administrando el correo electrónico, calendario, contactos, documentos, etc. Ofrece a sus usuarios, que cuentan con dispositivos con este SO, un acceso seguro a la intranet corporativa y aplicaciones personalizadas con HTML5. Se integra con las soluciones líderes de la empresa de mensajería instantánea para mantener a los empleados en contacto unos con otros. Permite a los empleados de una empresa acceder de forma segura, editar y sincronizar los archivos de ambos repositorios de archivos locales y corporativos como SharePoint y caja.
- **Debilidades:** está muy orientado para la empresa, siendo un problema a la hora de vender sus productos a usuarios particulares. Ha quedado desbancado por otros competidores y poco a poco se está perdiendo su uso.
- **Aplicaciones:** su tienda de aplicaciones se denomina BlackBerry World, es muy inferior respecto a las de sus competidores, cuenta con muy pocas aplicaciones y ninguna destaca en el mercado. No ha tenido mucho apoyo de los desarrolladores de aplicaciones y es una de las tiendas menos valoradas.
- **Dispositivos:** se trata de una plataforma exclusiva de la marca BlackBerry. Cuenta con dispositivos con teclado físico y pantalla táctil.

Bibliografía [\[5\]](#) [\[6\]](#) [\[7\]](#) [\[8\]](#) [\[9\]](#) [\[10\]](#)

2.2.3 Elección de una plataforma para el trabajo

Para la elección final se ha tenido muy en cuenta el número de aplicaciones disponibles en tiendas móviles estos últimos meses, la [cuota de mercado](#) y el crecimiento de estos SO móviles.

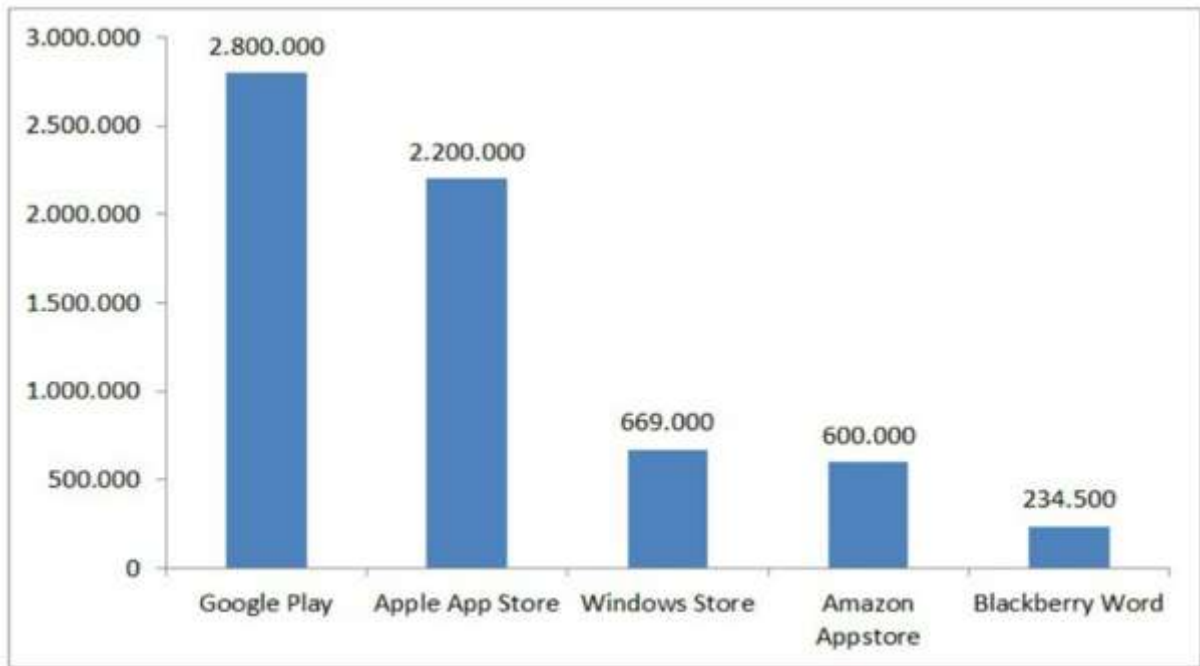


Ilustración 11: N° de aplicaciones disponibles en tiendas móviles. Marzo 2017

Teniendo en cuenta la caracterización realizada en apartados anteriores, es fácil desde un inicio descartar uno de los cuatro SO móviles que no nos servirán para la realización con éxito del trabajo. En concreto se descarta BlackBerry, ya que no cuenta con una tienda de aplicaciones a la altura de sus competidores y está destinada a usuarios de empresas.

El siguiente SO móvil en descartarse sería Windows Phone, ya que comparándolo con sus otros competidores, Android e iOS, está muy por debajo.

La decisión de usar Android o iOS queda en manos del usuario desarrollador. Se deberá elegir la que más se aproxime a sus preferencias y/o requisitos.

En este caso se va a usar Android, ya que se trata de un SO móvil de código abierto y no se necesita adquirir una licencia. Además, cuenta con la tienda con más aplicaciones en el mercado.

En apartados siguientes, se procederá a realizar un amplio resumen de este SO móvil, así como las herramientas de trabajo utilizadas a lo largo del trabajo.

2.3 Repaso a la historia de tiendas online

Una tienda en línea (también conocida como tienda online, tienda virtual o tienda electrónica) se refiere a un tipo de comercio que usa como medio principal para realizar sus transacciones un sitio web o una aplicación conectada a internet.

Los vendedores de productos y servicios ponen a disposición de sus clientes una página web (o aplicación informática) en la que se pueden observar imágenes de los productos, leer especificaciones y finalmente, adquirirlos. Este servicio le da al cliente

rapidez en la compra, la posibilidad de hacerlo desde cualquier lugar del mundo y a cualquier hora del día. Algunas tiendas en línea incluyen dentro de la propia página del producto los manuales de usuario de manera que el cliente puede tener una idea de antemano de lo que está adquiriendo. Igualmente, muchas tiendas en línea ofrecen a los compradores la posibilidad de calificar y evaluar el producto. Estas valoraciones se pueden llevar a cabo a través de la propia página o a través de aplicaciones de terceros, estando siempre visibles para cualquier visitante de la web.

Normalmente estos productos se pagan mediante tarjeta de crédito o servicios de pago como Paypal o Redsys, para posteriormente enviarlos al cliente por correo o agencia de transporte, aunque según el país y la tienda puede haber otras opciones, como recoger en la tienda. También suelen ofrecerse métodos de pago alternativos como la transferencia bancaria o el pago contra reembolso (pago en destino al recibir el pedido).

La inmensa mayoría de tiendas en línea requieren la creación de una cuenta de usuario en el sitio web a partir de datos como nombre, dirección y correo electrónico. Este último a veces es utilizado como medio de validación.

2.3.1 Información general

Una tienda virtual es un sitio web que vende productos o servicios y, por lo general, tiene un carrito de compra asociado a ella. Con la popularidad de Internet, se ha producido un rápido aumento de tiendas en línea. Las compras en línea se han convertido en ventaja para los propietarios de tiendas al por menor. En ellas las personas pueden comprar desde sus casas logrando tener más opciones ya que tienen una gran variedad de alternativas para elegir y no necesita caminar grandes distancias para llegar a otras tiendas.

Al estar en internet se puede vender a personas de todo el mundo, aunque se pueden hacer envíos internacionales (por ejemplo entre países de la Unión Europea), se debe tener en cuenta que estos envíos internacionales dificultan las devoluciones y los reclamos por garantía incrementando costos, salvo si se trata de productos digitales. El comercio electrónico también ha sido de gran beneficio para muchas pequeñas y medianas empresas, ya que pueden tener una tienda abierta para todo el mundo a un coste inferior comparado con la inversión que debería realizar para llegar a más lugares con diferentes sucursales. Inclusive pequeños comerciantes se pueden ver beneficiados de tiendas virtuales a medida, prediseñadas, donde solo deben escoger su plantilla y publicar sus productos.

2.3.2 Privacidad y seguridad

Debido a las amenazas de la privacidad de datos en internet y la amenaza de robo de identidad es muy importante hacer compras en línea solamente en sitios reconocidos y de buena reputación. Igualmente es recomendable no proporcionar datos personales ni de tarjeta de crédito si no se está utilizando una conexión segura.

Para asegurarse que la tienda visitada es legítima, se puede comprobar, entre otros, los siguientes elementos:

- Presencia de condiciones de uso y aviso legal.
- Datos de contacto completos, como el nombre y la dirección de la empresa.

Elementos que otorgan seriedad y confianza:

- Sello de confianza reconocido o recomendación por la FECEMD (Federación Española de Marketing Directo y Comercio Electrónico): no es necesario para que la tienda sea legal y su uso va disminuyendo con el paso del tiempo. Inicialmente estaban pensados para aportar seriedad y confianza, pero actualmente la mayoría de sellos de confianza en línea requieren un elevado pago anual por tenerlo y cada día está más discutido este método como imagen de seriedad para un comercio electrónico.
- Existencia de una tienda física: muchos negocios físicos se complementan con una tienda en línea y esto hace que los compradores tengan más confianza a la hora de comprar en este tipo de comercios.
- Aviso de consumidores en sitios externos (comparadores, foros, directorios de tiendas, etc.).
- Ganador de premios de comercio electrónico: no es necesario para que una tienda en línea sea legal. Sería imposible crear una tienda y para ser legal recibir un premio antes de abrirla. Recibir un premio de comercio electrónico puede requerir años, incluso no recibirlo nunca, muchas tiendas importantes no tienen ninguno.

2.3.3 Historia

En materia de Internet y comercio electrónico, una tienda virtual (o comercio virtual) es una página web cuyo objetivo es la venta a terceros de productos o servicios.

A principios de la década de los años 70 surgieron las primeras relaciones comerciales que usaban un ordenador para transmitir datos. Este tipo de intercambio de información incluía entre otros la transferencia de documentos, como facturas y órdenes de compra. Como resultado, se experimentaron grandes mejoras en este tipo de empresas.

A mediados de los años 80 surgió la venta por catálogo o venta directa. De esta manera, los productos eran mostrados con mayor realismo con la posibilidad de exhibirlos al público, resaltando sus características. La venta se solía realizar mediante un teléfono, mientras el pago era realizado mediante una tarjeta de crédito.

En 1989 aparece un nuevo servicio, la WWW. Cuatro años después las empresas y el público en general se dan cuenta de su potencial. No fue hasta mediados de la década de los 90 que se inauguraron las primeras tiendas en línea, entre ellas Amazon y eBay. A finales de la década, la oferta de comercio electrónico creció de manera muy considerable, incluso por encima de lo que el mercado daba de sí y, tras un repentino retroceso durante lo que se conoció como el estallido de la burbuja punto com (años 2000-2003), continuó ininterrumpidamente su crecimiento durante la década siguiente y hasta la actualidad.



Ilustración 12: Amazon en 1995

En 2014, y de acuerdo con datos de la CMNC (Comisión Nacional de los Mercados y la Competencia) la facturación de las tiendas en línea en España crece a niveles que rondan el 25% interanual.

En el año 2016, el comercio electrónico supera los 24 mil millones de euros en España, un 20,8% más que en el 2015.

Bibliografía [\[11\]](#)

3 PLATAFORMAS DE TRABAJO: WORDPRESS Y ANDROID

A continuación se van a definir las plataformas que se han escogido para el desarrollo de las aplicaciones web y Android, definiendo el sistema, los requisitos funcionales, rendimiento, usabilidad, seguridad, fiabilidad, compatibilidad, apariencia y requerimiento de software y hardware.

3.1 WordPress

WordPress es una potente plataforma para sitios web de publicación con una gran cantidad de características diseñadas para que publicar contenidos sea fácil y sencillo.

Algunas de las características de WordPress son:

- Puede actuar como gestor de contenidos, como blog o como ambos simultáneamente, lo que le permite disponer de un sitio web empresarial con su blog corporativo, todo ello gestionado de forma sencilla por una única herramienta.
- Páginas y artículos se generan dinámicamente a medida que se publica, por lo que la actualización es fácil y rápida.
- Al tener una base internacional, permite tener blogs y sitios web en casi cualquier idioma.
- Es fácilmente integrable con las redes sociales favoritas del desarrollador y las de sus visitantes.
- Puede o no aceptar comentarios en páginas y artículos y, si se aceptan, se pueden moderar, por lo que las capacidades de comunicación y obtención de información de los visitantes aumentan.
- Los artículos se organizan en categorías, lo que facilita el trabajo tanto al desarrollador como a los visitantes.
- La herramienta de búsqueda integrada facilita la navegación a los usuarios.
- Puede programar la publicación de artículos y páginas, lo que le permitirá organizar el trabajo y/o el del equipo de redacción de forma más eficiente en caso de existir.
- Permite crear diferentes usuarios con distintos derechos de administración y edición de contenidos.
- La utilización de plantillas y temas permite hacer rediseños de todo el sitio más rápida y fácilmente que página a página.
- La comunidad de WordPress pone continuamente a disposición de sus usuarios gran cantidad de plugins que aumentan la capacidad básica de la aplicación.

- Los contenidos están organizados fácilmente mediante feedRSS para facilitar su distribución y publicación en otras webs y plataformas.
- Funciona sobre PHP y MySQL.
- Se puede instalar localmente en un servidor propio, lo que proporciona un mayor control sobre su configuración y una fase de pruebas ilimitada.

Además de estas características de publicación WordPress cuenta con:

- **Multisitio:** admite un sitio por instalación, pero gracias a que el sitio puede ser extendido por medio de complementos específicos es fácil administrar y configurar múltiples sitios desde una sola instalación. Esta característica está implementada en el núcleo de WordPress desde la versión 3.0.5.

Después de habilitarse la opción de Multisitio, se puede crear una red (WordPress Network), por la que podrán administrarse varios sitios dentro de una misma instalación, compartiendo temas, plantillas, plugins y dominios. Se puede acceder a cada sitio dentro de un subdirectorio o subdominio del dominio principal.

- **Plantillas:** las plantillas o temas proporcionan un diseño que sirve para establecer la apariencia y estructura de tu blog o página web.

Hay una gran comunidad oficial, tanto profesional como de usuarios, dedicada al diseño de estas plantillas que se suelen listar en el sitio oficial de temas de WordPress una vez han sido comprobadas y aprobadas oficialmente. Aunque la filosofía de WordPress apuesta por un mercado válido según las directrices del W3C, las posibilidades de este sistema, tanto a nivel de diseño, estructura o gestión, y la flexibilidad del sistema de plantillas y widgets en concreto, son enormes y prácticamente permiten tener desde un simple blog hasta un CMS personalizado.

- **Widgets:** WordPress incorpora un sistema de widgets para sus plantillas desde la versión 2.2 que ofrece numerosas posibilidades y flexibilidad para el diseño y estructura de sus blogs y sitios web. Si bien son sumamente útiles, no todas las plantillas los soportan.
- **Complementos (Plugins):** hay un gran número de complementos que potencian el uso de WordPress más allá de un simple blog y que lo hacen un sistema flexible y prácticamente de propósito general. También conocidos como Plugin, son herramientas que extienden la funcionalidad del WordPress. Los hay gratuitos y de pago, y los comprobados y aprobados por WordPress se encuentran listados en la página oficial de plugins de WordPress.

3.1.1 Arquitectura de WordPress

Conocer a fondo la estructura de archivos y carpetas de WordPress es necesario si se quiere llevar correctamente la seguridad de un sitio web. Por otro lado, conocer a fondo como funciona y qué archivos y carpetas se usan, ayudará en la detección y corrección de problemas en sitios web creados con WordPress. Su estructura no ha sufrido muchos cambios en los últimos años con el paso de las versiones.

En esta parte del trabajo se hablará de la estructura de archivos y carpetas predeterminada de WordPress y de algunos casos que se pueden dar al trabajar con algunos plugins.

Una vez que se haya descargado WordPress, se descomprime el .zip y se visualizarán los siguientes archivos y carpetas:

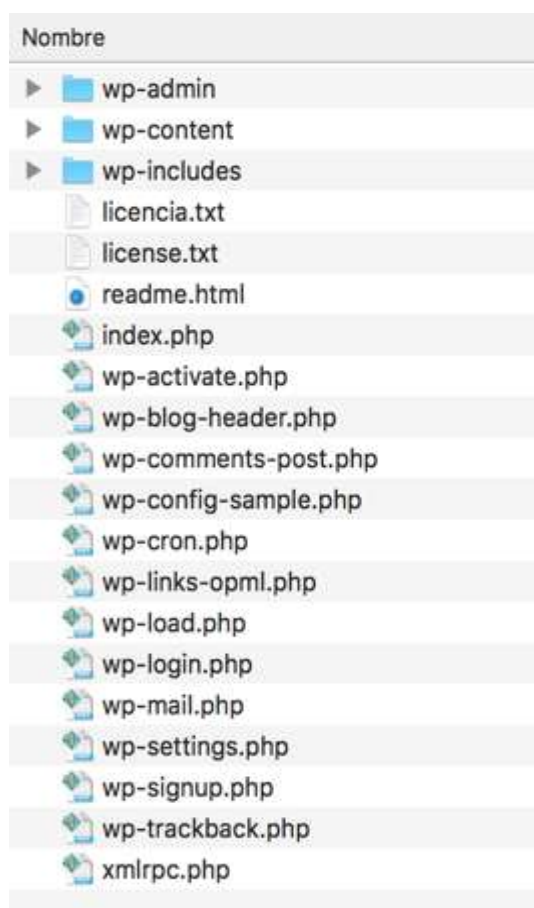


Ilustración 14: Carpetas WordPress

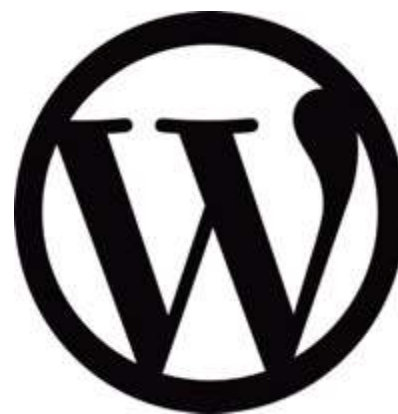


Ilustración 13: Icono de WordPress

- **Carpeta WP-CONTENT:** Es la carpeta donde se guarda todo el contenido en formato de archivos (no base de datos). Si queremos hacer un backup (copia de seguridad) del contenido, además de la base de datos de WordPress, también tenemos que guardar esta carpeta.
- **Archivo INDEX.PHP:** De este archivo poco se puede decir, solo que es el archivo principal al que se accede y desde donde se cargan el resto de partes de WordPress.
- **Archivo WP-CONFIG-SAMPLE.PHP:** Este archivo es la plantilla de lo que finalmente será el archivo WP-CONFIG.PHP tras la instalación del CMS. Éste es el archivo principal de configuración de WordPress, donde se guardan los datos de conexión con la base de datos de WordPress y los parámetros necesarios para conectar con la base de datos.

- **Archivo XMLRPC.PHP:** Se trata de un archivo que ofrece la comunicación mediante el protocoloXMLRPC.PHP. Actualmente WordPress recibe muchos ataques a través de este archivo, por lo que es importante hacer hincapié en su seguridad y su protección.
- **Carpeta WP-ADMIN:** Carpeta donde se guardan los archivos del back-end⁸ de WordPress. Esta parte de la instalación nunca se modifica.
- **Carpeta WP-INCLUDES:** Es una carpeta de archivos que necesita WordPress para funcionar, su API y las librerías principales se encuentran en esta carpeta que tampoco se modificará nunca.
- **Archivo WP-LOGIN.PHP:** Es un archivo bastante importante, ya que es el que se encarga de gestionar el login de los usuarios, tanto usuarios normales como administradores.

Una vez que se ha analizado la raíz de una instalación de WordPress, se procederá a hablar del contenido de la carpeta WP-CONTENT, la carpeta de contenido de WordPress.

Dentro de esta carpeta se podrá encontrar por defecto el siguiente contenido:

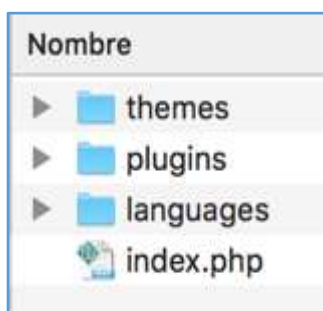


Ilustración 15: Carpeta wp-content

- **Carpeta LANGUAGES:** En esta carpeta se guardan los archivos de traducción tanto de temas y plugins, como del propio WordPress.
- **Carpeta THEMES:** Aquí se guardan los themes (temas), tanto el theme principal activo como los themes instalados.
- **Carpeta PLUGINS:** Aquí se guardan los plugins de la instalación, tanto los plugins activos como los que no lo están.

Normalmente, en una instalación en funcionamiento, dentro de la carpeta WP-CONTENT, nos vamos a encontrar algunas carpetas y archivos demás, dependiendo de qué se añada al trabajo y qué se éste utilizando. Aquí algunos ejemplos:

- **Carpeta CACHE:** Es una carpeta donde los plugins de caché guardan en discos los archivos que se generan con los distintos tipos de caché.

⁸ Back-end: En diseño web (o desarrollo web) hace referencia a la visualización del usuario visitante por un lado (front-end), y del administrador del sitio con sus respectivos sistemas por el otro (back-end). Véase en: https://es.wikipedia.org/wiki/Front-end_y_back-end

- **Carpeta BLOGS.DIR:** Esta carpeta solo existe cuando WordPress Multisite está activado. En esos casos el contenido de los blogs, hijos de la instalación del multisitio, se guarda dentro de wp-content en blogs.dir.
- **Carpeta UPLOADS:** Es la carpeta principal donde se guarda el contenido, almacenando todas las imágenes de forma predeterminada, es decir, todo lo que controla el Media Manager de WordPress.
- **Carpeta UPGRADE:** guarda algunos archivos de las actualizaciones de WordPress; suele ser una carpeta bastante temporal y suele estar vacía.
- **Archivo ADVANCED-CACHE.PHP:** Es un archivo que usan algunos plugins de caché para el caché de página en disco. Normalmente lo usan los plugins de caché más eficientes y el archivo solo está en WP-CONTENT cuando se está utilizando.

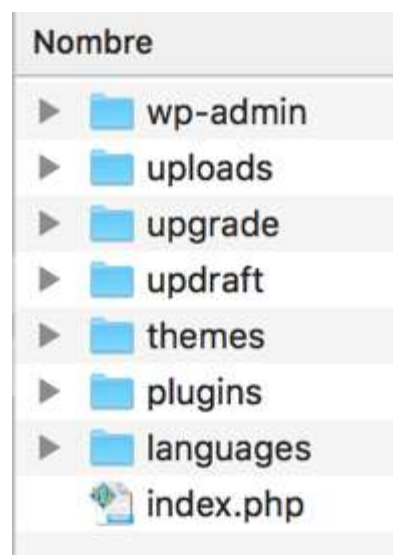


Ilustración 16: Carpeta wp-content +

- **Archivo OBJECT-CACHE.PHP:** Es muy similar a ADVANCED-CACHE.PHP pero con el caché de objetos. Al igual que el caso anterior forma parte del API de WordPress y solo está presente cuando el caché de objetos esta activado.
- **Archivo DB.PHP:** Es un archivo que controla (o hace de intermediario) la conexión con la base de datos MySQL desde PHP (WordPress), aunque también es usado para implementar caché de consultas a la base de datos en forma de plugin.

Evidentemente, WP-CONTENT, al ser una carpeta de contenido siempre tendrá bastantes archivos, es más, dependiendo de los plugins instalados, va a tener unos archivos u otros; estos son algunos ejemplos:

- **Carpeta W3TC-CONFIG:** Aquí se guardan las configuraciones del plugin W3 Total Cache; debemos tener mucho cuidado con eliminar esta carpeta en caso de que el plugin W3 Total Cache no esté totalmente desactivado.
- **Carpeta WP-SIMPLE-AUCTION:** Aquí se encuentran las configuraciones del plugin Simple Auctions para realizar subastas.
- **Carpeta WC-LOGS:** En esta carpeta se guardan logs y registros de algunos plugins que trabajan con WooCommerce; en pocas ocasiones es utilizada, aunque teóricamente el propio WooCommerce también puede escribir en ella.

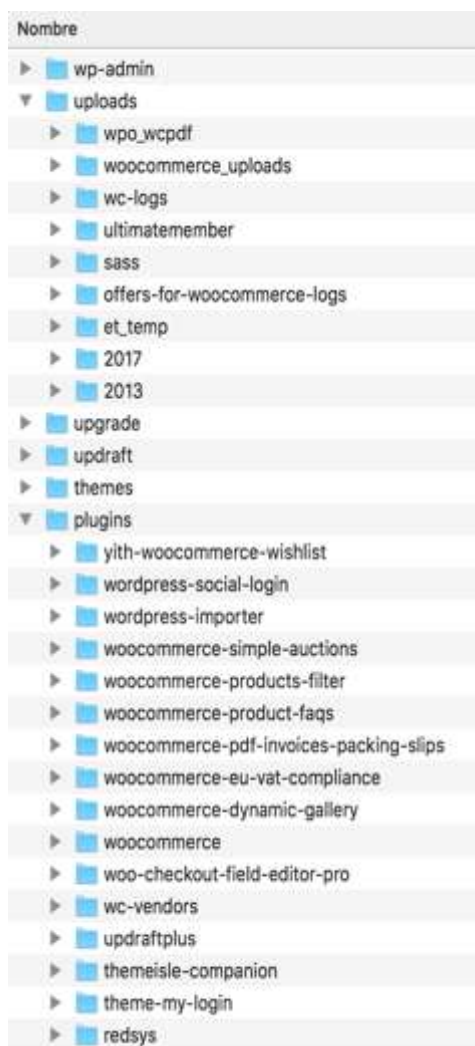


Ilustración 17: Carpeta wp-content ++

- **Carpeta WOOCOMMERCE-UPLOADS:** Aquí se guardan los archivos de imágenes y similares que se suben al Media Manager de WordPress desde WooCommerce.

- **Carpeta WC-VENDORS:** Es la carpeta que se crea al instalar el plugin WC-VENDORS, que transforma WooCommerce en un Marketplace.

- **Carpeta WOOCOMMERCE-PRODUCT-FAQS:** Es la carpeta que se crea al instalar el plugin WooCommerce Product Faqs para realizar preguntas directamente al vendedor sobre un artículo.

Es imposible listar todas las posibilidades que podemos encontrar, ya que como se especificado anteriormente, wp-content es una carpeta de uso común para contenido y puede ser usada tanto por plugins como por themes.

Esto no ocurre con las carpetas wp-include y wp-admin que son inalterables.

Bibliografía [\[17\]](#)

3.1.2 Métodos de instalación de WordPress

Una vez se conocen las carpetas y archivos clave para que WordPress funcione, se va a dar la oportunidad al desarrollador de escoger uno de los dos métodos de instalación.

3.1.2.1 Instalación de WordPress en un Servidor LOCAL

Con este tipo de instalación se podrá desarrollar el sitio web dentro de nuestro propio equipo pudiendo configurarlo hasta cumplir los objetivos propuestos para el trabajo.

Una vez el sitio esté completo, se podrá seguir con el método de instalación en un hosting, por el cual, podremos subir todos los datos de nuestro equipo a un hosting público o privado y utilizar un dominio para que el sitio sea público y accesible.

Este tipo de instalación se recomienda para hacer todo tipo de pruebas y configuraciones del sitio antes de publicarlo. Se recomienda empezar por este tipo de instalación y una vez se obtengan conocimientos sobre WordPress, publicar el sitio.

Se deben seguir los siguientes pasos para la instalación de WordPress en un servidor local.

- 1) Descargar la última versión de WordPress (gratuita) desde la página oficial en español (<https://es.wordpress.org>) y descomprimir el fichero .zip. El resultado es una carpeta con los archivos de WordPress expuestos en el apartado anterior.
- 2) Una vez descargado WordPress, se debe instalar un servidor local para alojar los archivos descargados. En el [Anexo 1](#) de este trabajo se definirán los pasos a seguir para la instalación de un servidor local. Para este trabajo se ha utilizado el servidor MAMP.
- 3) Para el servidor local es importante no olvidar tres datos que serán de gran importancia para el funcionamiento de WordPress: el directorio por defecto donde se instala WordPress (depende del SO del ordenador donde se desarrolle el sitio web), el servidor SMTP ('localhost' por defecto) y el nombre de usuario y clave de MySQL ('root' y 'root' por defecto).
- 4) Base de datos: WordPress como la mayoría de CMS, utiliza una base de datos para funcionar. Una base de datos es una serie de archivos compuestos por textos donde se guardan las configuraciones y contenido del CMS y mediante programación, se van utilizando estos datos según se vayan necesitando. Para acceder al servidor local desde el navegador, una vez iniciado el servidor, hay que escribir <http://localhost> y se verá lo siguiente:

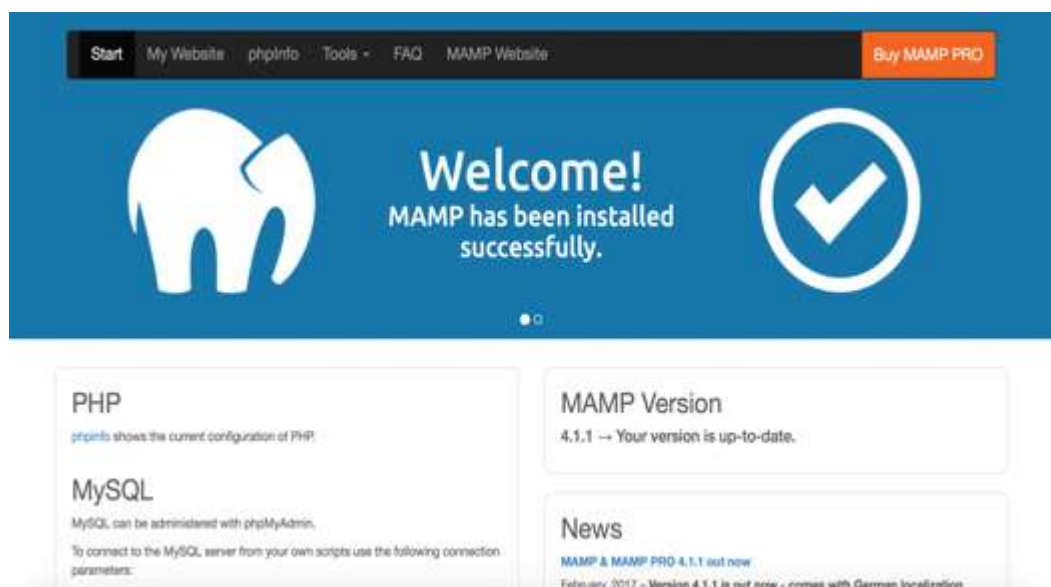


Ilustración 18: Acceso a la página del servidor local

- 5) A continuación se debe acceder a phpMyAdmin pinchando en el enlace donde pone MySQL (Ilustración 18). En caso de no haber enlace se deberá poner en la barra de direcciones: `http://localhost/phpmyadmin` y se podrá acceder. PhpMyAdmin es una herramienta básica que incluyen todos los servidores creada para gestionar la administración de las bases de datos MySQL y así poder crear, eliminar y modificar bases de datos con ella. Dentro, una vez se dé el nombre de usuario y clave, se podrás ver lo siguiente:

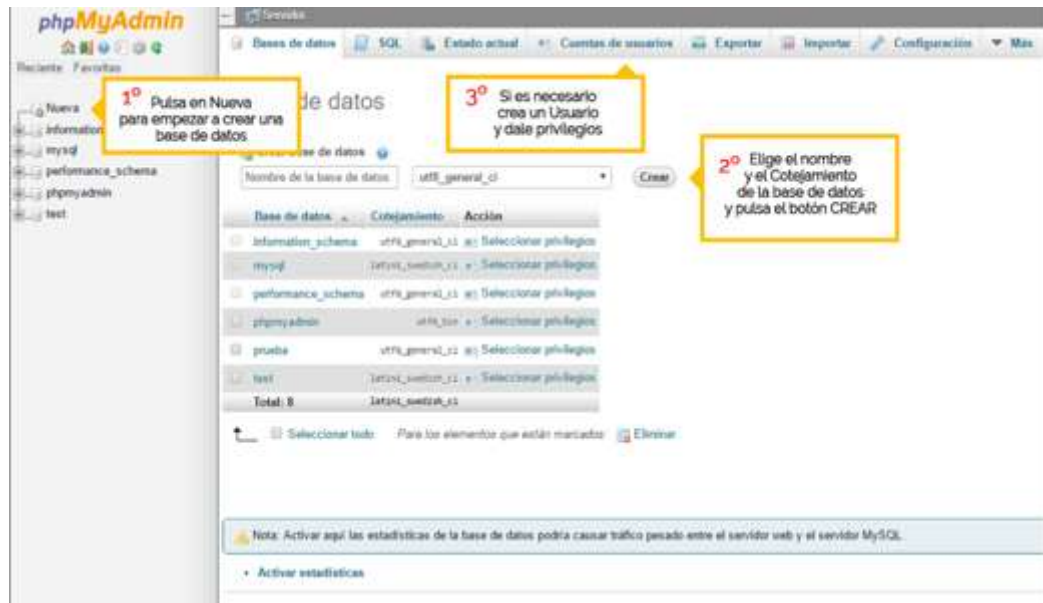


Ilustración 19: PhpMyAdmin

- 6) Se deben seguir los pasos indicados en la ilustración 19 para crear una base de datos. Se recomienda ponerle el nombre a la base de datos 'wordpress' para evitar futuros problemas. Así mismo, es recomendable realizar el tercer paso visible en la Ilustración 19.
- 7) Una vez creada la base de datos (sin introducir nada en ella) hay que dirigirse a la carpeta del servidor local, en este caso `"/Aplicaciones/MAMP"`, hay que desplegarla y se verá un directorio llamado 'htdocs'. Aquí es donde se pega toda carpeta de Wordpress descargada con anterioridad en el paso 1.
- 8) Una vez se han realizado los 7 pasos anteriores, se procede a la instalación de WordPress en el servidor local.

En la carpeta `Aplicaciones/MAMP/htdocs/wordpress/` hay un archivo llamado `wp-config-sample.php`. Hay que modificar el nombre del archivo por `wp-config.php`. Para ello se puede usar cualquier editor de texto. Dentro, se podrá ver una parte como esta:

```
// ** Ajustes de MySQL. Solicita estos datos a tu proveedor de alojamiento
web. ** //

/** El nombre de tu base de datos de WordPress */
define('DB_NAME', 'nombre_de_tu_base_de_datos');

/** Tu nombre de usuario de MySQL */
define('DB_USER', 'nombre_de_usuario_de_MySQL');

/** Tu contraseña de MySQL */
define('DB_PASSWORD', 'clave_MySQL');

/** Host de MySQL (es muy probable que no necesites cambiarlo) */
define('DB_HOST', 'localhost');
```

Tabla 5: Configuración wp-config

Se tienen que cambiar los campos donde pone: 'nombre_de_tu_base_de_datos', 'nombre_de_usuario_de_MySQL' y 'clave_MySQL'. El primero será el nombre de la base de datos que se crea en el apartado 6; el segundo el nombre del usuario al que se esté conectado en phpMyAdmin y el tercero es la clave de phpMyAdmin. Hay que tener cuidado con las comillas. Por defecto, los dos últimos se dejarán como 'root' y 'root'.

Una vez se introduzcan los datos, se guarda el archivo y se cierra.

- 9) A continuación, se escribe en el navegador: 'http://localhost/wordpress', para acceder al instalador de WordPress. Se deben seguir los pasos de configuración y una vez terminado se llegará a la pantalla principal de WordPress y se dará por finalizada la instalación de WordPress.



Ilustración 20: Página principal de WordPress

3.1.2.2 *Instalación de WordPress en un Servidor o Hosting*

Este método subirá todos los archivos necesarios para que funcione WordPress en un hosting público o privado. Se podrá empezar con este tipo de instalación y configurar WordPress seguidamente. Este paso se tendrá que realizar tarde o temprano para publicar nuestro sitio y hacer que sea visible, pero se recomienda, previamente, la instalación en un servidor local para aprender a configurar WordPress. Esto queda en manos del desarrollador.

Antes de instalar WordPress en un hosting se recomienda saberlos requisitos que se necesitan para llevar a cabo esta instalación de forma correcta:

- **Dominio:** es una dirección web, que normalmente coincide con el nombre de la web. Pueden ser gratuitos y/o de pago. Ejemplos de dominio: www.ejemplo.com, www.ejemplo.es, www.ejemplo.online, etc. Los dominios son únicos para cada sitio.
- **Hosting:** es un servidor en internet donde va a estar alojada nuestra web. El alojamiento web debe soportar programación en PHP y al menos una base de datos en MySQL; hoy en día casi todos los hosting lo incluyen. Pueden ser gratuitos y/o de pago (públicos o privados).
- **Base de Datos:** WordPress almacena la información en una base de datos, por lo tanto tenemos que crear una o el hosting donde alojemos la web proporcionará la opción de crearla automáticamente.
- **El Programa WordPress:** será el gestor de contenidos web que se usará en este trabajo y podrá ser descargado gratuitamente desde la página oficial.

Dados los requisitos necesarios para instalar WordPress en un hosting, queda en manos del desarrollador elegir el procedimiento que le permita llegar a los objetivos impuestos para el trabajo. Aquí se muestran algunos ejemplos de instalación de WordPress en un hosting con sus pros y contras.

- 1) Comprando un dominio y alojando el sitio en un hosting privado.
 - Pros: el dominio, independientemente del nombre del sitio web, será común para los usuarios que lo visiten. También se contará con una gran seguridad ya que normalmente los hosting de pago proporcionan el certificado SSL⁹. Se tendrá más espacio para subir archivos, más bases de datos, etc.
 - Contras: si se quieren contar con todas las prestaciones habrá que desembolsar una cantidad importante.

⁹ Certificado SSL: es una manera de decirles a los clientes que el sitio es auténtico, real y confiable para ingresar datos personales. SSL es un protocolo de seguridad que hace que la transmisión de los datos entre un usuario y el servidor web sea cifrada o encriptada. Véase en: <https://www.certsuperior.com/QueesunCertificadoSSL.aspx>

- 2) Utilizar el dominio que te proporciona un hosting privado.
 - Pros: no se necesitará pensar en si el dominio estará disponible o no ya que el hosting te incluye la terminación del sitio, solo se tendrá que pensar en el nombre de la página. Al ser un hosting privado contaremos con seguridad del certificado SSL.
 - Contras: Al igual que el anterior, este método es de pago.
- 3) Utilizar el dominio que te proporciona un hosting público.
 - Pros: tanto el dominio como el hosting son gratuitos.
 - Contras: al ser gratuito se tendrá un límite de espacio en el servidor para albergar los datos, solo se dispondrá de un número limitado de bases de datos, no contará con tanta seguridad, etc. Todo esto depende del hosting que se elija, algunos ofrecen mejores prestaciones.

Como pueden verse en estos ejemplos hay muchas maneras de llevar a cabo la instalación de WordPress. Todo este proceso queda en manos del desarrollador que deberá estudiar y elegir la mejor opción para la realización del trabajo.

En el caso de este trabajo, se ha trabajado sobre un servidor local durante un periodo de tiempo para familiarizarse con el entorno. Una vez comprendido el funcionamiento de este CMS y buscando siempre completar los objetivos impuestos, se ha hecho la instalación de WordPress en un hosting público con la opción de elegir un dominio libre por la empresa que nos da el servicio de hosting gratuito.

Bibliografía [\[18\]](#) [\[19\]](#)

3.1.3 Base de datos de WordPress

Resumen de la base de datos que necesita WordPress.

3.1.3.1 Tablas

Este punto es una descripción general de las tablas que WordPress 4.8 genera durante una instalación normal en la base de datos.

Nombre de la tabla	Descripción
wp_comments	Los comentarios en WordPress se mantienen aquí.
wp_commentmeta	Cada comentario incorpora datos llamados meta data que se almacenan aquí.
wp_links	Se guarda la información relativa a los enlaces añadidos en

	la funcionalidad de Enlaces de WordPress, obsoleta actualmente aunque puede volver a usarse con el plugin Links Manager.
wp_options	Los ajustes que se configuran en la página de ajustes de la administración de WordPress se almacenan en esta tabla. Ejemplo la configuración de los widget o plugins.
wp_postmeta	Cada entrada contiene información extra denominada data que se almacena aquí. Hay plugins que también añaden su propia información a esta tabla.
wp_post	Las entradas (posts), páginas (pages) y elementos de menú se guardan aquí.
wp_termmeta	Cada entrada tiene información extra que se almacena aquí.
wp_terms	Las categorías tanto de las entradas como de los enlaces y las etiquetas de las entradas se almacenan
wp_term_relationships	Las entradas y enlaces están asociadas a categorías y/o etiquetas de la tabla wp_terms, y esta asociación se guarda aquí.
wp_term_taxonomy	Esta tabla describe la taxonomía (categoría, enlace o etiqueta) de cada línea de la tabla wp_terms.
wp_user_meta	Cada usuario contiene información denominada data y se almacena aquí.
wp_users	La lista de usuarios se guarda aquí

Tabla 6: Tablas base de datos de WordPress

Bibliografía [\[20\]](#) [\[21\]](#)

3.1.3.2 Diagrama Entidad-Relación de las tablas de WordPress

Diagrama entidad-relación de la base de datos que muestra las tablas con sus variables y claves primarias, y que necesita WordPress para funcionar.

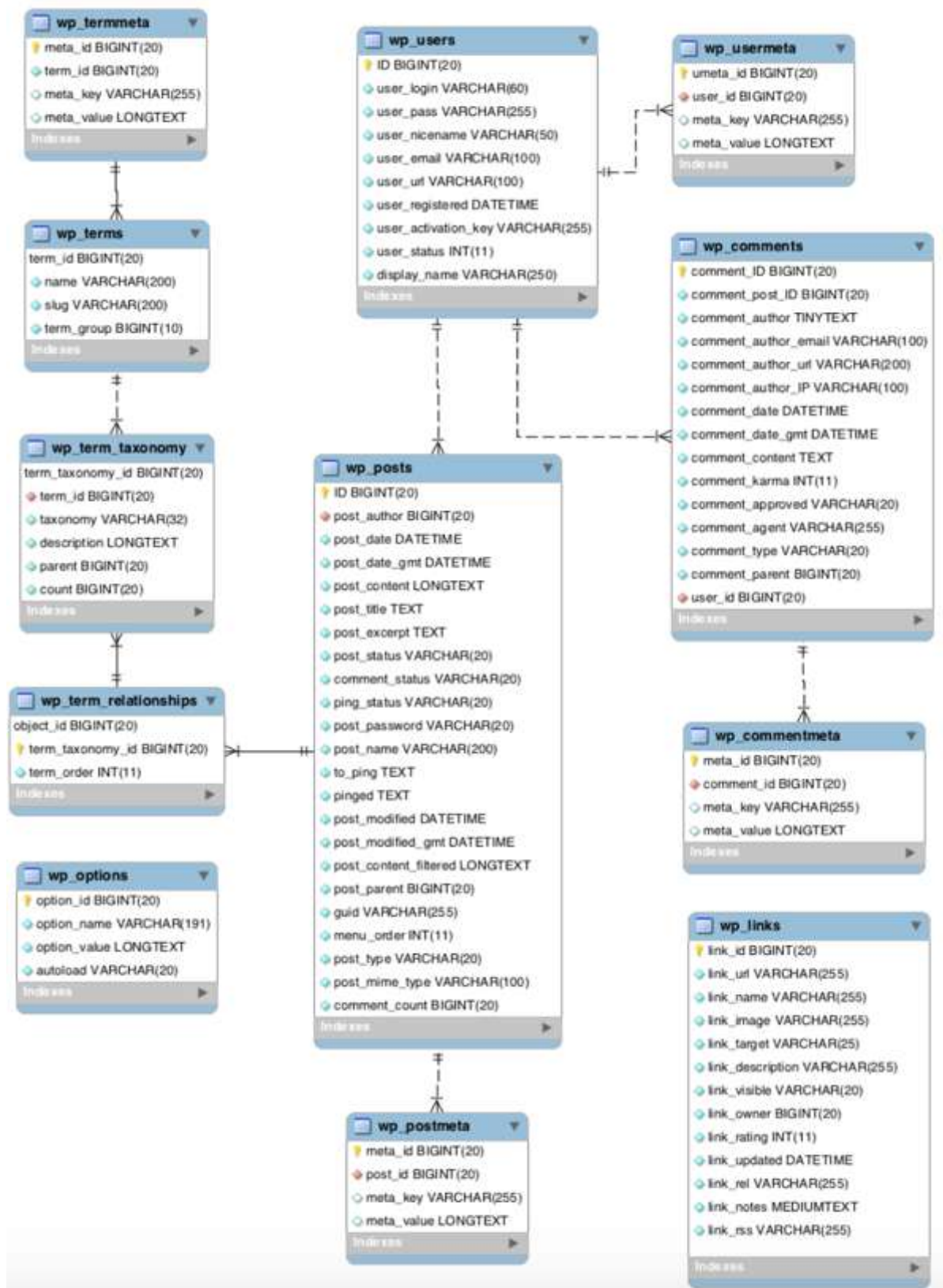


Ilustración 21: Diagrama E-R de la base de datos de WordPress

3.2 Android

Como hemos comentado anteriormente, existen muchas plataformas para móviles (iOS, Windows Phone, BlackBerry, etc.); sin embargo Android presenta una serie de características que lo hace diferente. Es el primero que combina, en una misma solución, las siguientes cualidades:

- **Plataforma realmente abierta.** Es una plataforma de desarrollo libre basada en Linux y de código abierto. Una de sus grandes ventajas es que se puede usar y customizar¹⁰ el sistema sin pagar licencias.
- **Portabilidad asegurada.** Las aplicaciones finales son desarrolladas en Java, lo que nos asegura que podrán ser ejecutadas en una gran variedad de dispositivos, tanto presentes como futuros. Esto se consigue gracias al concepto de máquina virtual.
- **Arquitectura basada en componentes inspirados en Internet.** Por ejemplo, el diseño de la interfaz de usuario se hace en XML, lo que permite que una misma aplicación se ejecute en un móvil de pantalla reducida o en un netbook.
- **Filosofía de dispositivo siempre conectado a Internet.**
- **Gran cantidad de servicios incorporados.** Por ejemplo, localización basada tanto en GPS como en redes, bases de datos con SQL, reconocimiento y síntesis de voz, navegador, multimedia, etc.
- **Aceptable nivel de seguridad.** Los programas se encuentran aislados unos de otros gracias al concepto de ejecución dentro de una capa que hereda de Linux. Además, cada aplicación dispone de una serie de permisos que limitan su rango de actuación (servicios de localización, acceso a internet, etc.).
- **Optimizado para baja potencia y poca memoria.** Por ejemplo, Android utiliza la Máquina Virtual Dalvik. Se trata de una implementación de Google de la máquina virtual de Java optimizada para dispositivos móviles.
- **Alta calidad de gráficos y sonido.** Gráficos vectoriales suavizados, animaciones inspiradas en Flash, gráficos en 3 dimensiones basados en OpenGL. Incorpora los códecs¹¹ estándar más comunes de audio y vídeo, incluyendo H.264 (AVC), MP3, AAC, etc.

¹⁰ Customizar: modificar algo de acuerdo a la preferencias personales. Véase en: <http://definicion.de/customizar/>

¹¹ Códecs: programa capaz de codificar y descodificar una señal o flujo de datos digitales. Véase en: <https://es.wikipedia.org/wiki/Códec>

Google adquiere Android Inc. en el año 2005. Se trataba de una pequeña compañía que acababa de ser creada, orientada a la producción de aplicaciones para terminales móviles. Ese mismo año empiezan a trabajar en la creación de una máquina virtual Java optimizada para móviles (Dalvik VM). En el año 2007 se crea el consorcio Handset Alliance¹ con el objetivo de desarrollar estándares abiertos para móviles. Está formado por Google, Intel, Texas Instruments, Motorola, Samsung, Ericsson, Vodafone, entre otros. Una pieza clave de los objetivos de esta alianza es promover el diseño y difusión de la plataforma Android. Sus miembros se han comprometido a publicar una parte importante de su propiedad intelectual como código abierto bajo la licencia Apache v2.0.

En noviembre del 2007 se lanza una primera versión del Android Software Development Kit (SDK). Al año siguiente aparece el primer móvil con Android (T-Mobile G1). En octubre, Google libera el código fuente de Android, principalmente bajo licencia de código abierto Apache (licencia GPL v2 para el núcleo). Ese mismo mes, se abre Android Market para la descarga de aplicaciones. En abril del 2009, Google lanza la versión 1.5 del SDK que incorpora



Ilustración 22: T-mobile G1, primer dispositivo Android

nuevas características como el teclado en pantalla. A finales del 2009 se lanza la versión 2.0 y durante el 2010 las versiones 2.1, 2.2 y 2.3. Durante el año 2010, Android se consolida como uno de los sistemas operativos para móviles más utilizados, con resultados cercanos al iPhone, e incluso superando al sistema de Apple en EE.UU (Estados Unidos).

En el 2011 se lanzan la versión 3.x específica para tabletas y 4.x tanto para móviles como para tabletas. Durante este año, Android se consolida como la plataforma para móviles más importante, alcanzando una cuota de mercado superior al 50%. En 2012, Google cambia su estrategia en su tienda de descargas online, reemplazando Android Market por Google Play Store, donde en un solo portal unifica la descarga de aplicaciones como de contenidos.

3.2.1 Arquitectura de Android

El siguiente gráfico muestra la arquitectura de Android. Como se puede ver, está formada por cuatro capas. Una de las características más importantes es que todas ellas están basadas en software libre.

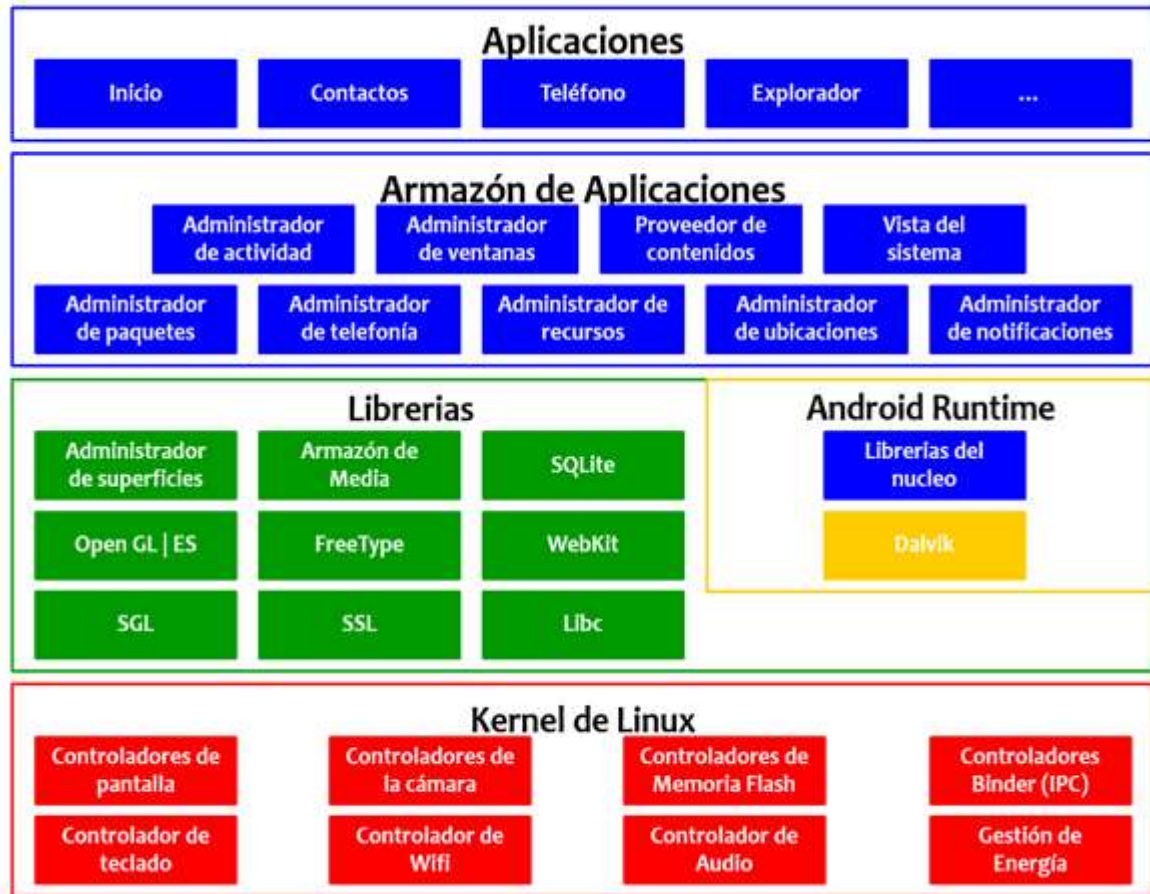


Ilustración 23: Arquitectura de Android

3.2.1.1 Kernel de Linux

El núcleo de Android está formado por el sistema operativo Linux. Esta capa proporciona servicios como la seguridad, el manejo de la memoria, el multiproceso, la pila de protocolos y el soporte de drivers¹² para dispositivos.

Esta capa del modelo actúa como capa de abstracción entre el hardware y el resto de la pila. Por lo tanto, es la única que es dependiente del hardware.

3.2.1.2 Runtime de Android (RTA)

Está basada en el concepto de máquina virtual¹³ utilizado en Java. Dadas las limitaciones de los dispositivos donde ha de ejecutarse Android (con poca memoria y procesador limitado) no fue posible utilizar una máquina virtual Java estándar. Google tomó la decisión de crear una nueva, la máquina virtual Dalvik, que respondería mejor a

¹² Drivers: programas cuya finalidad es relacionar el sistema operativo con los dispositivos hardware. Véase en: <https://basicoyfacil.wordpress.com/2008/12/16/que-es-un-driver/>

¹³ Máquina virtual: software que simula otro dispositivo y que puede ejecutar aplicaciones como si fuese ese dispositivo. Véase en: https://es.wikipedia.org/wiki/Máquina_virtual

estas limitaciones. Algunas características de la máquina virtual Dalvik que facilitan esta optimización de recursos son: que ejecuta ficheros Dalvik ejecutables (.dex) (formato optimizado para ahorrar memoria). Además, 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 y el manejo de la memoria a bajo nivel. También se incluye en el RTA el “core libraries” con la mayoría de las librerías disponibles en el lenguaje Java.

3.2.1.3 Librerías

Incluye un conjunto de librerías en C/C++ usadas en varios componentes de Android. Están compiladas en el código nativo del procesador. Muchas de las librerías utilizan proyectos de código abierto. Algunas de estas librerías son:

- **System C library:** una derivación de la librería BSD de C estándar, adaptada para dispositivos embebidos basados en Linux.
- **Media Framework:** librería basada en PacketVideo's OpenCORE; soporta códecs de reproducción y grabación de multitud de formatos de audio, vídeo e imágenes MPEG4, H.264, MP3, AAC, AMR, JPG y PNG.
- **Surface Manager:** maneja el acceso al subsistema de representación gráfica en 2D y 3D.
- **WebKit:** soporta un moderno navegador web utilizado en el navegador Android y en la vista WebView. Se trata de la misma librería que utiliza Google Chrome y Safari de Apple.
- **SGL:** motor de gráficos 2D.
- **Librerías 3D:** implementación basada en OpenGL ES 1.0 API. Las librerías utilizan el acelerador hardware 3D si está disponible, o el software altamente optimizado de proyección 3D.
- **FreeType:** fuentes en bitmap y renderizado vectorial.
- **SQLite:** potente y ligero motor de bases de datos relacionales disponible para todas las aplicaciones.
- **SSL:** proporciona servicios de encriptación Secure Socket Layer.

3.2.1.4 Armazón de aplicaciones o entorno de aplicación

Proporciona una plataforma de desarrollo libre para aplicaciones con gran riqueza e innovaciones (sensores, localización, servicios, barra de notificaciones, etc.). Esta capa ha sido diseñada para simplificar la reutilización de componentes. Las aplicaciones pueden publicar sus capacidades y otras pueden hacer uso de ellas (sujetas a las restricciones de seguridad). Este mismo mecanismo permite a los usuarios remplazar

componentes. Una de las mayores fortalezas del entorno de aplicación de Android es que se aprovecha el lenguaje de programación Java. El SDK de Android no acaba de ofrecer todo lo disponible para su estándar del Java Runtime Environment o entorno de ejecución Java (JRE), pero es compatible con una fracción muy significativa de la misma. Los servicios más importantes que incluye son:

- **Views:** extenso conjunto de vistas (parte visual de los componentes).
- **Resource Manager:** proporciona acceso a recursos que no son en el código.
- **Activity Manager:** maneja el ciclo de vida de las aplicaciones y proporciona un sistema de navegación entre ellas.
- **Notification Manager:** permite a las aplicaciones mostrar alertas personalizadas en la barra de estado.
- **Content Providers:** mecanismo sencillo para acceder a datos de otras aplicaciones (como los contactos).

3.2.1.5 Aplicaciones

Este nivel está formado por el conjunto de aplicaciones instaladas en una máquina Android. Todas las aplicaciones han de ser ejecutadas en la máquina virtual Dalvik para garantizar la seguridad del sistema.

Normalmente, las aplicaciones Android están escritas en Java. Para desarrollar aplicaciones en Java podemos utilizar el Android SDK. Existe otra opción consistente en desarrollar las aplicaciones utilizando C/C++. Para esta opción podemos utilizar el Android NDK (Native Development Kit).

3.2.2 Entornos de desarrollo

Para el desarrollo de las aplicaciones vamos a poder utilizar un potente y moderno entorno de desarrollo. Al igual que Android, todas las herramientas están basadas en software libre. Aunque existen varias alternativas para desarrollar aplicaciones en Android:

- Java Runtime Environment 5.0 o superior.
- Eclipse (Eclipse IDE for Java Developers).
- Android SDK (Google).
- Eclipse Plug-in (Android Development Toolkit- ADT).
- Android Studio.

Para la realización de este trabajo contaremos con Android Studio.

3.2.3 Versiones de Android

Antes de empezar un proyecto en Android hay que elegir la versión del sistema para la que deseamos realizar la aplicación. Es muy importante observar que hay clases y métodos que están disponibles a partir de una versión. Si las vamos a usar, hemos de conocer la versión mínima necesaria.

Cuando se ha lanzado una nueva plataforma siempre ha sido compatible con las versiones anteriores. Es decir, solo se añaden nuevas funcionalidades y, en el caso de modificar alguna funcionalidad, no se elimina, se etiquetan como obsoletas pero se pueden continuar utilizando.

A continuación se describen las plataformas lanzadas hasta la fecha con una breve descripción de las novedades introducidas. Las plataformas se identifican de tres formas alternativas: versión, nivel de API y nombre. El nivel de API corresponde a números enteros comenzando desde 1. Para los nombres se han elegido postres en orden alfabético Cupcake (v1.5), Donut (v1.6), Éclair (v2.1), Froyo (v2.2), Gingerbread (v2.3), Honeycomb (v3.0), Ice Cream (v2.4), etc. Las dos primeras versiones, que hubieran correspondido a las letras A y B, no recibieron nombre.

- **Android 1.0 Nivel de API 1 (septiembre 2008)**

Primera versión de Android. Nunca se utilizó comercialmente, por lo que no tiene mucho sentido desarrollar para esta plataforma.

- **Android 1.1 Nivel de API 2 (febrero 2009)**

No se añadieron apenas funcionalidades, simplemente se fijaron algunos errores de la versión anterior. Es la opción a escoger si queremos desarrollar una aplicación compatible con todos los dispositivos Android. No obstante apenas existen usuarios con esta versión.

- **Android 1.5 Nivel de API 3 (abril 2009, Cupcake)**

Es la primera versión con algún usuario (un 1,3% en julio del 2011). Como novedades, incorpora la posibilidad de teclado en pantalla con predicción de texto, los terminales ya no tienen que tener un teclado físico, así como la capacidad de grabación avanzada de audio y vídeo. También aparecen los widgets de escritorio y live folders¹⁴. Incorpora soporte para Bluetooth estéreo, por lo que permite conectarse automáticamente a auriculares Bluetooth. Las transiciones entre ventanas se realizan mediante animaciones.



Ilustración 24: Android 1.5 Cupcake

¹⁴ Live folders: organizar mejor las aplicaciones agrupándolas en diferentes carpetas. Véase en: <http://www.muywindows.com/2014/08/04/asi-funciona-live-folders-en-windows-phone-8-1-gdr-1>

- **Android 1.6 Nivel de API 4 (diciembre 2009, Donut)**



Ilustración 25: Android 1.6 Donut

Permite capacidades de búsqueda avanzada en todo el dispositivo. También incorpora gestos y multi-touch. Permite la síntesis de texto a voz. También facilita que una aplicación pueda trabajar con diferentes densidades de pantalla.

Soporte para resolución de pantallas WVGA. Aparece un nuevo atributo XML, `OnClick`, que puede especificarse en una vista. Se mejora la Play Store, antes Android Market, permitiendo una búsqueda más sencilla de aplicaciones. Soporte para CDMA/EVDO, 802. 1X y Virtual Private Network (VPN)¹⁵. Mejoras en la aplicación de la cámara.

- **Android 2.0/2.1 Nivel de API 5-7 (octubre 2009, enero 2010 Éclair)**

Esta versión de API apenas cuenta con usuarios, dado que la mayoría de fabricantes pasaron directamente de la versión 1.6 a la 2.1. Como novedades cabría destacar que incorpora un API para manejar Bluetooth 2.1. Nueva funcionalidad que permite sincronizar adaptadores para conectarlo a cualquier dispositivo. Ofrece un servicio centralizado



Ilustración 26: Android 2.0/2.1 Éclair

de manejo de cuentas. Mejora la gestión de contactos y ofrece más ajustes en la cámara. Se optimiza la velocidad de hardware. Se aumenta el número de tamaños de ventana y resoluciones soportadas. Nueva interfaz del navegador y soporte para HTML5. Mejoras en el calendario y soporte para Microsoft Exchange. La clase Motion Event ahora soporta eventos en pantallas multitáctil.

La siguiente actualización, la 2.1, se considera una actualización menor, por lo que le siguieron llamando Éclair. Destacamos el reconocimiento de voz que permite introducir un campo de texto sin necesidad de utilizar el teclado. También permite desarrollar fondos de pantalla animados. Se puede obtener información sobre la señal de la red actual que posea el dispositivo. En el paquete WebKit se incluyen nuevos métodos para manipular bases de datos almacenadas en web. También se permite obtener permisos de geolocalización y modificarlos en WebView. Incorpora mecanismos para administrar la configuración de la caché de aplicaciones, almacenamiento web y modificar la resolución de la pantalla

¹⁵ VPN: Red Privada Virtual que permite una extensión segura de la LAN sobre un red pública o no controlada. Véase en: https://es.wikipedia.org/wiki/Red_privada_virtual

- **Android 2.2 Nivel de API 8 (mayo 2010, Froyo)**



Ilustración 27: Android 2.2 Froyo

Como característica más destacada se puede indicar la mejora de velocidad de ejecución de las aplicaciones (ejecución del código de la CPU de 2 a 5 veces más rápido que en la versión 2.1). Esto se consigue con la introducción de un nuevo compilador JIT de la máquina Dalvik.

Se añaden varias mejoras relacionadas con el navegador Web, como el soporte de Adobe Flash 10.1 y la incorporación del motor JavaScript V8 utilizado en Chrome, o la incorporación del campo de “subir fichero” en un formulario.

El desarrollo de aplicaciones permite las siguientes novedades: se puede preguntar al usuario si desea instalar una aplicación en un medio de almacenamiento externo (como una tarjeta SD), como alternativa a la instalación en la memoria interna del dispositivo. Las aplicaciones se actualizan de forma automática cuando aparece una nueva versión. Proporciona un servicio para la copia de seguridad de datos que se puede realizar desde la propia aplicación para garantizar al usuario el mantenimiento de sus datos. Por último, facilita que las aplicaciones interaccionen con el reconocimiento de voz y que terceras partes proporcionen nuevos motores de reconocimiento.

Se mejora la conectividad: ahora podemos utilizar nuestro teléfono para dar acceso a Internet a otros dispositivos (tethering), tanto por USB como por Wi-Fi. También se añade el soporte a Wi-Fi IEEE 802.11n.

Se añaden varias mejoras en diferentes componentes: en el API gráfica OpenGL ES se pasa a soportar la versión 2.0. También puede realizar fotos o vídeos en cualquier orientación (incluso vertical) y configurar otros ajustes de la cámara. Para finalizar, permite definir modos de interfaz de usuario (“automóvil” y “noche”) para que las aplicaciones se configuren según el modo seleccionado por el usuario.

- **Android 2.3 Nivel de API 9-10 (diciembre 2010, Gingerbread)**

Debido al éxito de Android en las nuevas tabletas, ahora soporta mayores tamaños de pantalla y resoluciones (WXGA y superiores).

Incorpora un nuevo interfaz de usuario con un diseño actualizado. Dentro de las mejoras de la interfaz de usuario destacamos la mejora de la funcionalidad de “cortar, copiar y pegar” y un teclado en pantalla con capacidad multitáctil.

Incluye soporte nativo para varias cámaras, pensado en la segunda cámara usada en videoconferencia. La incorporación de esta segunda cámara ha propiciado la inclusión de reconocimiento facial para identificar al usuario del terminal.

La máquina virtual Dalvik para Android introduce un nuevo recolector de basura que minimiza las pausas de la aplicación, ayudando a garantizar una mejor animación y el aumento de la capacidad de respuesta en juegos y aplicaciones similares. Se trata de corregir así una de las lacras de este sistema operativo móvil, que en versiones previas no ha sido capaz de cerrar bien las aplicaciones en desuso. Se dispone de mayor apoyo para el desarrollo de código nativo (NDK). También se mejora la gestión de energía y control de aplicaciones. Y se cambia el sistema de ficheros, que pasa de YAFFS a EXT4.

Entre otras novedades destacamos el soporte nativo para telefonía sobre Internet VoIP/SIP. El soporte para reproducción de vídeo WebMA/P8 y codificación de audio AAC. El soporte para la tecnología NFC. Las facilidades en el audio, gráficos y entradas para los desarrolladores de juegos. El soporte nativo para más sensores (como giroscopios y barómetros) y un gestor de descargas para las descargas largas.

- **Android 3.0 Nivel de API 11-13 (febrero 2011, Honeycomb)**



Ilustración 29: Android 3.0 Honeycomb

Para mejorar la experiencia de Android en las nuevas tabletas se lanza la versión 3.0 optimizada para dispositivos con pantallas grandes. La nueva interfaz de usuario fue completamente rediseñada con paradigmas nuevos para la interacción, navegación y personalización. La nueva interfaz se pone a disposición de todas las aplicaciones, incluso las construidas para versiones anteriores de la plataforma.



Ilustración 28: Android 2.3 Gingerbread

Las principales novedades de este SDK son:

Con el objetivo de adaptar la interfaz de usuario a pantallas más grandes se incorporan las siguientes características: resolución por defecto WXGA(1280x800), escritorio 3D con widgets rediseñados, nuevos componentes y vistas, notificaciones mejoradas, arrastrar y soltar, nuevo cortar y pegar, barra de acciones para que las aplicaciones dispongan de un menú contextual siempre presente y otras características para aprovechar estas pantallas.

Se mejora la reproducción de animaciones 2D/3D gracias al renderizador OpenGL acelerado por hardware. El nuevo motor de gráficos Rederscript saca un gran rendimiento de los gráficos en Android e incorpora su propia API.

Primera versión de la plataforma que soporta procesadores multinúcleo. Lamáquina virtual Dalvik ha sido optimizada para permitir multiprocesado, lo que permite una ejecución más rápida de las aplicaciones, incluso aquellas que son de hilo único.

Se incorporan varias mejoras multimedia, como listas de reproducción M3U a través de HTTP Live Streaming, soporte a la protección de derechos musicales(DRM) y soporte para la transferencia de archivos multimedia a través de USB con los protocolos MTP y PTP.

En esta versión se añaden nuevas alternativas de conectividad, como las nuevas APIS de Bluetooth A2DP y HSP con streaming de audio. También se permite conectar teclados completos por USB o Bluetooth.

El uso de los dispositivos en un entorno empresarial es mejorado. Entre las novedades introducidas destacamos las nuevas políticas administrativas con encriptación del almacenamiento, caducidad de contraseña y mejoras para administrar los dispositivos de empresa de forma eficaz.

A pesar de la nueva interfaz gráfica optimizada para tabletas, Android 3.0 es compatible con las aplicaciones creadas para versiones anteriores. La tecla de menú, inexistente en las nuevas tabletas, es remplazada por un menú que aparece en la barra de acción.

- **Android 3.1 Nivel de API 12 (mayo 2011)**

Permite manejar dispositivos conectados por USB (tanto host como dispositivo).Protocolo de transferencia de fotos y vídeo (PTP/MTP) y de tiempo real (RTP).

- **Android 3.2 Nivel de API 13 (julio 2011)**

Optimizaciones para distintos tipos de tableta. Zoom compatible para aplicaciones de tamaño fijo. Sincronización multimedia desde SD.

- **Android 4.0 Nivel de API 14 (octubre 2011, Ice Cream Sandwich)**

La característica más importante es que se unifican las dos versiones anteriores (2.x para teléfonos y 3.x para tabletas) en una sola compatible con cualquier tipo de dispositivo. Entre las características más interesantes destacamos: se introduce un nuevo interfaz de usuario totalmente renovado. Por ejemplo, se reemplazan los botones físicos por botones en pantalla (como ocurría en las versiones 3.x).



Ilustración 30: Android 4.0 Ice Cream Sandwich

Nuevo API de reconocedor facial que permite, entre otras muchas aplicaciones, desbloquear el teléfono. También se mejora en el reconocimiento de voz. Por ejemplo, se puede empezar a hablar en cuanto pulsamos el botón.

Aparece un nuevo gestor de tráfico de datos por internet, donde podremos ver el consumo de forma gráfica y donde podemos definir los límites a ese consumo para evitar cargos inesperados con la operadora. Incorpora herramientas para la edición de imágenes en tiempo real, con herramientas para distorsionar, manipular e interactuar con la imagen al momento de ser capturada. Se mejora el API para comunicaciones por Near Field Communication (NFC) y la integración con redes sociales.

En diciembre del 2011 aparece una actualización de mantenimiento (versión 4.0.2) que no aumenta el nivel de API.

- **Android 4.0.3 Nivel de API 15 (diciembre 2011)**

Se introducen ligeras mejoras en algunas APIs incluyendo el de redes sociales, calendario, revisor ortográfico, texto a voz y bases de datos entre otros. En marzo de 2012 aparece la actualización 4.0.4.

- **Android 4.1 Nivel de API 16-18 (julio 2012, Jelly Bean)**



Ilustración 31: Android 4.1 Jelly Bean

Android Jelly Bean hereda las características desarrolladas en antiguas versiones de Android.

Incorporó una serie de nuevas funciones entre las que destacan el soporte de Bluetooth de baja energía de OpenGL ES 3.0 y de resolución 4K. Por primera vez en Android se incluyen los idiomas hebreo, árabe, africano, indio y zulú. Posee un sistema de localización Wi-Fi en segundo plano, mejoras en la seguridad y en la escritura, así como modo de perfiles con acceso restringido.

Además se incluye la función de autocompletado en el Dial pad para facilitar el marcado de números, Google Now como asistente personal que proporciona información contextualizada, y la posibilidad, desde un Nexus 7 y un Nexus 10 de ver el contenido en una televisión.

En cuanto a los desarrolladores, se han incorporado nuevas opciones para creadores de aplicaciones, sistema de logging, mejoras en el análisis de compilación de aplicaciones y APIs DRM de mayor calidad. Mejoras en la seguridad y encriptación Wi-Fi, mejoras en los codificadores de vídeo e inclusión de códec VP8, monitorización de notificaciones.

- **Android 4.4 Nivel de API 19-20 (octubre 2013, KitKat)**



Ilustración 32: Android 4.4 KitKat

Esta versión trae consigo muchas novedades como tanto en diseño, como en funcionalidad y rendimiento. Además, destaca por funcionar con sólo 512MB de RAM y reducir el gasto de batería haciéndolo una buena opción. Desde su lanzamiento, se han lanzado cuatro nuevas actualizaciones: "4.4.1", "4.4.2", "4.4.3", "4.4.4".

- **Android 5.0 Nivel de API 21-22 (noviembre 2014, Lollipop)**

Los cambios más prominentes en Lollipop incluyen una interfaz de usuario rediseñada construida sobre un diseño de lenguaje responsivo denominado como "Material Design", así como mejoras en el sistema de notificaciones que permiten que este sea accedido desde la pantalla de bloqueo, y mostrado junto con otras aplicaciones como banners en la parte superior de la pantalla. También se hicieron cambios internos, al usar ART como el tiempo de ejecución por defecto, y cambios cuyo objetivo es mejorar y optimizar el consumo de batería que va de la mano con el Project Volta, el cual busca dar un mejor rendimiento a la batería y aumentar su uso en hasta un 90%.



Android 5.0, Lollipop

Ilustración 33: Android 5.0 Lollipop

- **Android 6.0 Nivel de API 23 (octubre 2015, Marshmallow)**



Ilustración 34: Android 6.0 Marshmallow

Android 6.0 Marshmallow introduce un modelo de permisos rediseñado: ahora hay únicamente ocho categorías de permisos, y las aplicaciones ya no conceden automáticamente todos sus permisos específicos en el momento de la instalación. Se utiliza ahora un sistema "opt-in", en el que los usuarios pueden conceder o denegar permisos individuales a una aplicación cuando lo

requieran, tales como el acceso a la cámara o el micrófono. Las aplicaciones recuerdan las concesiones de permisos de cada usuario, y pueden ser ajustados en cualquier momento. Sólo las aplicaciones compiladas para Android 6.0 Marshmallow con su SDK utilizarán el nuevo sistema de permisos, mientras que todas las demás aplicaciones continuarán utilizando el modelo de permisos anterior.

Android Marshmallow ofrece soporte nativo para el reconocimiento de huellas digitales, lo que permite el uso de las mismas para desbloquear los dispositivos, y para la autenticación de Play Store o Android Pay; una API estándar también está disponible para la implementación de la autenticación basada en huellas digitales en otras aplicaciones. Un nuevo sistema de administración de energía conocida como 'doze' reduce las actividades en fondo del dispositivo cuando el sistema operativo detecta que el dispositivo no está siendo tocado físicamente. Android 6.0 Marshmallow es compatible con USB Tipo-C, lo que implica la capacidad de carga a través de USB, hasta cinco veces más rápida, dependiendo del terminal. También introduce "enlaces verificados" que se pueden configurar para abrir directamente una aplicación especificada, sin la necesidad de mostrar mensajes de usuario.

- **Android 7.0 Nivel de API 24-25 (agosto 2016, Nougat)**

Android Nougat incluye una característica llamada multiventana, que permite usar dos aplicaciones a la vez con la pantalla dividida. Fabricantes como Samsung implementó la multiventana desde el Galaxy S3. Pero ahora es compatible con cualquier dispositivo Android.

Esta versión de Android soporta de forma nativa la realidad virtual. Aunque la realidad virtual fue implementada en Smartphones como el Galaxy S6 y S7. Ahora es compatible de forma nativa, es decir, con cualquier dispositivo Android.



Ilustración 35: Android 7.0 Nougat

Incluye DayDrema que es un nuevo Play Store pero dedicado a la realidad virtual. Desde aquí podremos descargar Juegos o Apps destinados específicamente para la realidad virtual. Actualmente existen pocos dispositivos compatibles: (Google) Pixel, Pixel XL, todas las variantes de Motorola Moto Z (excepto moto Z play) y recientemente ZTE Axon7.

Doze on the go es una mejora de Doze (introducido por primera vez en Android Marshmallow). Ahora Doze no solo se activa cuando el dispositivo está encima de la mesa, ahora también entra en acción a la hora de apagar la pantalla.

- **Android 8.0 Nivel de API 26 (marzo 2017, Android O)**



Ilustración 36: Android 8.0 Android O

Ahora, el sistema limitará más los procesos en segundo plano, ahorrando por lo consiguiente mucha batería.

Se han mejorado las notificaciones en dos aspectos:

- 1) Ahora las notificaciones se podrán ordenar según categorías (noticias, tecnología...) y se podrá definir la frecuencia con que se reciben.
- 2) También al deslizar la notificación hacia la izquierda aparecerá, a parte de los ajustes para controlar las notificaciones de la aplicación específica, un reloj con el que podremos definir un periodo de tiempo en el que la notificación volverá a aparecer.

Ahora junto a los iconos de las aplicaciones aparecerá un número (función propia de Apple con IOS) dando a entender las tareas o notificaciones pendientes o sin leer.

En cuanto al diseño se ha mejorado:

- 1) La aplicación "Configuración" cuenta con un nuevo diseño, con un tema blanco y una categorización más profunda de los diferentes ajustes.

- 2) Nuevas guías de diseño para los iconos para unificar los diseños. El icono constará de dos capas, la primera (circular) será el icono de la aplicación, la segunda (cuadrada) será un fondo. Estas dos capas supuestamente interactúan mediante animaciones a la hora de abrirlas, cerrarlas o moverlas.
- 3) Android TV incluye un nuevo launcher.
- 4) Android "O" contiene soporte integrado para modos picture-in-picture.
- 5) La adición de un sonido de timbre personalizado, alarma o notificación se simplifica.

Android "O" agregará soporte para redes inalámbricas LAN para Wi-Fi basadas en Wi-Fi Aware¹⁶, gamas de color amplias en aplicaciones, una API para autofills¹⁷, multiprocesos y soporte de navegación segura de Google para WebViews, una API que permite al sistema aplicaciones de Voice Over IP (VoIP¹⁸) o voz sobre IP y actividades de lanzamiento en pantallas remotas. ART ofrece mejoras de rendimiento. Contiene límites adicionales en las actividades de fondo de las aplicaciones para mejorar la duración de la batería. Las aplicaciones pueden especificar "iconos adaptables" para contenedores de formas diferentes especificados por temas, como círculos, cuadrados y ardillas.

Android "O" admite nuevos amojis (emoticonos) que se incluirán en el estándar Unicode 10. Una nueva fuente de emoji también fue introducida, que notablemente rediseña sus figuras de cara para usar una forma circular tradicional, a diferencia del diseño "blob" que fue introducido en Android KitKat.

La arquitectura subyacente de Android se está revisando para que el código de bajo nivel específico del proveedor para soportar el hardware de un dispositivo se separe del sistema operativo Android utilizando una capa de abstracción de hardware conocida como "interfaz de proveedor". Se requerirá que las interfaces de proveedores sean compatibles con futuras versiones de Android; debido a estos cambios, los fabricantes de equipos originales solo necesitarán realizar las modificaciones necesarias en el marco del sistema operativo y en las aplicaciones agrupadas para actualizar un dispositivo a una futura versión de Android, manteniendo la misma interfaz de proveedor.

¹⁶ Wifi Aware: permitirá que dispositivos Wi-Fi cercanos se comuniquen entre sí sin intervención humana, y sin necesidad de conectarse a un punto de acceso ni a Internet. Véase en: <http://computerhoy.com/noticias/internet/wi-fi-aware-nuevo-wi-fi-internet-31615>

¹⁷ Autofills: llenado automático de datos para navegadores. Véase en: <https://play.google.com/store/apps/details?id=com.fillr&hl=es>

¹⁸ VoIP: conjunto de recursos que hacen posible que la señal de voz viaje a través de Internet empleando el protocolo IP. Véase en: https://es.wikipedia.org/wiki/Voz_sobre_protocolo_de_internet

El sistema operativo ofrecerá una distribución adaptada para dispositivos de gama baja conocidos como Android Go, que se utilizará en todos los dispositivos con 1GB de RAM o menos. Estos dispositivos se suministrarán con optimizaciones de la plataforma diseñadas para reducir el uso de datos móviles (incluida la habilitación del modo ahorro de datos de forma predeterminada) y un conjunto especial de servicios móviles de Google diseñados para reducir el uso de recursos y ancho de banda. Google Play Store también destacará aplicaciones ligeras adecuadas para estos dispositivos.

En el siguiente gráfico se refleja la cuota de mercado de cada una de las versiones anteriormente comentadas hasta enero de 2017. Para la obtención estos resultados se ha tenido en cuenta el número de actualizaciones realizadas y el número de dispositivos que se introducen en el mercado con cada versión.

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	1.0%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	1.1%
4.1.x	Jelly Bean	16	4.0%
4.2.x		17	5.9%
4.3		18	1.7%
4.4	KitKat	19	22.6%
5.0	Lollipop	21	10.1%
5.1		22	23.3%
6.0	Marshmallow	23	29.6%
7.0	Nougat	24	0.5%
7.1		25	0.2%

Ilustración 37: Cuota de mercado versiones Android

3.2.4 Componentes de una aplicación Android

Existen una serie de elementos clave que resultan imprescindibles para desarrollar aplicaciones en Android. En este apartado se va a realizar una descripción inicial de algunos de los más importantes.

3.2.4.1 Vista (View)

Las vistas son los elementos que componen la interfaz de usuario de una aplicación. Son por ejemplo, un botón, una entrada de texto,... Todas las vistas van a ser objetos descendientes de la clase View, y por tanto, pueden ser definidos utilizando código Java. Sin embargo, lo habitual va a ser definir las vistas utilizando un fichero XML y dejar que el sistema cree los objetos por nosotros a partir de este fichero. Esta forma de trabajar es muy similar a la definición de una página web utilizando código HTML.

3.2.4.2 Layout

Un Layout es un conjunto de vistas agrupadas de una determinada forma. Vamos a disponer de diferentes tipos de Layouts para organizar las vistas de forma lineal, en cuadrícula o indicando la posición absoluta de cada vista. Los Layouts también son objetos descendientes de la clase View. Igual que las vistas, los Layouts pueden ser definidos en código, aunque la forma habitual de definirlos es utilizando código XML.

3.2.4.3 Actividad (Activity)

Una aplicación en Android va a estar formada por un conjunto de elementos básicos de visualización, coloquialmente conocidos como pantallas de la aplicación. En Android cada uno de estos elementos, o pantallas, se conoce como actividad. Su función principal es la creación de la interfaz de usuario. Una aplicación suele necesitar varias actividades para crear la interfaz de usuario. Las diferentes actividades creadas serán independientes entre sí, aunque todas trabajarán para un objetivo común. Toda actividad ha de pertenecer a una clase descendiente de Activity.

3.2.4.4 Servicio (Service)

Un servicio es un proceso que se ejecuta “detrás”, sin la necesidad de una interacción con el usuario. Es algo parecido a un demonio en Unix o a un servicio en Windows. En Android disponemos de dos tipos de servicios: servicios locales, que pueden ser utilizados por aplicaciones del mismo terminal y servicios remotos, que pueden ser utilizados desde otros terminales.

3.2.4.5 Intención (Intent)

Una intención representa la voluntad de realizar alguna acción; como realizar una llamada de teléfono, visualizar una página web. Se utiliza cada vez que queramos:

- Lanzar una actividad.
- Lanzar un servicio.

- Lanzar un anuncio de tipo broadcast.
- Comunicarnos con un servicio.

Los componentes lanzados pueden ser internos o externos a nuestra aplicación. También utilizaremos las intenciones para el intercambio de información entre estos componentes.

En muchas ocasiones una intención no será inicializada por la aplicación, si no por el sistema, por ejemplo, cuando pedimos visualizar una página web. En otras ocasiones será necesario que la aplicación inicialice su propia intención. Para ello se creará un objeto de la clase Intent.

3.2.4.6 Receptor de anuncios (Broadcast receiver)

Un receptor de anuncios recibe y reacciona ante anuncios de tipo broadcast. Existen muchos originados por el sistema, como por ejemplo batería baja, llamada entrante,... Aunque las aplicaciones también puede lanzar un anuncio broadcast. No tiene interfaz de usuario, aunque pueden iniciar una actividad para atender a un anuncio.

3.2.4.7 Proveedores de contenido (Content Provider)

La compartición de información entre teléfonos móviles resulta un tema vital. Android define un mecanismo estándar para que las aplicaciones puedan compartir datos sin necesidad de comprometer la seguridad del sistema de ficheros. Con este mecanismo podremos acceder a datos de otras aplicaciones, como la lista de contactos, o proporcionar datos a otras aplicaciones.

Bibliografía [\[12\]](#) [\[13\]](#)

3.2.5 Ciclo de vida de una aplicación Android

El ciclo de vida de una aplicación Android es bastante diferente al ciclo de vida de una aplicación en otros S.O., como Windows. La mayor diferencia es que, en Android el ciclo de vida es controlado principalmente por el sistema, en lugar de ser controlado directamente por el usuario.

Una aplicación en Android está formada por un conjunto de elementos básicos de interacción con el usuario, conocidos como actividades. Además de varias actividades, una aplicación también puede contener servicios. Son las actividades las que realmente controlan el ciclo de vida de las aplicaciones, dado que el usuario no cambia de aplicación, sino de actividad. El sistema mantiene una pila con las actividades previamente visualizadas, de forma que el usuario puede regresar a la actividad anterior pulsando la tecla “retorno”.

Una aplicación Android corre dentro de su propio proceso Linux. Este proceso se crea con la aplicación y continuará vivo hasta que ya no sea requerido y el sistema reclame su memoria para asignársela a otra aplicación.

Una característica importante, y poco usual, de Android es que la destrucción de un proceso no es controlada directamente por la aplicación, sino que es el sistema el que determina cuándo destruir el proceso. Lo hace basándose en el conocimiento que tiene de las partes de la aplicación que están corriendo (actividades y servicios), en la importancia de dichas partes para el usuario y en cuánta memoria disponible hay en un determinado momento.

Si tras eliminar el proceso de una aplicación, el usuario vuelve a ella, se crea de nuevo el proceso, pero se habrá perdido el estado que tenía esa aplicación. En estos casos, será responsabilidad del programador almacenar el estado de las actividades, si queremos que cuando sean reiniciadas conserven su estado.

Como vemos, Android es sensible al ciclo de vida de una actividad; por lo tanto, se necesita comprender y manejar los eventos relacionados con el ciclo de vida si se quiere crear aplicaciones estables.

Una actividad en Android puede estar en uno de estos cuatro estados:

Activa (*Running*): la actividad está encima 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. El programador debe guardar el estado de la interfaz de usuario, preferencias, etc.

Destruída (*Destroyed*): cuando la actividad termina al invocarse el método *finish()*, o es eliminada por el sistema.

Cada vez que una actividad cambia de estado se van a generar eventos que podrán ser capturados por ciertos métodos de la actividad. A continuación se muestra un esquema que ilustra los métodos que capturan estos eventos.

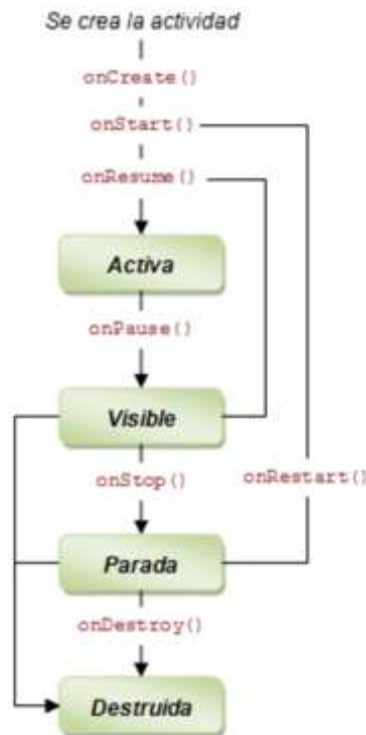


Ilustración 38: Ciclo de vida de una actividad Android

onCreate(Bundle): se llama en la creación de la actividad. Se utiliza para realizar todo tipo de inicializaciones, como la creación de la interfaz de usuario o la inicialización de estructuras de datos. Puede recibir información de estado de la actividad (en una instancia de la clase Bundle), por si se reanuda desde una actividad que ha sido destruida y vuelta a crear.

onStart(): nos indica que la actividad está a punto de ser mostrada al usuario.

onResume(): se llama cuando la actividad va a comenzar a interactuar con el usuario. Es un buen lugar para lanzar las animaciones y música.

onPause(): indica que la actividad está a punto de ser lanzada a segundo plano, normalmente porque otra actividad es lanzada. Es el lugar adecuado para detener animaciones, música o almacenar los datos que estaban en edición.

onStop(): la actividad ya no va a ser visible para el usuario. Cuidado, si hay muy poca memoria, es posible que la actividad se destruya sin llamar a este método.

onRestart(): indica que la actividad va a volver a ser representada después de haber pasado por onStop().

onDestroy(): se llama antes de que la actividad sea totalmente destruida. Por ejemplo, cuando el usuario pulsa el botón de volver o cuando se llama al método *finish()*. Cuidado, si hay muy poca memoria, es posible que la actividad se destruya sin llamar a este método. Bibliografía [\[14\]](#)

3.2.6 Android Studio

Todos los elementos necesarios para el desarrollo de aplicaciones para Android están disponibles gratuitamente en internet para todas las plataformas. El IDE elegido para este trabajo es Android Studio. Está basado en IntelliJ IDEA de la compañía JetBrains, que proporciona varias mejoras con respecto al plugin ADT (Android Developer Tools) para Eclipse. Android Studio utiliza una licencia de software libre Apache 2.0, está programado en Java y es multiplataforma.

A continuación se muestra una tabla con los requerimientos del sistema en diferentes SO.

Windows	macOS	Linux
Windows Vista/7/8/10	Mac OS X 10.8.5 o superior	GNOME o entorno de escritorio KDE
Mínimo 2GB de memoria RAM, recomendado 4GB		
2GB (se recomienda más) de espacio en el disco duro		
Resolución mínima de pantalla de 1280 x 800		
Java Development Kit 7 o superior		

Tabla 7: Requerimientos de Android Studio

Una vez todo esté instalado y configurado, se podrá comenzar a desarrollar cualquier aplicación para Android. A continuación se muestra un ejemplo del entorno dónde se va a trabajar durante la realización de este trabajo, en este caso se muestran dos capturas de pantalla en la plataforma macOS Sierra donde se ha llevado a cabo el desarrollo de la aplicación.

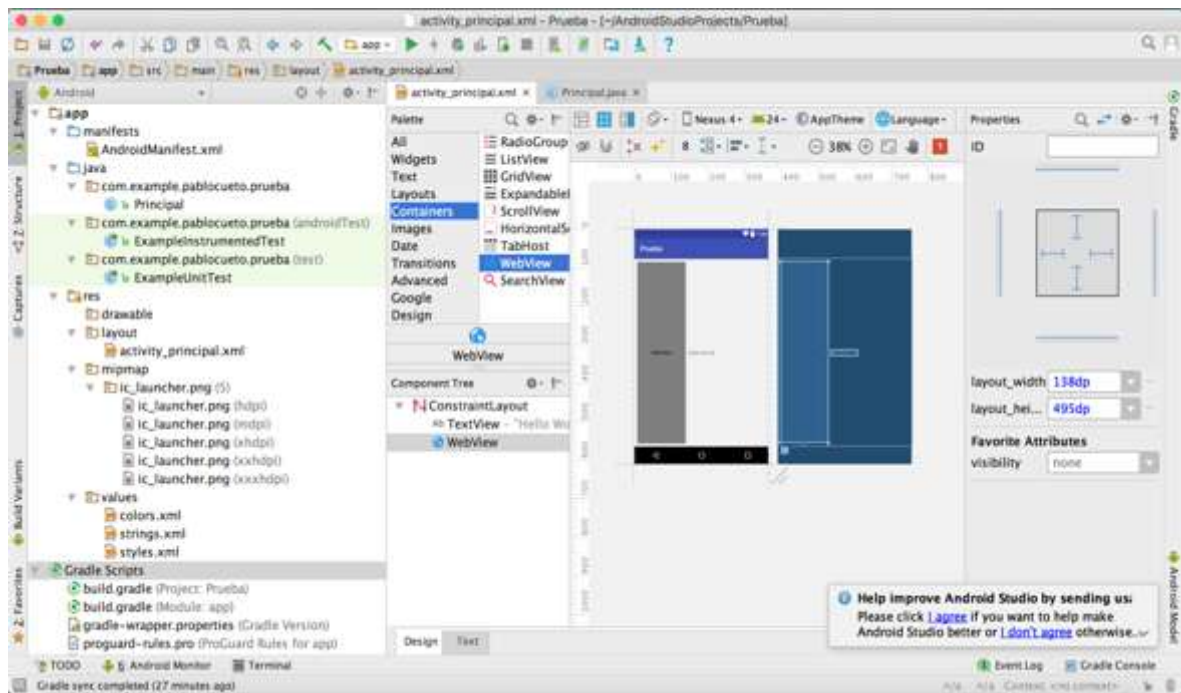


Ilustración 39: Entorno usado

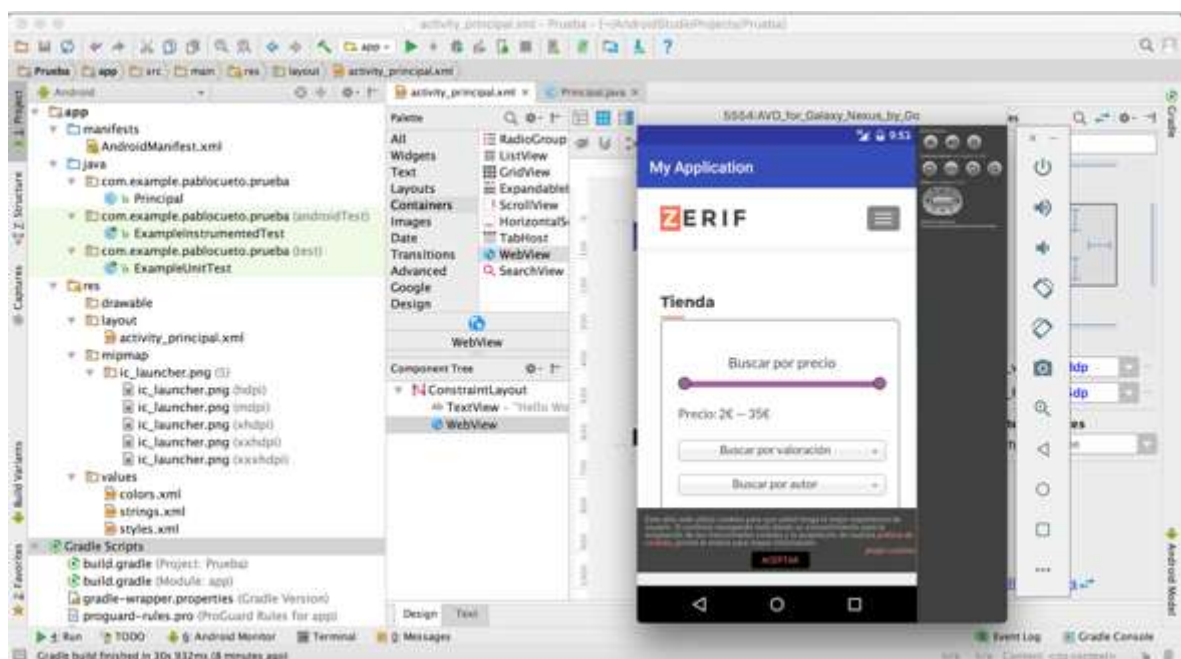


Ilustración 40: Entorno usado 2

Descarga gratuita de Android Studio: <https://developer.android.com/studio/index.html>

4 ANÁLISIS Y DISEÑO

Después de haber analizado los conceptos indicados a lo largo de este trabajo incluyendo el Anexo 1, se va a pasar a comentar la fase de desarrollo del trabajo abordando su análisis y diseño.

En primer lugar, en el análisis se estudiarán los requisitos de usuario de las aplicaciones así como los diferentes casos de uso que se pueden dar. Con ello se pasará a la fase de diseño del trabajo entrando en la parte de la arquitectura del mismo.

4.1 Análisis

En esta sección se incluirán los requisitos de usuario así como los casos de uso de los mismos.

4.1.1 Planteamiento del problema

A continuación se describirán diferentes ejemplos de situaciones en las que sea interesante desarrollar aplicaciones para la compra/venta de productos.

- **Entorno empresarial:** Empresas de todo tipo que quieran dar la oportunidad de comprar y vender productos o servicios a usuarios, no solo en tiendas físicas o páginas web, sino en cualquier parte gracias a los dispositivos móviles. Se desea permitir al usuario poder realizar estas acciones sin necesidad de desplazarse a la tienda física y sin necesidad de un ordenador. Se permitirá comprar y vender sin necesidad de conexión Wi-Fi gracias a los datos móviles, en cualquier hora del día y en cualquier parte en la que se disponga de cobertura a dichos datos.
- **Aplicaciones comerciales:** En empresas pequeñas y medianas que estén interesadas en aumentar el número de ventas y poder llegar a todas ciudades del país o del mundo donde no se encuentre una tienda física. Incluso es una gran oportunidad de hacer publicidad de sus productos dando la opción a usuarios de valorar y comentar prestaciones y características de los productos que adquieren para que futuros usuarios puedan comparar opiniones y decidir que acción llevar a cabo.

4.1.2 Planteamiento de la solución

Dado el planteamiento del problema, se debe buscar la solución más económica y eficiente que se adapte a las necesidades para lograr los objetivos e intentar hacerlo con la mayor sencillez posible.

Dado que hoy día la mayoría de la población cuenta con acceso a internet en sus casas o ciudades y debido al alto porcentaje de personas que cuenta con un Smartphone

o Tablet y a las múltiples herramientas de las que se dispone para desarrollar aplicaciones para ellos, se ha decidido usar un CMS y una plataforma móvil para el desarrollo de unas aplicaciones que, como se expuso anteriormente, permita a usuarios comprar y vender productos y servicios desde cualquier parte mientras exista la posibilidad de conectarse a internet.

Para esta compra/venta de productos y servicios, se ha optado por almacenar en una base de datos todo lo necesario para el funcionamiento de las aplicaciones, como los datos de usuarios para llevar el control de registro y así tener más seguridad, y características de productos, los cuales, son fundamentales para los usuarios que estén interesados en ellos.

Una vez elegida la plataforma y el método por el que se almacena y se solicita la información, solo queda desarrollar las aplicaciones y conseguir que sean robustas y sin errores.

4.1.3 Requisitos y marco regulatorio

Para el desarrollo de las aplicaciones se tiene que tener en cuenta si existe algún tipo de restricción que inflija algún problema, así como alguna normativa legal.

4.1.3.1 Requisitos

Los requisitos de las aplicaciones vienen principalmente determinados por las limitaciones de hardware y software de los dispositivos.

A nivel de hardware, para la realización de la aplicación web, se encuentra el inconveniente de no contar con un ordenador capaz de lanzar el CMS elegido necesario para el desarrollo de aplicaciones. Por ello mismo se recomienda ver los requisitos hardware que necesitan los diferentes software y utilizar el mejor dispositivo disponible capaz de lanzarlo o usar un CMS diferente para el desarrollo. Para la realización de la aplicación Android, se encuentra el inconveniente de contar con dispositivos de diferentes tamaños y las dos orientaciones de pantalla disponibles (vertical y horizontal). Esto lo soluciona el desarrollador programando una aplicación capaz de adaptarse a los diferentes tamaños de los dispositivos y a las diferentes orientaciones, ya sean Smartphones o Tablets.

A nivel software, para la realización de la página web, se puede encontrar con el problema de los servidores donde se aloja la información. Como se explicó en el apartado 3.1.2.1 de este trabajo, se recomienda usar un servidor local si se es un desarrollador principiante. Este tipo de servidor puede dar bastantes problemas a la hora de su instalación y configuración por lo que se recomienda, en estos casos, la utilización de un

servidor público gratuito y hacer las pruebas necesarias en él. Luego se tendrá la opción de poder comprar un mejor alojamiento para lanzar la aplicación operativa al 100%. En cuanto a los problemas que nos podemos encontrar para el desarrollo de la aplicación Android, estos vienen dados por la compatibilidad con las distintas versiones existentes de Android comentadas en el apartado 3.2.3 de este trabajo y a los permisos necesarios para el funcionamiento de la aplicación en cada versión (como el acceso a la cámara o galería). Si desarrollamos la aplicación para las últimas versiones lanzadas solo serán compatibles con los últimos dispositivos lanzados al mercado y no se dará la oportunidad a usuarios que tengan dispositivos más antiguos y que no soportan versiones más nuevas, poder acceder a la aplicación. Por esto, el desarrollador tendrá que programar la aplicación para que sea compatible con todos los dispositivos Android del mercado. En cuanto a los permisos, será necesario, por parte del desarrollador, dar los permisos necesarios a todas las versiones y no solo a las actuales.

4.1.3.2 *Marco regulatorio*

Ya que este trabajo es de tipo teórico/experimental no es necesario implementar toda la legislación necesaria para el desarrollo de las aplicaciones; pero sí es de gran interés intentar adaptar la solución lo más próxima posible a un resultado real por si en un futuro se decide ponerlas en funcionamiento.

Es de gran importancia conocer la legislación vigente en los países en los que las aplicaciones vayan a ser lanzadas con el objetivo de evitar problemas de violación de leyes. Por motivos de desconocimiento de las leyes de otros países, las aplicaciones, inicialmente, solo serán lanzadas en España y se deberá seguir la normativa vigente (normativa vigente hasta mediados de 2017).

Toda esta información está recogida en el Anexo 2 de este trabajo.

[\[Anexo 2\]](#)

4.1.4 *Requisitos de usuario*

A continuación se especificarán los requisitos de usuario dividiéndolos en dos categorías, requisitos de usuario capacitivos y requisitos de usuario restrictivos. Estos requisitos se especifican a continuación siguiendo un formato tabular en el cual se incluye el siguiente contenido para cada uno de ellos:

- Identificador: identifica de manera unívoca a un requisito. Deberá seguir la siguiente nomenclatura: Requisito-CapacitivoXX para los requisitos capacitivos y Requisito-RestictivoXX para los restrictivos, donde XX corresponde al número de requisito.

- Descripción: especificación detallada del requisito.
- Necesidad: establece la relevancia del requisito para el trabajo. Tomará valores entre 1 y 5, siendo 5 la necesidad más alta y 1 la más baja.
- Prioridad: establece la prioridad del requisito dentro del trabajo. Tomará valores entre 1 y 5 siendo 5 la prioridad más alta y 1 la más baja.
- Estabilidad: especifica la sensibilidad del requisito a ser modificado, tomando valores entre 'estable' y 'no estable'.

4.1.4.1 Requisitos de usuario de capacidad

A continuación se muestran los requisitos de capacidad que deben cumplir las aplicaciones. Dado que se cuenta con dos aplicaciones que ofrecen los mismos servicios, siempre que se hable de requisitos se dará por entendido que se habla de ambas.

Identificador	Requisito-Capacitivo01. Conexión a la página web y aplicación Android
Descripción	El usuario debe ser capaz de poder conectarse a través de internet a la tienda independientemente de si se hace a través de la página web o la aplicación Android.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 8: Requisitos de usuario de capacidad 1

Identificador	Requisito-Capacitivo02. Publicar noticias en el Blog
Descripción	Solo los usuarios que se postulen como administradores podrán publicar noticias en el Blog.
Necesidad	1
Prioridad	1
Estabilidad	Estable

Tabla 9: Requisitos de usuario de capacidad 2

Identificador	Requisito-Capacitivo03. Registro de usuarios nuevos
Descripción	Los usuarios que lo deseen puede registrarse dado un nombre de usuario y su cuenta de correo electrónico.
Necesidad	5
Prioridad	5

Estabilidad	Estable
--------------------	---------

Tabla 10: Requisitos de usuario de capacidad 3

Identificador	Requisito-Capacitivo04. Postulación como comprador y vendedor
Descripción	Los usuarios que se registren en la página serán postulados como compradores y vendedores por defecto.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 11: Requisitos de usuario de capacidad 4

Identificador	Requisito-Capacitivo05. Publicar productos en la tienda
Descripción	Todos los usuarios registrados podrán publicar los productos que deseen, dando la opción de poner la documentación que el usuario desee.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 12: Requisitos de usuario de capacidad 5

Identificador	Requisito-Capacitivo06. Contactar con el dueño de la tienda
Descripción	Todos los usuarios que visiten la tienda online o que tengan instalada la aplicación, podrán ponerse en contacto con el dueño de la tienda sin necesidad de estar registrado y con una cuenta de correo electrónico.
Necesidad	1
Prioridad	1
Estabilidad	Estable

Tabla 13: Requisitos de usuario de capacidad 6

Identificador	Requisito-Capacitivo07. Categorías
Descripción	Solo los usuarios que estén postulados como administradores podrán crear o eliminar categorías de productos.
Necesidad	1
Prioridad	1
Estabilidad	Estable

Tabla 14: Requisitos de usuario de capacidad 7

Identificador	Requisito-Capacitivo08. Lista de deseos
Descripción	Todos los usuarios que visiten la tienda online o que tengan instalada la aplicación, podrán añadir productos a la lista de deseos.
Necesidad	1
Prioridad	1
Estabilidad	Estable

Tabla 15: Requisitos de usuario de capacidad 8

Identificador	Requisito-Capacitivo09. Cuenta
Descripción	Todos los usuarios registrados tendrán acceso a la página de Mi Cuenta, donde se podrán modificar los datos personales y direcciones de facturación y envío, y se podrán comprobar los pedidos realizados.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 16: Requisitos de usuario de capacidad 9

Identificador	Requisito-Capacitivo10. Cambio de contraseña
Descripción	Todos los usuarios registrados podrán cambiar la contraseña dentro de la página Mi Cuenta.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 17: Requisitos de usuario de capacidad 10

Identificador	Requisito-Capacitivo11. Recuperar clave de usuario
Descripción	En caso de que un usuario olvide su clave de usuario para acceder a los servicios, podrá cambiarla con su dirección de correo.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 18: Requisitos de usuario de capacidad 11

Identificador	Requisito-Capacitivo12. Escritorio del vendedor
Descripción	Todos los usuarios registrados podrán acceder a su Escritorio del vendedor donde podrán acceder a su tienda personal y podrán crear productos nuevos, modificar productos ya creados, dar información sobre su tienda y poner la cuenta de Paypal donde se ingresarán los costes de las ventas de sus artículos.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 19: Requisitos de usuario de capacidad 12

Identificador	Requisito-Capacitivo13. Escritorio del Vendedor (Productos)
Descripción	Cada usuario, dentro de la página Escritorio del Vendedor, solo podrá modificar los productos ofrecidos por dicho usuario.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 20: Requisitos de usuario de capacidad 13

Identificador	Requisito-Capacitivo14. Nuevo Producto
Descripción	En el Escritorio del Vendedor, en la pestaña Nuevo Producto, se podrán crear productos indicando las características del producto: añadir fotografías para facilitar su venta, crear pujas, definir el producto en las categorías correspondiente según sus características, definir las unidades que hay de ese producto, los impuestos aplicados al producto, si es nuevo o usado, etc.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 21: Requisitos de usuario de capacidad 14

Identificador	Requisito-Capacitivo15. Modificar productos
Descripción	En el Escritorio del Vendedor se podrán modificar los productos ofrecidos en la tienda sin importar el motivo de su modificación. Por ejemplo: precio elevado, error en la descripción, editar imágenes, etc.

Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 22: Requisitos de usuario de capacidad 15

Identificador	Requisito-Capacitivo16. Modificación del perfil
Descripción	Se podrá modificar el perfil en la sección Perfil, dentro de la página: Escritorio del Vendedor → Añadir o modificar productos → Perfil, en la parte izquierda. Como se comentó antes, en la página Mi Cuenta también se pueden modificar los datos del perfil del usuario.
Necesidad	1
Prioridad	1
Estabilidad	Estable

Tabla 23: Requisitos de usuario de capacidad 16

Identificador	Requisito-Capacitivo17. Desconexión de sesiones abiertas
Descripción	Todos los usuarios registrados que tengan abiertas sus cuentas en diferentes dispositivos podrán cerrar sus cuentas en dichos dispositivos para mantener la seguridad de sus cuentas.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 24: Requisitos de usuario de capacidad 17

4.1.4.2 Requisitos de usuario restrictivos

En las siguientes tablas se muestran los requisitos de restricción que se deben tener en cuenta a la hora del desarrollo de las aplicaciones. Dado que se cuenta con dos aplicaciones que ofrecen los mismos servicios, siempre que se hable de requisitos se dará por entendido que se habla de ambas.

Identificador	Requisito-Restictivo01. Plataforma Android
Descripción	Los dispositivos en los que la aplicación correrá deben tener una versión Android 3.0 o superior.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 25: Requisitos de usuario restrictivos 1

Identificador	Requisito-Restritivo02. Permisos en la aplicación móvil
Descripción	A la hora de instalar la aplicación en un Smartphone o Tablet se deberán conceder una serie de permisos que serán necesarios para su correcto funcionamiento en el terminal. Estos permisos serán: • Acceso a la cámara. • Acceso a los archivos internos del dispositivo.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 26: Requisitos de usuario restrictivos 2

Identificador	Requisito-Restritivo03. Conectividad
Descripción	Para poder utilizar todas las funcionalidades que ofrecen las aplicaciones es necesario tener una conexión a internet bien por Wi-Fi o datos móviles (3G, 4G, etc.)
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 27: Requisitos de usuario restrictivos 3

Identificador	Requisito-Restritivo04. Idioma de la Interfaz
Descripción	La interfaz mostrará todo su contenido por defecto en Español.
Necesidad	3
Prioridad	3
Estabilidad	Estable

Tabla 28: Requisitos de usuario restrictivos 4

Identificador	Requisito-Restritivo05. Cuenta de correo
Descripción	Par registrarse en la aplicación es indispensable que el usuario cuente con una cuenta de correo electrónico.
Necesidad	5
Prioridad	5
Estabilidad	Estable

Tabla 29: Requisitos de usuario restrictivos 5

4.1.5 Casos de uso

En este apartado se incluirán los diferentes casos de uso¹⁹ planteados para el desarrollo de las aplicaciones. Para ello se utilizará un diagrama en el que se presentará al usuario como actor del caso interaccionando con las diferentes actividades.

En nuestro caso se definen tres tipos de actores:

- Usuario no registrado: serán todos los usuarios que accedan a la tienda online o dispongan de la aplicación móvil.
- Usuario registrado: serán aquellos usuarios que se han registrado en el sistema creando una cuenta y además acceden a la aplicación mediante su correo electrónico o nombre de usuario y contraseña.
- Administrador: encargado de llevar el control de la tienda en ambas aplicaciones.

Además se incluirá una descripción de formato tabular en la que se describirán los pasos necesarios para los diferentes escenarios planteados conteniendo los siguientes pasos:

- Identificador: Identifica de manera unívoca a un caso de uso. Deberá seguir la nomenclatura CasoXX, donde XX corresponde al número de caso de uso.
- Actor: Agente externo que interactúa con el sistema participando en el caso de uso.
- Objetivo: Especificación breve del caso de uso.
- Pre-condición: Condiciones que deben darse para la realización del caso de uso.
- Escenario: Especificación detallada de los pasos que realiza el usuario dentro del escenario.
- Escenario Alternativo: Especificación detallada de los pasos que realiza el usuario cuando puede darse una bifurcación en la ejecución normal de las aplicaciones.
- Post condición: Condiciones que son el resultado de la ejecución del caso de uso.

¹⁹Casos de uso: técnica para especificar el comportamiento de un sistema. Es una secuencia de interacciones entre un sistema y alguien o algo que usa alguno de sus servicios. Véase en: http://www-2.dc.uba.ar/materias/isoft1/2001_2/apuntes/CasosDeUso.pdf

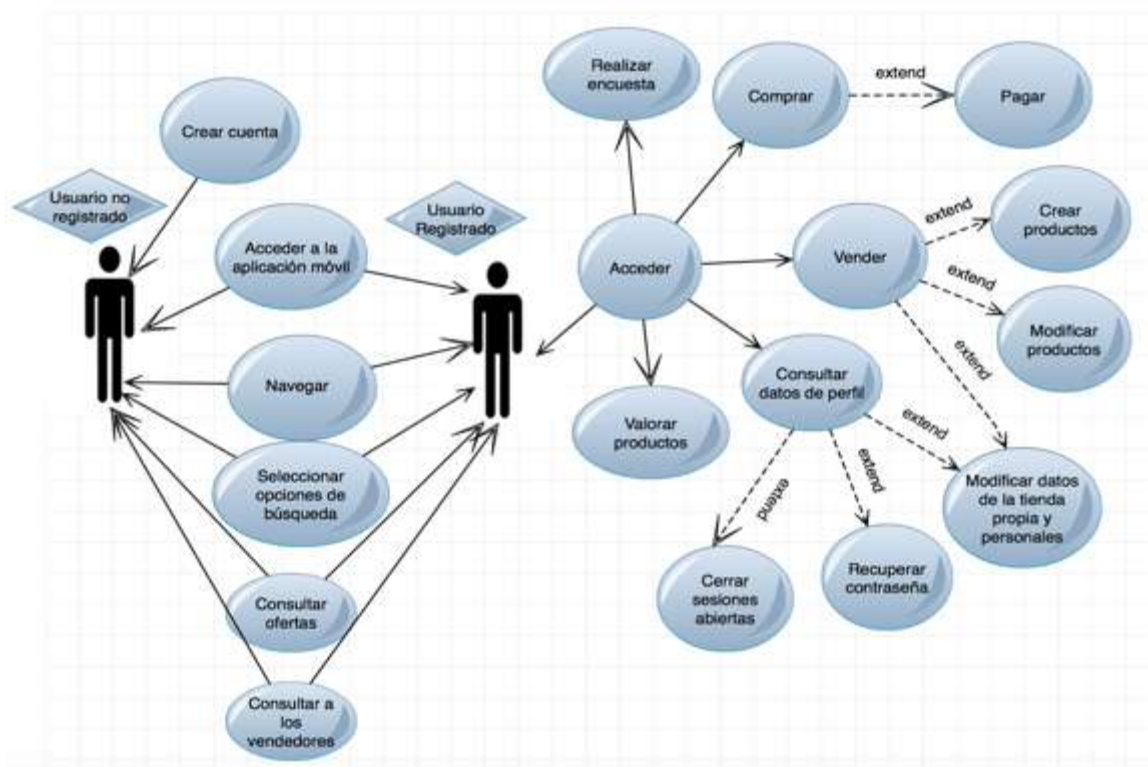


Ilustración 41: Casos de uso

Identificador	Caso01. Crear cuenta
Actor	Usuario no Registrado.
Objetivo	Darse de alta para acceder a los servicios dado un correo electrónico y una contraseña.
Pre-condición	- No debe haber una cuenta creada con el mismo correo electrónico. - El usuario debe tener la aplicación móvil instalada.
Escenario	- En el menú principal situado arriba a la derecha, pinchar en Registrarse. - Introducir cuenta propia de correo electrónico y nombre de usuario. - Al recibir un correo electrónico de la tienda, acceder a él y darle al enlace adjunto. - Introducir una nueva contraseña.
Escenario Alternativo	Con el plugin para registrarse a través de los datos de una de las redes sociales que se indiquen si está disponible.
Post-condición	Se creará la cuenta y se podrá acceder a los servicios proporcionados.

Tabla 30: Casos de uso 1

Identificador	Caso02. Acceder
Actor	Usuario registrado.
Objetivo	Acceder al perfil de usuario.
Pre-condición	Estar registrado.
Escenario	1. En el menú principal situado arriba a la derecha, pinchar en Acceder. 2. Introducir el nombre de usuario o correo electrónico y la contraseña.
Escenario Alternativo	Con el plugin para acceder a través de los datos de una de las redes sociales que se indiquen si está disponible.
Post-condición	Se pueden acceder a los servicios que se proporcionan.

Tabla 31: Casos de uso 2

Identificador	Caso03. Comprar (extend Acceder)
Actor	Usuario registrado.
Objetivo	Poder comprar productos.
Pre-condición	Estar registrado.
Escenario	1. En el menú principal situado arriba a la derecha, pinchar en Tienda o alguna de las categorías de la pestaña Categorías. 2. Pinchar sobre el producto que se quiera comprar. 3. Añadir al carrito. 4. Dentro de la página Carrito, darle a finalizar compra.
Escenario Alternativo	• En caso de subasta, esperar a recibir una notificación a través de un correo electrónico indicando que se ha ganado la subasta. • En caso de aceptarse ofertas por un producto, esperar a que el vendedor acepte la oferta.
Post-condición	El usuario habrá comprado un producto y recibirá la factura en su correo electrónico.

Tabla 32: Casos de uso 3

Identificador	Caso04. Pagar (extend Comprar)
Actor	Usuario registrado.

Objetivo	Pagar un producto a través de una pasarela de pago.
Pre-condición	Haber finalizado una compra.
Escenario	1. Después de finalizar la compra de uno o varios productos, elegir una de las pasarelas de pago. 2. Introducir los datos necesarios para efectuar el pago.
Escenario Alternativo	Ninguno.
Post-condición	Recibir el pedido.

Tabla 33: Casos de uso 4

Identificador	Caso05. Vender (extend Acceder)
Actor	Usuario registrado.
Objetivo	Vender un producto en la tienda y estar visible para todos los usuarios registrados y no registrados.
Pre-condición	Estar registrado.
Escenario	1. En la pestaña de Mi Cuenta del menú principal, acceder al Escritorio del Vendedor. 2. Al finalizar la creación de un nuevo producto estará disponible para su venta.
Escenario Alternativo	Ninguno.
Post-condición	Se añadirá el producto a la tienda y estará disponible para todos los usuarios.

Tabla 34: Casos de uso 5

Identificador	Caso06. Crear productos (extend Vender)
Actor	Usuario registrado.
Objetivo	Crear un producto para su venta.
Pre-condición	Estar registrado.
Escenario	1. En la pestaña de Mi Cuenta del menú principal, acceder al Escritorio del Vendedor. 2. Darle a la opción que se ofrece de nuevo producto. 3. Completar todos los campos necesarios para crear el producto como: nombre del producto, descripción del producto, tipo de producto, añadir una fotografía del

	producto, etc.
Escenario Alternativo	Ninguno.
Post-condición	Tener un producto en la tienda disponible para todos los usuarios.

Tabla 35: Casos de uso 6

Identificador	Caso07. Modificar productos (extend Vender)
Actor	Usuario registrado.
Objetivo	Modificar un producto añadido previamente a la tienda.
Pre-condición	Estar registrado y haber añadido un producto en la tienda.
Escenario	1. En la pestaña de Mi Cuenta del menú principal, acceder al Escritorio del Vendedor. 2. Darle a la opción que se ofrece de modificar producto. 3. Modificar todos los campos que se quieran del producto como: nombre del producto, descripción del producto, tipo de producto, añadir una fotografía del producto, etc.
Escenario Alternativo	Ninguno.
Post-condición	Producto modificado. Los usuarios podrán ver las nuevas características del producto.

Tabla 36: Casos de uso 7

Identificador	Caso08. Modificar datos de la tienda y personales (extend Vender y Consultar datos de perfil)
Actor	Usuario registrado.
Objetivo	Modificar los datos personales, las direcciones de facturación y envío, cambiar contraseña, modificar datos de la tienda, etc.
Pre-condición	Estar registrado.
Escenario	1. En la pestaña de Mi Cuenta del menú principal, acceder al Escritorio del Vendedor o página Mi Cuenta. 2. Entrar en los apartados direcciones, detalles de la cuenta o ajustes de la tienda. 3. Modificar los datos que se necesite y guardar cambios.
Escenario Alternativo	1. Pinchar en la pestaña de Nuevo producto dentro del Escritorio del vendedor. 2. En el menú de la derecha pinchar en la pestaña Perfil.

	3. Modificar los datos necesarios y guardar los cambios.
Post-condición	Modificación de los datos del usuario y su tienda personal.

Tabla 37: Casos de uso 8

Identificador	Caso09. Consultar datos del perfil (extend Acceder)
Actor	Usuario registrado.
Objetivo	Consultar los datos del perfil.
Pre-condición	Estar registrado.
Escenario	1. En la pestaña de Mi Cuenta del menú principal, acceder al Escritorio del Vendedor o página Mi Cuenta. 2. Entrar en los apartados direcciones, detalles de la cuenta o ajustes de la tienda. 3. Consultar los datos que se necesite.
Escenario Alternativo	1. Pinchar en la pestaña de Nuevo producto dentro del Escritorio del vendedor. 2. En el menú de la derecha pinchar en la pestaña Perfil. 3. Consultar los datos personales y de la tienda.
Post-condición	Comprobación de los datos del perfil.

Tabla 38: Casos de uso 9

Identificador	Caso10. Recuperar contraseña (extend Consultar datos de perfil)
Actor	Usuario registrado
Objetivo	Recuperar contraseña.
Pre-condición	Estar registrado.
Escenario	1. En el menú principal pinchar en Acceder. 2. Pinchar sobre contraseña perdida. 3. Introducir el nombre de usuario o correo electrónico de la cuenta creada con anterioridad. 4. En la bandeja de entrada del correo electrónico, abrir el correo y pinchar sobre el enlace adjunto. 5. Poner una clave nueva.
Escenario Alternativo	Ninguno
Post-condición	Obtener una nueva clave para poder acceder.

Tabla 39: Casos de uso 10

Identificador	Caso11. Cerrar sesiones abiertas (extend Consultar datos de perfil)
Actor	Usuario registrado
Objetivo	Cerrar sesiones abiertas en otros dispositivos u otros navegadores usados para acceder a la cuenta.
Pre-condición	Estar registrado y haber iniciado sesión en varios dispositivos.
Escenario	1. En la pestaña de Mi Cuenta del menú principal, acceder al Escritorio del Vendedor. 2. Darle a la opción que se ofrece de nuevo producto. 3. En el menú de la derecha acceder a la pestaña Perfil. 4. En la sección Sesiones, darle a desconectar en el resto de sitios.
Escenario Alternativo	Ninguno.
Post-condición	Sesión cerrada en otros dispositivos.

Tabla 40: Casos de uso 11

Identificador	Caso12. Acceder a la aplicación móvil
Actor	Usuario no registrado y registrado
Objetivo	Acceder a la aplicación móvil de la tienda.
Pre-condición	Descargar la aplicación y tenerla instalada.
Escenario	1. Descargar la aplicación de un correo electrónico. 2. Dar permisos para instalarla.
Escenario Alternativo	Ninguno
Post-condición	Poder acceder a la tienda a través de la aplicación móvil.

Tabla 41: Casos de uso 12

Identificador	Caso13. Navegar
Actor	Usuario no registrado y registrado
Objetivo	Navegar por la tienda.
Pre-condición	Poner la URL o tener instalada la aplicación.
Escenario	1. Introducir la siguiente URL en el navegador: https://idlashop.000webhostapp.com/ 2. Instalar la aplicación móvil en el dispositivo.

Escenario Alternativo	Ninguno.
Post-condición	Acceder a la tienda.

Tabla 42: Casos de uso 13

Identificador	Caso14. Seleccionar opciones de búsqueda
Actor	Usuario no registrado y registrado
Objetivo	Poder buscar el producto adecuado.
Pre-condición	1. Tener acceso a la tienda por cualquier navegador. 2. Tener la aplicación móvil instalada en un dispositivo.
Escenario	1. En la página principal elegir en el cuadro superior las diferentes opciones para encontrar un producto.
Escenario Alternativo	Buscar por las distintas páginas de la tienda el producto.
Post-condición	Poder encontrar el producto deseado.

Tabla 43: Casos de uso 14

Identificador	Caso15. Consultar ofertas
Actor	Usuario no registrado y registrado
Objetivo	Ver el precio de los productos, si tienen algún descuento, si están en subasta o si se aceptaría una oferta por un artículo.
Pre-condición	1. Tener acceso a la tienda por cualquier navegador. 2. Tener la aplicación móvil instalada en un dispositivo.
Escenario	1. En la página principal mirar los datos de cada una de las imágenes de productos.
Escenario Alternativo	Entrar dentro de un producto y visualizar sus características.
Post-condición	Obtener información acerca de los productos.

Tabla 44: Casos de uso 15

Identificador	Caso16. Consultar a los vendedores
Actor	Usuario no registrado y registrado
Objetivo	Consultar a los vendedores de productos.
Pre-condición	1. Tener acceso a la tienda por cualquier navegador. 2. Tener la aplicación móvil instalada en un dispositivo.

Escenario	1. Dentro de cada producto, en la pestaña de FAQ's, enviar un mensaje al dueño de la tienda que oferta el producto.
Escenario Alternativo	Ninguno.
Post-condición	Obtener información acerca de los productos.

Tabla 45: Casos de uso 16

Identificador	Caso17. Valorar productos
Actor	Usuario registrado.
Objetivo	Dejar una valoración sobre un producto.
Pre-condición	Estar registrado
Escenario	1. Dentro de cada producto, en la pestaña de valoraciones señalar el número de estrellas que se le otorga al producto y escribir una opinión acerca de él.
Escenario Alternativo	Ninguno.
Post-condición	Dejar información de un producto para que otros usuarios conozcan más sobre ese producto.

Tabla 46: Casos de uso 17

Identificador	Caso18. Realizar Encuesta de satisfacción
Actor	Usuario registrado
Objetivo	Posibilidad de realizar una encuesta de satisfacción.
Pre-condición	Estar registrado
Escenario	1. Acceder al perfil con nombre de usuario o correo electrónico, y contraseña. 2. Pinchar en el menú principal superior en 'Encuesta de satisfacción'. 3. Responder las preguntas.
Escenario Alternativo	Ninguno.
Post-condición	Dar opinión sobre el sitio.

Tabla 47: Casos de uso 18

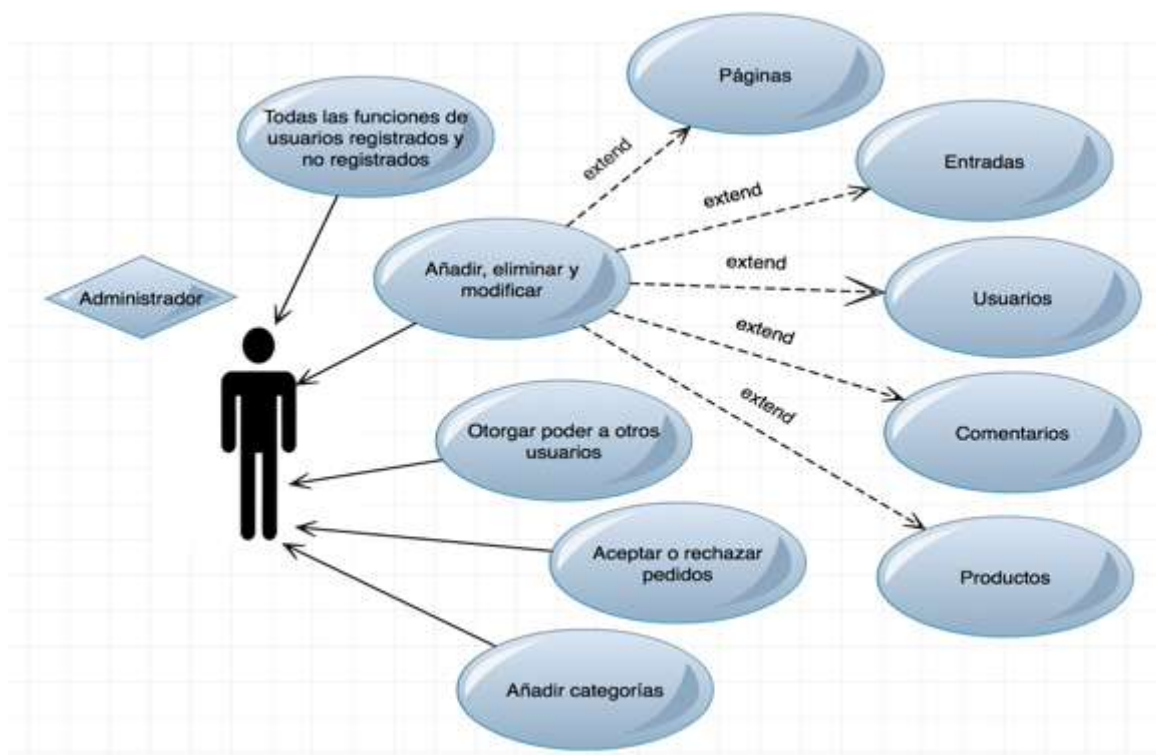


Ilustración 42: Casos de uso 2

Identificador	Caso19. Añadir, eliminar y modificar
Actor	Administrador.
Objetivo	Añadir, eliminar o modificar cualquier cosa de las aplicaciones.
Pre-condición	Ser administrador.
Escenario	1. En el menú de WordPress elegir la opción que se desee.
Escenario Alternativo	En la aplicación de WordPress configurar el sitio.
Post-condición	Mantenimiento del sitio.

Tabla 48: Casos de uso 19

Identificador	Caso20. Páginas (extend Añadir, eliminar y modificar)
Actor	Administrador.
Objetivo	Añadir, eliminar o modificar cualquiera de las páginas de la tienda.
Pre-condición	Ser administrador.
Escenario	1. En el menú de WordPress elegir Páginas. 2. Añadir, eliminar o modificar páginas.
Escenario Alternativo	Ninguno.
Post-condición	Añadido, eliminación o modificación de páginas de la tienda.

Tabla 49: Casos de uso 20

Identificador	Caso21. Entradas (extend Añadir, eliminar y modificar)
Actor	Administrador
Objetivo	Añadir, eliminar o modificar entradas de la tienda.
Pre-condición	Ser administrador.
Escenario	1. En el menú de WordPress acceder a la pestaña de Entradas. 2. Añadir, eliminar o modificar páginas.
Escenario Alternativo	Ninguno.
Post-condición	Añadido, eliminación o modificación de entradas de la tienda.

Tabla 50: Casos de uso 21

Identificador	Caso22. Usuarios (extend Añadir, eliminar y modificar)
Actor	Administrador.
Objetivo	Añadir, eliminar o modificar usuarios.
Pre-condición	Ser administrador.
Escenario	1. En el menú de WordPress acceder a la pestaña de Usuarios. 2. Añadir, eliminar o modificar usuarios.
Escenario Alternativo	Ninguno.
Post-condición	Añadido, eliminación o modificación de usuarios.

Tabla 51: Casos de uso 22

Identificador	Caso23. Comentarios (extend Añadir, eliminar y modificar)
Actor	Administrador.
Objetivo	Añadir, eliminar o modificar comentarios.
Pre-condición	Ser administrador.
Escenario	1. En el menú de WordPress acceder a la pestaña de Comentarios. 2. Añadir, eliminar o modificar comentarios.
Escenario Alternativo	Ninguno.
Post-condición	Añadido, eliminación o modificación de comentarios.

Tabla 52: Casos de uso 23

Identificador	Caso24. Productos (extend Añadir, eliminar y modificar)
Actor	Administrador
Objetivo	Añadir, eliminar o modificar productos.
Pre-condición	Ser administrador.
Escenario	1. En el menú de WordPress acceder a la pestaña de Productos. 2. Añadir, eliminar o modificar productos.
Escenario Alternativo	Ninguno.
Post-condición	Añadido, eliminación o modificación de productos.

Tabla 53: Casos de uso 24

Identificador	Caso25. Otorgar poder a otros usuarios
Actor	Administrador
Objetivo	Cambiar los privilegios de usuarios.
Pre-condición	Ser administrador.
Escenario	1. En el menú de WordPress acceder a la pestaña de Usuarios. 2. Elegir un usuario al que se desee dar privilegios. 3. Donde indica el perfil del usuario cambiar a Administrador o a cualquier otro tipo de perfil.
Escenario Alternativo	Ninguno.
Post-condición	Añadir o disminuir el nivel de usuario dentro de las aplicaciones.

Tabla 54: Casos de uso 25

Identificador	Caso26. Aceptar o rechazar pedidos
Actor	Administrador
Objetivo	Aceptar o rechazar pedidos realizados por cualquier usuario.
Pre-condición	Ser administrador.
Escenario	1. En el menú de WordPress acceder a la pestaña de WooCommerce/Pedidos. 2. Si se ha realizado alguna compra o venta de un producto se podrá aceptar o rechazar.
Escenario	Ninguno.

Alternativo	
Post-condición	Control sobre los productos vendidos.

Tabla 55: Casos de uso 26

Identificador	Caso27. Añadir categorías
Actor	Administrador.
Objetivo	Añadir tipos de categorías de productos.
Pre-condición	Ser administrador.
Escenario	1. En el menú de WordPress acceder a la pestaña de Productos/Categorías. 2. Crear, eliminar o modificar las categorías que se quiera.
Escenario Alternativo	Ninguno.
Post-condición	Poder elegir por parte de los usuarios la categoría de los productos que añaden a la tienda.

Tabla 56: Casos de uso 27

4.2 Diseño

Una vez realizado el análisis de los requisitos y establecer las funcionalidades de las aplicaciones, se debe definir un diseño. Para ello se describirá un diseño de la arquitectura de las aplicaciones web y Android, un diseño de la base de datos que ambas comparten y se comentarán los diferentes diseños realizados para la interfaz de usuario.

4.2.1 Diseño de la arquitectura de la aplicación Web y Android

En este apartado se analizará la arquitectura que deben tomar las aplicaciones, así como un estudio de los componentes que la formarán. Esto permitirá trazar los requisitos descritos en la fase de análisis de tal manera que se pueda verificar que la implementación realizada cumple con todas las funcionalidades previamente descritas.

Al tratarse de dos aplicaciones que ofrecen los mismos servicios y se comunican con el mismo servidor, se entenderá, a partir de ahora, que se habla de la arquitectura de ambas aplicaciones, exceptuando el dispositivo que solicita la información al servidor.

Para diseñar un diagrama de componentes y establecer los diferentes elementos que aparecerán en él, es necesario realizar una división a alto nivel de la aplicación. Ésta es dividida en dos módulos, cada uno correspondiente a las principales funcionalidades que deberá llevar a cabo. Por un lado deberá acceder a los servicios que proporciona la tienda y por otro, deberá comunicarse con el servidor que almacena la información de

WordPress y la base de datos. La siguiente figura muestra de manera gráfica esta división en módulos:

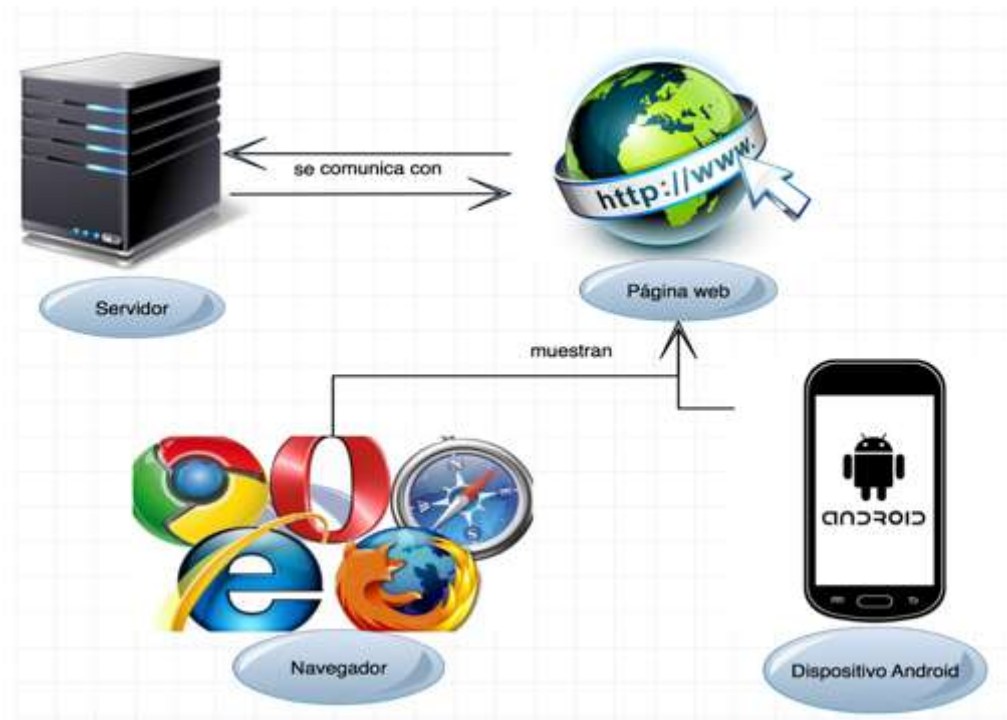


Ilustración 43: Arquitectura aplicaciones

A continuación se muestra un diagrama de los componentes de las aplicaciones y como se comunican entre sí.

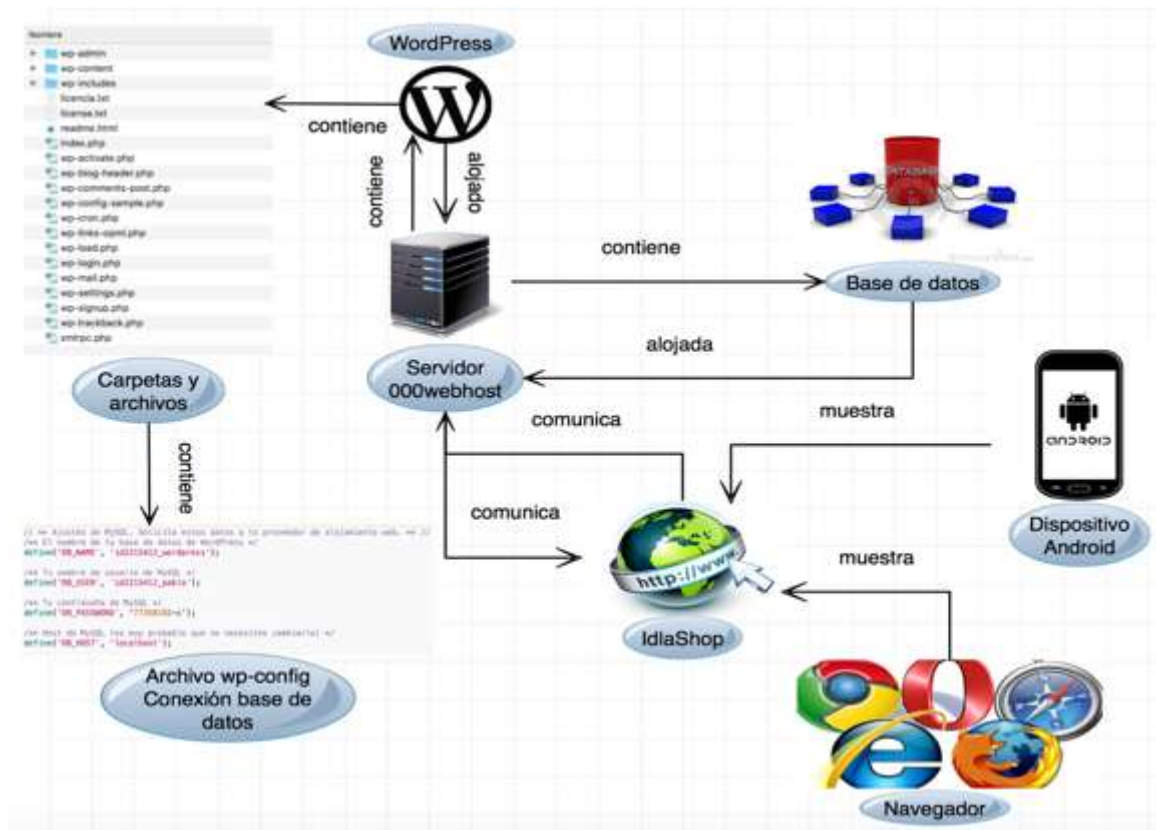


Ilustración 44: Conexión con el servidor

Para cada uno de estos elementos se incluirá una especificación siguiendo el siguiente formato tabular que incluirá la siguiente información:

- **Identificador:** identifica de manera unívoca un componente. Seguirá la nomenclatura ComponenteXX, donde XX corresponde al número del componente.
- **Descripción:** descripción del componente.
- **Acciones:** acciones que realiza el componente.
- **Dependencia:** componentes de los que depende.
- **Requisitos:** requisitos que genera este componente.

Identificador	Componente01. Navegador
Descripción	Se encargan de mostrar la página de la tienda. Se recomienda el uso de Google Chrome.
Acciones	- Mostrar la página de la tienda. - Ofrecer los servicios a usuarios.
Dependencia	Componente03.
Requisitos	Requisito-capacitivo01, Requisito-restrictivo03

Tabla 57: Arquitectura página web 1

Identificador	Componente02. Dispositivo Android
Descripción	Se encargan de mostrar la página de la tienda. Se requiere un dispositivo Android para ello. El usuario deberá dar permiso a la aplicación para acceder a sus archivos y el acceso a la cámara.
Acciones	• Mostrar la página de la tienda. • Ofrecer los mismos servicios que la aplicación web a usuarios. • Solicitar permisos para poder usarla.
Dependencia	Componente03.
Requisitos	Requisito-capacitivo01, Requisito-restrictivo03

Tabla 58: Arquitectura página web 2

Identificador	Componente3. IdlaShop
Descripción	Conjunto de páginas de la tienda que ofrecen los servicios necesarios para su funcionamiento.
Acciones	• Comprar y vender productos si se está registrado. • Cualquier usuario puede navegar por la página.
Dependencia	Componente01, Componente02, Componente04, Componente05, Componente06, Componente07.
Requisitos	Desde Requisito-capacitivo01 hasta Requisito-capacitivo17, Requisito-restrictivo03, Requisito-restrictivo05.

Tabla 59: Arquitectura página web 3

Identificador	Componente04. Servidor 000webhost
Descripción	Servidor o hosting GRATUITO donde se alojan todas las carpetas y archivos de WordPress y la base de datos. Ofrece los servicios necesarios para lanzar el sitio web con algunas restricciones: • 1000 MB de espacio en Disco. • 10000 MB de Ancho de Banda. • 2 Bases de datos MySQL. • 50Envíos de Correo. • 2 Sitios Web. • Creador de sitios. • Hosting Dominio Gratis. • Hosting WordPress. • SSL en servidor público.

	Si se quieren ver más características sobre este servidor acceder a: https://es.000webhost.com/web-hosting-premium#premium-pricing-table
Acciones	• Albergar todos los archivos necesarios para el funcionamiento de la tienda. • Comunicarse con un navegador para mostrar las páginas.
Dependencia	Componente01, Componente02, Componente03, Componente05, Componente 06, Componente07.
Requisitos	Requisito-capacitivo01, Requisito-restrictivo03, Requisito-restrictivo05.

Tabla 60: Arquitectura página web 4

Identificador	Componente05. Base de datos
Descripción	Base de datos que almacena toda la información sobre los usuarios registrados y los productos anunciados en la tienda.
Acciones	• Almacenar información necesaria para el correcto funcionamiento de la tienda. • Comunicarse con el servidor para guardar o mostrar información solicitada desde la página.
Dependencia	Componente03
Requisitos	Desde Requisito-capacitivo01 hasta Requisito-capacitivo17, Requisito-restrictivo03, Requisito-restrictivo05.

Tabla 61: Arquitectura página web 5

Identificador	Componente06. WordPress
Descripción	CMS almacenado en el hosting. Se encarga de guardar toda la información relacionada con el funcionamiento de la página. Por ejemplo: cada vez que se instala un plugin se crea una carpeta que alberga todos los archivos necesarios para su funcionamiento en WordPress.
Acciones	• Almacenar información relacionada con el funcionamiento de WordPress. • Sirve para modificar todo lo relacionado con el funcionamiento, apariencia y características de la página. • Se comunica con la base de datos que almacena una tabla nueva por cada plugin instalado e información de ajustes y

	características de la página.
Dependencia	Componente04, Componente05
Requisitos	Desde Requisito-capacitivo01 hasta Requisito-capacitivo17, Requisito-restrictivo03, Requisito-restrictivo05.

Tabla 62: Arquitectura página web 6

Identificador	Componente07. Carpetas y archivos de WordPress
Descripción	Información relacionada con WordPress. Son todas las carpetas y todos los archivos que necesita para funcionar.
Acciones	• Contener la información necesaria para que la tienda funcione y ofrezca los servicios sin errores.
Dependencia	Componente04, Componente05, Componente06.
Requisitos	Desde Requisito-capacitivo01 hasta Requisito-capacitivo17, Requisito-restrictivo03, Requisito-restrictivo05.

Tabla 63: Arquitectura página web 7

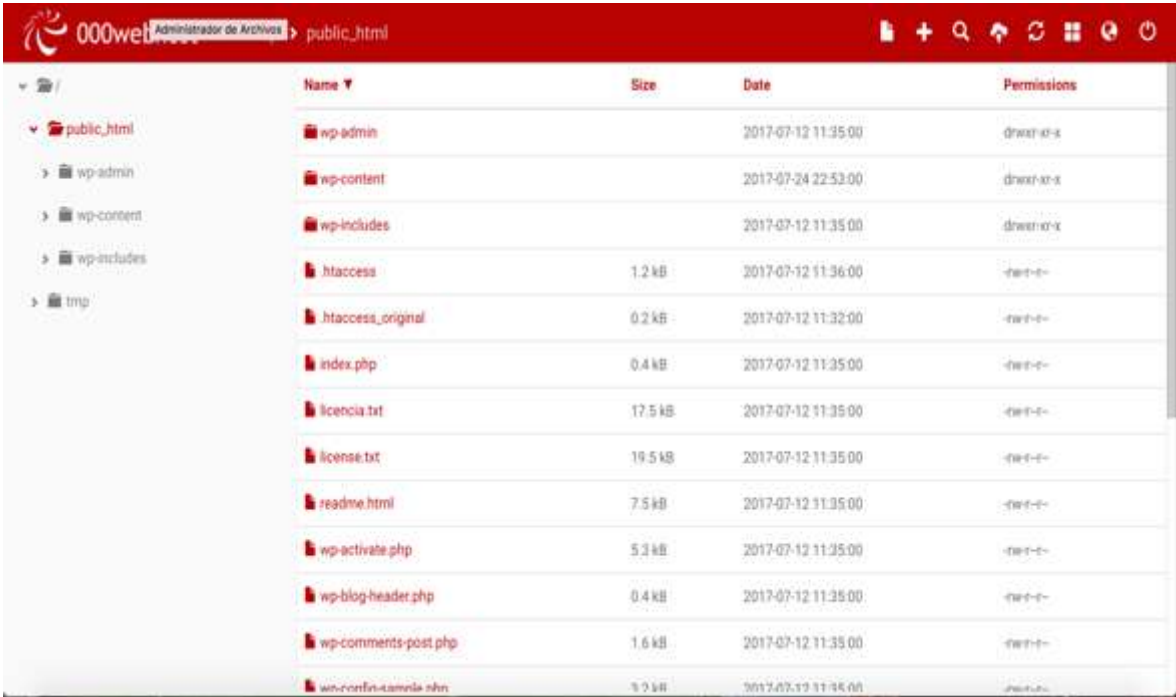
Identificador	Componente08. Archivo wp-config.php
Descripción	Archivo imprescindible para que la base de datos se comuniquen con el servidor.
Acciones	• Contener la información necesaria para comunicar la base de datos con el servidor.
Dependencia	Componente04, Componente05, Componente06, Componente07.
Requisitos	Requisito-capacitivo01, Requisito-restrictivo03, Requisito-restrictivo05.

Tabla 64: Arquitectura página web 8

4.2.2 Diseño de la base de datos

El uso de un sistema de almacenamiento para poder manejar la información es imprescindible. Para ello se contará con una base de datos que almacene toda la información de los usuarios registrados, los productos disponibles en la tienda y los ajustes de cada uno de los plugins instalados en WordPress. Además, serán necesarias una serie de carpetas y archivos alojados en el servidor que contendrán toda la información relacionada con los servicios y ajustes necesarios para que la tienda funcione correctamente como se habló anteriormente en el [punto 3.1.1.](#) de este trabajo.

En la siguiente ilustración se muestran todas las carpetas y archivos alojados en el servidor necesarios para el funcionamiento de la tienda:



Name	Size	Date	Permissions
wp-admin		2017-07-12 11:35:00	drwxr-xr-x
wp-content		2017-07-24 22:53:00	drwxr-xr-x
wp-includes		2017-07-12 11:35:00	drwxr-xr-x
.htaccess	1.2 kB	2017-07-12 11:36:00	-rwxr-xr-x
.htaccess_original	0.2 kB	2017-07-12 11:32:00	-rwxr-xr-x
index.php	0.4 kB	2017-07-12 11:35:00	-rwxr-xr-x
licencia.txt	17.5 kB	2017-07-12 11:35:00	-rwxr-xr-x
license.txt	19.5 kB	2017-07-12 11:35:00	-rwxr-xr-x
readme.html	7.5 kB	2017-07-12 11:35:00	-rwxr-xr-x
wp-activate.php	5.3 kB	2017-07-12 11:35:00	-rwxr-xr-x
wp-blog-header.php	0.4 kB	2017-07-12 11:35:00	-rwxr-xr-x
wp-comments-post.php	1.6 kB	2017-07-12 11:35:00	-rwxr-xr-x
wp-config-sample.php	5.0 kB	2017-07-12 11:35:00	drwxr-xr-x

Ilustración 45: Servidor

A continuación se muestran las tablas contenidas en la base de datos que está alojada en el servidor y que se comunica con las carpetas y archivos mostrados en la Ilustración 45.

The screenshot shows the phpMyAdmin interface for a database named 'id2215412_wp_8bba435ee45ect'. The left sidebar lists the tables, and the main area displays a table list with columns: Table, Action, Rows, Type, and Collation. The tables listed are:

Table	Action	Rows	Type	Collation
wp_commentmeta	☆ Browse Structure Search Insert Empty Drop	237	InnoDB	utf8mb4_unicode_ci
wp_comments	☆ Browse Structure Search Insert Empty Drop	67	InnoDB	utf8mb4_unicode_ci
wp_links	☆ Browse Structure Search Insert Empty Drop	8	InnoDB	utf8mb4_unicode_ci
wp_options	☆ Browse Structure Search Insert Empty Drop	944	InnoDB	utf8mb4_unicode_ci
wp_postmeta	☆ Browse Structure Search Insert Empty Drop	2,486	InnoDB	utf8mb4_unicode_ci
wp_posts	☆ Browse Structure Search Insert Empty Drop	261	InnoDB	utf8mb4_unicode_ci
wp_priv_commission	☆ Browse Structure Search Insert Empty Drop	3	InnoDB	utf8_unicode_ci
wp_simple_auction_log	☆ Browse Structure Search Insert Empty Drop	8	InnoDB	utf8_unicode_ci
wp_termmeta	☆ Browse Structure Search Insert Empty Drop	46	InnoDB	utf8mb4_unicode_ci
wp_terms	☆ Browse Structure Search Insert Empty Drop	32	InnoDB	utf8mb4_unicode_ci
wp_term_relationships	☆ Browse Structure Search Insert Empty Drop	119	InnoDB	utf8mb4_unicode_ci
wp_term_taxonomy	☆ Browse Structure Search Insert Empty Drop	32	InnoDB	utf8mb4_unicode_ci
wp_usermeta	☆ Browse Structure Search Insert Empty Drop	151	InnoDB	utf8mb4_unicode_ci
wp_users	☆ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_unicode_ci
wp_wcpdf_invoice_number	☆ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_api_keys	☆ Browse Structure Search Insert Empty Drop	8	InnoDB	utf8mb4_unicode_ci

Ilustración 46: Tablas base de datos 1

The screenshot shows the phpMyAdmin interface for the same database, displaying the second set of tables. The tables listed are:

Table	Action	Rows	Type	Collation
wp_woocommerce_attribute_taxonomies	☆ Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_downloadable_product_permissions	☆ Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_log	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_order_itemmeta	☆ Browse Structure Search Insert Empty Drop	224	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_order_items	☆ Browse Structure Search Insert Empty Drop	47	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_payment_tokenmeta	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_payment_tokens	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_sessions	☆ Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_shipping_zones	☆ Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_shipping_zone_locations	☆ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_shipping_zone_methods	☆ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_tax_rates	☆ Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_tax_rate_locations	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_tax_rate	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci
wp_woocommerce_query_cache	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci
wp_wsluserscontacts	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci
wp_wslusersprofiles	☆ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8_unicode_ci
wp_yith_wowl	☆ Browse Structure Search Insert Empty Drop	3	InnoDB	utf8_general_ci
wp_yith_wowl_lists	☆ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8_general_ci

Ilustración 47: Tablas base de datos 2

Dadas las tablas que conforman esta base de datos, se presenta el diagrama Entidad-Relación para comprender las conexiones existentes entre tablas

4.2.3 Diseño de la interfaz

Una vez descrita la función principal de las aplicaciones en puntos anteriores de este trabajo, hay que hacer ciertas aclaraciones para realizar un diseño de la interfaz sencilla y comentar su funcionamiento de tal manera que sirva como un pequeño manual de usuario.

Al tratarse de una tienda online, hay que elegir todos los detalles poniéndose en lugar de un usuario para que todo quede lo más simple posible a hora de navegar por la tienda y aumente su atracción por ella.

Lo primero de todo es elegir un tema acorde con el tipo de trabajo, en este caso una tienda online. Para ello, WordPress ofrece una gran cantidad de temas gratuitos y de pago para poder diseñar una página web al gusto del desarrollador. Debido a que la página hará uso del plugin WooCommerce, el tema elegido deberá ser compatible con dicho plugin y deberá ser gratuito. El tema elegido para el diseño de la tienda online ha sido Zerif Lite.



Ilustración 49: Tema Zerif Lite

La descripción que nos ofrecen al descargar este tema es la siguiente:

Zerif Lite es un tema gratuito para WordPress. Es perfecto para agencias de negocios, empresas corporativas y todo tipo de negocios, sitios para fotografías y freelancer. Se basa en BootStrap con soporte de parallax, es sensible, limpio, moderno, plano y mínimo. Zerif Lite es un ecommerce compatible con WooCommerce, WPML, RTL, Retina-Ready, SEO Friendly y con parallax, la imagen a pantalla completa es uno de los mejores temas de negocio.

Una vez elegido el tema, se tendrá que plantear que servicios ofrecerá la tienda para diseñar una interfaz acorde. Ya que se trata de una tienda donde se puede comprar y vender productos se deberá diseñar un menú específico para este fin y que sea fácil e intuitivo. También, se deberá añadir un buscador de productos para facilitar a los usuarios encontrar el producto ideal. Además, se hará uso de múltiples plugins que proporcionen los servicios necesarios para que el funcionamiento de la página sea óptimo. A continuación se muestran algunos ejemplos de servicios:

- MarketPlace: será una tienda donde los usuarios no solo podrán comprar, también se les dará la opción de poder vender sus productos, ya sean nuevos o usados.
- Subastas: se utilizará un plugin especial que permita hacer subastas de productos.
- Ofertas: se usará un plugin que ofrezca los servicios necesarios para que se puedan realizar ofertas por productos.
- Redes sociales: proporcionará un servicio, en el cual, los usuarios se podrán registrar y podrán acceder con una cuenta propia de alguna de las redes sociales que se ofrecen. También proporcionará la opción de compartir un producto en una red social. (Plugin gratuito pero inviable inserción en la página debido a que se usa un servidor que proporciona un dominio gratuito).

Después de haber aclarado que tema será usado, el tipo de menú que debe haber y expuesto alguno de los servicios más importantes que se ofrecen, se ha decidido elegir como página principal, la página donde se muestran los productos junto con el buscador. Así los usuarios tendrán a su disposición los productos y no tendrán que navegar en la web en busca de ellos.

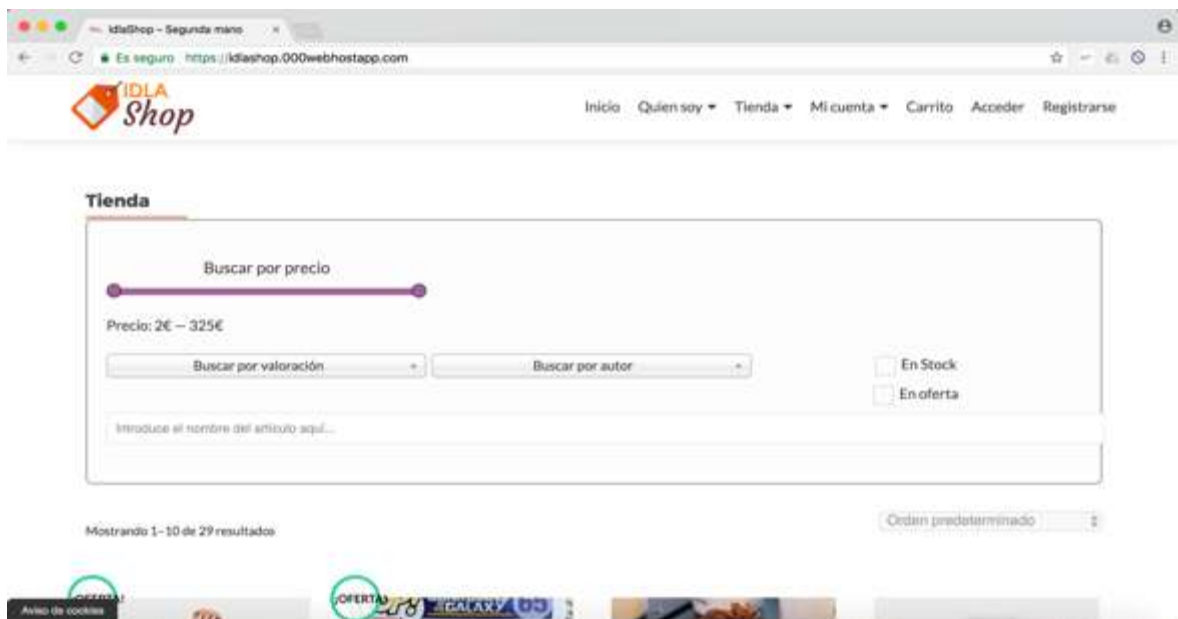


Ilustración 50: Página principal

En la parte superior derecha (Ilustración 50) puede verse el menú principal de la tienda. Cada una de las opciones que se muestran son páginas que han sido diseñadas

Ilustración 51: Página de Contacto

para una función específica. A continuación se muestra el diseño de alguna de ellas:

Esta página (Ilustración 51) está diseñada para que cualquier usuario, registrado o no, pueda comunicarse con el dueño de la tienda añadiendo los datos que se piden.

PRODUCTO	CANTIDAD	COMISIÓN	PORCENTAJE	
iPhone	1	312.00€	96.00%	Ver pedidos
Tutorial de WordPress	1	10.53€	96.00%	Ver pedidos
Totales	2	322.53€		

Ilustración 52: Escritorio de vendedor

Esta página (Ilustración 52) muestra las opciones que tienen los usuarios en su 'tienda personal'. Aquí el usuario podrá ver los datos sobre las ventas y pedidos realizados, así como dar opción a crear y modificar sus propios productos, configurar los ajustes de su tienda y visualizar los productos que se han añadido a su 'tienda personal'.

Como puede verse en la Ilustración 52, en la sección de Ventas/Porcentaje, se refiere al porcentaje de beneficios que el usuario obtiene por vender un producto suyo, en este caso el 96%. El 4% restante de los beneficios serán para el dueño de la tienda (abierto a posibles modificaciones).

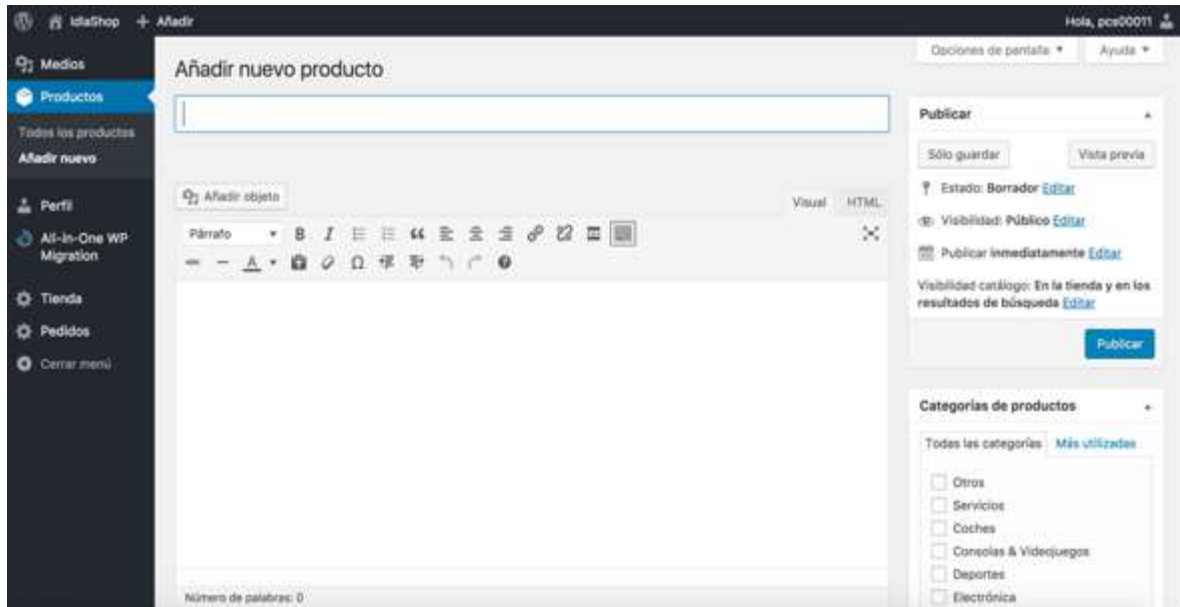


Ilustración 53: Página de nuevo producto

En el 'Escritorio del vendedor' (Ilustración 52), pulsando sobre la opción 'Nuevo producto', nos redirigirá a la página mostrada en la Ilustración 53. Aquí es donde un usuario podrá añadir un producto dando todas las características posibles para facilitar su venta. Una vez creado el producto, el usuario solo deberá darle al botón 'Publicar' y ese producto lo podrán visualizar todos los usuarios que visiten la tienda. También, desde esta página, el usuario podrá acceder a los datos de su perfil, a la información de los pedidos de su 'tienda personal' y a la opción de ver y modificar los productos añadidos anteriormente.

Se puede apreciar como la interfaz de esta página cambia con respecto a las demás mostradas. Esto es debido a que se hace uso de plugins gratuitos de WordPress. Para mostrar el mismo tema de las demás páginas habría que comprar los plugins que se requieran para dicho fin. Aún así, este tipo de interfaz es claro y simple y cualquier usuario podrá añadir productos sin ningún problema.

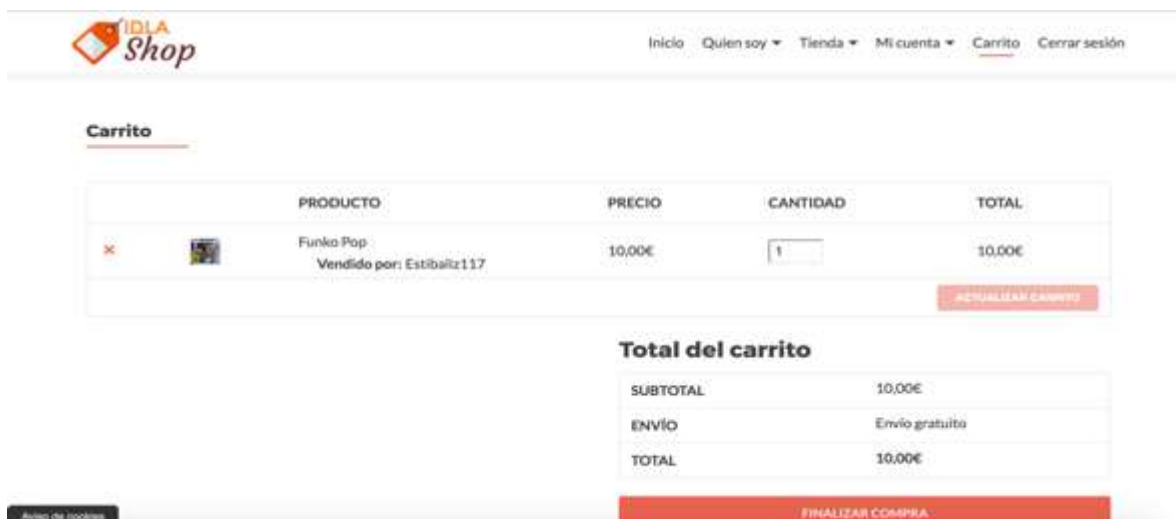


Ilustración 54: Página del carrito

En esta página (Ilustración 54) se muestran los productos añadidos al carrito por el usuario para su futura compra. Una vez el usuario añade los productos que quiera comprar, solo tendrá que pulsar sobre el botón 'Finalizar compra' y se mostrará lo siguiente:

Finalizar compra

Detalles de facturación

Nombre:

Apellidos:

Nombre de la empresa:

País:

Dirección de la calle:

Apartamento, habitación, etc. (opcional):

Localidad / Ciudad:

FINALIZAR COMPRA

Ilustración 55: Página de finalizar compra 1

Aquí (Ilustración 55) es donde el usuario que va a comprar el/los productos añade las direcciones de envío y facturación junto con sus datos personales y elegirá un método de pago.

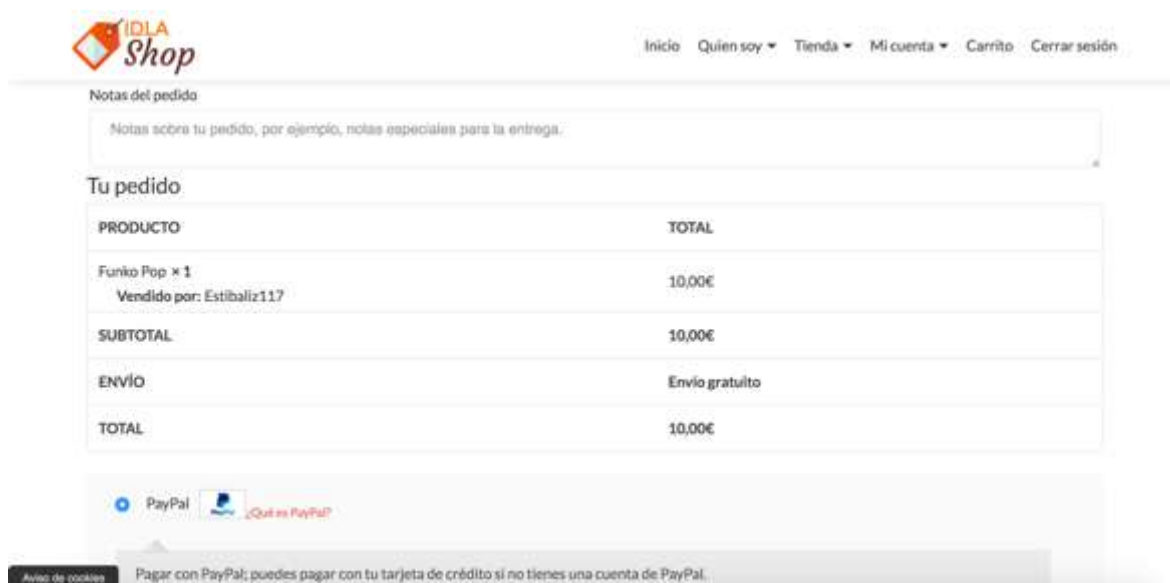


Ilustración 56: Página de finalizar compra 2

Una vez se elija el método de pago (recomendado Paypal) se redirigirá al usuario comprador hacia la página oficial de Paypal donde pondrá los datos de su cuenta y efectuará el pago. El usuario vendedor recibirá el 96% del importe del producto en la cuenta de Paypal definida en el Escritorio del vendedor/Ajustes de tienda y el dueño de la tienda recibirá el 4% en su correspondiente cuenta de Paypal (estos porcentajes pueden ser modificados en los ajustes generales de WooCommerce).

Dados algunos ejemplos de interfaz de la página web, ahora se añadirán algunos ejemplos de páginas vistas desde un dispositivo móvil emulado por Android Studio.



Ilustración 57: Menú principal y páginas de inicio

En la Ilustración 57 se puede apreciar el orden que se debe seguir una vez la aplicación haya sido instalada. Al iniciar la aplicación, aceptar el mensaje de cookies y descartar el mensaje superior, tendremos la vista de la página principal. Al tener las mismas funcionalidades que la aplicación web es sencillo navegar por la aplicación.

A continuación se muestran algunos ejemplos de páginas vistas desde la aplicación Android.

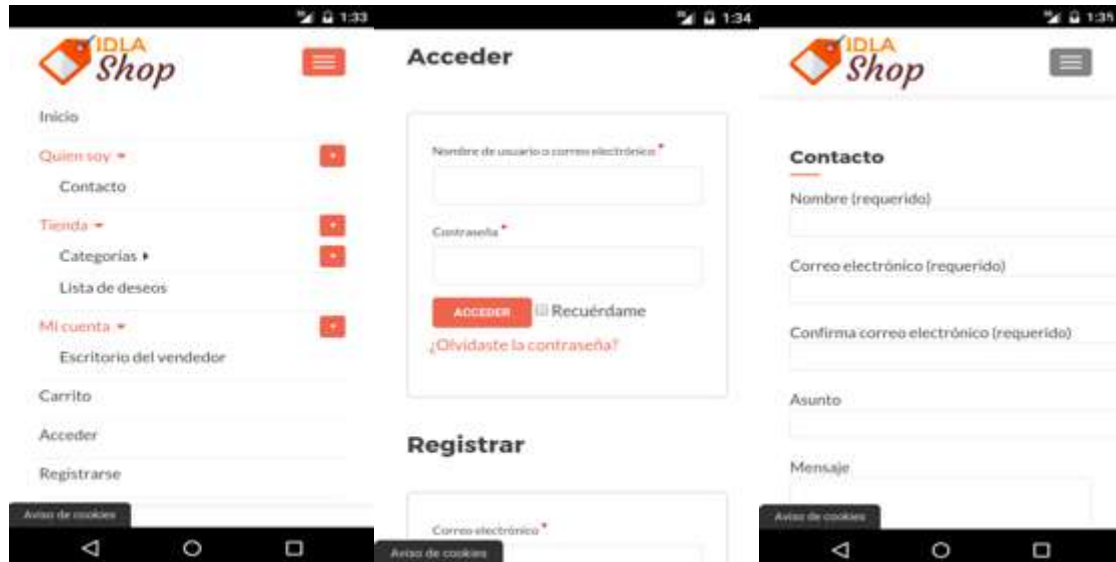


Ilustración 58: Páginas Menú, Acceder y Contacto

5 DESARROLLO

En esta sección se van a comentar los aspectos más relevantes respecto a la implementación de las aplicaciones, así como las consideraciones tomadas durante el desarrollo de las mismas. No se pretende crear un manual de desarrollo extenso por lo que aspectos básicos de implementación relacionados con WordPress y Android que no se encuentren en este trabajo pueden encontrarse en tutoriales ofrecidos en sus respectivas páginas de desarrollo.

5.1 Aplicación Web con WordPress (IdlaShop)

Esta aplicación, es quizás, la más importante del trabajo debido a que la mayor parte de los esfuerzos para alcanzar los objetivos están puestos en ella, ya que dada ésta, se construirá la aplicación Android. En WordPress se tendrán todos los archivos necesarios para el funcionamiento de la tienda y estarán alojados en el servidor junto con la base de datos usada.

Para el desarrollo de esta aplicación se ha necesitado la instalación de WordPress en un hosting gratuito (para hacer este trabajo lo más real posible) para que almacene todos sus archivos y facilite el intercambio de la información con una base de datos. El servidor elegido ha sido 000webhost (<https://es.000webhost.com/>). La misma empresa que ofrece los servicios de hosting indica que pasos seguir para lograr hacer un sitio web. Los pasos son los siguientes:



Ilustración 59: Pasos del hosting

Una vez completados los pasos 1 y 2, el paso número 3 dará las opciones de construir el sitio web con WordPress desde cero o subir nuestros archivos directamente si ya se tiene el desarrollo de una aplicación web o parte de ella. Para la realización de este trabajo se ha optado por subir WordPress al servidor y empezar de cero ya que pueden surgir problemas con las versiones de MySQL y PHP que se tienen instaladas en el ordenador para su uso como servidor local.



Ilustración 60: Elección de WordPress en el hosting

Una vez elegida la opción 'Construir el sitio web de WordPress', el paso número 4 estará finalizado al subir todas las carpetas y archivos de WordPress al hosting.

En el punto 5 y 6, la empresa del hosting muestra las diferentes maneras de cambiar el dominio de nuestro sitio pero para el desarrollo de la aplicación no será necesario ya que se trata de comprar un dominio.

Una vez los archivos de WordPress estén alojados en el servidor se añadirá una base de datos:

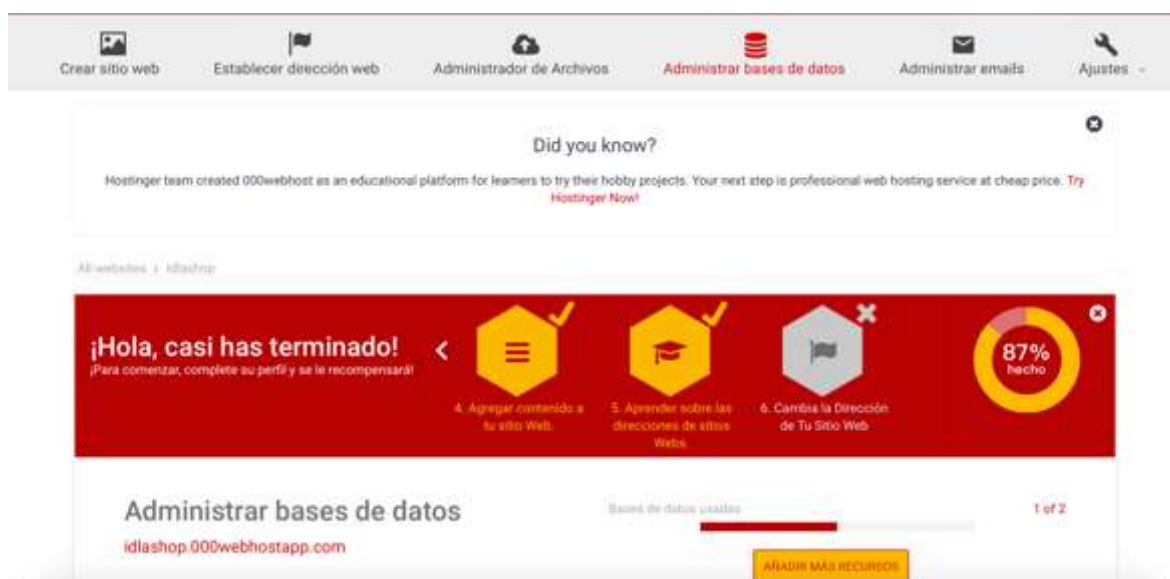


Ilustración 61: Añadir base de datos al hosting

Una vez creada la base de datos se deberá configurar el archivo wp-config.php (como se muestra en el apartado [3.1.2.1](#) punto número 8 de este trabajo). Finalizado este paso, la base de datos y hosting estarán comunicados. También se podrá dar un nombre al sitio con la terminación 000webhostapp.com. El sitio creado para este trabajo se llamará IdlaShop y su URL principal será: <http://idlashop.000webhostapp.com>.

El último paso, y posiblemente más importante, será el desarrollo de la aplicación para que dé los servicios que se requieran. Para ello se hace uso de los plugins gratuitos

de WordPress que se pueden descargar en su página. Al ser administrador del sitio se tendrá acceso a la siguiente página donde se podrá configurar todo lo relacionado con el sitio. En la pestaña Plugins/Añadir nuevo, se podrán descargar los plugins.

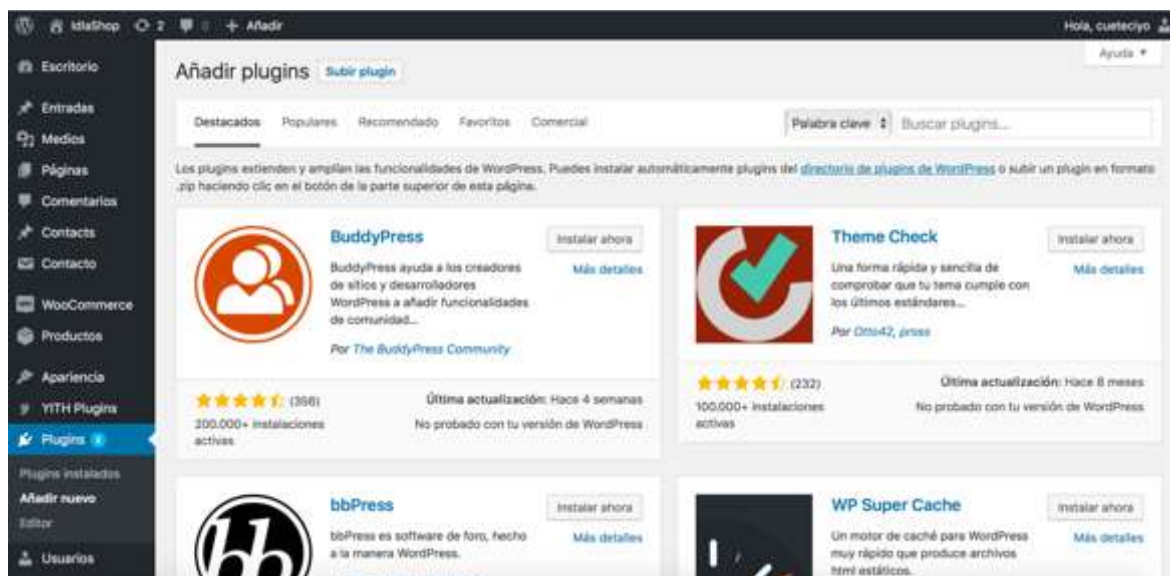


Ilustración 62: Plugins de WordPress

Para el desarrollo de la aplicación web ha sido necesaria la descarga y posterior instalación de los siguientes plugins con el objetivo de convertirla en una tienda online de compra/venta de artículos nuevos y/o usados.

Para cada uno de estos plugins se incluirá una especificación siguiendo el siguiente formato tabular que incluirá la siguiente información:

- Identificador: identifica de manera unívoca un plugin. Seguirá la nomenclatura PluginXX, donde XX corresponde al número del componente.
- Descripción: descripción del plugin.
- Acciones: acciones que realiza el plugin.

Identificador	Plugin01. WooCommerce
Descripción	Herramienta usada para el comercio electrónico.
Acciones	Transforma WordPress para su uso como tienda online.

Tabla 65: Plugin WooCommerce

Identificador	Plugin02. Advanced CSS
Descripción	Editor CSS.
Acciones	Permite escribir código CSS personalizado para cada dispositivo directamente desde WordPress.

Tabla 66: Plugin Advanced CSS

Identificador	Plugin03. Asesor de Cookies
Descripción	Aporta el mensaje del uso de Cookies.
Acciones	Avisa a los nuevos visitantes de la web sobre la utilización de cookies en la página y proporciona los textos iniciales para poder crear una política de cookies correcta.

Tabla 67: Plugin Asesor de Cookies

Identificador	Plugin04. Contact Form
Descripción	Editor de formularios.
Acciones	Añadir o editar formularios necesarios para páginas del sitio como el que puede verse en la página Contacto.

Tabla 68: Plugin Contact Form

Identificador	Plugin05. Pirate Forms
Descripción	Editor de formularios.
Acciones	Añadir o editar formularios necesarios para páginas del sitio como el que puede verse en la página Contacto.

Tabla 69: Plugin Pirate Forms

Identificador	Plugin06. Loco Translate
Descripción	Traductor.
Acciones	Da la posibilidad de cambiar los textos incluidos en temas o plugins de otros idiomas por el que se necesite.

Tabla 70: Plugin Loco Translate

Identificador	Plugin07. Ofertas para WooCommerce
Descripción	Posibilidad de hacer ofertas por productos.
Acciones	Otorga la posibilidad de hacer ofertas por productos determinados. Esta opción solo está disponible para productos añadidos por el dueño de la tienda IdlaShop ya que se trata de la versión gratis.

Tabla 71: Plugin Ofertas para WooCommerce

Identificador	Plugin08. Redsys WooCommerce
Descripción	Añade el método de pago por Redsys.
Acciones	Da la posibilidad de pagar los productos por Redsys. El plugin se

	ha añadido pero no está disponible debido a problemas con el dominio.
--	---

Tabla 72: Plugin Redsys WooCommerce

Identificador	Plugin09. Theme my login
Descripción	Añade funciones a WordPress.
Acciones	Adapta el inicio de sesión, el registro y la página de contraseña olvidada con el tema elegido.

Tabla 73: Plugin Theme my login

Identificador	Plugin10. WooCommerce Vendors
Descripción	Convierte la tienda en un Marketplace.
Acciones	Da la oportunidad a los usuarios registrados vender sus propios productos. Se tiene la versión gratis instalada por lo que no se tendrán todas las funcionalidades posibles.

Tabla 74: Plugin WooCommerce Vendors

Identificador	Plugin11. Woo Checkout Field Editor
Descripción	Modificar campos al pagar.
Acciones	Permite la modificación de los campos necesarios para pagar un producto en la página de finalizar compra.

Tabla 75: Plugin Woo Checkout Field Editor

Identificador	Plugin12. WooCommerce Dynamic Galery LITE
Descripción	Galerías de imágenes de los productos.
Acciones	Permite la modificación de los ajustes de la galería de imágenes sobre cómo se muestran las imágenes de un determinado producto.

Tabla 76: Plugin WooCommerce Dynamic Galery LITE

Identificador	Plugin13. WooCommerce PDF Invoices & Packing Slips
Descripción	Facturas en PDF.
Acciones	Crea, imprime y manda por correo electrónico facturas en PDF y comprobantes de envío de los pedidos de productos.

Tabla 77: Plugin WooCommerce PDF Invoices & Packing Slips

Identificador	Plugin14. WooCommerce Product FAQs
Descripción	Preguntas al vendedor del producto.
Acciones	Permite preguntar directamente al vendedor de un objeto a través de correo electrónico.

Tabla 78: Plugin WooCommerce Product FAQs

Identificador	Plugin15. WooCommerce Simple Auctions
Descripción	Creador de subastas.
Acciones	Con este plugin se puede elegir la opción de subasta al añadir un producto nuevo en la tienda y definir todos los datos necesarios como el tiempo que dura, el precio inicial, cuanto se puede incrementar como mínimo el precio, etc.

Tabla 79: Plugin WooCommerce Simple Auctions

Identificador	Plugin16. WooCommerce Products Filter
Descripción	Filtro para los productos.
Acciones	Añade el filtro que se puede ver en la página de inicio de la tienda. Al tratarse de la versión gratuita no se podrán utilizar algunas características especiales.

Tabla 80: Plugin WooCommerce Products Filter

Identificador	Plugin17. YITH WooCommerce Social Login
Descripción	Añade la posibilidad de registrarse con datos de redes sociales.
Acciones	Permite iniciar sesión y registrarse en la tienda solo a través de Twitter, Facebook o Google al tratarse de la versión gratuita. No está implementado debido a que a la hora de configurar las redes sociales da un error con respecto al dominio del sitio.

Tabla 81: Plugin YITH WooCommerce Social Login

Identificador	Plugin18. YITH WooCommerce Wishlist
Descripción	Lista de deseos.
Acciones	Añade la página de 'Lista de deseos' para guardar productos para futuras compras. Al tratarse de la versión gratuita no se cuentan con todas las opciones posibles.

Tabla 82: Plugin YITH WooCommerce Wishlist

Identificador	Plugin19. Wp-Pro-Quiz
Descripción	Realizar encuestas
Acciones	Añade la página de 'Encuesta de satisfacción' para dar la oportunidad a usuarios poder completar una encuesta en la que se le preguntará por todo lo relacionado con las aplicaciones web y Android.

Tabla 83: Plugin wp-pro-quiz

Una vez instalados todos estos plugins será necesario cambiar los ajustes que vienen por defecto por los que se necesiten, para que todo funcione correctamente. También, será necesario y aconsejable modificar los ajustes que vienen por defecto de WooCommerce ya que se trata del plugin que transforma el sitio web en una tienda online.



Ilustración 63: Ajustes de WooCommerce

Una vez todos los ajustes de todos los plugins, y los de WordPress y WooCommerce estén bien configurados al gusto del desarrollador y se den los mejores servicios a los usuarios sin errores, se pasará a describir el desarrollo que se ha llevado a cabo para la aplicación Android.

5.2 Aplicación Android con Android Studio

Esta aplicación ha sido desarrollada a partir de la aplicación web dado que ambas deben tener la misma apariencia, los mismos contenidos y deben ofrecer los mismos servicios.

Para el desarrollo de la aplicación con Android Studio ha sido vital buscar la manera de que ambas aplicaciones compartan la misma base de datos para así poder ver, por ejemplo, si un usuario pone en venta un producto desde la aplicación web, todos los usuarios que tengan instalada la aplicación Android en su dispositivo puedan visualizar ese producto casi de inmediato.

Dado que ya se cuenta con una gran base (la aplicación web), se ha de buscar la mejor manera para aprovecharla. Después de una búsqueda exhaustiva sobre cómo desarrollar aplicaciones Android dada una página web y que fuesen casi idénticas, se ha optado por desarrollar esta aplicación con el elemento WebView, usado por una gran cantidad de empresas para sus aplicaciones móviles.

La creación de la aplicación se ha iniciado con un proyecto desde cero en Android Studio. Para más información acerca de cómo se ha creado vea el [Anexo 3](#).

Una vez completados los pasos, Android Studio creará el proyecto, en el cual, se tendrá que programar lo que se necesite para que la aplicación funcione. La estructura

que se genera por defecto es la siguiente:

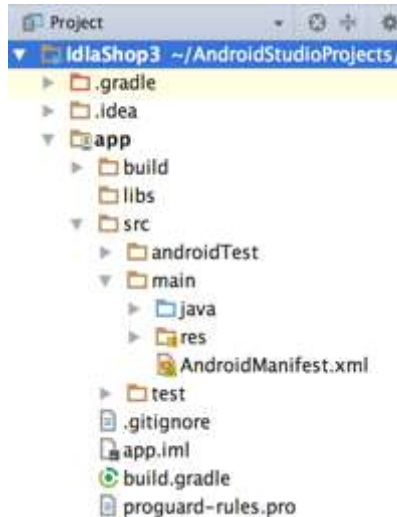


Ilustración 64: Carpetas proyecto Android Studio 1



Ilustración 65: Carpetas proyecto Android Studio 2

- **Carpeta app:** es donde se encuentran las carpetas y archivos más importantes del proyecto. Casi toda la actividad se centrará en esta carpeta.

- **Carpeta src:** en ella se debe incluir todo el código fuente de la aplicación.

- **Capeta main:** contiene las subcarpetas java y res y el archivo AndroidManifest.xml que serán claves para el desarrollo de la aplicación.

- **Carpeta java:** contiene las 'Activities'. Por defecto se crea una que representa la primera pantalla.

- **Carpeta res:** contiene los layout de la aplicación y los iconos de la aplicación.

En cuanto a los archivos que se crean por defecto, se tendrán que tener en cuenta cuatro de ellos:

- **MainActivity.java:** esta clase pertenece a la 'activity' que se crea por defecto y ésta, está formada por dos partes: la parte lógica y la parte gráfica.

La parte lógica es un archivo .java que es la clase que se crea para poder manipular, interactuar y colocar el código de esa actividad.

La parte gráfica es un XML que tiene todos los elementos que se ven en una pantalla declarados con etiquetas parecidas a las de HTML (layouts).

- **Activity_main.xml:** este archivo se aloja en la carpeta layout. Es la estructura visual para una interfaz de usuario, es decir, aquello que hace de intermediario entre el terminal móvil y el usuario. Un elemento de una interfaz de usuario (layouts) se

puede declarar desde un fichero XML o en tiempo de ejecución de la aplicación.

- **Styles.xml:** es un conjunto de reglas que determinan la apariencia y formato de un View o Layout.
- **AndroidManifest.xml:** es un archivo de configuración donde se pueden aplicar las configuraciones básicas de una aplicación. Su configuración puede realizarse a través de una interfaz gráfica, pero se recomienda conocer la sintaxis ya que en muchas ocasiones será más fácil y rápido hacerlo desde el propio XML.

Una vez que queden claros la estructura y archivos de un proyecto en Android Studio, se va a proceder a ampliar este proyecto para que se pueda usar en todos los dispositivos móviles o tablets independientemente de su tamaño y orientación, y que sea compatible con todas las versiones de Android, dando la oportunidad de instalar la aplicación a usuarios que disponen de versiones antiguas o nuevas.

El primer problema que se encuentra en este trabajo es decidir para que dispositivos será compatible la aplicación. Dado que para cumplir los objetivos se incluye el hecho de que debe ser lo más real posible, la aplicación se deberá adaptar a la máxima cantidad de dispositivos posibles con sus respectivas orientaciones (vertical y horizontal). Para ello, se deben añadir layouts en la capeta 'res' del proyecto indicando las dimensiones y orientaciones de los dispositivos.

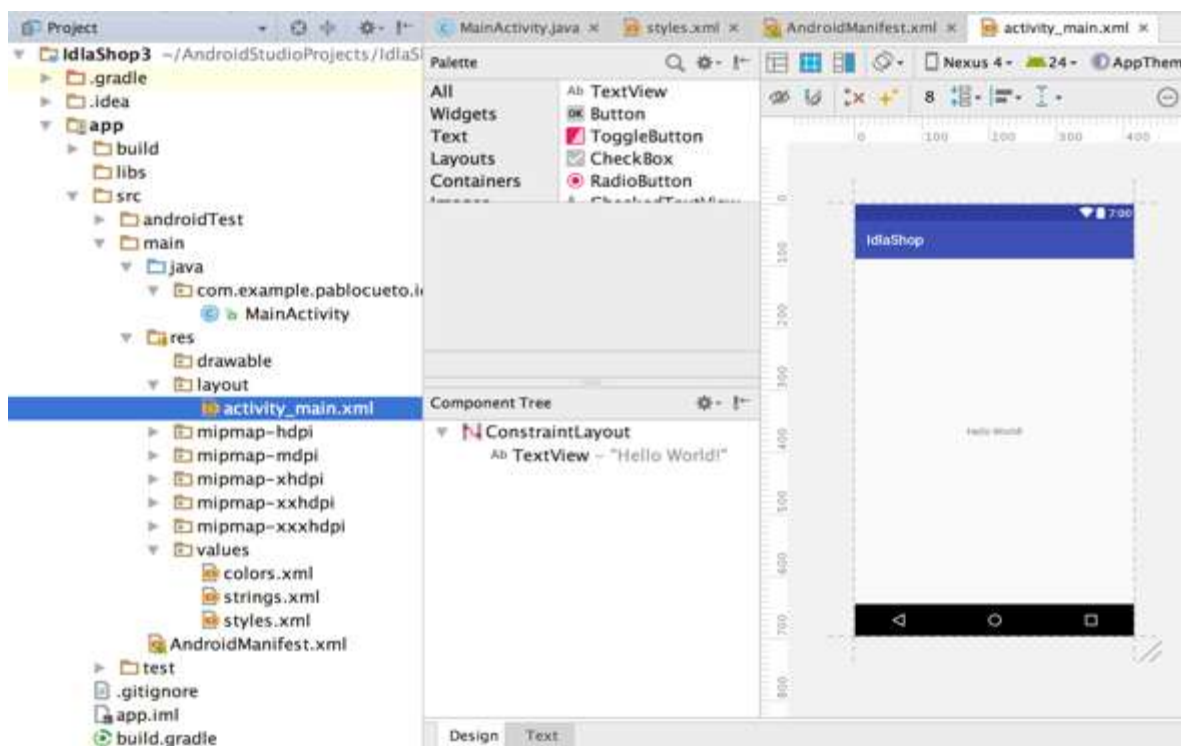


Ilustración 66: Vista de layout vertical

Como se muestra en la Ilustración XX, al crear el proyecto se genera una vista por defecto con un tamaño normal y orientación vertical. Para crear una vista horizontal en dispositivos de tamaño normal, se necesita crear una carpeta nueva dentro de la carpeta 'res'. Se debe elegir la opción New/Android resource directory y en las opciones que se muestran a continuación trasladar al cuadro derecho las opciones 'Size' y 'Orientation'. Una vez hecho esto, se debe elegir el tamaño en la opción 'Screen size' y la opción 'Landscape'. Por último se selecciona la opción 'layout' en la pestaña 'Resource type'.

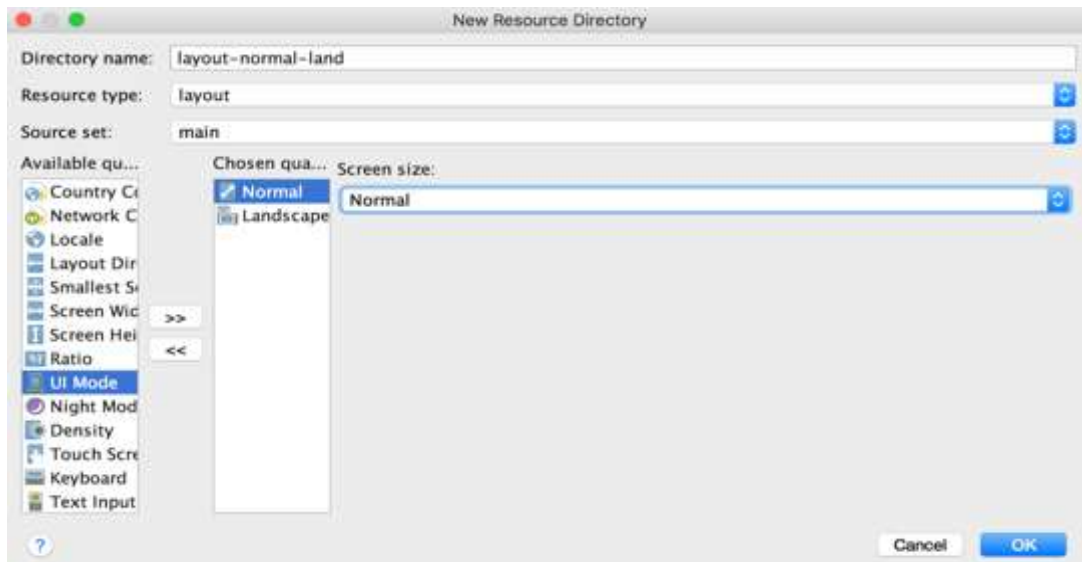


Ilustración 67: Ajustes de layout

Una vez completada la creación de la carpeta, se deberá crear un archivo .xml de igual nombre al creado por defecto, para que se tome el mismo nombre dadas todas las pantallas y todas las orientaciones. Para este proyecto se ha creado otro activity_main.xml. Al abrir el nuevo archivo muestra el mismo dispositivo que el creado por defecto pero en horizontal.

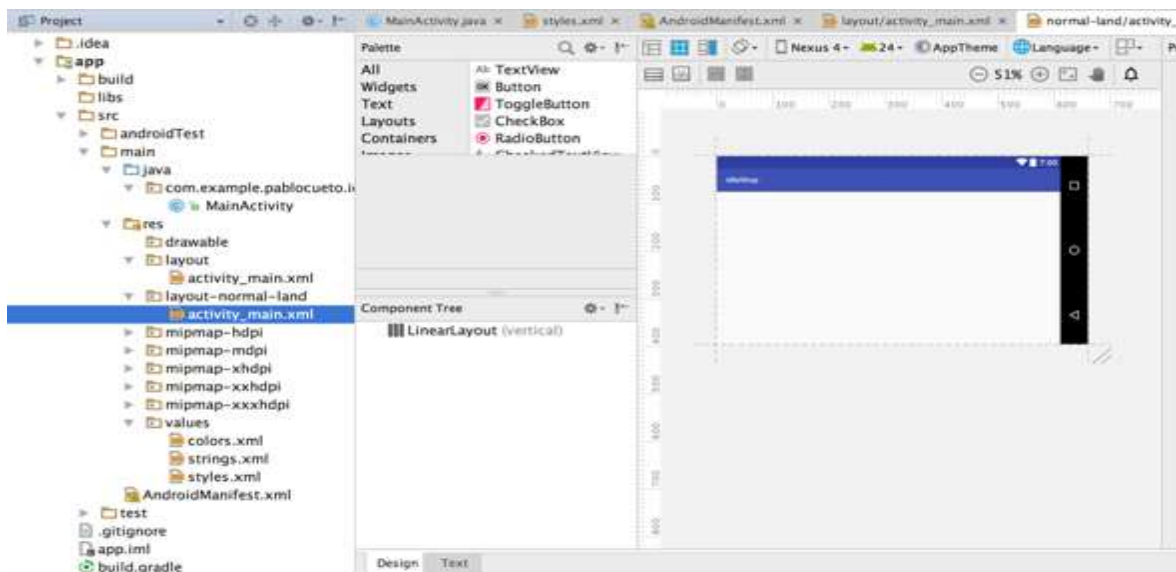


Ilustración 68: Vista de layout horizontal

Este proceso será necesario repetirlo para todos los tamaños y orientaciones para que la aplicación sea compatible con todos ellos.

Una vez resuelto el problema de los dispositivos y sus orientaciones, surge el problema de que se tiene que mostrar en la pantalla la tienda. Dado que se cuenta con una aplicación web, la mejor manera de crear una aplicación será usando el componente WebView. Éste, deberá ser añadido a todos los dispositivos con sus dos orientaciones creadas con anterioridad, deberá ocupar toda la pantalla y se les dará el nombre de 'WebView'. Este nombre será importante para el desarrollo de la aplicación ya que sobre este componente se programarán ciertos ajustes necesarios.

A partir de ahora se mostrará y hablará únicamente del layout que crea Android Studio por defecto y se dará por entendido que todos los pasos que se ejecuten sobre la aplicación afectarán a todos ellos de igual manera.

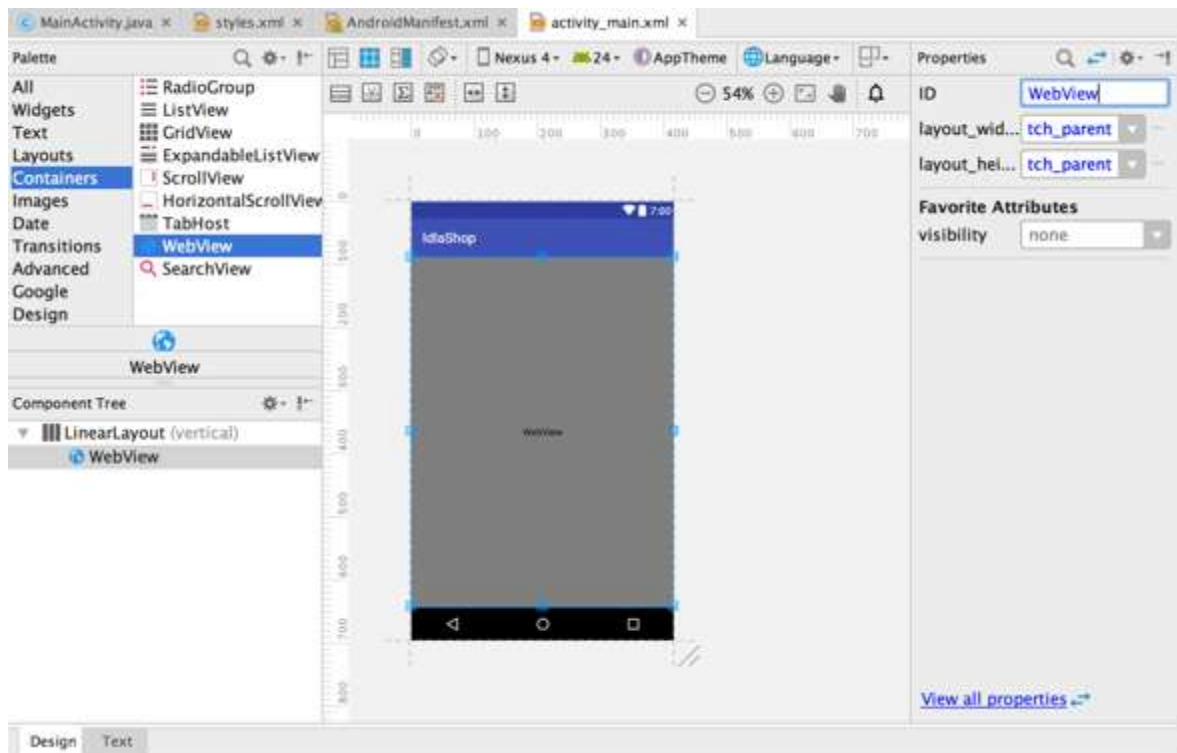


Ilustración 69: Incorporación de WebView

Una vez añadido este componente a todas las pantallas se puede observar la barra de color azul con el nombre de la aplicación. Para que la tienda ocupe toda la pantalla del dispositivo se recomienda eliminar dicha barra. Para ello se deberá añadir al código en el archivo AndroidManifest.xml la siguiente línea:

`android:theme="@style/Theme.AppCompat.NoActionBar"`

Una vez añadida, para que tenga efecto es necesario sincronizar el proyecto y la barra azul desaparecerá.

También se desea eliminar la barra del navegador que saldrá por defecto una vez se lance la aplicación. Para eliminarla se deberá añadir la siguiente línea en el archivo Style.xml dentro de la etiqueta style:

```
<item name ="android:windowNoTitle">true</item>
```

A partir de ahora se tendrán completadas todas las configuraciones de pantalla y solo quedará configurar la aplicación para que muestre la tienda. El componente WebView necesita unas indicaciones para mostrar una página concreta. Para darle estas indicaciones se deberá programar sobre la clase MainActivity.java quedando lo siguiente:

```
package com.example.pablocueto.idlashop;
```

```
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.webkit.WebViewClient;
public class MainActivity extends AppCompatActivity {
    //Variable webView dado el nombre del componente
    WebView webView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        // Se activa JavaScript
        WebSettings webSettings = webView.getSettings();
        webSettings.setJavaScriptEnabled(true);
        //URL que carga la aplicación
        webView.loadUrl("http://idlashop.000webhostapp.com");
        //Se fuerza el webview para que abra los enlaces internos dentro de la aplicación
        webView.setWebViewClient(new WebViewClient());
        //Se fuerza el webview para que abra los enlaces externos en el navegador
        webView.setWebViewClient(new MyAppWebViewClient());
    }
}
```

Dado que este código dará un error en la última línea, lo que se debe hacer es crear la clase MyAppWebViewClient extend WebViewClient y añadir el siguiente código en ella:

```
import android.content.Intent;
import android.net.Uri;
import android.webkit.WebView;
import android.webkit.WebViewClient;

public class MyAppWebViewClient extends WebViewClient {
    @Override
    public boolean shouldOverrideUrlLoading(WebView view, String url) {
        // Url base de la aplicación. Poner como se muestra, sin https://
        if(Uri.parse(url).getHost().endsWith("idlashop.000webhostapp.com")) {
            return false;
        }
        Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse(url));
        view.getContext().startActivity(intent);
    }
}
```

```

return true;
    }
}

```

Con este código siempre se navegará dentro de la de la aplicación web aunque se pulse repetidas veces el botón de 'atrás' de los dispositivos hasta que se salga de ella.

Si se está navegando por la tienda desde la aplicación y se quiere volver a la página anterior se necesitará pulsar el botón 'atrás' del dispositivo. Para que la aplicación no se cierre al pulsarlo es necesario implementar el siguiente código en la clase MainActivity.java

// Método para que al darle al botón de atrás no salga de la //aplicación

```

@Override
public boolean onKeyDown(int keyCode, @NonNull KeyEvent event){
    if(event.getAction() == KeyEvent.ACTION_DOWN){
        switch(keyCode){
            case KeyEvent.KEYCODE_BACK:
                if(webView.canGoBack()){
                    webView.goBack();
                }else{
                    finish();
                }
            }
        return true;
    }
}
return super.onKeyDown(keyCode, event);
}

```

Debido a que se está visualizando una página web se deberá dar permiso a la aplicación para que se conecte a internet desde ella. Se debe añadir la siguiente línea en el archivo AndroidManifest.xml dentro de la etiqueta manifest:

```
<uses-permission android:name="android.permission.INTERNET" />
```

Solo se encuentra un problema más relacionado con la vista de la aplicación en los dispositivos y es, que cuando se gira la pantalla, la aplicación se reinicia automáticamente. Esto puede resultar un problema ya que es necesario girar la pantalla, por cualquier motivo, y se tiene algún campo escrito, la información se perderá. Para solucionar este problema se debe añadir la siguiente línea en el archivo AndroidManifest.xml dentro de la etiqueta activity:

```

android:configChanges="keyboardHidden|orientation|screenSize"
android:label="@IdlaShop"

```

Con todas estas configuraciones, la tienda se podrá mostrar en todos los dispositivos y en las dos orientaciones.

El siguiente problema que surge es que la aplicación no responde al intentar subir archivos cuando se está creando un nuevo producto en la tienda. Esto se debe a que la aplicación necesita unos permisos para acceder a la cámara (en caso de querer tomar

una foto del producto) y a la galería de imágenes (si es donde se tiene guardada la imagen que se necesita). Esto se soluciona dando permisos a la aplicación para que pueda acceder añadiendo las siguientes líneas en el archivo AndroidManifest.xml:

```
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

Pero además de estos permisos, se debe configurar el acceso para diferentes versiones de Android. Debido a que todos los dispositivos no tienen instalada la misma versión, será necesario configurar el acceso para todas. Se añadirá el código necesario a la clase MainActivity.java para que sea compatible con las versiones superiores a la 3.0 ya que, como se puede observar en la Ilustración 37 de este trabajo, el 99% de los dispositivos Android tienen instalada esta versión o superiores.

// Variables necesarias

```
private String mCM;
private ValueCallback<Uri> mUM;
private ValueCallback<Uri[]> mUMA;
private final static int FCR=1;
// Método necesario para acceder a la cámara o a los archivos internos del dispositivo. En caso de
// no haber archivos internos se deberá abrir la cámara.
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent intent){
    super.onActivityResult(requestCode, resultCode, intent);
    if(Build.VERSION.SDK_INT >= 21){
        Uri[] results = null;
        //Comprobar si hay respuesta o no
        if(resultCode== Activity.RESULT_OK){
            if(requestCode == FCR){
                if(null == mUMA){
                    return;
                }
            }
            if(intent == null){
                //Echar foto y si no hay imágenes disponibles
                if(mCM != null){
                    results = new Uri[]{Uri.parse(mCM)};
                }
                else{
                    String dataString = intent.getDataString();
                    if(dataString != null){
                        results = new Uri[]{Uri.parse(dataString)};
                    }
                }
            }
        }
        mUMA.onReceiveValue(results);
        mUMA = null;
    }else{
        if(requestCode == FCR){
            if(null == mUM) return;
            Uri result = intent == null || resultCode != RESULT_OK ? null : intent.getData();
            mUM.onReceiveValue(result);
            mUM = null;
        }
    }
}
```

```

    }
}

// Dentro del método onCreate
//Código necesario para que se active en diferentes versiones de Android
if (Build.VERSION.SDK_INT >= 23 && (ContextCompat.checkSelfPermission(this,
Manifest.permission.WRITE_EXTERNAL_STORAGE) !=
PackageManager.PERMISSION_GRANTED || ContextCompat.checkSelfPermission(this,
Manifest.permission.CAMERA) != PackageManager.PERMISSION_GRANTED)) {
    ActivityCompat.requestPermissions(MainActivity.this, new
String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE, Manifest.permission.CAMERA}, 1);
}
if (Build.VERSION.SDK_INT >= 21) {
    webViewSettings.setMixedContentMode(0);
webView.setLayerType(View.LAYER_TYPE_HARDWARE, null);
} else if (Build.VERSION.SDK_INT >= 19) {
webView.setLayerType(View.LAYER_TYPE_HARDWARE, null);
} else if (Build.VERSION.SDK_INT < 19) {
webView.setLayerType(View.LAYER_TYPE_SOFTWARE, null);
//Para Android 3.0+
public void openFileChooser(ValueCallback<Uri> uploadMsg) {
mUM = uploadMsg;
Intent i = new Intent(Intent.ACTION_GET_CONTENT);
i.addCategory(Intent.CATEGORY_OPENABLE);
i.setType("*/*");
MainActivity.this.startActivityForResult(Intent.createChooser(i, "File Chooser"), FCR);
}

// Para Android 3.0+, por si el metodo anterior no es compatible
public void openFileChooser(ValueCallback uploadMsg, String acceptType) {
mUM = uploadMsg;
Intent i = new Intent(Intent.ACTION_GET_CONTENT);
i.addCategory(Intent.CATEGORY_OPENABLE);
i.setType("*/*");
MainActivity.this.startActivityForResult(
Intent.createChooser(i, "File Browser"),
FCR);
}

//Para Android 4.1+
public void openFileChooser(ValueCallback<Uri> uploadMsg, String acceptType, String capture) {
mUM = uploadMsg;
Intent i = new Intent(Intent.ACTION_GET_CONTENT);
i.addCategory(Intent.CATEGORY_OPENABLE);
i.setType("*/*");
MainActivity.this.startActivityForResult(Intent.createChooser(i, "File Chooser"),
MainActivity.FCR);
}

//Para Android 5.0+
public boolean onShowFileChooser(
WebView webView, ValueCallback<Uri> filePathCallback,
WebChromeClient.FileChooserParams fileChooserParams) {
if (mUMA != null) {
mUMA.onReceiveValue(null);
}
mUMA = filePathCallback;
Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
if (takePictureIntent.resolveActivity(MainActivity.this.getPackageManager()) != null) {

```

```

        File photoFile = null;
    try {
        photoFile = createImageFile();
        takePictureIntent.putExtra("PhotoPath", mCM);
    } catch (IOException ex) {
        Log.e(TAG, "Image file creation failed", ex);
    }
    if (photoFile != null) {
        mCM = "file:" + photoFile.getAbsolutePath();
        takePictureIntent.putExtra(MediaStore.EXTRA_OUTPUT, Uri.fromFile(photoFile));
    } else {
        takePictureIntent = null;
    }
    Intent contentSelectionIntent = new Intent(Intent.ACTION_GET_CONTENT);
    contentSelectionIntent.addCategory(Intent.CATEGORY_OPENABLE);
    contentSelectionIntent.setType("*/*");
    Intent[] intentArray;
    if (takePictureIntent != null) {
        intentArray = new Intent[]{takePictureIntent};
    } else {
        intentArray = new Intent[0];
    }

    Intent chooserIntent = new Intent(Intent.ACTION_CHOOSER);
    chooserIntent.putExtra(Intent.EXTRA_INTENT, contentSelectionIntent);
    chooserIntent.putExtra(Intent.EXTRA_TITLE, "Image Chooser");
    chooserIntent.putExtra(Intent.EXTRA_INITIAL_INTENTS, intentArray);
    startActivityResult(chooserIntent, FCR);
    return true;
}

});

```

Una vez solucionado este problema, la aplicación tendrá los permisos necesarios para acceder a la cámara y a los archivos internos del dispositivo aunque será necesaria la autorización del usuario. La primera vez que se abra la aplicación en un dispositivo aparecerá un mensaje pidiendo autorización para acceder a la cámara y a los archivos internos del dispositivo.

Otro problema que necesita ser corregido es que al comprar un producto descargable, como por ejemplo un libro, la aplicación no responde y no se descarga el archivo. Para solucionar esto se deben dar permisos a la aplicación añadiendo las siguientes líneas en el archivo AndroidManifest.xml:

```

<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.MEDIA_CONTENT_CONTROL" />

```

Además de incluir estos permisos será necesario incluir en la clase MainActivity.java, dentro del método onCreate el siguiente código:

```

webView.setDownloadListener(new DownloadListener() {
    public void onDownloadStart(String url, String userAgent, String
        contentDisposition, String mimetype, long contentLength) {
        //Para descargar directamente a través del gestor de descargas
        final DownloadManager.Request request = new DownloadManager.Request(Uri.parse(url));
        request.allowScanningByMediaScanner();
    }
});

```

```

        request.setMimeType(mimetype);
//-----Cookie-----
String cookies = CookieManager.getInstance().getCookie(url);
        request.addRequestHeader("cookie", cookies);
//-----Cookie-----
        request.addRequestHeader("User-Agent", userAgent);
        request.setDescription("Downloading file...");
        request.setTitle(URLUtil.guessFileName(url, contentDisposition, mimetype));
        request.allowScanningByMediaScanner();
        request.setNotificationVisibility(DownloadManager.Request.VISIBILITY_VISIBLE_NOTIFY_COMPLETED);
        request.setDestinationInExternalPublicDir(Environment.DIRECTORY_DOWNLOADS,
        URLUtil.guessFileName(url, contentDisposition, mimetype));
        final DownloadManager dm = (DownloadManager) getSystemService(DOWNLOAD_SERVICE);
        new Thread("Browser download") {
            public void run() {
                dm.enqueue(request);
            }
        }.start();
    }
});

```

Este código permitirá la descarga directa en segundo plano de los productos descargables que se compren en la tienda.

Por último se ha añadido un sistema que permite a los dispositivos que tengan la aplicación instalada recibir notificaciones push para, por ejemplo, avisar a los usuarios que existen nuevos productos disponibles en la tienda, que hay nuevas ofertas en productos determinados, etc. Para este sistema se ha utilizado la plataforma de desarrollo Firebase en la nube de Google. La información sobre esta plataforma y como crear un proyecto que se asocie con la aplicación Android de este trabajo se podrá consultar en el [Anexo 4](#) de este trabajo.

Una vez creado el proyecto en Firebase se necesitarán dos servicios de Android en la aplicación, uno que se encargue de recibir los mensajes de notificaciones y otro que gestione los token²⁰ que proporciona Firebase al instalar la aplicación en un dispositivo. Para ello se crearán dos clases .java en la carpeta app/src/main/java junto a la clase principal MainActivity y ambas deberán ser extensión del servicio de Firebase.

- **Clase MyFirebaseMessagingService:** será la encargada de gestionar como se reciben las notificaciones en el dispositivo. También debe implementar el método que tiene que ejecutarse cuando se reciban las notificaciones. Dicho método será onMessageReceived.

```

public class MyFirebaseMessagingService extends FirebaseMessagingService {
//Variable para mostrar en consola
public static final String TAP = "Notificaciones";
//Este método se ejecuta cuando se recibe una notificación

```

²⁰ Token: cuando una aplicación está configurada para recibir notificaciones, ésta genera un código único llamado token. Gracias a este token que funciona como identificador, el cliente web Firebase puede enviar un mensaje de notificación a un usuario específico o a todos los usuarios que tengan instalada la aplicación.

```

@Override
public void onMessageReceived(RemoteMessage remoteMessage) {
    super.onMessageReceived(remoteMessage);
    String from = remoteMessage.getFrom();
    //Clase Log para mostrar por consola que se reciben los mensajes. Para pruebas
    Log.d(TAP, "Mensaje recibido de: " + from);
    //Acceder al mensaje que se envía. Como existe la posibilidad de que sea nulo se hace la
    validación con el if/null
    if (remoteMessage.getNotification() != null) {
        //Mostrar en la consola
        Log.d(TAP, "Notificación: " + remoteMessage.getNotification().getBody());
        //Esto es para mostrar la notificación en el dispositivo. Recibe el título y el cuerpo del mensaje
        ShowNotification(remoteMessage.getNotification().getTitle(), remoteMessage.getNotification().getBody());
    }
    //Si se quiere enviar más información a parte del título y el cuerpo del mensaje
    if (remoteMessage.getData().size() > 0) {
        //Para ver en la consola
        Log.d(TAP, "Data: " + remoteMessage.getData());
    }
    //Método para mostrar la notificación en el dispositivo
    private void ShowNotification(String titulo, String mensaje) {
        //Esto es lo que hará cuando se presione sobre la notificación
        Intent intent = new Intent(this, MainActivity.class);
        intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
        PendingIntent pendingIntent =
            PendingIntent.getActivity(this, 0, intent, PendingIntent.FLAG_ONE_SHOT);
        //Para el sonido de la notificación
        Uri soundUri = RingtoneManager.getDefaultUri(RingtoneManager.TYPE_NOTIFICATION);
        /**Con NotificationCompat se puede especificar el icono de la aplicación que varía según la
        versión de Android del dispositivo. Para este caso se ha hecho para cualquier versión. Se añaden
        el título de la notificación, el mensaje, un sonido al recibirla y eliminarla una vez se pulse sobre
        ella. También se llama al intent encargado de lo que sucede cuando se pulsa sobre la notificación.
        */
        NotificationCompat.Builder notificationBuilder = new NotificationCompat.Builder(this)
            .setAutoCancel(true)
            .setSmallIcon(R.drawable.ic_launcher)
            .setContentTitle(titulo)
            .setContentText(mensaje)
            .setSound(soundUri)
            .setContentIntent(pendingIntent);
        //El gestor de notificaciones se encarga de mostrar la notificación
        NotificationManager notificationManager =
            (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);
        notificationManager.notify(0, notificationBuilder.build());
    }
}

```

- **Clase MyFirebaseInstanceIdService** que será la encargada de gestionar los token de los usuarios que tengan la aplicación instalada. También debe implementar el método que se ejecuta cuando se asignan los token. Este token puede cambiar con el tiempo y el método onTokenRefresh avisará cuando suceda.

```

public class MyFirebaseInstanceIdService extends FirebaseInstanceIdService {
    //Variable para mostrar en consola
    public static final String TAP = "Notificaciones";

```

```

//Este método se ejecuta cuando se asigna un token a un dispositivo que ha instalado la aplicación
o cuando se actualiza
@Override
public void onTokenRefresh() {
    super.onTokenRefresh();
    //Sentencia para acceder al token
    String refreshedToken = FirebaseInstanceId.getInstance().getToken();
    //Para ver por consola. Para pruebas
    Log.d(TAP, "Refreshed token: " + refreshedToken);
}
}

```

Además se deberán añadir al archivo AndroidManifest las siguientes líneas de código para que la aplicación use los servicios. Se añadirán justo al finalizar la sentencia 'activity':

```

</activity>
<!-- Ésta es la orden a la aplicación para que el servicio de Android que se encarga de recibir las
notificaciones esté activo. Hace referencia a su clase java e implementa la acción
messaging_event
-->
<service
    android:name="com.example.pablocueto.idlashop.MyFirebaseMessagingService">
    <intent-filter>
        <action android:name="com.google.firebase.MESSAGING_EVENT"/>
    </intent-filter>
</service>
<!-- Ésta es la orden a la aplicación para que el servicio de Android pueda gestionar el token que
proporciona Firebase al instalar la aplicación. Hace referencia a su clase java e implementa la
acción instance_id_event
-->
<service android:name="com.example.pablocueto.idlashop.MyFirebaseInstanceIdService">
    <intent-filter>
        <action android:name="com.google.firebase.INSTANCE_ID_EVENT"/>
    </intent-filter>
</service>

```

Una vez se haya añadido todo el código anterior al proyecto en Android Studio e instalado la aplicación en un dispositivo real (o instalado en el emulador de Android Studio) ya se podrán enviar notificaciones push.

Dentro de la página del proyecto, en la página de Firebase, se debe pinchar sobre la opción 'Notifications' del menú de la izquierda y pinchar sobre 'Nuevo mensaje'. Una vez introducidos los datos de la notificación se envía y se comprueba que se ha recibido.

Texto del mensaje

Notificación

Etiqueta del mensaje (opcional) ?

Notificación de prueba

Fecha de entrega ?

Enviar ahora

Enviar más tarde

Destino

☒ Segmento de usuarios ☐ Tema ☐ Un único dispositivo

Dirigir al usuario si...

Aplicación

IdlaShop

Ilustración 70: Datos de una notificación

Mensaje	Estado ?	Fecha de entrega ?	Plataforma	Estimación de destinos ?	% de mensajes abiertos ?
Notificación de prueba Notificación	✓ Completada	10 ago. 2017 17:32		-	-

Ilustración 71: Envío de una notificación

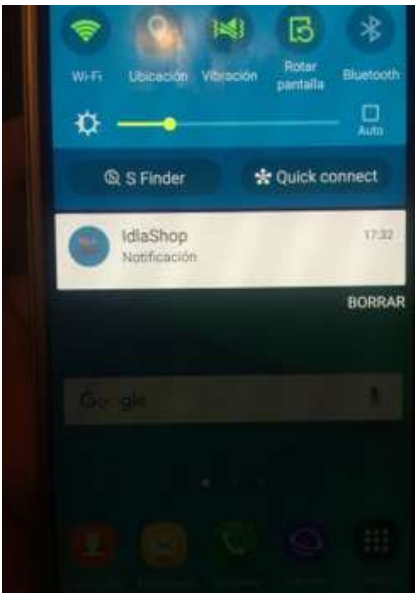


Ilustración 72: Comprobación de la notificación

Como puede verse en la Ilustración 72, la notificación se ha recibido correctamente en un dispositivo que tiene la aplicación instalada. Si se pulsa sobre ella, se abrirá la aplicación.

6 PLANIFICACIÓN Y PRESUPUESTO

En esta sección se incluirá un desglose de las actividades que se han llevado a cabo para llegar a los objetivos y sus duraciones. También, se añadirá el presupuesto por la realización del trabajo.

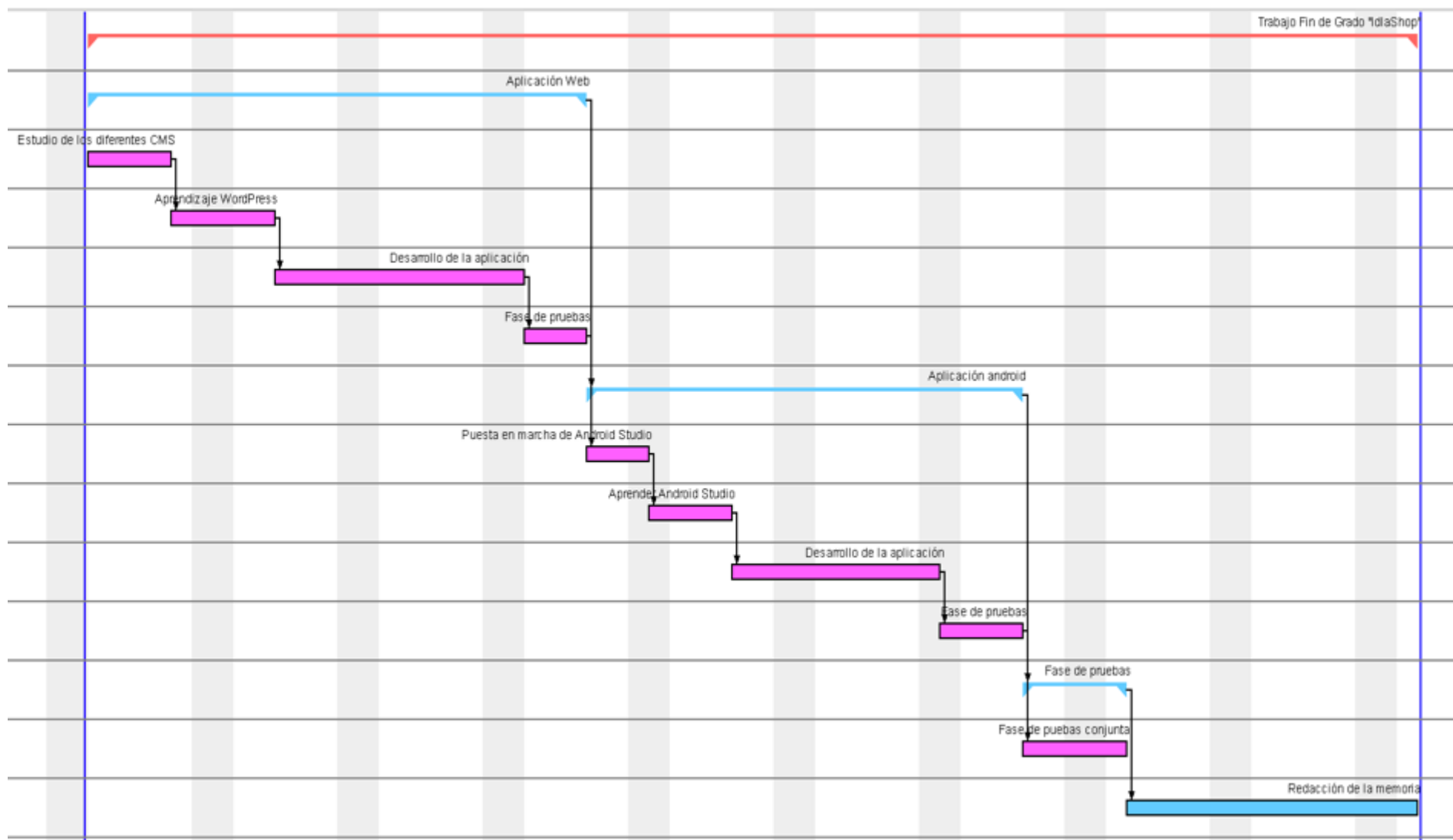
6.1 Planificación

En este punto se va a establecer la duración de las distintas tareas o actividades relacionadas a lo largo del trabajo. Esta planificación se definirá en base a la herramienta del Diagrama de Gantt que es una de las técnicas de planificación más comunes. En él están todas las actividades o tareas en las que se ha dividido el proyecto, junto con la duración de cada una de ellas y los hitos a cumplir para garantizar la finalización del trabajo en el tiempo previsto.

La siguiente tabla muestra las tareas realizadas a lo largo del trabajo estableciendo un inicio y un final que darán lugar a la realización del diagrama.

Nombre de la tarea	Duración
• Trabajo Fin de Grado	Total: 300 horas
○ Aplicación Web	110 horas
❖ Estudio de lo diferentes CMS	10 horas
❖ Aprendizaje de WordPress	20 horas
❖ Desarrollo de la aplicación	75 horas
❖ Fase de pruebas	5 horas
○ Aplicación Android	90 horas
❖ Puesta en marcha de Android Studio	5 horas
❖ Aprender Android Studio	20 horas
❖ Desarrollo de la aplicación	60 horas
❖ Fase de pruebas	5 horas
○ Fase de pruebas (Dispositivos reales)	10 horas
❖ Fase de pruebas conjunta	10 horas
○ Redacción de la memoria	90 horas
❖ Redacción de la memoria	90 horas

Tabla 84: Tareas del trabajo



Ilustración

73:

Diagrama

de

Gantt

6.2 Presupuesto

Una vez se ha establecido la planificación del trabajo, se presenta el presupuesto para la realización del mismo. Los costes se desglosarán en dos partes:

- Costes personales: se refiere a los costes de la mano de obra. El trabajo ha sido desarrollado por un ingeniero técnico en un total de 300 horas. Dado que se ha trabajado un total de 4 horas diarias (media jornada), el total de días utilizados para la elaboración del trabajo han sido 75, lo que supone un total de 3 meses (incluidos los sábados).
- Costes materiales: son los costes del material usado para la elaboración del trabajo. Se han utilizado un ordenador de altas prestaciones y un dispositivo móvil.

A continuación se expone el desglose del presupuesto:

- Costes personales:
 - Duración (días): 75
 - Horas diarias: 4
 - Dedicación del ingeniero (horas/mes): 100
 - Coste del ingeniero al mes (media jornada): 1.200 €

$$\text{Costes personales} = \frac{\text{duración días} \times \text{horas diarias}}{\text{dedicación ingeniero mes}} \times \text{coste ingeniero mes} = 3.600 \text{ €} \quad (1)$$

El coste personal del trabajo asciende a **TRES MIL SEISCIENTOS EUROS**.

- Costes de equipo (amortización del equipo):
 - Ordenador portátil MacBook Pro de 13' con un coste de 1.430 € sin IVA con un uso de 2 años (periodo de depreciación 60 meses).
 - Smartphone Samsung Galaxy Note 6 con un coste de 275€ sin IVA con un uso de 2 años (periodo de depreciación 60 meses).

$$\text{Amortización} = \frac{\text{Nº meses uso (total)}}{\text{periodo depreciación}} \times \text{coste equipos} \times \% \text{ uso en trabajo} = 682 \text{ €} \quad (2)$$

- Uso de licencias:
 - Microsoft Office 2016: 150 €

El coste material del trabajo asciende a **OCHOCIENTOS TREINTA Y DOS EUROS**.

Establecidos los costes personales y materiales de este trabajo, la siguiente tabla resume cada uno de ellos, estableciendo el coste total para el proyecto.

Descripción del Presupuesto (costes totales)	Presupuesto costes totales
Costes Personales	3600 €
Costes de amortización	682 €
Costes funcionales	150 €
Costes indirectos (20%)	886,40 €
Total sin IVA	2713,60 €
Total con IVA (21%)	3283,46 €

Tabla 85: Presupuesto

El presupuesto total por la realización del trabajo asciende a la cantidad de **TRES MIL DOSCIENTOS OCHENTA Y TRES EUROS CON CUARENTA Y SEIS CÉNTIMOS DE EURO.**

Jaén, a 31 de Julio de 2017

El ingeniero proyectista

FIRMA

Fdo. Pablo Cueto Siles

7 CONCLUSIONES Y LÍNEAS FUTURAS

En este punto se describirán las conclusiones que se han obtenido tras la realización de este trabajo. A continuación se hablará de las líneas futuras de este trabajo para nuevas mejoras y desarrollos.

7.1 Conclusiones

El desarrollo de este trabajo no ha sido tan fácil como se esperaba debido a que se han tenido que superar una serie de retos y ha sido necesario tomar diferentes decisiones para llegar a cumplir los objetivos que se habían propuesto. Lo único que se tenía claro desde el principio era que se quería desarrollar una aplicación web y una aplicación Android para una tienda en donde se pudiera comprar y vender productos nuevos y/o usados.

Con esto surgieron las dudas sobre qué plataformas de desarrollo serían las escogidas para dicho fin. Tras estudiar el mercado y analizar los puntos fuertes y débiles de las distintas plataformas de desarrollo, se optó por elegir WordPress para la aplicación Web y Android Studio para la aplicación Android. Esto originó el necesario estudio de ambas plataformas para ver cómo funcionaban y conocer sus estructuras ya que se trataba de software desconocido, aunque se tenía conocimientos previos en lenguajes de programación usados en ambas plataformas, lo que facilitó el aprendizaje.

A continuación vinieron dudas acerca de qué aplicación debía ser desarrollada en primer lugar. Después de un largo estudio sobre cuál sería el orden, se optó por desarrollar la aplicación Web junto con la base de datos y sobre ésta, desarrollar la aplicación Android.

A pesar de estos inconvenientes, posiblemente el mayor fue decidir qué servicios incorporar ya que existen tiendas de todo tipo y que ofrecen todos los que se tenían en mente. Se buscaba la intención de crear algo nuevo y novedoso, y tras un estudio comparando las tiendas más grandes e importantes en España y el mundo, se decidió que sería una tienda Marketplace, una tienda donde se pudieran hacer subastas y donde se pudieran hacer ofertas tipo trueque por algún producto en concreto. Además, se añadieron servicios secundarios para aumentar el nivel y para que se pudieran captar más clientes.

Con estas ideas se pensó que la mejor manera de empezar a desarrollar la aplicación Web era subiendo WordPress directamente a un hosting gratuito y empezar de cero para evitar problemas con las versiones de MySQL y PHP. Esto dificultó la tarea de desarrollo ya que el hosting ofrecido, aunque era el mejor de los gratuitos, proporcionaba

una velocidad reducida y por tanto se tardó más tiempo, pero al final se cumplió el objetivo, crear una tienda online funcional.

Una vez la aplicación Web se encontraba operativa y funcional, empezó el desarrollo de la aplicación Android. Aquí es donde se hizo un mayor esfuerzo ya que es donde se requerían más conocimientos de programación. Tras una larga búsqueda y un largo estudio sobre como programar aplicaciones Android en foros, tutoriales, videos, etc., se encontró la solución y a partir de ella se elaboró el trabajo.

Una vez se tenían las aplicaciones funcionales se llevaron a cabo pruebas de todo tipo combinando ambas y forzando a errores para corregirlos. Después de comprobar que todo estuviese funcional y sin errores, ya se habían alcanzado los objetivos de este trabajo.

Desde un punto de vista personal, ver como los conocimientos adquiridos de diferentes asignaturas cursadas en la universidad han permitido desarrollar unas aplicaciones totalmente funcionales, han supuesto una alta satisfacción personal. Poder comprobar cómo funcionan este tipo de aplicaciones, no solo en el aspecto técnico, sino en aspectos más externos como las leyes que se deben cumplir a la hora de lanzar este tipo de aplicaciones, la seguridad que deben llevar implantadas y deben cumplir, el coste que requiere mantenerlas a flote y que hay que desembolsar para ofrecer todo tipo de servicios, etc.

Desde el punto de vista del desarrollo, se ha podido ver un largo crecimiento de la tecnología en un plazo de tiempo muy corto. Se ha podido observar como el desarrollo de aplicaciones para dispositivos móviles va en aumento año tras año y se puede dar por entendido que esto seguirá así y aumentando. También se ha podido observar que el mercado de las aplicaciones es uno de los mercados más grandes del mundo y ahora, y en un futuro, las empresas demandarán programadores para desarrollar más y más aplicaciones.

Todo esto da qué pensar ya que se trata de un mercado joven. Podría ser factible en un futuro no muy lejano poder emprender y dedicarse a este sector que está en auge y con posibilidad de aumentar considerablemente.

7.2 Líneas futuras

Dado este trabajo, existen diversas mejoras y futuras líneas de trabajo, tanto en la mejora de la aplicación Web como en la mejora de la aplicación Android. A continuación se detallan posibles futuras líneas de trabajo y mejoras que serian interesantes a la hora de querer emprender.

- Con respecto a la aplicación web se propone:
 - Comprar licencias de los plugin de WordPress para dar los mejores servicios. El hecho de que se hayan usado plugins gratuitos para el desarrollo en este trabajo no significa que no se cumplan los objetivos.
Es de entender, que para poder tener todas las opciones y configuraciones posibles se necesiten comprar licencias.
- Con respecto a la seguridad:
 - Alojar los datos en un servidor privado (de pago) para, además de tener más espacio donde alojarlos, se tenga la mejor seguridad posible y sea fiable. En este trabajo, al tratarse de una tienda, se necesita la mejor seguridad entre las comunicaciones entre cliente y servidor. Ésta seguridad es proporcionada por el certificado SSL y sería necesario contratar un hosting privado para obtenerlo.
- Con respecto a la aplicación Android se propone:
 - Actualizar la aplicación cuando aparezcan las nuevas versiones del SO para que sea compatible y adaptarla a futuros dispositivos. También, si fuese necesario tras la actualización de la compatibilidad, otorgar permisos a la aplicación.

8 ANEXOS

8.1 Anexo 1: Como instalar un servidor local

Para comenzar a trabajar en tu proyecto online con WordPress, será necesario contar con un dominio y un servidor o hosting donde alojar los archivos. Si se trata de una página web muy extensa o el desarrollador no tiene los conocimientos necesarios para usar WordPress, lo recomendable es comenzar en un servidor de pruebas o servidor local ya que, si se contratara un hosting de pago, no se podría tomar todo el tiempo que se quisiera para realizar pruebas y se tendría que evitar hacer un mal SEO²¹ (Search Engine Optimization u Optimización en Motores de Búsqueda) que en el futuro se traducirá en errores 404.

Al instalar MAMP en un ordenador se asume el riesgo de que no funcione y que otros programas instalados den problemas si no se realiza la instalación correctamente. Es muy importante leer y entender las instrucciones que aparecen en la página web (<https://www.mamp.info/en/>) durante el proceso de instalación para su correcto funcionamiento.

Un servidor local es simplemente un ordenador con una serie de aplicaciones instaladas para que pueda ser utilizarlo como servidor de prueba. De esta manera se pueden corregir las imperfecciones y realizar los cambios necesarios en la página web hasta que se esté 100% seguro de que puede ser subida al servidor remoto definitivo.

Las aplicaciones necesarias para que un ordenador funcione como servidor local son:

- **Sistema operativo.** Generalmente el más utilizado es Windows pero se puede utilizar un ordenador Mac o Linux.
- **Apache.** Es la aplicación madre que permite que el ordenador se convierta en servidor.
- **MySQL.** Es una aplicación de bases de datos controlada por Apache que permite crear todas las bases de datos necesarias para nuestro trabajo.
- **PHP.** Es la tecnología de programación utilizada en el servidor.

²¹SEO: es el proceso técnico mediante el cual se realizan cambios en la estructura e información de una página web, con el objetivo de mejorar la visibilidad del sitio en los resultados obtenidos de los diferentes buscadores. Véase en: https://es.wikipedia.org/wiki/Posicionamiento_en_buscadores

Todas estas aplicaciones han sido desarrolladas después del lanzamiento de Linux y son de código abierto. Pueden ser instaladas por separado cada una de ellas aunque hoy en día existe un paquete que las engloba a todas.

Originalmente este paquete de instalación se creó bajo el nombre de LAMP (Linux, Apache, MySQL y PHP) aunque hoy en día ya se puede encontrar para Mac (MAMP) y para Windows (WAMP).

En este trabajo se usará la versión para Mac.

Lo primero que hay que hacer es descargar MAMP desde su página oficial <https://www.mamp.info/en/> e instalarlo.

Una vez instalado, el programa creará un icono de acceso directo en el Launchpad para que arranques el servidor cada vez que lo quieras utilizar. Al arrancar MAMP, se verá una pantalla con dos cuadrados que representa el servidor Apache y el servidor MySQL. Previa inicialización del servidor es recomendable cambiar los puertos

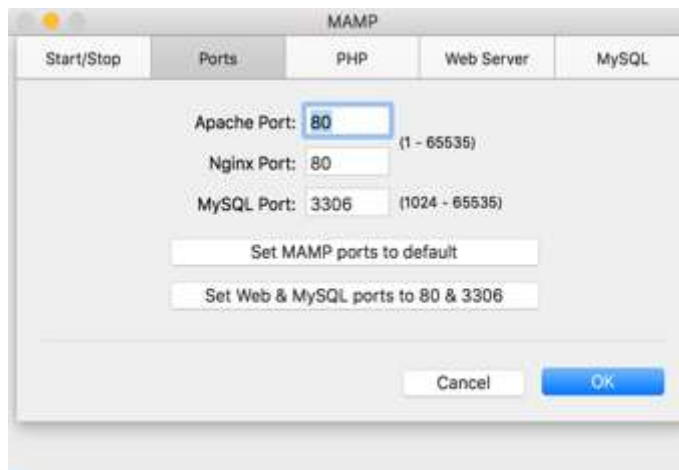


Ilustración 74: MAMP 1

que vienen por defecto a los puertos 80 y 3306 pulsando en el icono de Preferences....

Una vez cambiados los puertos y dejando la configuración por defecto de los demás campos, se puede iniciar el servidor.

Si los dos cuadrados del servidor Apache y MySQL se activan, quiere decir que el servidor está activo y funcionando.

En la pantalla principal de MAMP se podrá acceder a phpMyAdmin pulsando en el icono Open WebStart page.

A partir de aquí podremos realizar las pruebas necesarias en nuestro trabajo.

Bibliografía [\[Anexo 1\]](#)

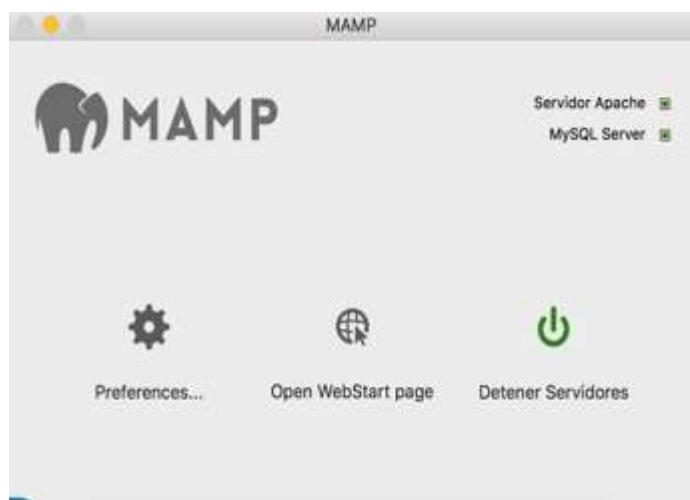


Ilustración 75: MAMP 2

8.2 Anexo 2:Legislación

En éste anexo se darán a conocer los requisitos legales que se deben cumplir a la hora de montar una tienda online, desde la adaptación a la LOPD (Ley Oficial de Protección de Datos) hasta los requisitos de la LSSI (Ley de Servicios de la Sociedad de la Información y de Comercio Electrónico), como el deber de información, la normativa de la contratación electrónica o la política de cookies, pasando por los artículos de la LOCM (Ley de Ordenación del Comercio Minorista) que afectan directamente a un comercio electrónico y los nuevos cambios implementados en la Ley de Consumidores.

Si la tienda online es complementaria a la tienda física, el dueño de dicha tienda, deberá darse de alta en un nuevo epígrafe del IAE (Impuesto de Actividades Económicas), ya que se estará comenzando una nueva actividad. Si, por el contrario, se va a comenzar la actividad desde cero, deberá cursar el alta censal en Hacienda con el modelo 036 y darse de alta como autónomo en la Seguridad Social.

Para darse de alta como una sociedad debe inscribirse en el Registro Mercantil y solicitar su CIF (Código de Identificación Fiscal) en un plazo máximo de 30 días desde el registro como sociedad.

- **LOCM:** En líneas generales, los requisitos legales para vender por Internet son los mismos que los de una tienda física, con la diferencia de que no es necesaria una licencia de apertura.

Sin embargo, las tiendas online deben cumplir una serie de condiciones legales específicas, ya que la relación entre proveedor y cliente se realizan sin la presencia física simultánea, por lo que los artículos que afectan especialmente a las tiendas virtuales son los correspondientes a las Ventas a Distancia y entre los que cabe destacar:

- Plazo de ejecución y pago: De no indicarse en la oferta el plazo de envío del pedido, la entrega deberá realizarse en un máximo de 30 días desde la celebración del contrato.
- Derecho de desistimiento: El comprador podrá desistir libremente del contrato, sin necesidad de alegar ninguna causa, dentro del plazo de 14 días contados desde la fecha de recepción del producto. El importe ha de ser devuelto en un plazo máximo de 14 días tras el desistimiento. En caso de que esta información no aparezca expresa durante la compra el plazo de devolución se amplía a un año.
- Pago mediante tarjeta de crédito: Cuando el importe de una compra sea cargado utilizando el número de una tarjeta de crédito sin que ésta hubiese sido presentada directamente o identificada electrónicamente, su titular podrá exigir la inmediata anulación del cargo y el reembolso se deberá

efectuar con la mayor brevedad. En el caso de que la compra hubiese sido efectivamente realizada por el titular de la tarjeta y éste hubiese exigido indebidamente la anulación, quedará obligado frente al vendedor al resarcimiento de los daños y perjuicios ocasionados como consecuencia de dicha anulación.

- **LSSI:** Regula las obligaciones a la hora de comprar por internet:
 - El deber de la información: Se aplica a empresas y profesionales que desarrollan una actividad económica de comercio electrónico por Internet y establece la necesidad de que plataforma de ecommerce albergue en un lugar visible y accesible a cualquier usuario los datos básicos del negocio, tales como:

- Nombre o denominación social y datos de contacto.
- Nº de inscripción del registro en el que esté inscrito el negocio.
- NIF

Además de:

- Precio de los productos, indicando si van incluidos los impuestos, gastos de envío, etc.
- Si se precisa o no de una autorización administrativa previa.
- Datos del Colegio profesional, número de colegiado, título académico y Estado de la UE en el que se expidió, con su homologación, si se tuviese y fuese necesario.

- Contratación online: Si se realizan contratos de carácter electrónico, se tendrá el deber de facilitar al cliente, información referente al proceso de contratación electrónica, en los instantes anterior y posterior a la celebración del contrato.

Así, de forma previa, se deberá incluir la siguiente información:

- Trámites que deben seguirse para contratar “on-line”.
- Si el documento electrónico del contrato se va a archivar y si este será accesible.
- Medios técnicos para identificar y corregir errores en la introducción de datos.
- Lengua o lenguas en que podrá formalizarse el contrato.
- Condiciones generales a las que, en su caso, se sujete el contrato.

De forma posterior a la formalización del contrato, se deberá:

- Confirmar que se ha recibido la aceptación de compra enviando una notificación de recibo mediante correo electrónico durante las 24 horas siguientes a la dirección facilitada por el usuario.

y/o

- Confirmar que se ha recibido la aceptación de compra por un medio equivalente al usado en la contratación, siempre que dicha confirmación pueda ser archivada por el usuario y de forma inmediata a su aceptación.
- Política de Cookies: Tras la reciente actualización de la LSSI se deberá, además, adaptar el sitio a las nuevas condiciones establecidas para el uso de cookies, es decir, aquellas porciones de información enviadas por un sitio web y almacenadas en el navegador del usuario que permite a dicho sitio web consultar la actividad previa del usuario. La ley de cookies es de aplicación a los prestadores de servicios de la sociedad de la información (tanto empresas como particulares que realizan actividades económicas a través de Internet) establecidos en España y a los servicios prestados por ellos.

Para ello, se deberá incluir un procedimiento de consentimiento informado previo a la utilización de cookies mediante una política de cookies en la que deberá incluir, por ejemplo, los tipos de cookies utilizadas por la plataforma de ecommerce y la forma de desactivarlas.

Es importante contar con un procedimiento, que se ejecute a través de un pop-up, una landing page o visible en la cabecera o en el footer de la web, en el que:

- Se informe de manera visible, accesible y sin necesidad de hacer scroll de uso y política de cookies del sitio.
 - Utilizar una fórmula en la que se recabe el consentimiento previo informado del usuario que visita la página antes de instalar cualquier archivo para recabar información.
 - Explicar qué es una cookie, el tipo de cookies que está utilizando el sitio web y la finalidad de las mismas.
 - Proveer a los usuarios de instrucciones sobre cómo deshabilitar las cookies desde los diferentes navegadores existentes.
 - Ofrecer referencias sobre lugares en los que conseguir más información.
- **LOPD:** Normalmente, cualquier negocio basado en comercio electrónico trata a diario con datos personales de sus clientes y, como cualquier empresa que

recabe datos de carácter personal, se estará obligado a cumplir con la LOPD. Por ello, se debe adaptar el negocio a los requisitos establecidos por la LOPD para garantizar la protección y buen tratamiento de datos de carácter personal

Los pasos para el proceso de implantación de la LOPD serán:

- Identificación de los ficheros que contengan datos de carácter personal (empleados, clientes, proveedores, etc.).
- Identificación del nivel de seguridad que se les aplica.
- Identificación del Administrador del Fichero.
- Elaboración del Documento de Seguridad.
- Formación al Responsable del Fichero.
- Información a los propietarios de los datos, sobre la existencia de los ficheros.
- Inscripción de los ficheros en el Registro de la Agencia Española de Protección de Datos.
- Elaborar una política de privacidad para nuestra empresa
- Contar con un formulario de recolección de datos que permita el consentimiento previo y expreso.

Cabe recordar que existen diferentes niveles de protección y tratamiento de datos. En el caso de un comercio electrónico, se deberá adaptar al nivel básico o al intermedio en el caso de almacenar la información relacionada con los datos bancarios.

- Nivel básico: datos identificativos, como el NIF, Nº de la Seguridad Social, nombre, apellidos, dirección, teléfono, firma, imagen, e-mail, nombre de usuario, número de tarjeta, matrícula, etc.
- Nivel medio: datos a cerca de infracciones administrativas o penales, solvencia o crédito, datos tributarios o de la Seguridad Social, datos de prestación servicios financieros, y datos referentes a la personalidad o comportamiento de las personas, como gustos, costumbres aficiones etc.

Debe recordarse, además, que la LSSI prohíbe de forma expresa el envío de correos electrónicos publicitarios no solicitados o expresamente consentidos.

- **Condiciones de uso:** Las condiciones del servicio recogen los derechos y obligaciones de los clientes y usuarios y deben ser aceptadas de forma previa y expresamente por el usuario antes de adquirir cualquier producto o servicio de la plataforma de ecommerce.

De la misma forma que los datos básicos de la empresa, deben colocarse en un lugar visible y de fácil acceso y estar redactadas de manera clara y concisa.

Estas condiciones deben recoger todos los aspectos que regulan el servicio ofrecido en el comercio electrónico:

- Normativa de uso del sitio.
 - Propiedad intelectual.
 - Condiciones de la compra.
 - Derechos del usuario.
 - Obligaciones del Usuario.
 - Formas de pago.
 - Política de Devoluciones.
 - Política de Privacidad.
- **Ley de consumidores y el comercio electrónico:** El comercio electrónico es uno de los sectores que mayor crecimiento ha experimentado en nuestro país en los últimos años. Es por ello, y con el objetivo de adaptarse a las últimas exigencias europeas, que en los últimos meses se han producido diferentes modificaciones en la Ley de Consumidores que se aplican especialmente a las tiendas online. Básicamente, cualquier tienda virtual debe cumplir los siguientes requisitos:
 - 1) Se deberá mostrar de forma clara e inequívoca el precio final del producto o servicio contratado antes de que se concluya la transacción y deberá ser aceptado expresamente por el consumidor. En caso de que el cliente no tenga acceso al precio definitivo desde el comienzo de la transacción, podrá recuperar la diferencia entre el coste inicial y el final. Además, con los bienes y servicios cuyo precio no puede calcularse de antemano o se hacen mediante presupuesto, se deberá informar de cuáles son los criterios para determinar dicho precio.
 - 2) Se amplía el plazo de devolución de productos entre los siguientes 7 y 14 días naturales. Además, el empresario habrá de cerciorarse de que el consumidor ha quedado suficientemente informado. En caso de que el consumidor no sea informado, el plazo podría ampliarse a 12 meses, contados desde la fecha de expiración del periodo inicial.
 - 3) Será obligatorio poner a disposición del comprador un formulario de desistimiento, común en toda Europa y que deberá ser facilitado junto con la información previa al contrato de compra.
 - 4) En el comercio electrónico, el comprador ha de ser debidamente informado, hasta el último paso de la transacción o proceso de compra, de que la aceptación de la oferta obliga al pago por su parte.
 - 5) El empresario o vendedor será el encargado de asumir los riesgos que pudiera sufrir el producto durante el transporte hasta que sea entregado al consumidor.

- 6) El vendedor no podrá cobrar un recargo sobre el precio del producto a los consumidores o clientes por pagar mediante tarjeta de crédito o cualquier otro medio de pago una cantidad superior de lo que cuesta ofrecer esos servicios de pago.
- **La llegada del RGPD (Reglamento General de Protección de Datos):** Todas las empresas y autónomos tendrán hasta el 25 de mayo de 2018 para adaptarse a la normativa europea, que supondrá mayor protección y privacidad para los datos de los clientes y un esfuerzo añadido por parte de todas los negocios, sin importar su tamaño, de cara a la regularización de su información privada.

Para el caso de la aplicación Android, a parte de los datos anteriormente mencionados, será necesario lo siguiente:

- Es importante cumplir con la Propiedad intelectual y Ley de marcas, al necesitar de una marca, nombre comercial o logo para la aplicación, se deberá registrar el sitio para poder protegerlo y explotarlo correctamente, y para que sean reconocidos los derechos de autor. Si es necesario colgar contenidos de terceros se deberá pedir autorización de los autores y tener en cuenta sus licencias.
- Al tratarse de programas informáticos o software, se deberá tener en cuenta la ley de patentes, ya que se puede registrar como patente para su explotación comercial y para que no sea plagiado.
- Menores de 14 años. Los problemas en esta materia se derivan especialmente con las aplicaciones que van dirigidas a un target de menores de 14 años, como juegos o aplicaciones educativas. Este colectivo es un grupo especialmente cuidado por la legislación de consumidores y usuarios. En tema de derechos de imagen como en protección de datos. También hay que tener muy presente que un problema de esta índole afectaría muy negativamente a la reputación de la aplicación si este tema se gestiona mal legalmente.
- Permisos. Al tratarse de aplicaciones que se instalan y ejecutan en dispositivos móviles, es importante ser claros y explícitos al solicitar permisos al usuario. Muchas veces la aplicación móvil va a necesitar acceder a los contactos de la agenda o a contenidos del móvil, ya sea por cuestión de pagos, cesión de datos o instalación de cookies o simplemente compartir contenidos.

El usuario ha de ser informado y deben ser validados por el mismo de forma sencilla y lo más clara posible antes de su instalación con la opción de cambiar de opinión o configuración en caso de que lo necesite el usuario de la aplicación.

Bibliografía [\[Anexo 2\]](#).

8.3 Anexo 3: Pasos para crear un proyecto en Android Studio

Para crear un nuevo proyecto en Android Studio, después de haberlo descargado de la página oficial (<https://developer.android.com/studio/index.html?hl=es-419>) e instalado correctamente, se deberán seguir los siguiente pasos:

- 1) Abrir Android Studio y pulsar sobre al botón ‘Start a new Android Studio project’.
- 2) Poner en el apartado ‘Application name’ el nombre de la aplicación. En el caso de este trabajo se llamará IdlaShop. Los demás campos se dejarán por defecto.



Ilustración 77: Proyecto Android Studio 2

- 4) En este paso se debe escoger el Activity que se necesite. Esto es básicamente la plantilla de inicio de la aplicación que se va a crear. Para este trabajo se ha escogido la Activity ‘Empty Activity’.

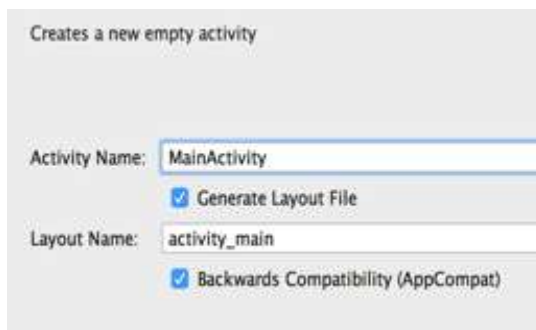


Ilustración 79: Proyecto Android Studio 5



Ilustración 76: Proyecto Android Studio 1

- 3) A continuación, se debe elegir el/los dispositivos para desarrollar la aplicación y la versión mínima de Android que deben soportar. En el caso para este trabajo se escogerá ‘Phone and Tablet’ y la versión Android 6.0 (Marshmallow).



Ilustración 78: Proyecto Android Studio 4

- 5) Los campos ‘Activity name’ y ‘Layout name’ se pueden modificar pero para este trabajo se han dejado por defecto.

8.4 Anexo 4: Firebase y como crear un proyecto

Firebase es una plataforma creada por Google. Su principal función es desarrollar y facilitar la creación de aplicaciones de alta calidad, con el fin de que aumenten los usuarios interesados en instalar la aplicación y poder ganar más dinero. La plataforma está subida en la nube y está disponible para diferentes plataformas como iOS, Android y web. Contiene diversas funciones para que cualquier desarrollador pueda combinar y adaptar la plataforma a medida de sus necesidades.

Sus principales características son las siguientes:

- **Desarrollo:** permite la creación de mejores aplicaciones, minimizando el tiempo de optimización y desarrollo, mediante diferentes funciones, entre las que destacan la detección de errores y de testeo, que supone poder dar un salto de calidad a la aplicación. Poder almacenar todo en la nube, testear la aplicación o poder configurarla de manera remota, son características destacables de la plataforma.
- **Analítica:** permite tener un control máximo del rendimiento de la aplicación mediante analíticas, todo desde un único panel y de forma gratuita. Los datos analíticos que facilita Firebase, facilitan la toma de decisiones basadas y fundamentadas en datos reales.
- **Poder de crecimiento:** permite gestionar de manera fácil todos los usuarios de las aplicaciones, con el añadido de que se pueden captar nuevos usuarios, mediante invitaciones o notificaciones.
- **Monetización:** mediante AdMob, Firebase permite que se pueda ganar dinero.
- **Rapidez:** implementar Firebase puede ser fácil y rápido, gracias a su API que es muy intuitiva, sostenida en un solo SDK. Con Firebase se pueden centrar los esfuerzos en resolver los problemas de los clientes y así evitar la pérdida de tiempo en la creación de una infraestructura más compleja.
- **Agilidad:** Firebase ofrece aplicaciones multiplataforma con unas APIs integradas a SDK individuales para iOS, Android y JavaScript, de tal forma que se pueden gestionar diferentes aplicaciones sin necesidad de salir de la propia plataforma.

El procedimiento para crear un proyecto en la plataforma ha sido el siguiente:

- 1) Como la plataforma Firebase pertenece a Google, basta con tener una cuenta de Google para poder utilizarla.
- 2) Una vez dentro de la página principal de Firebase se debe pinchar en la opción 'Ir a la consola' en la parte superior derecha y a continuación se creará un nuevo proyecto otorgándole un nombre y el país donde serán usados los servicios que elijamos para la aplicación.

- 3) Al trabajar con Android, se pinchará en la opción 'Añade Firebase a tu aplicación de Android'. El asistente pedirá el nombre del paquete de la aplicación. Éste será el que se eligió al crear el proyecto en Android Studio. En el caso de este trabajo será:

`com.example.pablocueto.idlashop`

También pedirá el certificado de firma de depuración SHA-1 pero se dejará en blanco por defecto.

- 4) Una vez introducidos los datos anteriores se descargará el archivo `google-service.json` que contendrá la información necesaria para la comunicación entre la aplicación y el servidor de Firebase. Éste archivo se deberá añadir en la carpeta 'app' del proyecto en Android Studio.

- 5) Después de añadir el archivo anterior, se deberán añadir las siguientes líneas de código en los archivos `build.gradle`:

- Archivo `build.grandle` del proyecto (en este caso se abrirá como `IdlaShop`) dentro de la sentencia 'dependencies':

`classpath 'com.google.gms:google-services:3.1.0'`

- Archivo `build.grandle` dentro de la carpeta `app` (se abrirá con el nombre de `app`) dentro la sentencia 'dependencies':

`compile 'com.google.firebase:firebase-messaging:10.2.6'`

`compile 'com.android.volley:volley:1.0.0'`

Fuera de la última llave del archivo:

`apply plugin: 'com.google.gms.google-services'`

Una vez añadidas estas líneas de código se pinchará sobre la pestaña amarilla en la opción 'Sync now'.

- 6) Completado estos pasos se habrá terminado la creación del proyecto en Firebase y solo faltaría la configuración del código en de la aplicación para implementar los servicios que se requieran.

Bibliografía [\[Anexo 4\]](#)

8.5 Anexo gráfico

Aquí se incluye la procedencia de las imágenes introducidas en este trabajo.

- Ilustración 1: Ejemplo de Web 1.0. Disponible en: <https://www.w3.org/2010/Talks/0921-html5-plh/web10.html>
- Ilustración 2: Web 3.0. Disponible en: <http://maureeninternet.blogspot.com.es>
- Ilustración 3: Uso CMS. Disponible en: <https://www.hiberus.com/crecemos-contigo/comparativa-entre-gestores-de-contenidos-cms/>
- Ilustración 4: Evolución de la telefonía móvil. Disponible en: <http://toyoutome.es/blog/la-evolucion-de-los-telefonos-moviles/5649>
- Ilustración 5: Evolución conectividad móvil. Disponible en: <http://omicro.no.espanol.com/2013/01/todo-sobre-el-4g-lte-que-es-para-que-sirve-y-cuando-llegara-a-espana/>
- Ilustración 6: Cuota de mercado SO móviles. Disponible en: <https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=8&qpcustomd=1>
- Ilustración 7: Android. Disponible en: <http://systope.blogspot.com.es/p/sistemas-operativos-moviles.html>
- Ilustración 8: iOS. Disponible en: <https://www.mediatrends.es/a/29630/siete-sistemas-operativos-moviles-alternativos-android-2015/>
- Ilustración 9: Windows Phone. Disponible en: <https://www.indalcasa.com/dispositivos-moviles/los-sistemas-operativos-moviles-para-smartphones-android-vs-ios-vs-symbian-vs-windows-phone/>
- Ilustración 10: BlackBerry. Disponible en: <http://iphoneandord.com/los-5-mejores-sistemas-operativos-para-celulares/>
- Ilustración 11: Nº de aplicaciones disponibles en tiendas móviles. Marzo 2017. Disponible en: https://elpais.com/tecnologia/2017/05/31/actualidad/1496242186_229624.html
- Ilustración 12: Amazon en el año 1995. Disponible en: <http://www.businessinsider.com/this-is-what-amazon-looked-like-in-1995-2013-10>
- Ilustración 13: Icono de WordPress. Disponible en: http://www.freepik.es/iconos-gratis/wordpress-pequeno-circulo-social_700866.htm
- Ilustraciones 14-15-16-17-18-19-20: Pantallazos propios.
- Ilustración 21: Diagrama ER de la base de datos de WordPress. Disponible en: <https://codex.wordpress.org/File:WP4.4.2-ERD.png>
- Ilustración 22: T-mobile G1, primer dispositivo Android. Disponible en: <http://www.knowyourmobile.com/products/7450/t-mobile-g1-preview>

- Ilustración 23: Arquitectura de Android. Disponible en: <http://www.androidestudio.com/2015/08/conceptos-y-generalidades-de-android.html>
- Ilustración 24-25-26-27-28-29-30-31-32-33-34: Versiones de Android. Disponibles en: <http://peru.com/epic/epic-users/android-mira-evolucion-logos-sistema-operativo-fotos-noticia-389332-1222226>
- Ilustración 35: Versión 7.0 Android Nougat. Disponible en: <http://androidemix.com/2016/08/algunos-usuarios-reportan-problemas-tras-actualizar-su-nexus-a-android-nougat/>
- Ilustración 36: Versión 8.0 Android O. Disponible en: <https://www.xataka.com/moviles/android-o-estas-son-todas-las-novedades-del-nuevo-sistema-operativo-de-google>
- Ilustración 37: Estadística versiones Android, Enero 2017. Disponible en: <https://tecnogeek.net/2017/01/byebye-froyo-no-va-mas-la-distribucion-android/>
- Ilustración 38: Ciclo de vida de una actividad Android. Disponible en: <http://www.androidcurso.com/index.php/tutoriales-android/37-unidad-6-multimedia-y-ciclo-de-vida/158-ciclo-de-vida-de-una-actividad>
- Ilustraciones desde la 39 hasta la 79: Pantallazos propios.

9 REFERENCIAS

Para la redacción de referencias bibliográficas y citas se ha seguido lo aplicado a la Norma [UNE-ISO 690:2013](#).

- [1] *La evolución de la web 1.0, 2.0 y 3.0*, Enfoque Comunicación, © 2015 [consulta: 15 julio 2017]. Disponible en: <http://www.enfoquecomunicacion.com/blog/la-evolucion-de-la-web-1-0-2-0-y-3-0>
- [2] *Los mejores CMS en 2016/2017: una comparativa de los gestores de contenido*, 1&1 Internet España S.L.U., © 2017 [consulta: 15 julio 2017]. Disponible en: <https://www.1and1.es/digitalguide/hosting/cms/cms-en-comparativa-los-gestores-de-contenido-mas-usados/>
- [3] *Sistema de gestión de contenidos historia y su evolución*, OC&C Agencia de marketing 360°, © 2017 [consulta: 15 julio 2017]. Disponible en: <https://cursodemarketingdigitalocc.es/sistemas-de-gestion-de-contenidos-historia-y-su-evolucion/>
- [4] *Comienzos de la telefonía móvil*, #MASblog, © 2017 [consulta: 15 julio 2017]. Disponible en: <http://blog.masmovil.es/conoce-como-ha-sido-la-evolucion-de-la-telefoniamovil/>
- [5] *Características y tabla comparativa de los sistemas operativos móviles más usados*, Blog sobre nuevas tecnologías, © 2017 [consulta: 15 julio 2017]. Disponible en: <http://jmacuna73.blogspot.com.es/2017/03/sistemas-operativos-moviles.html>
- [6] *Dispositivos móviles y sus sistemas operativos*, Aprendiendo sobre tecnología, [consulta: 15 julio 2017]. Disponible en: <https://aprendiendotecnologiaadmonb.wordpress.com/dispositivos-moviles-y-sus-sistemas-operativos/>
- [7] *Sistemas operativo móvil*, Wikipedia, Wikipedia® [consulta: 15 julio 2017]. Disponible en: https://es.wikipedia.org/wiki/Sistema_operativo_móvil
- [8] *Comparación de sistemas operativos móviles*, Ciencia-Edu, [consulta: 16 julio 2017]. Disponible en: <https://cienciaedu.wordpress.com/2012/09/20/comparacion-de-sistemas-operativos-moviles/>
- [9] *WINDOWS PHONE ventajas y desventajas*, Academia, Academia © 2017 [consulta: 17 julio 2017]. Disponible en: http://www.academia.edu/11840046/WINDOWS_PHONEventajas_y_desventajas
- [10] *Ventajas de BlackBerry*, BlackBerryEspañol. Copyright © 2014 [consulta: 17 julio 2017]. Disponible en: <http://www.blackberryesp.com.ve/2017/03/cuatro-desventajas-de-lostrabajadores.html>

- [11] *Tienda en línea*, Wikipedia, Wikipedia® [consulta: 17 julio 2017]. Disponible en: https://es.wikipedia.org/wiki/Tienda_en_l%C3%ADnea
- [12] *Historial de versiones Android*, Wikipedia, Wikipedia® [consulta: 17 julio 2017]. Disponible en: https://es.wikipedia.org/wiki/Anexo:Historial_de_versiones_de_Android
- [13] Tomás, Jesús; Alfaomega Grupo Editor, S.A. de C.V., México, julio 2012, *El gran libro de Android*, Segunda edición, ISBN: 978-84-267-1832-7, edición en español publicada por MARCOMBO, S A., Barcelona, España, Derechos reservados © MARCOMBO, S A.
- [14] *Ciclo de vida de una actividad*, Universidad Politécnica de Valencia, Master en Desarrollo de aplicaciones Android © 2017 [consulta: 17 julio 2017]. Disponible en: <http://www.androidcurso.com/index.php/tutoriales-android/37-unidad-6-multimedia-y-ciclo-de-vida/158-ciclo-de-vida-de-una-actividad>
- [15] *Características del gestor de contenidos WordPress*, Esfera Hosting, © Esfera Hosting [consulta: 18 julio 2017]. Disponible en: <https://www.esferahosting.com/wordpress>
- [16] *WordPress*, Wikipedia, Wikipedia® [consulta: 18 julio 2017]. Disponible en: https://es.wikipedia.org/wiki/WordPress#Caracter.C3.ADsticas_principales
- [17] Estructura básica de archivos y carpetas de WordPress, Raiola Network, [consulta: 18 julio 2017]. Disponible en: <https://raiolanetworks.es/blog/estructura-basica-archivos-carpetas-wordpress/>
- [18] *Instalar WordPress en un servidor local*, Gian Oliveira, © Gian Oliveira, [consulta: 18 julio 2017], Disponible en: <https://www.gianoliveira.com/instalar-wordpress-servidor-local.html>
- [19] *Ayuda WordPress*, Ayuda WordPress, © Ayuda WordPress, [consulta: 18 julio 2017]. Disponible en: <https://ayudawp.com/como-instalar-wordpress-paso-a-paso/>
- [20] *Database Description*, WordPress, [consulta: 18 julio 2017]. Disponible en: https://codex.wordpress.org/es:Database_Description
- [21] *Diagrama base de datos WordPress*, Powered by frikipandi.com, © [consulta: 18 julio 2017]. Disponible en: <http://www.frikipandi.com/public/post/diagrama-base-de-datos-de-wordpress/>
- [Anexo 1] *¿Cómo instalar un servidor local?*, Gian Oliveira, © Gian Oliveira, [consulta: 18 julio 2017], Disponible en: <https://www.gianoliveira.com/como-instalar-un-servidor-local.html>
- [Anexo 2] *Tecnología e Internet*, Infoautónomos, Copyright© 2010 - 2017 Infoautónomos Marca registrada por Infoautónomos 2009 S.L. [consulta: 22 julio 2017]. Disponible en: <http://infoautonomos.eleconomista.es/tecnologia-pymes-autonomos/requisitos-legales-para-una-tienda-online/>

Decálogo de buenas prácticas: Aspectos legales de las aplicaciones móviles, YeePLY, [consulta: 22 julio 2012]. Disponible en: <https://www.yeeply.com/blog/decalogo-de-buenas-practicas-aspectos-legales-de-las-aplicaciones-moviles/>

[Anexo 4] *Firebase, qué es y para que sirve*, Innovation & Entrepreneurship Bussines School ©2017 [consulta: 25 julio 2017]. Disponible es: <http://www.iebschool.com/blog/firebase-que-es-para-que-sirve-la-plataforma-desarrolladores-google-seo-sem/>

