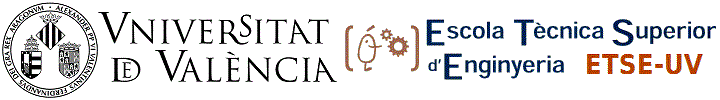
****

**Grau en Enginyeria Informàtica**

**Treball de Fi de Grau**

**<Títol del projecte>**

**autor:**

**Traver i Fos, Alejandro**

**tutora:**

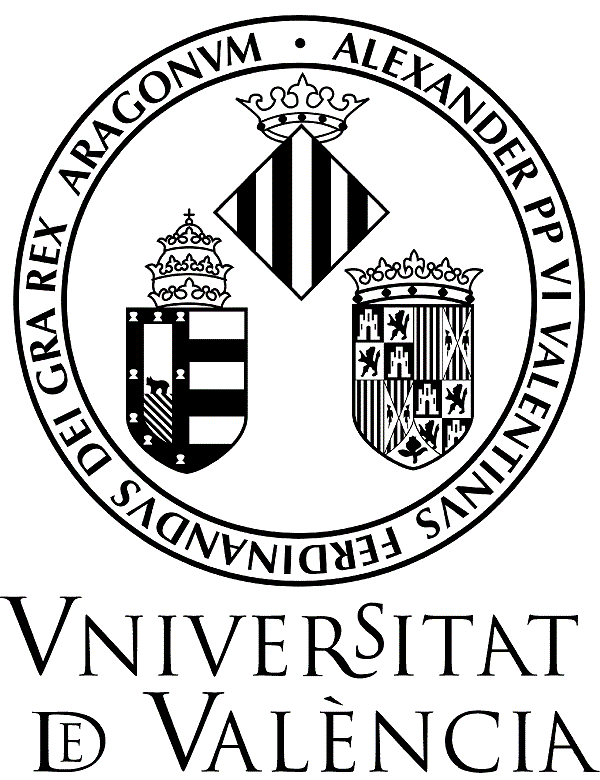
**Rueda Pascual, Silvia**

**Tribunal:**

|  |  |
| --- | --- |
| President: | Vocal 1: |
| Vocal 2: | **Data de defensa:**  **Qualificació:** |

****

**Grau en Enginyeria Informàtica**

****

**Treball de Fi de Grau**

**<Títol del projecte>**

**<autor/a>:**

**<nom i cognoms>**

**<tutor/a/s/es>:**

**<nom i cognoms>**

**<mes, any>**

Índice

[Capítulo 1. Introducción 7](#_Toc424676725)

[Capítulo 2. Motivación y objetivos 9](#_Toc424676726)

[Subcapítulo 1. Motivación 9](#_Toc424676727)

[Subcapítulo 2. Objetivos 9](#_Toc424676728)

[Capítulo 3. Estado del Arte 11](#_Toc424676729)

[Subcapítulo 3.1 Casos de estudio 11](#_Toc424676730)

[1. WhatsApp 11](#_Toc424676731)

[2. Telegram 12](#_Toc424676732)

[3. Comparativa 12](#_Toc424676733)

[Subcapítulo 3.2 Selección de la plataforma del lado del cliente 12](#_Toc424676734)

[1. iOs 13](#_Toc424676735)

[2. Android 13](#_Toc424676736)

[3. Multiplataforma 13](#_Toc424676737)

[4. Comparativa 15](#_Toc424676738)

[Subcapítulo 3.3 Plataforma del lado del Servidor 15](#_Toc424676739)

[1. Arquitectura del Sistema 15](#_Toc424676740)

[2. Lenguaje de programación 16](#_Toc424676741)

[3. Sistema de Gestión de Base de Datos 17](#_Toc424676742)

[4. Tecnologías de comunicación 18](#_Toc424676743)

[Subcapítulo 3.4 Seguridad 19](#_Toc424676744)

[Capítulo 4. Especificación 21](#_Toc424676745)

[Subcapítulo 4.1 Definición de requisitos de usuario 21](#_Toc424676746)

[1. Requisitos Funcionales 21](#_Toc424676747)

[2. Requisitos No Funcionales 23](#_Toc424676748)

[Subcapítulo 4.2 Especificación del Sistema 25](#_Toc424676749)

[Subcapítulo 4.3 Planificación y estimación de costes 26](#_Toc424676750)

[1. Descomposición del trabajo en tareas 26](#_Toc424676751)

[2. Identificación de hitos 26](#_Toc424676752)

[3. Planificación temporal 26](#_Toc424676753)

[4. Estimación de costes 26](#_Toc424676754)

[Subcapítulo 4.4 Viabilidad legal 26](#_Toc424676755)

[Subcapítulo 4.5 Viabilidad económica 26](#_Toc424676756)

[Capítulo 5 Desarrollo del Proyecto 27](#_Toc424676757)

[Subcapítulo 5.1 Análisis 27](#_Toc424676758)

[Subcapítulo 5.2 Diseño del Sistema 27](#_Toc424676759)

[Subcapítulo 5.3 Implementación del Sistema 27](#_Toc424676760)

[Capítulo 6 Pruebas y resultados 29](#_Toc424676761)

[Subcapítulo 6.1 Plan de Pruebas de Funcionalidad 29](#_Toc424676762)

[Subcapítulo 6.2 Plan de Pruebas de Usabilidad 29](#_Toc424676763)

[Subcapítulo 6.3 Plan de Pruebas de Rendimiento 29](#_Toc424676764)

[Capítulo 7 Conclusiones y trabajo futuro 31](#_Toc424676765)

[ANEXOS 33](#_Toc424676766)

[Bibliografía 35](#_Toc424676767)

[No hay ninguna fuente en el documento actual. 35](#_Toc424676768)

[Notas 36](#_Toc424676769)

**Figuras**

[Figura 1. Porcentaje de Uso de los diferentes Sistemas Operativos para Dispositivos Móviles. [i1] 4](file:///C:\Users\Alejandro\Google%20Drive\TFG\Memoria-TFG.docx#_Toc418617677)

[Figura 2. Modelo de Arquitectura Cliente/SGBD 6](#_Toc418617678)

[Figura 3. Modelo de Arquitectura Cliente/Servidor 6](#_Toc418617679)

[Figura 4. Modelo de capas. [i2] 7](#_Toc418617680)

# Capítulo 1. Introducción

La Sociedad en la que vivimos está sufriendo desde hace ya bastantes años una serie de profundos cambios sociales. Muchos refieren que nos dirigimos a la Sociedad de la Información aunque también se han acuñado términos como era tecnológica o aldea global. Esta evolución se ha producido gracias a la introducción de las nuevas tecnologías.

Hemos sido de testigos de una evolución tecnológica sin parangón, en la cual las tecnologías de la información y la Comunicación (TIC) han sido de las más desarrolladas. Los canales de los que dispone un ciudadano para comunicarse son numerosos; tenemos correo tradicional, la llamada telefónica, el envío de mensajes cortos SMS, el correo electrónico, cualquier servicio de mensajería instantánea (como Telegram o WhatsApp), una video llamada ... Casi todos disponibles desde un dispositivo móvil.

Sólo hemos de pensar en los dispositivos móviles que usábamos hace tan solo unos años para darnos cuenta del cambio. Nada que ver con los terminales actuales. El Smartphone ha supuesto una verdadera revolución en nuestra vida cotidiana y en la forma en la que hacemos las cosas.

La repercusión que tiene en nuestras vidas este pequeño dispositivo de tan solo unas pocas pulgadas es realmente grande, hasta el punto en el que lo usamos para realizar todo tipo de actividades. En la red se bromea al respecto.



Las estadísticas confirman esta realidad social, con una tasa de penetración del 66% [1], siendo la franja de edad que más lo usa la comprendida entre 16 y 24. Esta franja de edad comprende a gran parte de la población universitaria, formada por los estudiantes.

En el contexto de la Universidad de Valencia, los canales de comunicación existentes entre profesores y alumnos son el correo electrónico y el foro del que dispone cada asignatura. Estos canales carecen de la flexibilidad y la inmediatez que podemos encontrar en servicios de mensajería instantánea como Line o Telegram. Por otro lado estos servicios no se pueden usarse en el contexto de la Universidad, por revelar los datos personales a una empresa externa.

Sin embargo, si analizamos los medios de los que disponen los profesores y los alumnos para comunicarse la situación es completamente distinta. Si un alumno quiere resolver una duda ha de usar necesariamente el correo electrónico de la universidad o el foro de la asignatura. Como veremos más adelante, este medio tiene ciertos inconvenientes que no permiten una comunicación fluida.

# Capítulo 2. Motivación y objetivos

## Subcapítulo 1. Motivación

Este proyecto viene motivado por la necesidad que se ha identificado de mejorar la comunicación entre los alumnos y los profesores.

Recientemente la Universidad de Valencia ha desarrollado una aplicación para Android que permite acceder al Aula Virtual y al correo electrónico, entre otras funcionalidades. Aunque esta aplicación facilita y agiliza la comunicación sigue siendo limitada, heredando los inconvenientes que tiene el correo electrónico.

El Sistema agilizaría el aprendizaje y permitiría una difusión mediante el uso de Tutorías Virtuales, evitando así que los profesores tengan que contestar de forma individual a la misma cuestión realizada por diferentes alumnos.

Los potenciales usuarios del futuro Sistema, es decir, los profesores y alumnos de la Universidad de Valencia, han mostrado interés en la aplicación y han valorado de forma positiva su implantación.

## Subcapítulo 2. Objetivos

Vista la motivación que justifica la realización de este proyecto, resumimos en una frase el objetivo principal del mismo:

*“Desarrollar un sistema de mensajería instantánea exclusivo para profesores y alumnos que pertenecen a la Universidad de Valencia.”*

El sistema aspira a ser un Sistema que permita prescindir del correo electrónico en algunos de sus usos. Por ejemplo podría servir para dudas puntuales que los alumnas quieran hacer al profesor o para que un profesor resuelva una duda común a varios alumnos.

Para ello será necesario cumplir una serie de objetivos secundarios:

* Implementar todas las funcionalidades que permitan cumplir el objetivo principal.
* Lograr desarrollar un Sistema robusto que resulte ser un canal fiable de comunicación entre profesores y alumnos.
* Conseguir una buena difusión entre los potenciales usuarios para garantizar que se utilice. Si sus usuarios no la utilizan, no se puede cumplir el objetivo principal.
* Dotar a la aplicación de una interfaz amigable e intuitiva que permita usar la aplicación de forma fácil y cómoda.
* Comprobar que el Sistema cumple los requisitos de calidad y es mantenible, facilitando la agregación de nuevas funcionalidades en el futuro.

# Capítulo 3. Estado del Arte

Actualmente, cuando abordamos cualquier tipo de proyecto encontramos en el mercado Sistemas similares al que vamos a desarrollar o al menos, que contiene algunas funcionalidades de las que queremos dotar a nuestro Sistema.

En este Capítulo vamos a analizar las tecnologías disponibles y a seleccionar las que se adapten mejor a las necesidades que tengamos. Está dividido en 4 subcapítulos; primero se hablará de las aplicaciones comerciales existentes, posteriormente se hablará de la plataforma del lado del cliente y por último las tecnologías que formarán parte del Servidor. Adicionalmente se ha dedicado un capítulo a la forma en la que se pretende implementar mecanismos de seguridad.

## Subcapítulo 3.1 Casos de estudio

Antes de comenzar el estudio de las tecnologías disponibles que pueden usarse para el desarrollo de este proyecto, se van a analizar las aplicaciones que han tenido una buena acogida por parte de los usuarios para conocer las funcionalidades que ofrecen.

Es importante conocer las aplicaciones más utilizadas del mercado. Si han triunfado quiere decir que tienen características interesantes para los usuarios. Además, desde la perspectiva de la usabilidad también nos interesa analizarlas ya que puede ser parte de su éxito. En cualquier caso los potenciales usuarios de la aplicación que se va a desarrollar conocen estas aplicaciones y están acostumbrados a su interfaz.

Hay dos servicios de mensajería instantánea que me parecen interesantes para analizar sus funcionalidades: WhatsApp y Telegram.

### WhatsApp

Esta aplicación tiene sus orígenes ligados casi al nacimiento de los propios dispositivos móviles inteligentes. Se ha convertido en todo un fenómeno social a causa de su gran popularidad. Su interfaz sencilla y su alta disponibilidad han sido dos de los pilares sobre los que ha construido su éxito.

En sus orígenes era una aplicación con unas funcionalidades muy concretas y reducidas que ofrecía una alternativa gratuita al SMS(Short Messege Service) y MMS (Multimedia Messege Service), posteriormente pasó a costar 1$ anual. Permitía el envío y recepción de mensajes individuales además de los chats grupales. Actualmente ofrece más opciones:

* Versión Web
* Llamadas mediante la aplicación
* Ajustes sobre privacidad

#### Arquitectura

A nivel interno, WhatsApp utiliza un protocolo de la capa de Aplicación llamado XMPP (Extensible Messaging and Presence Protocol). Es un protocolo abierto y extensible que presenta una arquitectura Cliente/Servidor descentralizada. Hablare de este protocolo con más detalle en el contexto de las Tecnologías del lado del Servidor.

Los Servidores no almacenan el estado de las comunicaciones, sino que son los propios clientes los que mantienen sus SGBD.

### Telegram

Telegram es una aplicación creada hace unos años (especificar) para competir con Whatsapp. Pese a no haber tenido el éxito de aquélla es mucho más rica en funcionalidades. Desde el primer momento permitió las conversaciones cifradas (hay un concurso que ofrece 200.00 $ al que rompa el cifrado, que todavía no ha conseguido nadie).

Pero lo realmente interesante de este Sistema es que es de código abierto, por tanto podemos analizar su funcionamiento con cierta profundidad y aprender cómo implementa algunas de las funcionalidades de las que queremos dotar a nuestro Sistema.

### Comparativa

Ahora vamos a realizar una comparación de ambas aplicaciones para ver las características de cada una.

|  |  |  |
| --- | --- | --- |
|  | Telegram | WhatsApp |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Subcapítulo 3.2 Selección de la plataforma del lado del cliente

En este apartado voy a realizar un estudio sobre las plataformas disponibles para dispositivos móviles analizando las ventajas y desventajas que presenta cada una. Posteriormente seleccionaré una de las plataformas justificando las razones que me han llevado a decantarme por esa opción.

Actualmente hay dos Sistemas Operativos que incorporan el 94.8 % de los dispositivos móviles actualmente, como puede apreciarse en la Figura 1.

Figura 1. Cuota de Mercado de Sistemas Operativos Móviles en España

### iOs

El sistema iOs es el Sistema Operativo desarrollado por Apple Inc. que incorporan exclusivamente los dispositivos móviles (tablets y Smartphones) de la misma compañía. Es un Sistema Operativo que destaca por su interfaz fluida e intuitiva.

Desarrollar para este esta plataforma requiere una cuota anual de 99$.

### Android

El sistema Android es un Sistema Operativo desarrollado por Android Inc. bajo licencia GNU GPL.

Las aplicaciones desarrolladas para Android se desarrollan en Java, usando las librerías nativas del Sistema Operativo. Desarrollar para la plataforma no requiere más que un entorno de desarrollo apropiado y un emulador o dispositivo físico; para poder publicar aplicaciones en la tienda virtual del sistema es necesario crearse una cuenta de desarrollador. Esta cuenta tiene un coste de 25$ anuales.

### Multiplataforma

La alternativa a desarrollar una aplicación en el lenguaje nativo de una plataforma es desarrollar una aplicación que se pueda usar desde distintas plataformas. Vamos a analizar las ventajas y desventajas de desarrollar la aplicación como aplicación web y veremos las opciones que nos ofrece Cordova.

#### Apache Cordova

Apache Cordova es un Framework en el que se desarrollan las aplicaciones usando tecnologías Web (Javascript, HTML5 y CSS3) para después exportarla y compilarla usando los lenguajes y librerías nativas de cada plataforma. Es una herramienta muy potente si se desea desarrollar un determinado software para distintas plataformas.

El rendimiento que tienen las aplicaciones desarrolladas con este Framework tienen un rendimiento muy inferior a las desarrolladas en el lenguaje nativo de la plataforma.

#### Aplicación Web

Una Aplicación Web es un software que se ejecuta en el contexto de un navegador. Se suele considerar que son multiplataforma. Se ejecutan en el contexto de un navegador web (o browser). Por tanto los requisitos necesarios para ejecutarlas es disponer de conexión a internet y tener un navegador instalado.

La principal ventaja que presenta esta opción es que un único desarrollo adaptado para dispositivos móviles podría ejecutarse igualmente en un terminal Android, iOs u otro Sistema Operativo. Dependiendo del Servidor elegido de entre las opciones que hay en el mercado, podría suponer un coste extra en licencias si se opta por un desarrollo propietario. No obstante las opciones de software de código abierto son muy populares entre las cuales el líder indiscutible es Apache Server.

Cuando un cliente solicita una página a un servidor Web, se decarga gran cantidad de documentos (documentos HTML, Hojas de estilo, código JavaScript…). Esto no supone un inconveniente para los ordenadores, pero para un Smartphone puede resultar pesada. Asimismo, dado que las compañías suelen establecer un máximo de tráfico de descarga mensual, el tráfico generado por una aplicación Web puede suponer una merma importante en el gasto de dicha cuota.

El principal inconveniente de esta opción es que conlleva un tráfico de datos eleva del consumo de datos, muy importante a la hora de desarrollar aplicaciones para móviles. Una aplicación web requiere mucho más tráfico en consecuencia puede derivar en costes adicionales para el usuario o que supere el límite de datos que tenga con su operadora.

También cabe destacar que no es una buena opción desde el punto de vista de la usabilidad. Para acceder a una aplicación web, primero es necesario abrir un navegador; después escribir la URL de la aplicación a la que deseamos acceder; por último es necesario introducir el usuario y contraseña en el caso de no almacenar cookies. En contraposición a una aplicación, que solo tenemos que seleccionar la aplicación en el menú para abrirla y empezar a usarla.

### Comparativa

## Subcapítulo 3.3 Plataforma del lado del Servidor

### Arquitectura del Sistema

/\*

El patrón a seguir para este proyecto es un Cliente/Servidor clásico; la opción P2P no resulta práctica(...).

\*/

A la hora de seleccionar un modelo de aplicación se van a considerar dos posibilidades.

#### Arquitectura Cliente/SGBD

En este modelo

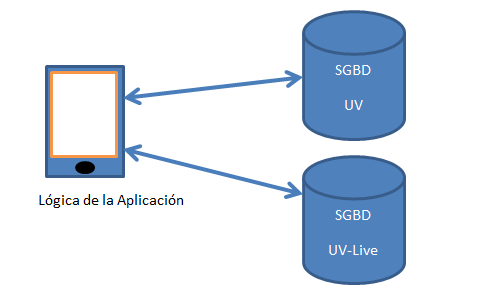


Figura 2. Modelo de Arquitectura Cliente/SGBD

#### Arquitectura Cliente/Servidor

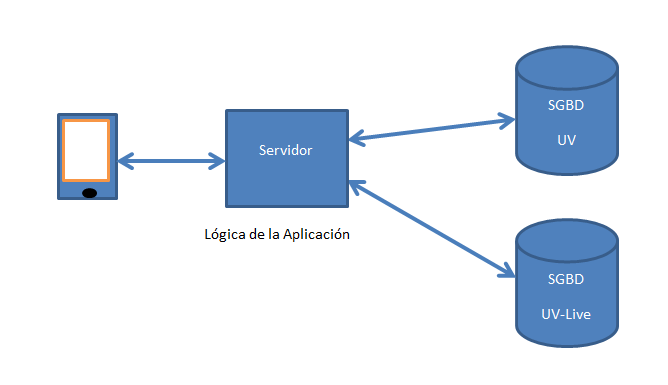


Figura 3. Modelo de Arquitectura Cliente/Servidor

### Lenguaje de programación

Vamos a seleccionar un lenguaje de programación con el que desarrollar nuestro Servidor. Para ello vamos a tener en cuenta la curva de aprendizaje.Dos de los lenguajes más extendidos, mejor documentados y que poseen más librerías son Java y PHP. Ambos son lenguajes que siguen el paradigma de Programación Orientada a Objetos.

#### Java

Es un lenguaje de programación que goza de gran popularidad. Está diseñado conforme el paradigma de programación orientado a objetos. Su principal característica es que el código es independiente de la plataforma sobre la que finalmente se ejecute. Esta característica se consigue compilando el código fuente a un código intermedio (llamado *bytecode*) que es interpretado por un intérprete, la JVM (Java Virtual Machine).

Recolector de basura

Eficiencia

Licencia

La mayor crítica que se le ha hecho a Java es la eficiencia. En los últimos años se ha trabajado para mejorar el funcionamiento de la JVM.

Librerías

Uno de los grandes beneficios que aporta Java es la API. En el núcleo del lenguaje están implementadas gran cantidad de funcionalidades, lo que permite una producción muy rápida, favoreciendo que el desarrollo se centre en la lógica de negocio y obviando los detalles de implementación.

Servlets

Es un lenguaje especialmente apto para la programación del lado del Servidor. Los Servlets son componentes que permiten gestionar las peticiones en el contexto del Servidor.

#### PHP

PHP es un lenguaje de programación interpretado utilizado para la generación de contenido Web dinámico.

Variable de sesión

Sintaxis sencilla

Al ser interpretado, es más difícil de depurar al no tener un compilador que detecte los errores de sintaxis.

Es más fácil introducir modificaciones al no tener que recompilar el código cada vez.

### Sistema de Gestión de Base de Datos

El Sistema de Gestión de Bases de Datos es uno de los elementos más importantes de la arquitectura de la aplicación.

#### MaríaDB

MaríaDB es un Sistema de Gestión de Bases de Datos basado en mySQL que se distribuye bajo licencia GPL.

/\*

Ventajas:

-sintaxis conocida

-

\*/

#### MongoDB

MongoDB es un Sistema de Gestión de Bases de Datos desarrollado en C++ bajo licencia GNU de código abierto. El nombre proviene de la palabra “humongous” que significa gigantesco. Su principal particularidad es que es no-SQL sino que implementa un lenguaje de manipulación propio.

Es un sistema orientado a documentos, por lo tanto no se definen estructuras de tablas como en las bases de datos relacionales sino que cada registro se almacena con su estructura propia y no necesariamente ha de repetirse.

Velocidad y potente sistema de consulta.

MongoDB se diseñó desde el principio para lograr escalabilidad horizontal. El Sistema de Gestión se encarga de equilibrar los datos entre los nodos y redistribuye los documentos de forma automática.

En mongoDB no existe el concepto de registro sino que se almacenan documentos que no tienen una estructura predefinida.En mongoDB -> un registro se denomina documento. -> Flexibilidad

Los docuementos se agrupan en colecciones ( como las tablas).

Se pueden crear índices para algunos atributos de los documentos.

### Tecnologías de comunicación

En esta sección vamos a hablar de las tecnologías disponibles para realizar la comunicación de la aplicación cliente con la aplicación servidor. Realizaré la selección de los protocolos de Internet que se adapten más a la aplicación.

#### HTTP vs Protocolo de Aplicación propio

HTTP es el protocolo de la capa de aplicación que se utiliza para ofrecer servicios Web. Utiliza como protocolo de transporte un protocolo confiable, como es TCP.

#### JSON

JSON es un formato ampliamente utilizado actualmente para el intercambio de información en el contexto de aplicaciones web. Ofrece la ventaja de que permite serializar objetos de Java para poder ser encapsulados, por ejemplo en peticiones o respuestas HTTP. Veamos un ejemplo.

Supongamos que deseamos codificar en formato JSON una clase Libro con los siguientes atributos:

class Libro {

private String titulo, editorial, isbn;

Libro( String t , String e , String isbn){

titulo=t;

editorial=e;

this.isbn=isbn;

}

ArrayList<Libro> libros = new ArrayList<Libro>();

libro.add(new Libro(“Don Quijote de la Mancha” , ”Libro Hobby Club, S.A.”, “ISBN 10: 8497364678”));

Se generaría una cadena de texto similar a esta:

{"libros":[  
    {"titulo":“Don Quijote de la Mancha”,"editorial":"Libro Hobby Club, S.A.",isbn:” ISBN 10: 8497364678”}   
]}

#### XMPP

Es un protocolo basado en el intercambio de XML, ideado para sistemas de mensajería instantánea y videoconferencia. Es utilizado en la implementación de los chats de redes sociales como son Facebook… (hay más)

## Subcapítulo 3.4 Seguridad

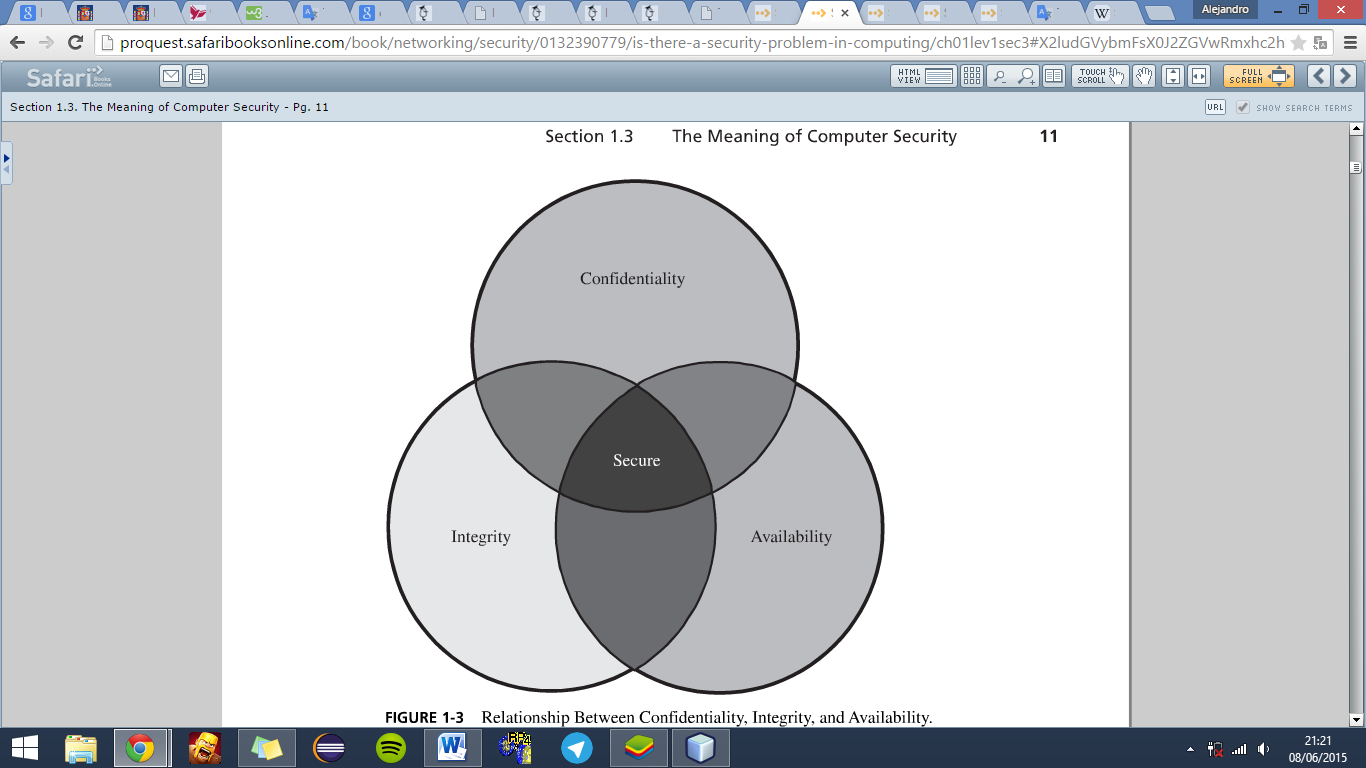
La aplicación fruto de este proyecto va a manipular datos sensibles de la Universidad de Valencia. /\*Por esta razón, es importante conocer e implementar los protocolos de seguridad que tenga la universidad.\*/

Cuando hablamos de seguridad no podemos caer en el error de asumir que un Sistema Informático es seguro o no como si fuera una variable booleana. Pfleeger nos da una definición.

“When we talk about computer security, we mean that we are addressing three important aspects of any computer-related system: confidentiality, integrity, and availability.”

En Seguridad Informática son 3 los requisitos de que deben cumplirse para considerar a un sistema seguro:

* Confidencialidad
  + Es la capacidad del Sistema de garantizar que solamente los usuarios autorizados pueden acceder a un determinado recurso o servicio.
* Integridad
  + Es la capacidad del Sistema de garantizar que los recursos solamente puedan ser modificados por usuarios autorizados.
* Disponibilidad
  + Es la característica del Sistema que determina que el Sistema se encuentra accesible para los usuarios autorizados.



Como se muestra en la figura, es importante cumplir simultáneamente estas tres propiedades, para considerar que el Sistema es seguro.

Para cumplir con los objetivos de seguridad, se van a tener en cuenta una serie de recomendaciones

El servidor va a estar alojado en la red de la Universidad de Valencia. La seguridad de la red y del nodo que alojará el Servidor no se encuentra dentro del alcance del proyecto, por tanto no se va a detallar la seguridad del nodo que alojará al Servidor ni la seguridad de la red.

/\* Protocolos de seguridad de la uv \*/

/\* Cifrado con HTTP y TLS. \*/

# Capítulo 4. Especificación

La especificación del Sistema aborda qué solución se va a llevar a cabo y cómo se va a implementar. Tras el estudio de las tecnologías disponibles que pueden aportar alguna característica o dar una solución a algún problema se va a definir el conjunto de tecnologías que se ha elegido.

## Subcapítulo 4.1 Definición de requisitos de usuario

El objetivo de esta sección es determinar el alcance del proyecto. Para ello vamos a definir los requisitos de usuario. Es importante delimitar perfectamente qué es lo que abarca el desarrollo del Sistema que se va a desarrollar, para evitar retrasos en fases más avanzadas.

En nuestro caso, dado que no existe un cliente con el que debamos determinar los requisitos, consideraremos un conjunto de requisitos que podrían ser interesantes para un cliente real que deseara implantar el Sistema.

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO** | **CÓDIGO REQUISITO:** a |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

### Requisitos Funcionales

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  Deberá existir un grupo de debate por cada asignatura, de forma que cada alumno y profesor tendrá un grupo por cada asignatura que imparta o asista. | **CÓDIGO REQUISITO:** RF1 |
| **TIPO REQUISITO:** Cliente |
| **IMPORTANCIA:** Esencial |
| **ORIGEN:** |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  Si el receptor de un mensaje está desconectado o no tiene conexión a Internet en el momento de recibir ese mensaje, deberá recibirlo cuando se conecte. | **CÓDIGO REQUISITO:** RF2 |
| **TIPO REQUISITO:** Cliente |
| **IMPORTANCIA:** Alta |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  Sólo se permitirán 3 intentos de acceso al Sistema cada 30 segundos para evitar ataques de fuerza bruta. | **CÓDIGO REQUISITO:** RF3 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  La función de moderar los grupos estará disponible para los profesores, que podrán bannear temporalmente a un alumno. | **CÓDIGO REQUISITO:** RF4 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  Al finalizar el curso se actualizarán automáticamente los datos del Sistema para para comenzar el nuevo curso. | **CÓDIGO REQUISITO:** RF5 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  Las credenciales necesarias para acceder al Sistema deberán ser las de la Universidad de Valencia que ya poseen sus usuarios. | **CÓDIGO REQUISITO:** RF6 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  Los comercios de campus podrán anunciar sus promociones. Deberán tener acceso al Sistema para gestionar los mensajes publicitarios. | **CÓDIGO REQUISITO:** RF7 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  Los profesores tendrán la opción de moderar los grupos de discusión, pudiendo inhabilitar temporalmente a un alumno si ha tenido algún comportamiento inadecuado. | **CÓDIGO REQUISITO:** RF8 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  Al finalizar cada curso se actualizará automáticamente la BD con los datos del nuevo curso. | **CÓDIGO REQUISITO:** RF9 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

### Requisitos No Funcionales

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  La aplicación deberá ser compatible con la versión de Android más utilizada actualmente. | **CÓDIGO REQUISITO:** RNF1 |
| **TIPO REQUISITO:** Accesibilidad |
| **IMPORTANCIA:** Alta |
| **ORIGEN:** |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  El ámbito de la aplicación se reduce a la comunidad formada por profesores y alumnos de la Universidad de Valencia. | **CÓDIGO REQUISITO:** RNF2 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  La aplicación deberá visualizarse correctamente en todo tipo de dispositivos, independientemente del tamaño de la pantalla. (diseño responsive) | **CÓDIGO REQUISITO:** RNF3 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  El Servidor se ejecutará sobre un entorno Linux. | **CÓDIGO REQUISITO:** RNF4 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  Se usará un Sistema de Gestión de Bases de Datos para proporcionar almacenamiento persistente. | **CÓDIGO REQUISITO:** RNF5 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  Una vez al día se realizará un guardado automático (dump) del estado de la Base de Datos | **CÓDIGO REQUISITO:** RNF6 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  La longitud del Overhead que se introduzca para el envío de los mensajes debe ser la menor posible. | **CÓDIGO REQUISITO:** RNF7 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  La ejecución de la aplicación debe ser lo más eficiente posible, para que tenga el menor impacto posible en la duración de la batería del dispositivo. | **CÓDIGO REQUISITO:** RNF8 |
| **TIPO REQUISITO:** b |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

|  |  |
| --- | --- |
| **DESCRIPCIÓN REQUISITO**  El Sistema debe cumplir con la Ley de Protección de Datos (LOPD). | **CÓDIGO REQUISITO:** RNF9 |
| **TIPO REQUISITO:** Legal |
| **IMPORTANCIA:** c |
| **ORIGEN:** d |

## Subcapítulo 4.2 Especificación del Sistema

Definidas las características que debe poseer el Sistema vamos a presentar y detallar cada aspecto de la solución que se va a implementar.

* La arquitectura de la aplicación va a ser un esquema cliente/servidor centralizado. [imagen de estado del arte explicando]
* Para el intercambio de mensajes me he decantado por XMPP…
* El Sistema de Gestión de Bases de Datos que se va a utilizar es mySQL
* Definitivamente se va implementar una aplicación Android..

/\*

* Arquitectura de la aplicación
* Tecnología de intercambio de mensajes
* Lenguaje de programación del lado del Servidor
* Sistema de Gestión de Bases de Datos
* Tipo de aplicación que se va a implementar definitivamente (android, web)
* Prototipo de la interfaz
* Implantación del Sistema -> comunicación con otros Sistemas

\*/

## Subcapítulo 4.3 Planificación y estimación de costes

### Descomposición del trabajo en tareas

1. Inicio
2. Estudio previo
3. Planificación
4. Implementación
   1. 1ª Iteración (envío de un mensaje y login)
      1. Análisis
      2. Diseño
      3. Implementación
      4. Pruebas
   2. 2ª Iteración (listar conversaciones e iniciar)
   3. 3ª Iteración (backUp y bloqueo)
   4. 4ª Iteración (actualización y comerciantes)
5. Integración

### Identificación de hitos

1. Hito A) Aceptación del Inicio
2. Hito B) Aceptación del Estudio previo
3. Hito C) Aceptación de la Planificación
4. Hito D) Aceptación del primer ciclo de desarrollo
5. Hito E) Aceptación de la Implementación

### Planificación temporal

/\*Gantt\*/

### Estimación de costes

## Subcapítulo 4.4 Viabilidad legal

/\* LOPD \*/

## Subcapítulo 4.5 Viabilidad económica

# Capítulo 5 Desarrollo del Proyecto

## Subcapítulo 5.1 Análisis

## Subcapítulo 5.2 Diseño del Sistema

## Subcapítulo 5.3 Implementación del Sistema

**Subcapítulo 5.4 Contratos**

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Login | | |
| Endpoint | {{environment}}/user/login | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| userName | String | Si | Nombre de usuario con el que se trata de acceder |
| password | String | Si | Clave con la que se trata de acceder |
| loginType | String | Si | Rol con el que se quiere acceder |
| pushToken | String | No | Token para recibir mensajes y notificaciones |
| Modelo de salida | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| token | String | Si | Token único para identificarse en futuras peticiones |

|  |  |
| --- | --- |
| Operación | Logout |
| Endpoint | {{environment}}/user/logout |
| Método HTTP | POST |
| Modelo de entrada | |
| Vacío | |
| Modelo de salida | |
| Vacío | |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Get Conversations | | |
| Endpoint | {{environment}}/rolUV/conversations | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Vacío | | | |
| Modelo de salida | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| Conversations | List<Conversation> | No | Listado de conversaciones de las que dispone el usuario |
| Conversation | Conversation | No | Modelo de conversación |
| idConversation | int | Si | Identificador la conversación |
| name | String | Si | Nombre de la conversación |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Init Conversation | | |
| Endpoint | {{environment}}/rolUV/conversations/init | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| idUser | int | Si | Identificador del usuario con el que se quiere iniciar la conversación |
| Modelo de salida | | | |
| Vacío | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Get Users | | |
| Endpoint | {{environment}}/rolUV/users | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Vacío | | | |
| Modelo de salida | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| Users | List<User> | No | Listado de usuarios con los que se puede iniciar una conversación |
| User | User | Si | Modelo que representa a un usuario |
| userId | int | Si | Identificador único de usuario |
| firstname | String | Si | Nombre del usuario |
| lastname | String | Si | Apellido del usuario |
| username | String | Si | Nombre de usuario |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Get messages | | |
| Endpoint | {{environment}}/rolUV/messages | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| idConversacion | int | Si | Identificador único de conversación a la que pertenecen los mensajes solicitados |
| Modelo de salida | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| messages | List<Message> | No | Token único para identificarse en futuras peticiones |
| message | Message | Si | Mesaje modelo |
| idMessage | int | Si | Identificador único del mensaje |
| text | String | Si | Texto del mensaje |
| timestamp | int | Si | Marca de tiempo del mensaje |
| owner | String | Si | Nombre del usuario que ha enviado el mensaje |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Get previous messages | | |
| Endpoint | {{environment}}/rolUV/messages/previous | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| idConversacion | int | Si | Identificador único de conversación a la que pertenecen los mensajes solicitados |
| timestamp | int | Si | timestamp del último mensaje del que se dispone |
| Modelo de salida | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| messages | List<Message> | No | Token único para identificarse en futuras peticiones |
| message | Message | Si | Mesaje modelado |
| idMessage | int | Si | Identificador único del mensaje |
| text | String | Si | Texto del mensaje |
| timestamp | int | Si | Marca de tiempo del mensaje |
| owner | String | Si | Nombre del usuario que ha enviado el mensaje |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Get following messages | | |
| Endpoint | {{environment}}/rolUV/messages/following | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| idConversacion | int | Si | Identificador único de conversación a la que pertenecen los mensajes solicitados |
| timestamp | int | Si | timestamp del último mensaje del que se dispone |
| Modelo de salida | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| messages | List<Message> | No | Token único para identificarse en futuras peticiones |
| message | Message | Si | Mesaje modelado |
| idMessage | int | Si | Identificador único del mensaje |
| text | String | Si | Texto del mensaje |
| timestamp | int | Si | Marca de tiempo del mensaje |
| owner | String | Si | Nombre del usuario que ha enviado el mensaje |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Send message | | |
| Endpoint | {{environment}}/message/send | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| idConversacion | int | Si | Identificador único de conversación a la que pertenecen los mensajes solicitados |
| message | String | Si | Texto del mensaje que se está enviando |
| Modelo de salida | | | |
| Vacío | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Update push token | | |
| Endpoint | {{environment}}/rolUV/pushToken/update | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| pushToken | String | Si | Token para poder recibir notificaciones push |
| Modelo de salida | | | |
| Vacío | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Get logs | | |
| Endpoint | {{environment}}/admin/logs | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Vacío | | | |
| Modelo de salida | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| Logs | List<Log> | Si | Listado de Logs |
| log | Log | Si | Cada uno de los logs del listado |
| timestamp | String | Si | Marca de tiempo del momento exacto del log |
| level | String | Si | Tipo de log |
| clazz | String | Si | Clase en la que se ha producido el evento |
| message | String | Si | Message asociado al log |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Validate merchant username | | |
| Endpoint | {{environment}}/admin/merchantName/exists | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| userName | String | Si | Nombre de usuario a validar |
| Modelo de salida | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| status | Boolean | Si | True si existe ya el nombre del usuario proporcionado |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Register merchant | | |
| Endpoint | {{environment}}/admin/merchant/register | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| dni | String | Si | DNI del comerciante |
| firstName | String | Si | Nombre del comerciante |
| lastName | String | Si | Apellido del comerciante |
| userName | String | Si | Nombre de usuario del comerciante |
| password | String | Si | Contraseña del comerciante |
| Modelo de salida | | | |
| Vacío | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Update merchant | | |
| Endpoint | {{environment}}/admin/merchant/update | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| dni | String | Si | DNI del comerciante |
| firstName | String | Si | Nombre del comerciante |
| lastName | String | Si | Apellido del comerciante |
| userName | String | Si | Nombre de usuario del comerciante |
| password | String | Si | Contraseña del comerciante |
| Modelo de salida | | | |
| Vacío | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Register broadcast | | |
| Endpoint | {{environment}}/merchant/broadcast/register | | |
| Método HTTP | POST | | |
| Modelo de entrada | | | |
| Parámetro | Tipo | Obligatorio | Descripción |
| title | String | Si | Título del anuncio enviado en la notificación |
| broadcastMessage | String | Si | Mensaje del anuncio |
| Modelo de salida | | | |
| Vacío | | | |

# Capítulo 6 Pruebas y resultados

El Plan de Pruebas se va a componer de un Plan de Pruebas de Usabilidad y de un Plan de Pruebas de Rendimiento. Hay aspectos que son muy importantes cuando desarrollamos una aplicación para dispositivos móviles y que debemos valorar y cuantificar. El consumo de la batería, el uso de la memoria y el volumen del tráfico generado son aspectos, que de no cuidarlos pueden provocar el usuario no utilice la aplicación.

## Subcapítulo 6.1 Plan de Pruebas de Funcionalidad

## Subcapítulo 6.2 Plan de Pruebas de Usabilidad

Para estas pruebas se ha decidido realizar la evaluación mediante tests. Es un método sencillo que permite generar fácilmente gráficas sobre la satisfacción sobre cada aspecto que se pretende evaluar. Concretamente se utilizará un test de Medida de prestaciones, en el cual usuarios reales realizan tareas reales. En este caso las tareas se realizarán en un entorno de pruebas ya que forma parte de este proyecto la implantación del Sistema.

## Subcapítulo 6.3 Plan de Pruebas de Rendimiento

Vamos a realizar diversas pruebas para cuantificar el consumo que arroja la aplicación sobre el terminal de pruebas. Los aspectos que se van a evaluar son: el consumo de batería, el uso de la memoria y el uso de Red. Para comparar y poder asegurar que se encuentran en unos valores comedidos, se realizarán las mismas pruebas sobre aplicaciones del mercado de las mismas características, esperando estar en una horquilla aceptable.

# Capítulo 7 Conclusiones y trabajo futuro

# ANEXOS

# Bibliografía

## No hay ninguna fuente en el documento actual.

**Imágenes**

[i1] <http://www.actualidadiphone.com/cuales-han-sido-las-apps-para-smartphone-mas-usadas-en-el-2014> [04-05-2015]

[i2] <http://www.webanswers.com/post-images/1/19/1350D05B-12B1-FC60-E07213BA65654642.jpg>

<http://2.bp.blogspot.com/-49iztrPVbOA/UNxAMsfO2yI/AAAAAAAAAJM/-kFS05CZ9P0/s1600/esclavos-movil.jpg>

**Referencias**

[1] <http://www.20minutos.es/noticia/1900266/0/espana-lidera/uso-smartphones/66-penetracion/> [03/09/2014]

<http://es.wikipedia.org/wiki/IOS> [04-05-2015]

<http://es.wikipedia.org/wiki/Android> [04-05-2015]

<http://www.paradigmatecnologico.com/blog/por-que-phonegap-va-lento-2/> [04-05-2015]

<http://es.kantar.com/tech/m%C3%B3vil/2015/marzo-2015-cuota-de-mercado-de-sistemas-operativos-en-espa%C3%B1a-hasta-enero/> [06-05-2015]

<http://saladeprensa.telefonica.com/jsp/base.jsp?contenido=/jsp/notasdeprensa/notadetalle.jsp&id=0&idm=es&pais=1&elem=21187> [05-06-2015]

<http://es.wikipedia.org/wiki/MongoDB#Replicaci.C3.B3n> [09-06-2015]

<http://es.wikipedia.org/wiki/MongoDB#Replicaci.C3.B3n> [09-06-2015]

<http://www.uv.es/~montanan/redes/trabajos/WhatsApp.pdf> [12-06-2015]

Enlaces de interés

<http://es.wikipedia.org/wiki/Micron%C3%BAcleo>

<http://es.wikipedia.org/wiki/N%C3%BAcleo_monol%C3%ADtico>

<http://es.wikipedia.org/wiki/Aplicaci%C3%B3n_web>

<http://es.wikipedia.org/wiki/N%C3%BAcleo_(inform%C3%A1tica)>

<http://es.wikipedia.org/wiki/Exon%C3%BAcleo>

<http://es.wikipedia.org/wiki/PhoneGap>

<http://dominickm.com/?p=144>

<http://www.paradigmatecnologico.com/blog/por-que-phonegap-va-lento-2/>

<http://en.wikipedia.org/wiki/Cross-platform>

# Notas

* comentar la LOPD y qué es necesario para cumplir con ella