



Universidad  
Carlos III de Madrid

Grado en Ingeniería de Sistemas Audiovisuales

TRABAJO FIN DE GRADO

DESARROLLO DE UN  
BIBLIOTECARIO VIRTUAL  
PARA DISPOSITIVOS  
MÓVILES ANDROID

Autor: Marta Castro Llorente

Tutor: David Griol Barres

Leganés, septiembre de 2014



Título: DESARROLLO DE UN BIBLIOTECARIO VIRTUAL PARA DISPOSITIVOS ANDROID  
Autor: Marta Castro Llorente  
Director: David Griol Barres

## EL TRIBUNAL

Presidente: \_\_\_\_\_

Vocal: \_\_\_\_\_

Secretario: \_\_\_\_\_

Realizado el acto de defensa y lectura del Trabajo Fin de Grado el día \_\_ de \_\_\_\_\_ de 2014 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE



# Agradecimientos

En primer lugar quiero agradecer a mi tutor, David Griol, el brindarme la gran oportunidad de poder realizar este proyecto y contar con su apoyo y ánimos constantes. Agradecerle también la ayuda que me ha ofrecido durante el desarrollo de este proyecto y su dedicación. Gracias a esto he desarrollado este proyecto con una inmensa ilusión y optimismo.

También quiero expresar mi gratitud a toda mi familia, especialmente a mis padres y hermanos que siempre han estado pendientes de mi vida académica y nunca han dejado de apoyarme y decirme que podía.

Agradecer también a mis compañeros de la Universidad, particularmente a aquellos que han estado conmigo desde el principio de la carrera, tanto en ámbitos universitarios como personales, gracias a ellos realizar, esta carrera y este proyecto, ha sido una aventura espectacular. Gracias a vosotros, a vuestra comprensión y apoyo constante, habéis creado en mí un deseo por realizar este proyecto.

De nuevo, gracias a todos.



# Resumen

Actualmente, el uso de teléfonos móviles inteligentes está en continuo crecimiento debido a los numerosos avances tecnológicos del sector, que permiten traducir en una aplicación cualquier tarea cotidiana.

El presente Trabajo Fin de Grado se desarrolla para facilitar al usuario la gestión de consultas relativas a un servicio de biblioteca virtual, con la ventaja de ser una aplicación para dispositivos móviles basada en la interacción multimodal mediante la voz y los interfaces tradicionales (pantalla y teclado virtual).

Las consultas que resuelve el asistente multimodal son las relativas a la normativa de regulación de la biblioteca, la ubicación física de libros y distribución de materias, la búsqueda on-line de libros y materias, así como la disponibilidad y la posibilidad de reservarlos.

La aplicación usa otras tecnologías adicionales que se detallan en el contenido del presente Trabajo Fin de Grado, tales como el uso de bases de datos, el uso de librerías de Google, el reconocimiento automático del habla y el conversor de texto a voz. Para el desarrollo se ha llevado a cabo el uso de diferentes lenguajes de programación y servicios propios de Android.

El desarrollo de esta aplicación permite reducir el tiempo requerido para completar las operaciones descritas y conocer el estado de los diferentes servicios de la biblioteca, todo ello mediante la combinación de la voz e interfaces intuitivas para adecuarse a las preferencias y ubicación del usuario.

## **Palabras clave:**

Dispositivos móviles, Android, Sistemas de diálogo, Bibliotecas digitales, Asistentes multimodales, Reconocimiento automático del habla y Síntesis de Texto a Voz.





# Abstract

Currently the use of smart phones is continuously growing due to the many technological advances and industry needs that allow translating into application any everyday task.

This Final Project is developed to provide the required information in a library to the user. Consultations are managed by a virtual library service, with the advantage of a mobile application based on voice and interfaces.

The questions solved by the multimodal assistant are the rules relating to the regulation of the library, the physical location of books and distribution of materials, online search for books and materials, availability, and a pre-order.

Additional technologies used in the application are detailed in the content of this Final Project, such as the use of databases, automatic speech recognition and text to speech converter. Different programming languages and services of Android are used for the development and debugging.

The search time for books as well as the status of the services of the library is provided in order to help the user using their own voice or simple interfaces.

## **Keywords:**

Mobile devices, Android, Dialogue systems, Spoken dialog systems, Digital Libraries Data bases mobile, Operative systems, Multimodal assistant, Automatic Speech Recognition and Text-To-Speech.



# Índice general

## Contenido

<b>1. INTRODUCCIÓN Y OBJETIVOS .....</b>	<b>1</b>
1.1 Introducción .....	1
1.2 Objetivos del TFG .....	4
1.3 Planificación temporal .....	5
1.4 Medios empleados.....	9
1.5 Estructura de la memoria .....	9
<b>2. ESTADO DEL ARTE .....</b>	<b>12</b>
2.1 Introducción .....	13
2.2 Sistemas de diálogo.....	14
2.2.1 <i>Arquitectura y acciones básicas de un sistema de diálogo</i> .....	17
2.2.2 <i>Arquitectura de los sistemas de diálogo hablado</i> .....	18
2.2.3 <i>Reconocimiento Automático del habla</i> .....	19
2.2.4 <i>Síntesis de texto a voz</i> .....	20
2.2.5 <i>Sistemas de diálogo multimodal</i> .....	24
2.2.6 <i>Aplicaciones para dispositivos móviles con ASR y TTS</i> .....	25
2.3 Sistemas operativos móviles .....	27
2.3.1 <i>Sistemas operativos móviles más destacados</i> .....	28
2.4 Elección de la plataforma para el trabajo .....	32
2.5 Descripción detallada de la plataforma de trabajo usada: Android.....	33
<b>3. DESCRIPCIÓN GENERAL DEL SISTEMA .....</b>	<b>42</b>
3.1 Presentación del sistema .....	42
3.2 Herramientas usadas.....	44
3.2.1 <i>Eclipse y SDK Android</i> .....	45
3.2.2 <i>MySQL y PHP. Almacenamiento de datos en el módulo palabras clave y libros</i> .....	45
3.3 Recursos necesarios para el sistema.....	59
3.3.1 <i>Reconocimiento automático del habla</i> .....	59
3.3.2 <i>Síntesis de texto a voz</i> .....	60
3.3.3 <i>AsyncTask para el desarrollo de tareas en segundo plano</i> .....	60
<b>4. DESCRIPCIÓN DETALLADA DE LOS MÓDULOS DEL SISTEMA.....</b>	<b>63</b>
4.1 Módulo Inicio .....	63
4.1.1 <i>Funcionalidad</i> .....	63
4.1.2 <i>Flujo de datos</i> .....	66
4.2 Módulo Asistente por voz .....	67
4.2.1 <i>Funcionalidad</i> .....	67
4.2.2 <i>Flujo de datos</i> .....	67
4.3 Módulo Búsqueda de palabras clave .....	69
4.3.1 <i>Funcionalidad</i> .....	69
4.3.2 <i>Flujo de datos</i> .....	69
4.4 Módulo Búsqueda de libros.....	72
4.4.1 <i>Funcionalidad</i> .....	72
4.4.2 <i>Flujo de datos</i> .....	72
4.5 Módulo Consulta página web.....	77
4.5.1 <i>Funcionalidad</i> .....	77
4.5.2 <i>Flujo de datos</i> .....	77
4.6 Módulo Indicaciones-plano .....	78

4.6.1 <i>Funcionalidad</i> .....	78
4.6.2 <i>Arquitectura</i> .....	78
<b>5. EVALUACIÓN DE LA APLICACIÓN .....</b>	<b>84</b>
5.1 Metodología de evaluación .....	84
5.2 Resultados de la evaluación .....	84
<b>6. CONCLUSIONES, TRABAJO FUTURO Y PRESUPUESTO .....</b>	<b>94</b>
6.1 Conclusiones .....	94
6.2 Trabajo futuro.....	97
6.3 Presupuesto.....	97
Glosario .....	100
Bibliografía.....	104

# Índice de Figuras

FIGURA 1. SISTEMAS OPERATIVOS PARA DISPOSITIVOS MÓVILES .....	3
FIGURA 2. VERSIONES MÁS USADAS DE LOS DISTINTOS SISTEMAS OPERATIVOS MÓVILES ....	3
FIGURA 3. ESQUEMA DEL DESARROLLO DEL TRABAJO FIN DE GRADO .....	6
FIGURA 4. DIAGRAMA DE GANTT DE LA PLANIFICACIÓN TEMPORAL DEL TFG.....	8
FIGURA 5. ESQUEMA BÁSICO DE UN SISTEMA DE RECONOCIMIENTO AUTOMÁTICO DEL HABLA .....	14
FIGURA 6. DIAGRAMA DE ACCIONES DE UN SISTEMA DE DIÁLOGO .....	17
FIGURA 7. ARQUITECTURA MODULAR DE UN SISTEMA DE DIÁLOGO HABLADO.....	19
FIGURA 8. ACTIVAR EL USO DEL TTS EN LOS DISPOSITIVOS ANDROID.....	21
FIGURA 9. GRÁFICO CUOTA DE MERCADO .....	33
FIGURA 10. ARQUITECTURA ANDROID.....	34
FIGURA 11. CICLO DE VIDA DE UNA ACTIVITY .....	40
FIGURA 12. ESQUEMA GENERAL DE LA APLICACIÓN.....	43
FIGURA 13. ESQUEMA DEL TFG EN ECLIPSE .....	44
FIGURA 14. PANEL DE CONTROL DE XAMPP .....	48
FIGURA 15. REPRESENTACIÓN EN FORMATO JSON DE LOS VALORES OBTENIDOS AL REALIZAR LA PETICIÓN.....	48
FIGURA 16. ARQUITECTURA MYSQL.....	49
FIGURA 17. PHPMYADMIN .....	52
FIGURA 18. PRESENTACIÓN DE LA TABLA DE LIBROS.....	52
FIGURA 19. PRESENTACIÓN DE LA TABLA DE PALABRAS CLAVE. ....	53
FIGURA 20. IMPLEMENTACIÓN PASEO DEL JSON DE LIBROS .....	59
FIGURA 21. IMPLEMENTACIÓN DEL RECONOCEDOR AUTOMATICO DEL HABLA.....	59
FIGURA 22. IMPLEMENTACIÓN DE LA SINTESIS DE TEXTO A VOZ. ....	60
FIGURA 23. USO DEL MÉTODO DOINBACKGROUND(.....	61
FIGURA 24. CAPTURA DE PANTALLA DEL DIÁLOGO.....	64
FIGURA 25. CAPTURA DE PANTALLA DEL MÓDULO DE INICIO.....	64
FIGURA 26. ASISTENTE POR VOZ.....	65
FIGURA 27. ENTRADA POR TECLADO.....	65
FIGURA 28. ENTRADA POR PANTALLA.....	65
FIGURA 29. RECONOCEDOR AUTOMÁTICO DEL HABLA.....	67
FIGURA 30. IMPLEMENTACIÓN MÓDULO ASISTENTE POR VOZ .....	68
FIGURA 31. EJEMPLO DIÁLOGO APLICACIÓN Y USUARIO. ....	69
FIGURA 32. TABLA PALABRAS CLAVE .....	70
FIGURA 33. FLUJO DE DATOS DEL MÓDULO DE BÚSQUEDA DE PALABRAS CLAVE INTERACCIÓN MULTIMODAL .....	71
FIGURA 34. BOTÓN DEL MENÚ SELECCIONADO .....	71
FIGURA 35. PRESENTACIÓN DE LA TABLA DE LIBROS.....	73
FIGURA 36. PLANO LIBRO PROGRAMACIÓN EN JAVA .....	74
FIGURA 37. LISTA DE MATERIAS .....	75
FIGURA 38. LIBROS PERTENECIENTES A LA MATERIA ANÁLISIS Y DISEÑO DE CIRCUITOS. ..	76
FIGURA 39. INFORMACIÓN DETALLADA DEL LIBRO SELECCIONADO .....	76

FIGURA 40. ACCESO PÁGINA WEB UC3M .....	77
FIGURA 41. FLUJO DE DATOS MÓDULO CONSULTA PAGINA WEB .....	78
FIGURA 42. DIAGRAMA GENERAL DE LA APLICACIÓN. PRIMERAS OPCIONES.....	81
FIGURA 43. DIAGRAMA GENERAL DE LA APLICACIÓN. SALIR .....	82
FIGURA 44. ENCUESTA .....	86
FIGURA 45. ESTADÍSTICAS DE LOS RESULTADOS PARA CADA UNA DE LAS PREGUNTAS QUE SE PLANTEAN EN LA ENCUESTA DE VALORACIÓN DE LA APLICACIÓN.....	91
FIGURA 46. PRESUPUESTO DE LA APLICACIÓN .....	97

# Índice de Tablas

TABLA 1. TÉCNICAS QUE SOLVENTAN LAS DIFICULTADES EN EL RECONOCIMIENTO DEL HABLA.....	20
TABLA 2. ACTIVITIES DE LA APLICACIÓN .....	45
TABLA 3. TABLA PALABRAS CLAVE QUE CONTIENE LAS POSIBLES PALABRAS DICHAS POR EL USUARIO Y LA RESPUESTA ASOCIADA A CADA UNA .....	45
TABLA 4. TABLA LIBROS QUE CONTIENE LA INFORMACIÓN DETALLADA DE CADA LIBRO ...	46
TABLA 5. VENTAJAS Y DESVENTAJAS DE XAMPP .....	47
TABLA 6. VENTAJAS Y DESVENTAJAS DEL MYSQL.....	50
TABLA 7. TABLA PALABRAS CLAVE INDICANDO EL TIPO Y EL COTEJAMIENTO USADO.....	51
TABLA 8. TABLA LIBROS INDICANDO EL TIPO Y EL COTEJAMIENTO USADO .....	51
TABLA 9. CARACTERÍSTICAS PRINCIPALES DEL LENGUAJE .....	54
TABLA 10. CAMPOS PRINCIPALES DONDE SE USAN SCRIPTS DE PHP.....	54
TABLA 11. INCONVENIENTES DE PHP .....	55
TABLA 12. COMANDOS USADOS EN EL FICHERO .PHP .....	56
TABLA 13. COMANDOS USADOS DESDE EL LADO DEL CLIENTE PARA ESTABLECER LA CONEXIÓN HTTP. ....	56
TABLA 14. ESTRUCTURAS DE JSON.....	57
TABLA 15. MÉTODOS USADOS.....	58
TABLA 16. OPCIONES DEL MENÚ DESLIZABLE .....	66
TABLA 17. MÓDULOS UTILIZADOS .....	68
TABLA 18. EXPLICACIÓN COLUMNAS TABLA “PALABRAS CLAVE” .....	69
TABLA 19. DIÁLOGO ESTABLECIDO ENTRE EL USUARIO Y LA APLICACIÓN EN EL MÓDULO BÚSQUEDA DE LIBROS .....	73
TABLA 20. INFORMACIÓN SOLICITADA .....	80





# Capítulo 1

## Introducción y objetivos

El primer capítulo de la memoria expone los principales objetivos de la aplicación desarrollada para el Trabajo Fin de Grado, así como la motivación y los principales retos que se persiguen con su desarrollo. Se detalla además la finalidad de la aplicación, la planificación temporal seguida para completar su desarrollo, los medios hardware y software que se han utilizado, además de una pequeña descripción sobre el contenido del resto de capítulos que completan la memoria.

### 1.1 Introducción

El presente proyecto tiene como principal objetivo el desarrollo de una aplicación en Android basada en los sistemas de diálogo hablado que permita la interacción con los usuarios a través de dispositivos móviles con sistema operativo Android

Los dispositivos móviles [Smartphone] son, actualmente, parte indispensable de nuestras vidas, ya que, gracias a ellos se facilitan muchas de las nuestras tareas cotidianas para comunicarnos con los demás, buscar de información, conocer cómo llegar a distintos destinos, enviar correos o mensajes instantáneos, hacer videollamadas y muchas funcionalidades adicionales.

Estos dispositivos son capaces de llevar a cabo la mayoría de las tareas que se pueden realizar con un ordenador con Internet, pero de una manera más sencilla y más cómoda y desde prácticamente cualquier lugar y de forma instantánea.

En los últimos tiempos, con el principal objetivo de facilitar la accesibilidad a las funcionalidades comentadas, se están implementando los sistemas de diálogo hablados en los dispositivos móviles.

Un sistema de diálogo es un sistema automático capaz de emular a un ser humano en un diálogo con otra persona, [Griol2007] con el objetivo de que el sistema cumpla con una cierta tarea (normalmente suministrar una cierta información o proporcionar ciertos servicios).

La posibilidad de poder dar y recibir órdenes orales a través de estas aplicaciones, nos permite llevar a cabo determinadas tareas de forma más rápida y accesible. Este sistema permite sustituir interfaces tradicionales como el teclado, el ratón o la pantalla por la voz.

No obstante, construir sistemas de diálogo que comprendan perfectamente al usuario, que entiendan las distintas jergas y expresiones y sean capaces de responder con exactitud a la consulta solicitada supone un trabajo bastante complejo. Con el desarrollo de las tecnologías, estos sistemas han ido avanzando progresivamente, comenzando por un diálogo sencillo basado en pregunta-respuesta (el usuario pregunta a la aplicación y ésta le responde) a uno formado por pregunta-pregunta-respuesta-respuesta (el usuario pregunta a la aplicación y ésta le contesta con una pregunta, acotando así el rango de búsqueda de la primera pregunta. El usuario vuelve a preguntar, pero esta vez de una manera más concisa, facilitando así la respuesta a la aplicación).

A raíz de, existen estas complicaciones y limitaciones comentadas las llamadas aplicaciones multimodales, en las que el acceso a la información, no sólo depende de la interacción usuario-dispositivo mediante el habla, sino que también se pueden realizar las consultas mediante las interfaces tradicionales tales como teclado, pantalla, botones, así como interfaces de entrada y de salida, ofreciendo por estas vías distintas maneras de acceder a la información. Gracias a este tipo de aplicaciones, los usuarios con discapacidad, pueden acceder a la información por diferentes canales. En el caso de personas con discapacidad visual, lo podrían hacer realizando la consulta y recibiendo la respuesta de forma oral. Las personas con discapacidad auditiva podrían hacer estas tareas de forma escrita y pictográfica.

El sistema operativo móvil Android está basado en Linux, sistema operativo libre, gratuito y multiplataforma. El lenguaje de programación que se utiliza para desarrollar aplicaciones para este sistema es una variante del lenguaje Java llamada Dalvik. [ANDROID]

En la siguiente figura podemos observar la gráfica comparativa de los distintos SO móviles actuales. En primer lugar se encuentra Apple, con su iOS, 54,27% y en segundo lugar está a Android 35,41% y después, muy de lejos, Java ME y Symbian con un 3,18% y Blackberry y Kindle con un 1,56% y 0,95% respectivamente [UCHCEU]. La Figura 1 nos muestra los distintos sistemas operativos móviles, mientras que en la figura 2 se observan las versiones más usadas de los distintos sistemas operativos.

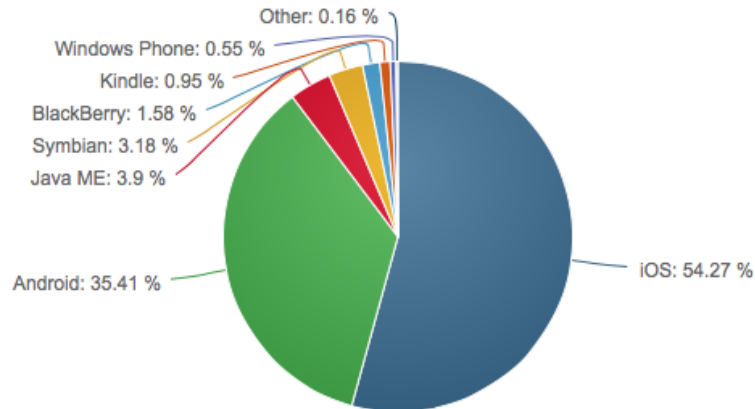


Figura 1. Sistemas operativos para dispositivos móviles

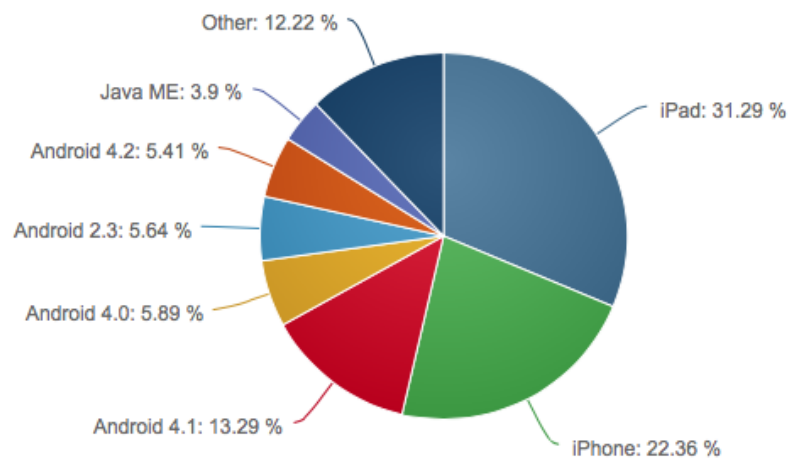


Figura 2. Versiones más usadas de los distintos sistemas operativos móviles

Android cuenta, por una parte con la funcionalidad de Reconocimiento automático del habla o ASR (Automatic Speech Recognition) para detectar la consulta del usuario y por otra parte con la síntesis de texto a voz o TTS (Text-To-Speech) que se encarga de convertir el texto en voz, lo cual sirve para contestar al usuario el resultado de su consulta. Éstas han sido dos de las clases de Android implementadas en el código del presente TFG. Además se ha hecho uso de servicios tales como *AsyncTask*, *RecognizerIntent*, *JSONObject*, *JSONArray*, *Intent*, *GridView*, *AlertDialog*, *ArrayList<HashMap<String, String>>*, *HashMap<String, String>*, etc. del lado del cliente y del lado del servidor PHP, MySQL, HTTP y JSON.

Actualmente se pueden encontrar distintas aplicaciones que hacen uso del reconocimiento de voz. Para el sistema operativo Android existen varias, tales como Cloe, Jennie, Edwin, etc. En cuanto al sistema operativo IOS destaca Siri, principal aplicación dentro de este ámbito.

Haciendo uso de este recurso, se ha llevado a cabo el desarrollo de la presente aplicación: *Desarrollo de un bibliotecario virtual para dispositivos móviles Android* cuyo objetivo es desarrollar un asistente multimodal que sea capaz de gestionar consultas relativas a un servicio de biblioteca digital, ya sea mediante el diálogo con él o mediante

el uso de interfaces tradicionales como puede ser la pantalla y el teclado usando el sistema operativo móvil Android.

Actualmente existen diversidad de opciones para crear bibliotecas digitales personales tales como: Google book, Mobipocket, Microsoft reader, Greenstone, etc. Por otro lado existen aplicaciones de Bibliotecas digitales que permiten tener un conocimiento de los volúmenes de que disponen cada una de ellas, hacer reservas para su uso, etc. [BNE] Incluso algunas permiten leer los libros mediante de programas de audio de los dispositivos [QuijoteQuijoteQuijoteQuijoteQuijote]. Pero en ningún caso existen en el mercado aplicaciones asociadas a dispositivos móviles que puedan gestionar la relación de la biblioteca con los usuarios a través de los sistemas de diálogo. Por estos motivos, el proyecto desarrollado supone una innovación dentro del tipo de aplicaciones diseñadas con objetivos similares.

## 1.2 Objetivos del TFG

Una vez analizada la importancia y principales ventajas de los sistemas multimodales se establece que el objetivo fundamental del presente Trabajo Final de Grado es desarrollar una aplicación multimodal para dispositivos móviles Android, cuya finalidad es la de proporcionar al usuario información relativa sobre una biblioteca virtual, en concreto para el presente TFG se ha tomado como modelo la Biblioteca de la Universidad Carlos III de Madrid de Leganés, modelando únicamente la bibliografía recomendada y los textos básicos.

El usuario decide mediante que modalidad (oral, visual o táctil) quiere acceder al contenido e información de la biblioteca, según sus preferencias o capacidades. Dispondrá de un botón preparado para escucharle, un menú en el que se muestran las distintas opciones y un campo en el que puede escribir la consulta, si prefiere usar la interfaz gráfica en vez de la voz.

La aplicación es capaz de contestar al usuario sobre todas las consultas posibles relacionadas con la Biblioteca de la Universidad Carlos III. Proporciona al usuario dicha información mediante el habla, el teclado y usando la interfaz gráfica, proporcionando así tres vías para informar al usuario y que éste entienda con claridad la respuesta a su consulta.

Las consultas que lleva a cabo la presente aplicación son:

- Información general sobre la biblioteca: cómo está distribuida, cuántas plantas tiene, dónde se encuentran los ordenadores, baños, salas de reuniones, etc.
- Información concreta de los fondos bibliográficos y normativa de regulación de la biblioteca:
  - o Procedimiento para reservar un libro y dónde llevarlo a cabo.
  - o Listado de libros de la biblioteca

- Indicaciones sobre cómo encontrar el libro que ha solicitado, incluyendo mapas gráficos e indicaciones de GPS.

En base a ese objetivo principal, se proponen los siguientes objetivos parciales:

- Realizar un estudio del sistema Android: arquitectura, lenguaje de programación y versiones existentes.
- Completar un estado del arte relativo a los sistemas de diálogo.
- Explicar del uso de las tecnologías adicionales y de los recursos utilizados.
- Diseñar y desarrollar algoritmos eficientes para satisfacer las consultas relacionadas con la biblioteca.
- Realizar pruebas unitarias para mejorar su funcionamiento y eficacia.
- Realizar un análisis de los resultados obtenidos y evaluarlos.

## 1.3 Planificación temporal

El desarrollo del presente proyecto se ha llevado a cabo en las fases que se describen a continuación:

### Fase 1: Documentación y planificación

Antes de comenzar con el código ha sido necesario el aprendizaje de las distintas tecnologías a utilizar, dentro de la variedad que existe en el mercado, cuáles usar, sus ventajas e inconvenientes.

- Estudio de los sistemas operativos móviles. Documentación previa al desarrollo de la aplicación sobre los distintos sistemas operativos actualmente en el mercado y la decisión de cuál elegir para el desarrollo de la aplicación.
- Estudio de la plataforma Android. Investigación sobre qué programas se utilizarán para la implementación del código y en qué versión se desarrollará.
- Estudio de los diferentes sistemas de diálogo. Documentación relacionada con los sistemas de diálogo, su expansión, su uso y su implementación (APIs de Android para poder implementar dichos sistemas).
- Planificación de los requisitos de la aplicación. Determinar los requisitos a cumplir de la aplicación.
- Estudio de las tecnologías adicionales referentes a la aplicación y a la base de datos.

### Fase 2: Desarrollo de la aplicación

Una vez finalizada la documentación acerca de las tecnologías a utilizar y el aprendizaje de éstas, se lleva a cabo el desarrollo de la aplicación:

- Diseño. División de los distintos módulos y de las funcionalidades del sistema.
- Implementación del código. Llevar a cabo la programación de la aplicación utilizando Eclipse para su desarrollo.
- Base de datos. Desarrollar la base de datos necesaria para la aplicación, en la cual se encontrarán todas las posibles consultas.
- Evaluación de la aplicación. Realizar una encuesta a 10 personas acerca de la aplicación y exponer los resultados.

### Fase 3: Redacción de la memoria y presentación

Tras finalizar el desarrollo de la aplicación se ha llevado a cabo:

- Memoria del Trabajo Fin de Grado. Redacción del presente documento.
- Preparación de la presentación.

En la Figura 3 se muestra el diagrama WBS que indica cómo se estructuró el presente Trabajo Fin de Grado.

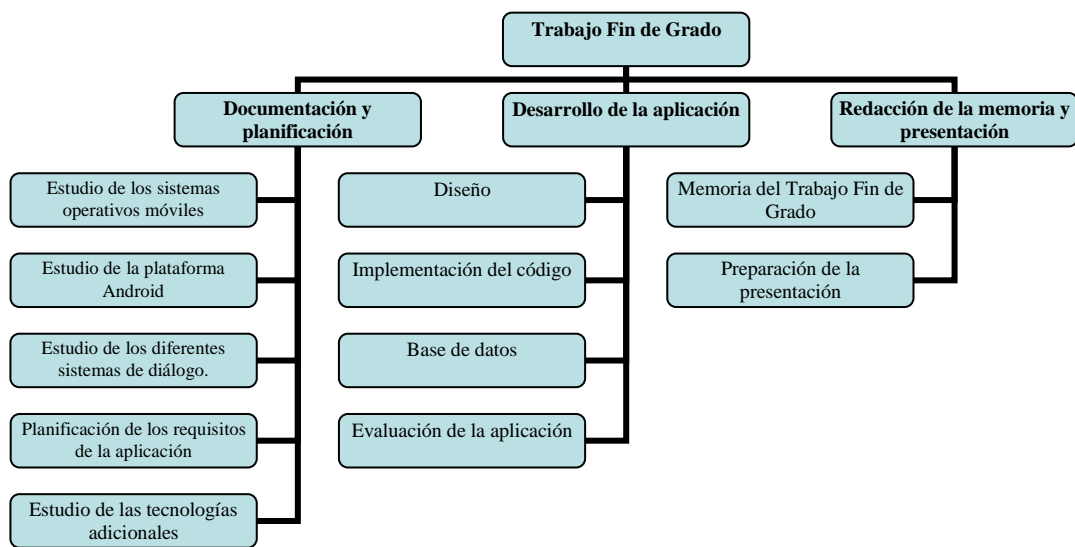
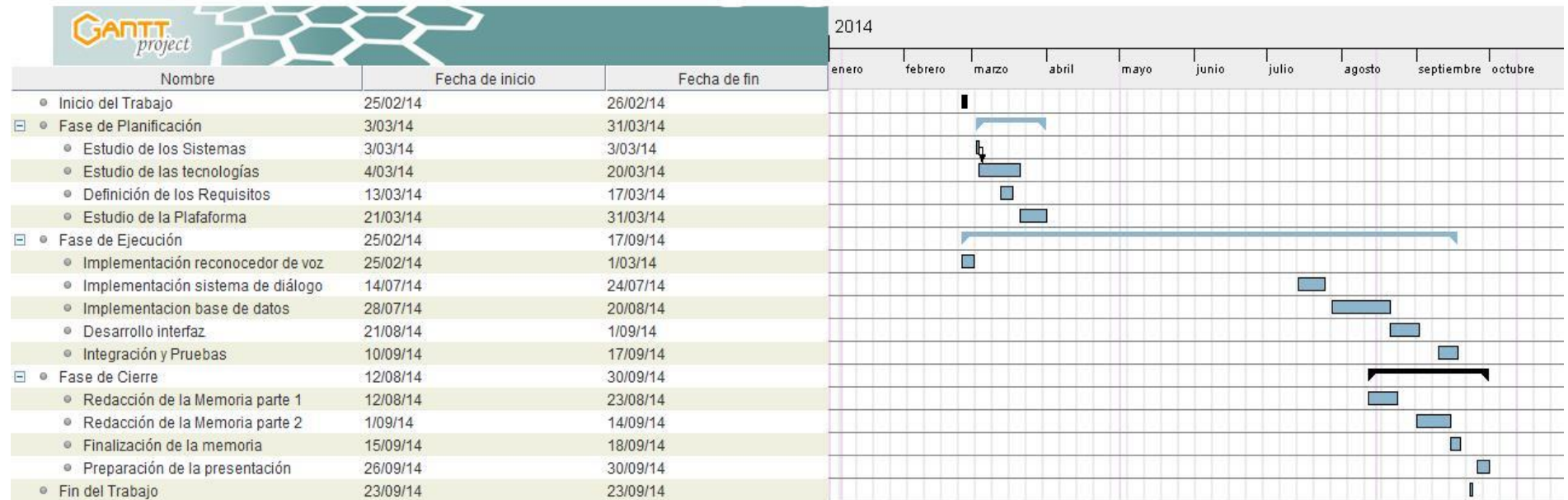


Figura 3. Esquema del desarrollo del Trabajo Fin de Grado

Una vez establecidas las fases del proyecto se ha realizado la planificación temporal de dichas fases mediante un *diagrama de Gantt*, [Gantt] herramienta gráfica cuyo objetivo es exponer el tiempo dedicado para cada una de las diferentes fases a lo largo de un tiempo total determinado.

En la Figura 4 se muestra el diagrama de Gantt de la planificación temporal del TFG. Como se observa su inicio comenzó el 25 de Febrero en el que se implementó el sistema de reconocimiento automático por un lado y el sintetizador de texto a voz por otro, pero no fue hasta el 14 de Julio cuando se comenzó realmente con el desarrollo del TFG. En el diagrama se introducen las duraciones de cada tarea específicamente considerando una jornada de trabajo de 5 horas de lunes a jueves y 10 horas de jueves a domingo, durante 5 semanas. Calculando un promedio de 500-600 horas para la realización del presente Trabajo Fin de Grado, realizado en su mayor parte del 14 de Julio al 14 de Septiembre.



**Figura 4. Diagrama de Gantt de la planificación temporal del TFG**



## 1.4 Medios empleados

Los medios empleados para llevar a cabo la realización del Trabajo Fin de Grado son:

- Hardware
  - Ordenador portátil HP.
  - Ordenador portátil MacBookAir.
  - Terminal móvil Samsung Galaxy.
  - Cable USB.
  - Servidor web
  
- Software
  - Entorno de desarrollo integrado de código abierto multiplataforma Eclipse.
  - SDK (Software Development Kit) de Android: kit de desarrollo de software para Android.
  - JDK (Java Development kit): kit de desarrollo de Java.
  - Plug-in ADT (Android Development Tools) para Eclipse: permite desarrollar aplicaciones Android utilizando Eclipse.
  - Aplicación de búsqueda por voz de Google.
  - Python.
  - Dropbox.
  - Paquete Microsoft Office 2010.
  - Adobe Photoshop.
  - Adobe Reader.
  - Editor de programación Notepad ++.

## 1.5 Estructura de la memoria

Tras este primer capítulo de Introducción, el contenido de la presente memoria se completa con 6 capítulos adicionales, los cuales se describen a continuación de forma resumida.

**Capítulo 1- Introducción** El presente documento comienza explicando la motivación por la cual se eligió este proyecto, cuáles son sus objetivos, las fases del desarrollo, los medios empleados para llevarlo a cabo y la estructura de la presente memoria.

**Capítulo 2- Estado del arte** En este capítulo se expone cuál es Estado del Arte de las Tecnologías del habla y en concreto de los sistemas de diálogos, se describe también el sistema operativo Android detallando su arquitectura, versiones y estructura de una aplicación. Se desarrolla también una descripción de los distintos sistemas operativos móviles.

**Capítulo 3- Descripción general del sistema** Este capítulo presenta una visión general del contexto en el que se enmarca la aplicación. Explica detalladamente la plataforma Android, sus versiones, su estructura y qué programa se ha utilizado para implementar el código. También se presentan las dos herramientas más importantes usadas para implementar el sistema de diálogo, de una manera resumida, pues su explicación exhaustiva se realizó en el estado del arte. Por último se define la base de datos utilizada.

**Capítulo 4- Descripción detallada de los módulos del sistema.** En el capítulo 4 se explican los distintos módulos que componen la aplicación, las funcionalidades principales de los mismos, la arquitectura utilizada para el desarrollo de la aplicación y se detallan los flujos de datos de cada uno.

**Capítulo 5- Evaluación de la aplicación.** En este capítulo se expone el cuestionario realizado a varios usuarios y las conclusiones obtenidas se representan en gráficos para determinar la calidad de la aplicación.

**Capítulo 6- Conclusiones, trabajos futuros y presupuesto.** Para finalizar la memoria, se exponen las conclusiones a las cuales se ha llegado tras la finalización del Trabajo Fin de Grado y los posibles trabajos futuros que se podrían implementar para ampliar la aplicación desarrollada. Este capítulo contiene el presupuesto calculado del presente TFG.

**Glosario**

**Bibliografía**



# Capítulo 2

## Estado del arte

En este capítulo se expondrá cuál es Estado del Arte de las Tecnologías del Habla y en concreto de los sistemas de diálogo, principales campos en los que se enmarca este trabajo, comenzando por una aproximación a su uso y posible evolución, analizando el mundo donde se desarrolla, las organizaciones nacionales e internacionales que lo estudian y estructuran, las últimas investigaciones españolas, los estándares actuales para dispositivos móviles, en qué consisten, y el análisis del que se va a utilizar para este trabajo.

Este Trabajo Fin de Grado ha desarrollado una aplicación para dispositivos móviles que permite la iteración oral con los usuarios. Dicha aplicación consiste en un “bibliotecario virtual” que es capaz de contestar al usuario todas aquellas cuestiones/preguntas que tenga sobre la biblioteca, ya sea mediante el diálogo con él, o mediante el uso de interfaces tradicionales como puede ser la pantalla y el teclado. Pero se puede suponer que en un corto plazo será necesario implementar un sistema real en la puerta de la biblioteca, posiblemente iniciándose con un sensor de acercamiento para reconocer la presencia de una persona que quiere utilizar los servicios de la biblioteca, en la que una voz le pregunte sus deseos y una vez confirmado le facilite el uso de un pequeño terminal que le permita realizar la gestión oportuna que deseaba sin intervención humana. Esto significaría el paso de utilizar las aplicaciones ya existentes para teléfonos inteligentes, ya desarrolladas y amortizadas económicamente, en dispositivos móviles especializados, tan baratos por la gran cantidad de su producción sin caer en aplicaciones desarrolladas ad-hoc muchísimo más caras.

## 2.1 Introducción

El uso generalizado de teléfonos inteligentes y tabletas portátiles de pequeño tamaño, su superior portabilidad con los denominados “ordenadores portátiles” y su frecuencia de uso, sobre todo, mientras se está realizando otra actividad, añadido a su conexión a través de internet con una inmensa cantidad de información hacen posible que se desarrollen una cantidad de aplicaciones que permitan a los usuarios realizar infinitud de funciones. Como en algunas se utiliza el reconocimiento de voz existe un creciente interés en el desarrollo de las Tecnologías del habla.

En cualquier tipo de comunicación son necesarios los usos de protocolos que permitan el entendimiento entre ambas interlocutores. Para ello se suelen crear protocolos o estándares que los hagan fácilmente accesibles a nivel internacional. Sin embargo para las Tecnologías del habla todavía no se dispone de ese protocolo y será necesario mucho trabajo y esfuerzo para conseguirlo. Hay avances continuos en el desarrollo de tecnologías de la información y se están realizando grandes esfuerzos en trabajos científicos promovidos por organizaciones nacionales e internacionales que impulsarán su crecimiento.

Existen tesis doctorales, como la de Syaheerah Lebai Lutfi [Tomei] que describen que existe una especie de frustración y aburrimiento con los sistemas de diálogo por parecer poco sensibles. Porque una vez que se empieza a hablar con la máquina, tan específico del ser humano, no resulta fácil entender que no es otra persona la que está al otro lado. Por tanto se podría decir, como se cita en dicha tesis, que actualmente la sociedad no se siente plenamente satisfecha con el nivel actual de relación con los sistemas automáticos a través del lenguaje.

Además, los usuarios quieren acceder a estos servicios en cualquier lugar y en cualquier momento de una manera natural, intuitiva y eficiente. Las interfaces basadas en el discurso se han convertido en una de las principales opciones para facilitar este tipo de comunicación, que es una buena solución para el tamaño reducido de dispositivos móviles, facilita la comunicación en entornos donde este acceso no es posible utilizar interfaces de entrada tradicionales (por ejemplo, teclado y ratón) y facilita el acceso a la información para personas con discapacidad visual o motora.

Por otra parte con los avances de las tecnologías video, imagen y voz, la interacción persona-ordenador (*Human-Computer Interaction* HCI) ha alcanzado una nueva etapa, en la cual la información multimodal [Tomei2013] es un punto clave para mejorar la comunicación entre humanos y máquinas. A diferencia del tradicional teclado y ratón basado en interfaces, las interfaces multimodales permiten una mayor flexibilidad en la entrada y salida, posibilitando a los usuarios emplear diferentes modalidades para la entrada así como para obtener respuestas a través de diferentes medios, por ejemplo, discurso, gestos y expresiones faciales. Esto es especialmente importante para los usuarios con necesidades especiales, para quienes las interfaces tradicionales podrían no ser adecuados.

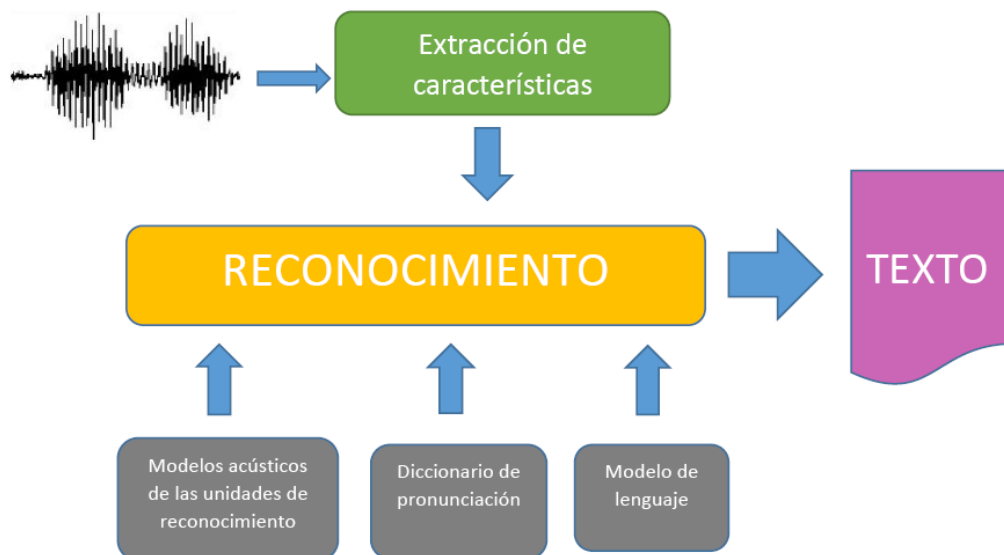
Hay una gran variedad de aplicaciones en las que se pueden utilizar sistemas de diálogo hablado; como por ejemplo aplicaciones informativas sobre vuelos o reservas turísticas, previsiones meteorológicas, sistemas bancarios o ayuda de conferencia. En

algunos casos la interacción oral puede ser la única manera de acceder a la información, como, por ejemplo, cuando la pantalla es demasiado pequeña para mostrar información (ej. dispositivos portátiles) o cuando los ojos del usuario están ocupados en otras tareas (por ejemplo el manejo del dispositivo). La interacción hablada también es útil para dispositivos de control remoto y robots, especialmente en entornos inteligentes.

## 2.2 Sistemas de diálogo

Un sistema de diálogo puede entenderse como un sistema automático capaz de emular a un ser humano en un diálogo con otra persona, [Griol2007] con el objetivo de que el sistema cumpla con una cierta tarea (normalmente suministrar una cierta información o proporcionar ciertos servicios).

En la Figura 5 se muestra Esquema básico de un sistema de reconocimiento automático del habla mostrando los distintos módulos que lo componen y su interrelación entre ellos.



**Figura 5. Esquema básico de un sistema de reconocimiento automático del habla**

Los sistemas de diálogo hablado (spoken dialogue systems) son sistemas informáticos que reciben como entrada frases del lenguaje natural expresadas de forma oral y generan como salida frases del lenguaje natural expresadas asimismo de forma oral [López\_Cózar2014]. La finalidad de estos sistemas es emular el comportamiento inteligente de un ser humano que realiza una tarea concreta.

Merece una especial mención la realidad nacional donde se han creado organizaciones como La Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN) [SEPLN] que promueven todo tipo de actividades relacionadas con el estudio del procesamiento de lenguaje natural.

Entre todos los trabajos que realiza esta sociedad, está la publicación de una revista cuyo objetivo principal es el de ofrecer a los investigadores que trabajan dentro del marco del Procesamiento del Lenguaje Natural (PLN) una oportunidad para presentar nuevos trabajos, comunicar resultados, discutir problemas y obstáculos encontrados durante su trayectoria investigadora. Por otro lado, permite intercambiar opiniones sobre directrices futuras de investigación básica y aplicación prevista por los expertos y contrastarlas con las necesidades reales del mercado. Reflexionar y debatir en profundidad sobre temas concretos de máxima actualidad tales como la extracción de información, la recuperación de información o la evaluación de sistemas de procesamiento del lenguaje natural.

También realiza congresos anuales, normalmente en las universidades españolas, donde la SEPLN pretende promover la difusión de las actividades de investigación, desarrollo e innovación que realizan en cualquiera de los ámbitos del procesamiento del lenguaje natural los diversos grupos e investigadores españoles y extranjeros. El congreso aspira a ofrecer un foro de discusión y comunicación en el que se favorezca el intercambio de la información y materiales científicos necesarios para promover la publicación de trabajos y la colaboración con instituciones nacionales e internacionales que actúen en el ámbito de interés del congreso.

La Comisión Interministerial de Ciencia y Tecnología (CICYT) subvencionó el proyecto DIHANA, [DIHANA] que tenía como principal objetivo el desarrollo de un sistema de diálogo para el acceso a un sistema mediante el habla espontánea. El sistema desarrollado en el proyecto está compuesto por siete módulos: un Reconocedor automático del habla, un módulo de comprensión del habla, un gestor del diálogo, un módulo de consulta a la base de datos, un generador de respuestas en lenguaje natural, un sintetizador de texto a voz y, finalmente, por un gestor central de comunicaciones. Para la implementación del sistema se optó por una arquitectura siguiendo el paradigma cliente-servidor, donde el gestor central actúa como cliente, gestionando las comunicaciones, y el resto de módulos actúan como servidores

El proyecto EDECAN, como continuación de DIHANA, explora técnicas para aumentar la robustez de un sistema de diálogo. EDECAN apunta al desarrollo de tecnologías para la adaptación y personalización para diferentes acústicas y contextos de aplicación. Estos elementos dependen no sólo del usuario, sino también del entorno físico en el que se desarrolla la aplicación. Por otro lado, el concepto de contexto de aplicación se refiere a la estructura semántica de los dominios en los que el diálogo es llevado a cabo.

El proyecto SD-TEAM que pretende la evolución de estos sistemas de diálogo cerrados, hacia sistemas de interacción humana dinámicos y flexibles, con una clara capacidad de evolución autónoma mediante el aprendizaje, evaluación y cooperación a través de la experiencia de uso [SD\_TEAM]. Estos nuevos sistemas "vivos" serían una evolución de sus predecesores, capaces de adaptarse, interactuar con usuarios y con otros sistemas y de aprender "sobre la marcha" para reconfigurarse. En este contexto, una arquitectura distribuida y flexible (evolución a su vez de la propuesta en el proyecto EDECÁN) permite la cooperación entre diferentes sistemas (módulos de comprensión, reconocedores, gestores de diálogo, etc. Desarrollados para diferentes entornos de funcionamiento) así como la interacción de naturaleza multimodal con usuarios, ambas

basadas en un cierto funcionamiento autónomo que permitirá una evolución constante e independiente.

A lo largo de los años se han desarrollado sistemas de diálogo "estáticos", donde el usuario se adapta hacia las necesidades del sistema. Por el contrario, cada día más, los sistemas tienden a ser "orgánicos", capaces de aprender, crecer, reconfigurarse y auto repararse ofreciendo de esta forma una mayor robustez frente a cualquier cambio en las condiciones de trabajo. En este contexto se hace imprescindible estudiar en profundidad los aspectos metodológicos y tecnológicos que faciliten la extensión de sistemas de diálogo hablado estáticos hacia sistemas "orgánicos" con capacidad de aprendizaje, evaluación y cooperación, en definitiva, sistemas que evolucionen de forma autónoma hasta lograr la plena satisfacción del usuario. Entre los usuarios potenciales se deben incluir aquellos colectivos para los cuales un sistema, que no tenga en cuenta una adecuada adaptación y personalización, no funcionaría.

En el ámbito internacional el Consorcio World Wide Web (W3C) [W3C] es una comunidad donde las organizaciones Miembro, el personal a tiempo completo y el público en general trabajan conjuntamente para desarrollar estándares Web. Su misión es guiar la Web hacia su máximo potencial y su principal actividad es desarrollar protocolos y directrices que aseguren el crecimiento de la Web a largo plazo. Los estándares del W3C definen las partes claves que hacen que la World Wide Web funcione.

El valor social que aporta la Web, es que ésta hace posible la comunicación humana, el comercio y las oportunidades para compartir conocimiento. Uno de los objetivos principales del W3C es hacer que estos beneficios estén disponibles para todo el mundo, independientemente del hardware, software, infraestructura de red, idioma, cultura, localización geográfica, o habilidad física o mental.

El W3C lleva a cabo su misión uniendo a diversos agentes sociales, bajo un proceso claro y efectivo basado en el consenso para desarrollar los estándares que toman como base las contribuciones aportadas por Miembros, el Equipo y la comunidad en general.

Para eliminar cualquier tipo de duda entre Web e Internet tomando la definición en Wikipedia "Internet es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos estándares TCP/IP". De esta forma, Internet se define mediante los estándares TCP/IP.

La Web, por otro lado, se define en el W3C como sigue: "La World Wide Web (WWW, o simplemente Web) es un espacio de información donde los elementos de interés, denominados como recursos, se identifican a través de identificadores globales llamados Identificadores de Recurso Uniforme (URI)." Así que en la Web se define mediante otras especificaciones. Las tres primeras especificaciones para las tecnologías Web fueron URLs, HTTP y HTML.



## 2.2.1 Arquitectura y acciones básicas de un sistema de diálogo

En la Figura 6 se muestran las acciones básicas que realiza un sistema de diálogo [Griol2007].

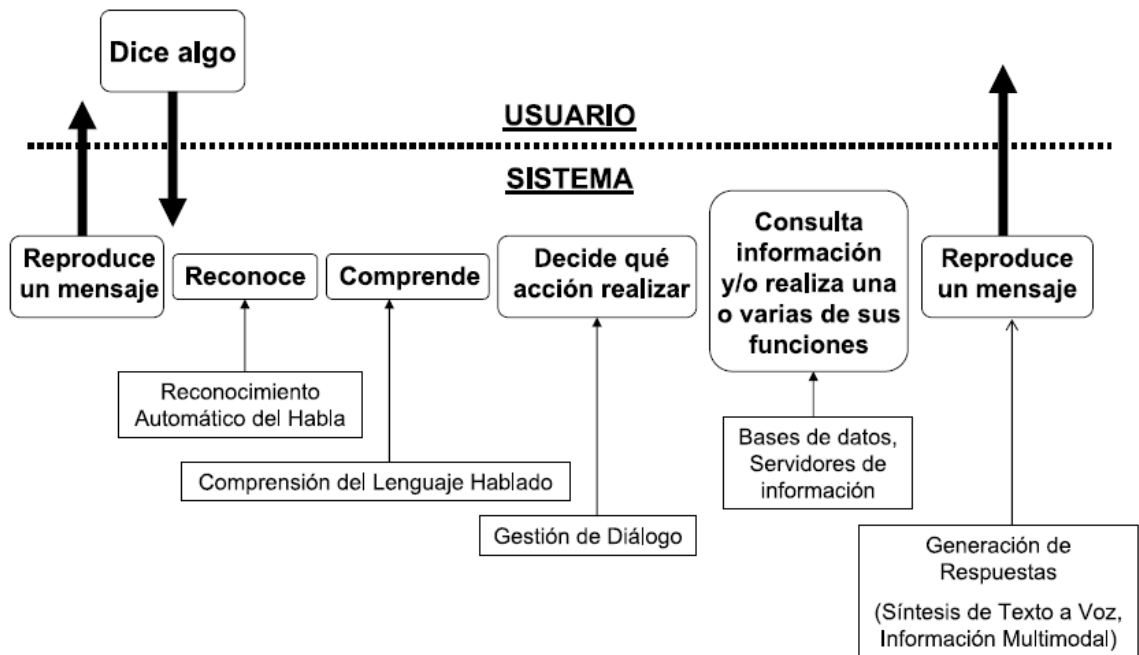


Figura 6. Diagrama de acciones de un sistema de diálogo

Como se aprecia, el sistema genera un mensaje inicial que sirve habitualmente para dar la bienvenida o informar al usuario sobre las características del sistema. Tras cada intervención del usuario, el sistema realiza un conjunto de *acciones básicas* que se repiten cíclicamente como respuesta a cada acción del usuario:

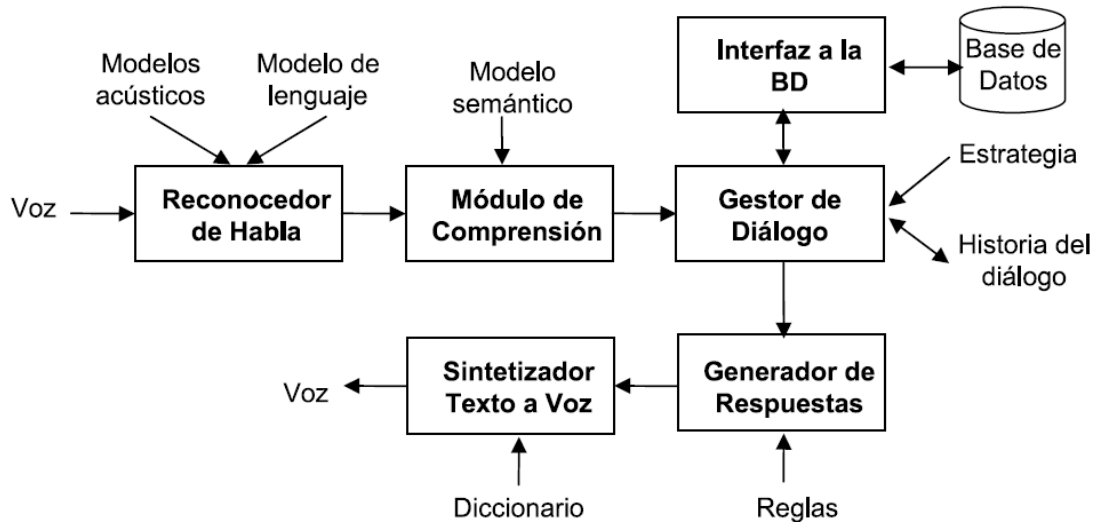
1. Reconocer la secuencia de palabras mencionadas por el usuario.
2. Extraer el significado de dichas palabras, es decir, comprender qué información es útil en el dominio del sistema.
3. Realizar operaciones de acceso a la base de datos u otros recursos del sistema, en los que se almacena la información que solicita el usuario o se registran las operaciones que desea conocer.
4. Decidir qué acción o acciones deben realizarse a continuación de cada solicitud del usuario, es decir, qué respuesta debe suministrar el sistema.
5. Reproducir un mensaje hablado que informe al usuario qué acción ha seleccionado el sistema.

## 2.2.2 Arquitectura de los sistemas de diálogo hablado

Los sistemas de diálogo se componen de una serie de módulos genéricos para tratar las múltiples fuentes de conocimiento y que deben coordinarse para responder a los requerimientos del usuario. A continuación se detallan los módulos.

- Módulo de Reconocimiento Automático del habla (RAH): Facilita la secuencia de palabras que se ajusta a la señal vocal de entrada con una mayor probabilidad.
- Módulo de comprensión (CH): Extrae el significado de las palabras reconocidas en el módulo anterior, expresándolo en términos de un lenguaje semántico especificado para la tarea.
- Gestor de diálogo (GD): Decide qué paso debe dar el sistema tras cada intervención del usuario. Para ello se basa en la interpretación semántica generada por comprensión, el historial del diálogo, el estado actual del sistema, información obtenida de la base de datos, la estrategia definida.
- Módulo de consulta a la base de datos de la aplicación (MCBD): Recibe peticiones de consulta a la base de datos por parte del gestor de diálogo, las procesa y devuelve el resultado al gestor.
- Módulo de generación de respuestas (GR): Tiene como función la generación de una frase, gramaticalmente correcta y en un lenguaje lo más cercano posible al lenguaje natural, que transmita el mensaje generado por el gestor de diálogo.
- Sintetizador de Texto a Voz. Transforma la frase de respuesta al usuario en señal de audio.
- Módulo central gestor de comunicaciones: Este módulo es el responsable de establecer las comunicaciones con el resto de módulos del sistema, encargándose de recibir la totalidad de mensajes enviados (excepto los de audio) y de encaminarlos al servidor destino.

En la Figura 7 se muestra la arquitectura modular de un sistema de diálogo hablado.



**Figura 7. Arquitectura modular de un sistema de diálogo hablado**

A continuación se desarrolla con mayor profundidad los módulos más relevantes de los mencionados anteriormente.

### 2.2.3 Reconocimiento Automático del Habla

El reconocimiento automático del habla (ASR) o reconocimiento automático de voz es una disciplina de la inteligencia artificial que tiene como objetivo permitir la comunicación hablada entre seres humanos y computadoras [RAHRAHRAH]. El problema que se plantea en un sistema de este tipo es el de hacer cooperar un conjunto de informaciones que provienen de diversas fuentes de conocimiento (acústica, fonética, fonológica, léxica, sintáctica, semántica y pragmática), en presencia de ambigüedades, incertidumbres y errores inevitables para llegar a obtener una interpretación aceptable del mensaje acústico recibido.

Estos sistemas están en continua mejora gracias al desarrollo de mejores algoritmos y modelos acústicos de los equipos de investigación y a la aparición de procesadores con mayor capacidad de cálculo

Para llevar a cabo esta conversión, los enfoques que han proporcionado los mejores resultados han sido los probabilísticos, basados en la Teoría de la Decisión de Bayes, la Teoría de la Información y en las Técnicas de Comparación de Patrones, de Programación Dinámica (DTW), de Modelos Ocultos de Markov (HMM) y de Redes Neuronales. Se trata de ser capaces de generar un conjunto de patrones (asociados a partes de habla) que puedan ser comparados con la señal acústica de entrada (a reconocer) devolviendo la secuencia de estos patrones que con mayor probabilidad "representan" a la misma. [Epistemowikia]

Existen grandes dificultades en el reconocimiento del habla, algunas de las causas son:

- Las variaciones de fonación, debidas a los hablantes.

- Las ambigüedades acústicas; palabras que se pronuncian igual pero se escriben de forma distinta (vaca y baca).
- Variaciones de producción y que le desvían del registro teórico ideal,
- Coarticulación. La señal acústica recibida se ve afectada debido al contexto en el que se encuentra.
- Variaciones de tiempo
- Ruidos e interferencias

Para conseguir mejores resultados referentes a este difícil problema se han desarrollado técnicas entre las que destacan las resumidas en la Tabla 1.

Técnicas	Descripción
<b>Técnicas de Programación Dinámica (DTW)</b>	Esta técnica consiste en realizar una comparación entre los patrones o plantillas de las que dispone el sistema con la señal acústica recibida como entrada, de esta forma se obtienen posibles candidatos a los que puede pertenecer la señal recibida
<b>Modelos Ocultos de Markov (HMM)</b>	Un modelo oculto de Markov se puede considerar como una especie de autómatas finito, ya que está formado por una serie de estados que tienen una conexión directa mediante transiciones
<b>Redes Neuronales</b>	Las redes neuronales son una estructura de procesamiento y aprendizaje de información, que está formada por un conjunto de nodos que se denominan neuronas, las cuales están conectadas mediante una serie de pesos. Cada neurona va a recibir una entrada a partir de las conexiones que tiene con el resto de neuronas, y va a producir una salida.

Tabla 1. Técnicas que solventan las dificultades en el reconocimiento del habla

### 2.2.4 Síntesis de texto a voz

Son sistemas que permiten la conversión de texto a voz sintética. Se basan en un sistema de voces pre-grabadas, pronunciación y entonación capaz de vocalizar cualquier combinación de letras en forma de palabra, y también considerando otros aspectos más específicos como la fonética y acento de cada tipo de lenguaje, intentando que sea de la forma más natural e inteligible posible. Los conversores de texto-voz son conocidos también con las siglas CTV o por las siglas en inglés TTS (Text To Speech) [CTV]. Internamente este proceso se compone de dos fases.

En la primera se realiza una representación lingüística simbólica, para ello se siguen tres procesos consecutivos:

- *Normalización del texto.* Se convierte la totalidad del texto a una forma textual convencional. Esto afecta principalmente a las cifras, abreviaturas, etc. A la normalización del texto también se la denomina pre-procesado o tokenización.
- *Conversión fonética.* Una vez normalizado el texto se asignan transcripciones fonéticas a cada palabra. Este proceso se denomina «conversión texto-fonema» (TTP en sus siglas en inglés de text-to-phoneme) o «conversión grafema-fonema» (GTP en sus siglas en inglés de grapheme-to-phoneme).
- *División prosódica.* Se divide el texto en unidades prosódicas, tales como unidades sintagmáticas, proposiciones y frases.

En la segunda fase, la que forma el sintetizador propiamente dicho, toma como entrada la representación lingüística simbólica y la transforma en voz sintética.

Por tanto, un sistema de síntesis de voz tiene que ajustar dos características principales: las características fonéticas que hacen que un discurso sea inteligible y las características acústicas que hacen que la salida sea lo más parecida a una onda acústica procedente de un ser humano.

El sistema Android incorpora un motor TTS instalado por defecto, con el cual las aplicaciones pueden leer textos en voz alta. Sin embargo, Android da la posibilidad de instalar y personalizar varios motores, aunque siempre eligiendo uno como motor TTS principal. Es importante saber que los asistentes virtuales estilo Jennie o S-Voice, clones de Siri, lo único que hacen es utilizar el motor TTS principal para realizar las lecturas de texto.

Su uso en el dispositivo se encuentra en Ajustes / Idioma e introducción / Salida de texto a voz se puede elegir y seleccionar el motor TTS a definir como principal, junto a opciones de personalización como se puede apreciar en la Figura 8.

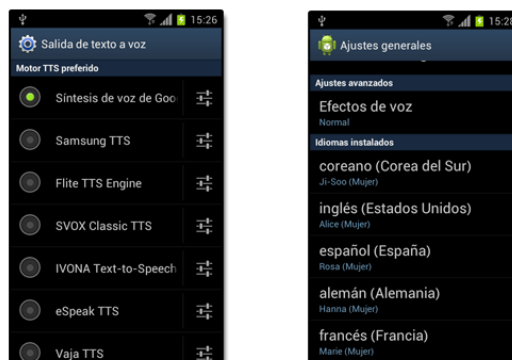


Figura 8. Activar el uso del TTS en los dispositivos Android.

Seguidamente se estudian varios sintetizadores de voz (motores TTS) para Android, con sus respectivos idiomas y características [EMEZETA].

### 2.2.4.1.1 Pico TTS

El sintetizador de voz por defecto en la mayoría de dispositivos Android es Pico TTS, [SPAESP] desarrollado por la compañía SVOX.



Antiguamente, en dispositivos con Android 2.2 (Froyo), bastaba con instalar SpeechSynthesis SPA-ESP Voice, un paquete con el idioma español para el sintetizador Pico TTS. Actualmente, la mayoría de los dispositivos lo traen con los idiomas predefinidos. La calidad de la voz no es mala, sin embargo es muy monótona. Se permite cambiar entre varios idiomas (inglés EEUU/UK, francés, italiano, alemán y español). Es gratuito, viene en español y está fabricado por: Google / SVOX.

### 2.2.4.1.2 IVONA TTS HQ

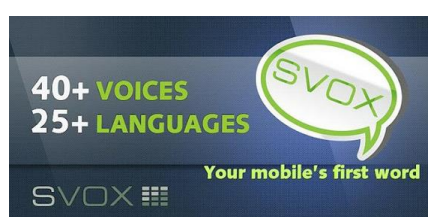
IVONA TTS HQ es, probablemente, uno de los mejores sintetizadores de voz que existen para Android junto a SVOX [¡Error! No se encuentra el origen de la referencia.]. IVONA es actualmente gratuito, ya que está en fase beta, por lo que se puede descargar e instalar sin ningún tipo de coste.



IVONA posee multitud de idiomas, que pueden ser comprobados en su propia página, para comprobar las virtudes del sintetizador. Se trata de una voz muy clara, realista y con buen acento. En español, se dispone de la voz de Conchita y Penélope (esta última, en fase beta, con acento de EEUU) y también de Enrique, voz masculina, aunque parece que aún no está disponible para Android. Es gratuito por ahora, viene en español y está fabricado por: IVONA.

### 2.2.4.1.3 SVOX Classic TTS

Otro de los más importantes sintetizadores de Android voz es SVOX [SVOX]. Al igual que IVONA tiene un amplio surtido de voces e idiomas, siendo además una de las empresas con más tiempo y experiencia en el sector.



En español la voz es de Noelia y Pablo, mientras que en mexicano es de Angélica y a Juan a y tienen un coste de 2'99 euros cada una. También se pueden instalar las versiones trial (de prueba) desde el Google Play, para probarlas antes de comprar. Las opciones de este sintetizador permiten además aumentar o reducir el volumen de voz, el tono o mejorar el rendimiento (para móviles de altas prestaciones). Está fabricado por: SVOX.

#### 2.2.4.1.4 Samsung TTS

Samsung incorpora un sintetizador de voz propio en los dispositivos Android de su marca [HTS], basados en los sintetizadores de HTS Working Group.



Entre las voces que incorpora, se encuentran Ji-Soo (coreano), Alice (inglés EEUU), Rosa (español), Hanna (alemán) y Marie (francés), todas ellas voces femeninas. Como curiosidad, mencionar que también permite activar ciertos efectos de voz (profunda, alta, fina, gruesa, robótica o con efecto de helio). Es gratuito, viene en español y está fabricado por: Samsung.

#### 2.2.4.1.5 CereProc

CereProc es una compañía que tiene varios sintetizadores de voz, [CEREPROC] entre los que se encuentran Caitlin (irlandés), Adam, Isabella y Katherine (inglés) y Stuart y Heather (escocés).

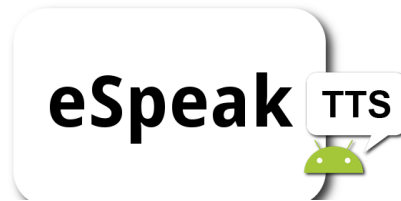


También tienen algunas voces gratuitas, inútiles pero curiosas, como PigLatin (un juego infantil, el juego de los cerdos), Dog (sintetizador a voz de perro), Dodo (una especie de broma similar al Pato Donald) y Nicole Sexy French. Está fabricado por: SVOX.

#### 2.2.4.1.6 eSpeak TTS

Este sintetizador de voz es un port del TTS open source eSpeak [ESPEAK], diseñado para funcionar en plataformas Android. Incluye una gran cantidad de voces en diferentes idiomas (incluso esperanto, swahili u otros más difíciles de encontrar).





Lamentablemente, la voz es muy robótica y casi incomprensible, aunque puede servir para emular voces robotizadas. Aún tiene algunos problemas con acentos y reproducción, sin embargo, va por buen camino. El sintetizador permite cambiar algunas opciones como tono, velocidad o género (voz masculina o voz femenina). Es gratuito por ahora, viene en español y está fabricado por: Eyes-Free Project.

### 2.2.5 Sistemas de diálogo multimodal

El objetivo fundamental de un sistema de diálogo multimodal (multimodal dialogue system) es superar las limitaciones de la interacción basada exclusivamente en el habla [López\_Cózar2014]. En una interacción multimodal el usuario no está restringido a utilizar el habla como único canal de comunicación, sino que puede utilizar varios dispositivos de entrada, como por ejemplo un teclado, un ratón, un micrófono, una cámara, una pantalla sensible al tacto, una PDA, etc. Así mismo, el sistema multimodal puede utilizar diversos canales de salida para proporcionar información al usuario como por ejemplo, voz, texto, gráficos o imágenes, con objeto de estimular varios de sentidos del usuario de forma simultánea. Algunos sistemas de diálogo multimodal permiten incluso que los usuarios puedan elegir entre las diversas modalidades de entrada para llevar a cabo la interacción, permitiendo así una cierta adaptación a las condiciones ambientales de luz, ruido, etc. Esta ventaja permite además que personas con determinadas discapacidades (p. e. personas invidentes) puedan usar estos sistemas mediante alguna de las modalidades de interacción disponibles.

Cabe destacar la importancia del empleo de la voz como uno de los modos con más auge por su comodidad, instinto, reducidas dimensiones del terminal, etc. Otras de las ocasiones en las que el uso de la voz es muy útil debido al impedimento o deficiente utilización de las manos para el manejo del teléfono móvil: en el coche con manos libres, realizando tareas manuales o alguna minusvalía física.

La manera de hacer que diferentes mecanismos de acceso a la información puedan interactuar conjuntamente ofreciéndonos una experiencia satisfactoria, se hace cada vez más difícil ante un mercado en expansión, en lo que se refiere a dispositivos de acceso a información, arquitecturas, sistemas operativos, etc. Por este motivo la estandarización adquiere una gran importancia.

Se muestran a continuación algunos sistemas de diálogo multimodal:

#### MASK. Multimodal-Multimedia Automated Service Kiosk

Sistema desarrollado en el LIMSI Spoken Language Processing Group (Francia) que proporciona horarios de tren, reservas, precios, etc. Para comunicarse con el sistema los usuarios usan el habla y una pantalla sensible al tacto, mientras que éste se puede comunicar con los usuarios mediante habla, gráficos, vídeo y sonido.



### AdApt.

Sistema desarrollado en el Centre for Speech Technology, Royal Institute of Technology (Suecia), proporciona información sobre apartamentos disponibles en la ciudad de Estocolmo. Los usuarios se pueden comunicar con el sistema oralmente y mediante un ratón, y el sistema se puede comunicar con ellos mediante un agente animado y un mapa acompañado de texto.

### August.

Sistema desarrollado también en el Centre for Speech Technology, Royal Institute of Technology (Suecia), para proporcionar información acerca del autor sueco August Strindberg y sobre otros servicios de la ciudad de Estocolmo.

## **2.2.6 Aplicaciones para dispositivos móviles con ASR y TTS**

Existen muchas aplicaciones de reconocimiento de voz que están actualmente disponibles para las diferentes plataformas (iOS, Android, BlackBerry, Nokia, etc.) y que responden a necesidades prácticas, especialmente en la vida profesional [Kioskea]. La mayoría de estas aplicaciones se usan para la traducción, la búsqueda Web, la Búsqueda de direcciones y de itinerarios (Google Maps) y actualizar estados en las redes sociales. También son utilizadas en configuración "manos libres" para los mandos de voz estándares como la búsqueda en la agenda, realizar llamadas, etc.

### **2.2.6.1 Aplicaciones disponibles para iOS**

#### Siri

Con Siri se puede usar la voz para enviar mensajes, programar reuniones, hacer llamadas y mucho más [SIRI]. Y no hace falta que grites, háblale en un tono normal: Siri lo entiende todo e incluso te responde. Y es tan fácil de usar y hace tanto por ti que cada día le encontrarás nuevas utilidades. Otra prestación sorprendente para hacer más cosas con tu voz: en lugar de teclear, toca el icono del micrófono en el teclado, di lo que quieras escribir y Siri tomará nota. Luego toca OK y tus palabras se convertirán en texto. Puedes utilizarse Dictado para escribir mensajes, apuntar cosas, buscar en Internet y un largo etcétera. Dictado también funciona con apps de terceros, así que úsalo para actualizar tu estado de Facebook, tuitear o publicar en Instagram.



### Vlingo

Es una aplicación gratuita del tipo mandos de voz general [*Vlingo*]. Se puede utilizar a partir de iOS 4.0 y también en Android, BlackBerry, Nokia. Sirve de asistente personal para ejecutar diferentes tareas, pero por ahora está disponible sólo en inglés para terminales de Apple.



### Vocalia

Es una aplicación que permite llamar a los contactos de la agenda del iPhone por su nombre [*Vocalia*], sin tener que haber registrado previamente ningún tipo de grabación para reconocerlos; Es decir, lee los nombres de la agenda y establece un sistema de conocimiento de voz gracias al cual puede adivinar qué contacto estamos pronunciando. Por ahora el programa funciona sólo con nombres en inglés y francés, pero se está trabajando en una versión en español que estará disponible en breve.



## **2.2.6.2 Aplicaciones adicionales disponibles para Android**

### Búsqueda por voz de Google

Esta aplicación está integrada en cualquier dispositivo Android. Permite efectuar búsquedas en Google en voz alta. Los resultados aparecen en una página del navegador.



### Traductor de Google

Además de una funcionalidad de traducción de texto, esta aplicación permite enunciar expresiones para traducir en 17 lenguas y obtener su equivalente en más de 60 lenguas. Además, las traducciones pueden ser escuchadas en 24 lenguas.



### Google Maps Navigation

Este popular servicio ofrece un sistema de navegación GPS por órdenes de voz (mediante un acceso a internet). Permite enunciar el lugar de destino en lugar de escribirlo. Muy útil para los momentos que se requiere la utilización del "manos libres".



Tras analizar las distintas aplicaciones se extrae que las características generales es la de realizar funciones como la posibilidad de llamar a sus contactos, redactar un texto y realizar búsquedas en un explorador.

En el presente proyecto se desarrolla una aplicación que no realiza estas funciones estándares, sino que simula a una persona que es capaz de contestar al usuario la respuesta a las consultas que tenga relacionadas con la biblioteca de la Universidad Carlos III de Madrid y, además, de una manera multimodal. Responde tanto a preguntas generales relacionadas con la distribución de la biblioteca, como a preguntas más concretas sobre la ubicación de cada libro mostrando un mapa para completar dicha explicación.

## 2.3 Sistemas operativos móviles

Un sistema operativo es un conjunto de órdenes y programas que controlan los procesos básicos de una computadora y que permiten el funcionamiento de otros programas. En concreto, un sistema operativo móvil o SO móvil es un sistema operativo que controla un dispositivo móvil al igual que los PCs utilizan Windows o Linux entre otros [SOMobiles]. Sin embargo, los sistemas operativos móviles son mucho más simples y están más orientados a la conectividad inalámbrica, los formatos multimedia para móviles y las diferentes maneras de introducir información en ellos.

Un sistema operativo móvil está formado por cuatro capas:

### 1. Kernel

El núcleo o kernel proporciona el acceso a los distintos elementos del hardware del dispositivo. Ofrece distintos servicios a las superiores como son los controladores o drivers para el hardware, la gestión de procesos, el sistema de archivos y el acceso y gestión de la memoria.

### 2. Middleware

El middleware es el conjunto de módulos que hacen posible la propia existencia de aplicaciones para móviles. Es totalmente transparente para el usuario y ofrece servicios claves como el motor de mensajería y comunicaciones, códecs multimedia, intérpretes de páginas web, gestión del dispositivo y seguridad.

### 3. Entorno de ejecución de aplicaciones

El entorno de ejecución de aplicaciones consiste en un gestor de aplicaciones y un conjunto de interfaces programables abiertas y programables por parte de los desarrolladores para facilitar la creación de software.

### 4. Interfaz de usuario

Las interfaces de usuario facilitan la interacción con el usuario y el diseño de la presentación visual de la aplicación. Los servicios que incluye son el de componentes gráficos (botones, pantallas, listas, etc.) y el del marco de interacción.

## **2.3.1 Sistemas operativos móviles más destacados**

Existen muchos sistemas operativos móviles, pero los más usados son [SOUsados]:



### **2.3.1.1 Android**

Android es un sistema operativo basado en Linux, desarrollado por la Open Handset Alliance, la cual lidera Google. Se trata de un sistema abierto, lo que le otorga múltiples ventajas. Su lanzamiento tuvo lugar en Octubre de 2008, su principal objetivo fue plantar cara al reciente sistema operativo móvil lanzado por Apple. Es el sistema operativo que más ha crecido en los últimos años. Ha experimentado un aumento de cuota de mercado del 145% en el último año, lo que le ha permitido alcanzar un 59%, superando con creces el 36.1% con el que contaba en 2011.

#### Ventajas

A la ya comentada de ser un sistema abierto, cuenta con una gran posibilidad de personalización, haciendo que cada terminal sea prácticamente único. Cuenta con una mayor facilidad de transferencia de archivos, sin necesidad de mediación de ningún programa. Cuenta con navegador web compatible con Flash Player. Su crecimiento, le ha convertido en una plataforma muy atractiva para desarrollar, ya que cuenta con un gran número de dispositivos en el mercado.

#### Inconvenientes

Su método de actualización del sistema operativo es un tanto lento, siendo un largo proceso en el cual Google debe lanzar la nueva versión y posteriormente esta debe pasar por las manos de fabricantes y proveedores, por lo que las actualizaciones tardan mucho tiempo en llegar o incluso en ocasiones estas no llegan.

#### Aplicaciones

El recientemente renovado Android Market ha pasado a llamarse Google Play, debido al gran número de usuarios con los que cuentan, muchos desarrolladores se han animado para usar esta plataforma, por lo que se cuenta con una gran cantidad y variedad de aplicaciones a unos precios muy bajos o incluso gratuitos.

### Teléfonos

Numerosas marcas han optado por incluir Android en sus equipos. Algunas de estas marcas son: Samsung, Sony Ericsson, LG, HTC, Huawei, etc. Hay gran variedad de dispositivos, desde terminales muy básicos a dispositivos de gama alta dispuestos a plantarle cara al iPhone.

Como conclusión se establece que Android es un sistema muy a tener en cuenta debido a su crecimiento, además permite una gran personalización. Existen infinidad de documentación y herramientas de desarrollo en internet. Además Google ha creado Android Developers, con consejos y tutoriales, de forma que se pueda sacar todo el partido posible a esta plataforma.



### **2.3.1.2 iOS**

Perteneciente a Apple y originalmente pensada para iPhone. Posteriormente se ha portado a otros dispositivos de la compañía como el iPad y el iPod Touch. Se trata de una versión reducida del sistema Mac OSX existente para ordenadores. Fue presentado en 2007 conjuntamente con el primer iPhone. Con este sistema operativo, Apple fue pionera con el concepto de teléfonos inteligentes tal y como los conocemos hoy. Según International Data Corporation cuenta con un 23% de cuota de Mercado en el primer cuatrimestre de 2012 (crecimiento del 88%).

### Ventajas

Al estar diseñado específicamente para un modelo consigue explotar al máximo sus capacidades. Para Apple la experiencia de usuario es muy importante y eso se ve reflejado en el iPhone, gran fluidez y un periodo de respuesta muy pequeño. Existe la posibilidad de eliminar las limitaciones impuestas por Apple mediante la realización de un jailbreak, el cual nos permite acceder al sistema de archivos del sistema. Gracias a iCloud puedes compartir archivos y calendarios entre varios dispositivos, como pueden ser el iPad y ordenadores.

### Inconvenientes

Apple no permite la instalación de aplicaciones que no hayan sido validadas por ella misma a través de la App Store y es necesario pagar una licencia, válida para un año y con un precio de 99\$. Otro inconveniente es la ausencia de Adobe Flash, ya que Steve Jobs lo consideraba inseguro, con errores y que incrementaba el consumo de batería, iOS apostó por HTML5 para sustituir Adobe Flash. Para programar solo permite el lenguaje Objective-C.

### Aplicaciones

Sin lugar a dudas la clave del éxito del iPhone, Apple fue la primera en apostar por una tienda de aplicaciones. Apple otorga el 70% de los ingresos al desarrollador y se queda con el 30%. El 3 de Marzo de 2012 cumplió 25 mil millones de descargas.

### Teléfonos

Solo disponible para iPhone. Actualmente se pueden adquirir tres modelos (3G, 4, 4S).

Por tanto, tras observar sus características se concluye estableciendo que IOS se trata de un gran sistema operativo, con un gran soporte que apenas cuenta con carencias. Existe una gran cantidad de información para poder desarrollar con esta plataforma.



### **2.3.1.3 BlackBerry** **BlackBerry**

Desarrollado por la empresa canadiense Research In Motion, tuvo sus inicios en el 1999. Incorporó a las funciones típicas de los teléfonos móviles el acceso al correo electrónico e internet. Aunque este sistema está orientado especialmente para un uso profesional (dar servicio a las empresas y profesionales), ha tenido un gran éxito entre los jóvenes debido a su servicio de mensajería instantánea. Su cuota de mercado ha sufrido una caída del 30%, siendo su cuota en el primer cuatrimestre de 2012 del 9.7%, frente al 13.6% con el que contaba apenas un año atrás.

### Ventajas

Para un determinado tipo de usuario puede ser una ventaja que la mayoría de terminales cuenten con un teclado físico, el cual es más cómodo para escribir. La mayoría de operadores cuentan con tarifas específicas para usuarios con BlackBerry. Permite varios lenguajes de programación para su desarrollo.

### Inconvenientes

Lo que para algunos casos era una ventaja, el teclado físico no está muy recomendado para navegar. Permite una escasa personalización de sus dispositivos. No posee tantas aplicaciones como, por ejemplo, Android e iOS.

### Aplicaciones

Su tienda de aplicaciones se denomina App World, es muy inferior respecto a las otras tiendas de aplicaciones, cuenta con muy pocas aplicaciones y casi ninguna de gran calidad. No ha tenido mucho apoyo de los desarrolladores y es una de las tiendas peor valoradas.

### Teléfonos

Se trata de una plataforma exclusiva de la marca BlackBerry. Se pueden encontrar modelos muy variados, desde dispositivos con pantalla táctil a teléfonos con teclado físico.

Blackberry es un sistema operativo muy orientado al mundo empresarial, que ofrece un gran gestor de correo y una buena agenda, pero no está muy indicado para el desarrollo de aplicaciones.

### 2.3.1.4 Windows Phone

Se trata del sucesor del sistema operativo Microsoft Mobile, el cual originalmente se llamó Pocket PC. Al contrario que BlackBerry OS está orientado a un mercado generalista. Se trata de un sistema más joven que sus rivales, y fue presentado en 2010. Microsoft continúa en su búsqueda de un sistema adecuado para los teléfonos móviles. En 2011 se anunció una alianza con Nokia, en la cual se convertirá en el principal sistema operativo de la compañía finlandesa. Su cuota de mercado ha aumentado un 26.9%, pero sigue siendo muy baja, ya que contaba con 2.6% en 2011, actualmente se cifra en un 3.3%.

#### Ventajas

Microsoft ha apostado fuerte por los servicios en la nube, cuenta con una gran compatibilidad con Windows Live. Se prevé una actualización para mediados de 2012, que se llamará Windows Phone 8 y que espera volverlo tan competitivo como iOS y Android.

#### Inconvenientes

Aún cuenta con muchas carencias, por ejemplo, no cuenta con centro de notificaciones, existe un problema para usuarios que se mudan de un país al otro ya que no admite el cambio del país de residencia en su cuenta de servicio, Microsoft ha sido denunciada por ello. Para transferir ficheros al móvil es necesario instalar Zune en el PC.

#### Aplicaciones

Cuenta con una gran tienda de reciente llegada pero de gran calidad, Market Place, aunque aún no existe gran variedad de aplicaciones. Microsoft permite dos formas distintas de implementar las aplicaciones, una es mediante Microsoft Silverlight y la otra es mediante Microsoft XNA Framework con la que también se pueden desarrollar juegos para XBOX 360.

#### Teléfonos

Cuenta con varios fabricantes. Samsung, LG, HTC cuentan con modelos con este sistema operativo, mientras que Nokia ha apostado por Windows Phone como su sistema operativo para sus nuevos dispositivos.

WindowsPhone cuenta con una gran plataforma, pero aún se tiene que consolidar en el mercado. Respecto al desarrollo, Microsoft cuenta con el Centro de Desarrollo de MSDN.

### 2.3.1.5 Symbian

Fue fruto de la alianza de varias empresas de telefonía móvil, entre las que se puede destacar Nokia, Sony Ericsson, Samsung, Siemens, etc. En sus inicios su objetivo era competir con Palm, un sistema operativo pensado para las PDAs. Es el sistema operativo que utilizan la mayoría de los teléfonos de Nokia, aunque, como se acaba de decir, esta

última ha pactado con Microsoft para incorporar Windows Phone en sus terminales. Pese a todo Nokia ha asegurado soporte hasta 2016. Su cuota de mercado ha caído un 60.6% respecto al año pasado. Actualmente su cuota de mercado es de 10.4% frente al 26% de 2011.

#### Ventajas

Su funcionamiento es correcto y se caracteriza por ser muy fácil de usar.

#### Inconvenientes

Se trata de un sistema operativo que tiene los días contados, por lo tanto, se desaconseja desarrollar para esta plataforma ya en decadencia, otro inconveniente es que solo permite C++ como lenguaje de programación.

#### Aplicaciones

La antigua Ovi Store ha sufrido un lavado de cara y cambio de nombre, ahora se conoce como Tienda Nokia. Se trata de una buena tienda de aplicaciones, pero no está a la altura de las de sus competidores.

#### Teléfonos

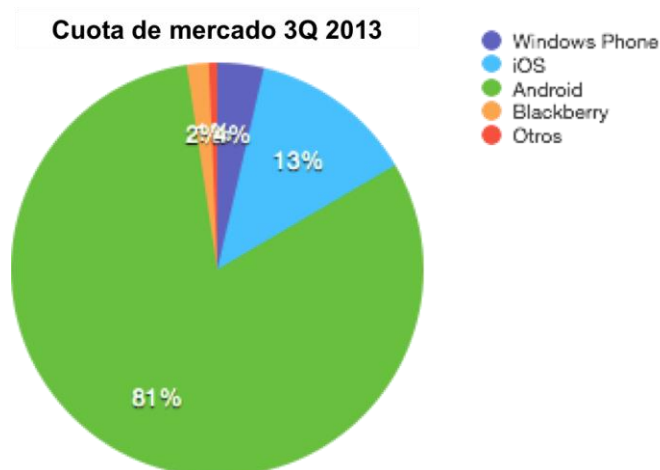
Se encuentra disponible en muchos de los teléfonos fabricados por Nokia, pero tras el pacto para incorporar Windows Phone en sus smartphones de gama alta, este sistema quedará relegado para los modelos de gama baja de Nokia.

Symbian es un sistema operativo muy robusto, pero escaso en opciones. Se ha quedado desfasado, por lo que Nokia optó por pactar con Microsoft. Debido a su próxima desaparición queda descartado desarrollar para esta plataforma.

## 2.4 Elección de la plataforma para el trabajo

Para la elección final del sistema operativo a emplear en el presente trabajo se ha tenido muy en cuenta la cuota de mercado y el crecimiento de estos sistemas operativos en los últimos años [SOMercado]. En la siguiente figura se puede observar la repartición del mercado de los teléfonos móviles.





**Figura 9. Gráfico cuota de mercado**

Si además se tiene en cuenta la caracterización realizada en apartados anteriores, es fácil desde un inicio descartar un par de plataformas, para las que no está teniendo éxito el desarrollar. En concreto se descarta BlackBerry y Symbian, ya que no cuentan con una tienda de aplicaciones a la altura de sus competidores y en el caso de Symbian está en pleno proceso de desaparición.

La siguiente plataforma en ser descartada es Windows Phone, se trata de una plataforma joven, tras los intentos de Microsoft de encontrar el sistema operativo perfecto que no pudo conseguir con Windows Mobile. Es probable que se asiente en el mercado y obtenga buenos resultados, pero aún está a mucha distancia de los dos sistemas operativos móviles principales: iOS y Android.

La lucha entre iOS y Android no tiene un vencedor claro y queda en manos del usuario decidir cuál se acerca más a sus preferencias y requisitos, en este caso se va a realizar sobre la plataforma Android, ya que permite la ejecución de la aplicación en el propio dispositivo sin la necesidad de adquirir una licencia. En el siguiente capítulo se explicará con detalle la plataforma Android.

## 2.5 Descripción detallada de la plataforma de trabajo usada: Android

Android está basado en Linux, sistema operativo libre, gratuito y multiplataforma. El lenguaje de programación que se utiliza para desarrollar aplicaciones para este sistema es una variante del lenguaje Java llamada Dalvik.

A continuación se describen las características y funcionalidades de Android en las que está basado el presente proyecto.

## Arquitectura de Android

Android [ANDROID] es una plataforma para dispositivos móviles que contiene una pila de software donde se incluye un sistema operativo, middleware y aplicaciones básicas para el usuario.

La arquitectura empleada en Android está estructura en capas. Cada una de estas capas utiliza servicios ofrecidos por las anteriores, y ofrece a su vez los suyos propios a las capas de niveles superiores, tal como muestra la siguiente figura:

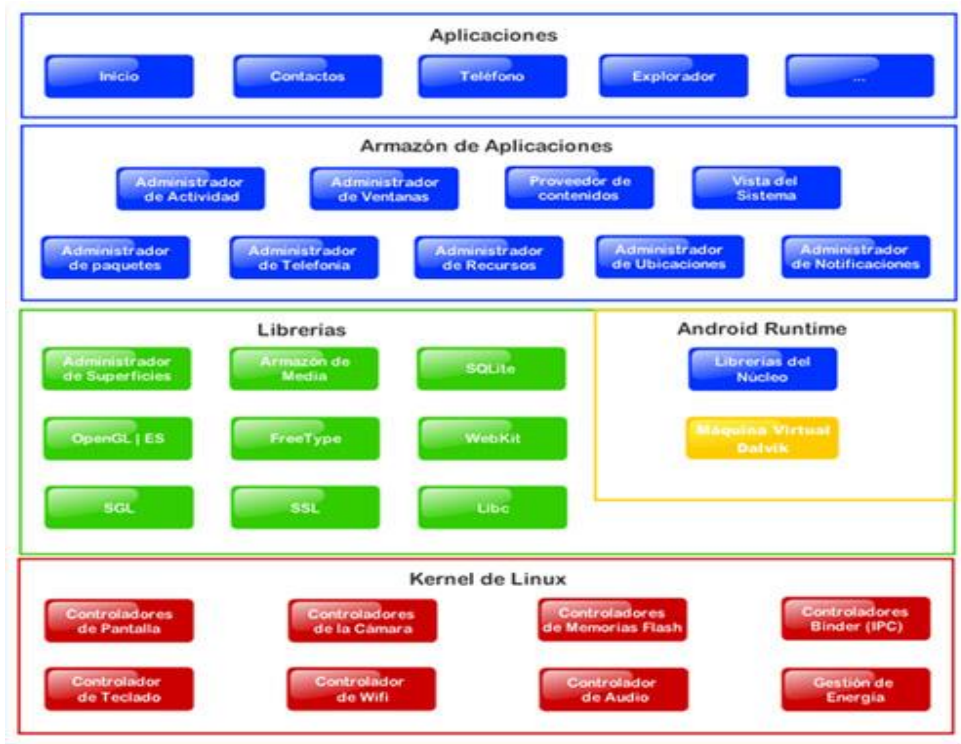


Figura 10. Arquitectura Android

## Núcleo Linux

Android utiliza el núcleo de Linux 2.6 como una capa de abstracción para el hardware disponible en los dispositivos móviles. Esta capa contiene los drivers necesarios para que cualquier componente hardware pueda ser utilizado mediante las llamadas correspondientes y proporciona servicios como la seguridad, el manejo de la memoria, el multiproceso, la pila de protocolos y el soporte de drivers para dispositivos.

Siempre que un fabricante incluye un nuevo elemento de hardware, lo primero que se debe realizar para que pueda ser utilizado desde Android es crear las librerías de control o drivers necesarios dentro de este kernel de Linux embebido en el propio Android.

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. El desarrollador no accede directamente a esta capa, sino que debe utilizar las librerías disponibles en capas superiores.

El kernel también se encarga de gestionar los diferentes recursos del teléfono (energía, memoria, etc.) y del sistema operativo en sí: procesos, elementos de comunicación (networking), etc.

## Librerías

La siguiente capa que se sitúa justo sobre el kernel la componen las bibliotecas nativas de Android, también llamadas librerías. Están escritas en C o C++ y compiladas para la arquitectura hardware específica del teléfono y proporcionan a Android la mayor parte de sus capacidades más características. El objetivo de las librerías es proporcionar funcionalidad a las aplicaciones para tareas que se repiten con frecuencia, evitando tener que codificarlas cada vez y garantizando que se llevan a cabo de la forma “más eficiente”. Junto al núcleo basado en Linux, estas librerías constituyen el corazón de Android.

Entre las librerías incluidas habitualmente se encuentran:

- Librería libc: Incluye todas las cabeceras y funciones según el estándar del lenguaje C. Todas las demás librerías se definen en este lenguaje.
- Librería Surface Manager: Es la encargada de componer los diferentes elementos de navegación de pantalla. Gestiona también las ventanas pertenecientes a las distintas aplicaciones activas en cada momento.
- OpenGL/SL y SGL (motor gráfico): Representan las librerías gráficas y, por tanto, sustentan la capacidad gráfica de Android. OpenGL/SL maneja gráficos en 3D y permite utilizar, en caso de que esté disponible en el propio dispositivo móvil, el hardware encargado de proporcionar gráficos 3D. Por otro lado, SGL proporciona gráficos en 2D, por lo que será la librería más habitualmente utilizada por la mayoría de las aplicaciones. Una característica importante de la capacidad gráfica de Android es que es posible desarrollar aplicaciones que combinen gráficos en 3D y 2D.
- Librería Media Libraries: Proporciona todos los códecs necesarios para el contenido multimedia soportado en Android (vídeo, audio, imágenes estáticas y animadas, etc.)
- FreeType (fuentes de texto): Permite trabajar de forma rápida y sencilla con distintos tipos de fuentes.
- Librería SSL (cifrado de comunicaciones): Posibilita la utilización de dicho protocolo para establecer comunicaciones seguras.
- Librería SQLite (base de datos): Creación y gestión de bases de datos relacionales.
- Librería WebKit (navegador): Proporciona un motor para las aplicaciones de tipo navegador y forma el núcleo del actual navegador incluido por defecto en la plataforma Android.

## Tiempo de ejecución de Android (Android Runtime)

Al mismo nivel que las librerías de Android se sitúa el entorno de ejecución, el entorno de ejecución de Android no se considera una capa en sí mismo. Éste lo constituyen las Core Libraries, que son librerías con multitud de clases con las funcionalidades habituales de Java así como otras específicas de Android y la máquina virtual Dalvik. Esta máquina virtual es el principal componente de ejecución de Android.

Las aplicaciones se codifican en Java y son compiladas en un formato específico para que esta máquina virtual las ejecute. Gracias a esto, las aplicaciones se compilan una única vez y de esta forma estarán listas para ejecutarse en cualquier dispositivo Android que disponga de la versión mínima del sistema operativo que requiera la aplicación.

### Framework de Aplicaciones

Representa fundamentalmente el conjunto de herramientas de desarrollo de cualquier aplicación, forma todas las clases y servicios que utilizan directamente las aplicaciones para realizar sus funciones y que se apoyan en las bibliotecas y en el entorno de ejecución que ya comentados. Toda aplicación que se desarrolle para Android utiliza el mismo conjunto de API y el mismo "framework", representado por este nivel. La mayoría de los componentes de esta capa son bibliotecas Java que acceden a los recursos a través de la máquina virtual Dalvik.

Entre las API más importantes ubicadas aquí, se pueden encontrar las siguientes:

- Activity Manager: Conjunto de API que gestiona el ciclo de vida de las aplicaciones en Android.
- Window Manager: Gestiona las ventanas de las aplicaciones y utiliza la librería Surface Manager.
- Telephone Manager: Incluye todas las API vinculadas a las funcionalidades propias del teléfono (llamadas, mensajes, etc.).
- Content Provider: Permite a cualquier aplicación compartir sus datos con las demás aplicaciones de Android.
- View System: Proporciona un gran número de elementos para poder construir interfaces de usuario (GUI), como listas, mosaicos, botones, "check-boxes", tamaño de ventanas, control de las interfaces mediante teclado, etc.
- Location Manager: Posibilita a las aplicaciones la obtención de información de localización y posicionamiento.
- Notification Manager: Comunican al usuario eventos que ocurran durante su ejecución: una llamada entrante, un mensaje recibido, etc.
- XMPP Service: Colección de API para utilizar este protocolo de intercambio de mensajes basado en XML.

### Aplicaciones

Este nivel contiene, tanto las incluidas por defecto de Android como aquellas que el usuario vaya añadiendo posteriormente, ya sean de terceras empresas o de su propio desarrollo. Todas estas aplicaciones utilizan los servicios, las API y librerías de los niveles anteriores.

#### 2.5.1.1 Versiones de Android

Una vez conocida la estructura de Android, es necesario conocer las distintas versiones que existen, pues cada una contiene clases y métodos que no contenía la anterior, para así saber cuál es la versión a utilizar para desarrollar la aplicación [ANDROIDver]. En el presente proyecto se ha usado el API 16 de Android y la versión Kernel 3.0.31.2429075.

A continuación se enumeran las versiones de Android y las principales características añadidas a cada una. Destacar que dichas versiones reciben nombres basados en postres en inglés, a cada versión le corresponde un postre que empieza por una letra distinta que sigue un orden alfabético.

### **Android 1.5 Nivel de API 3 (Cupcake)**

Es la primera versión con algún usuario. Incorpora la posibilidad de teclado en pantalla con predicción de texto y la capacidad de grabación de audio y vídeo. También aparecen los widgets de escritorio y live folders y permite conectarse automáticamente a auriculares bluetooth.

### **Android 1.6 Nivel de API 4 (Donut)**

Permite capacidades de búsqueda avanzada en todo el dispositivo. También se incorpora gestures y multi-touch. Permite la síntesis de texto a voz.

### **Android 2.0/2.1 Nivel de API 5 (Éclair)**

Destaca la incorporación de Google Maps. La versión 2.1 es una corrección de errores de la 2.0, motivo por el que conserva el mismo nombre. Incorpora una nueva funcionalidad que permite sincronizar adaptadores para conectarlo a cualquier dispositivo. Mejora la gestión de contactos y ofrece más ajustes en la cámara. Se ha optimizado la velocidad de hardware.

### **Android 2.2 Nivel de API 8 (Froyo)**

Rediseño de la pantalla de inicio, se aumenta el número de paneles de 3 a 5. 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 de acuerdo a varios benchmarks). Esto se consigue con la introducción de un nuevo compilador JIT de la máquina Dalvik.

### **Android 2.3 Nivel de API 9 (Gingerbread)**

La máquina virtual de Dalvik 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. Incorpora un nuevo interfaz de usuario con un diseño actualizado.

### **Android 3.x Nivel de API 11 (Honeycomb)**

Versión de Android diseñada específicamente para tabletas optimizada para dispositivos con pantallas grandes.

### **Android 4.0 Nivel de API 14 (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. Se introduce un nuevo interfaz de usuario totalmente renovado. Nuevo API

de reconocedor facial, permite entre otras muchas aplicaciones desbloquear el teléfono a su propietario. También se mejora en el reconocimiento de voz. Por ejemplo se puede empezar a hablar en cuanto se pulsa el botón.

### **Android 4.1 Nivel de API 16 (Jelly Bean)**

En esta versión se hace hincapié en mejorar un punto débil de Android: la fluidez del interfaz de usuario. Con este propósito se incorporan varias técnicas, como: sincronismo vertical, triple búfer y aumentar la velocidad del procesador al tocar la pantalla. Se mejoran las notificaciones con un sistema de información expandible personalizada. Se introducen varias mejoras en Google Search. Se potencia la búsqueda por voz con resultados en forma de ficha. La función Google Now permite utilizar información de posición, agenda y hora en las búsquedas.

### **Android 4.2 Nivel de API 17**

Se pueden crear varias cuentas de usuario en el mismo dispositivo (solo está disponible en tabletas). Los Widgets de escritorio pueden aparecer en la pantalla de bloqueo. Se incorpora un nuevo teclado predictivo deslizante. Nueva aplicación de cámara para hacer fotos panorámicas (en 360°).

### **Android 4.3 Nivel de API 18**

Los perfiles restringidos (disponible sólo en tabletas) que permiten controlar los derechos de los usuarios para ejecutar aplicaciones específicas y para tener acceso a datos específicos. Igualmente, los programadores pueden definir restricciones en las apps, que los propietarios puedan activar si quieren. Se mejora la seguridad para gestionar y ocultar las claves privadas y credenciales. Se incorpora un administrador de permisos, con el que se puede desactivar algunos de los permisos de las aplicaciones. En lo que se refiere a los gráficos, da soporte oficial a OpenGL ES 3.0.

### **Android 4.4 Nivel de API 19 (KitKat)**

Versión de Android en fase de desarrollo, todavía no ha sido presentada. Presentará nuevas funcionalidades tales como el modo de inmersión en pantalla completa oculta todas las interfaces del sistema (barras de navegación y de estado) de tal manera que una aplicación puede aprovechar el tamaño de la pantalla completa. WebViews (componentes de la interfaz de usuario para mostrar las páginas Web) se basa ahora en el software de Chrome de Google y por lo tanto puede mostrar contenido basado en HTML5. Todos los componentes principales de Android han sido recortados para reducir sus requerimientos de memoria, y se ha creado una nueva API que permite adaptar el comportamiento de la aplicación en dispositivos con poca memoria.

## **2.5.1.2 Estructura de una aplicación Android**

Las aplicaciones en Android están basadas en componentes [ANDROIDEstruc]. Existen cuatro tipos de componentes: *Activities*, *Services*, *Broadcast receivers*, *Content providers*. Una aplicación Android será una combinación de uno o más de estos componentes y deberán ser declarados de forma explícita en el fichero `AndroidManifest.xml`

## Activity

Modela una actividad llevada a cabo por una aplicación. Conjunto de elementos básicos de visualización, coloquialmente conocidos como pantallas de la aplicación.

Su función principal es la creación del interfaz de usuario. Una aplicación suelen necesitar varias actividades para crear el 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. Existe una actividad inicial que es la que se le presenta al usuario cuando se arranca la aplicación. Se implementan como subclase de la clase Activity.

## Service

Modela una actividad ejecutada en segundo plano sin interfaz gráfica. Se implementa como subclase de la clase Service. En Android existen de dos tipos de servicios: servicios locales, que son ejecutados en el mismo proceso y servicios remotos, que son ejecutados en procesos separados.

## Broadcast receiver (Receptor de anuncios)

No realizan ninguna acción, sólo reciben y reaccionan ante anuncios enviados por difusión. Normalmente reaccionan lanzando alguna actividad que procese la información recibida. Las aplicaciones pueden generar sus propios anuncios y difundirlos para avisar a otras aplicaciones. Extienden la clase BroadcastReceiver.

## Content provider

Modela un conjunto de datos de una aplicación para que puedan estar disponibles para otras aplicaciones, es decir, distintas aplicaciones comparten información. Los datos pueden estar almacenados en el sistema de ficheros, en una base de datos, o cualquier otro almacenamiento. Las aplicaciones no interactúan directamente con ellos, deben manejarlos a través de ContentResolver. Extiende la clase ContentProvider.

Para la *activación* de los tres primeros componentes comentados anteriormente es necesario el uso de Intents. Los Intents modelan mensajes asíncronos (clase Intent). Para activities y services nos indican la acción solicitada y una URI indicando dónde están los datos a los que se les aplica la acción. Para broadcast receivers nos indica la acción que debe ser anunciada. Content provider se activan a través de ContentResolver.

En cuanto a la *parada* de componentes, los content providers y broadcast receivers tienen por definición un tiempo limitado de actividad, no necesitan pararse explícitamente. Destacar que Android proporciona métodos para parar de forma ordenada activities y services que están en ejecución. Cualquier componente puede ser finalizado por el sistema cuando se detecta que no va a utilizarse o se necesitan recursos para activar nuevos componentes

Las activities, services y broadcast receivers tienen sus propios ciclos de vida. Este ciclo de vida se compone de tres estados [ANDROIDvida]:



- Activa o en ejecución: Está en primer plano y tiene el foco de interacción con el usuario, lo que quiere decir que es visible.
- Pausada: La actividad ha perdido el foco pero sigue siendo visible para el usuario. Sigue siendo una actividad “viva” pero podría ser eliminada por el sistema en cualquier momento (normalmente, situaciones extremas de baja memoria). Cuando una actividad está tapada por completo, pasa a estar parada.
- Parada: Está totalmente oculta por otra actividad. Se mantiene información de su estado por si es necesario recuperarla. El sistema frecuentemente elimina estas actividades para liberar memoria del sistema.
- Destroyed: Cuando la actividad termina al invocarse el método finish (), o es matada por el sistema.

Cuando una Activity cambia de estado se le informa invocando los siguientes métodos:

- *onCreate ()*, *onDestroy ()*: Abarcan todo el ciclo de vida. Representan el principio y fin de la activity.
- *onStart ()*, *onStop ()*: Representan la parte visible del ciclo de vida. Es visible aunque puede no tener el foco de acción con el usuario. Pueden invocarse múltiples veces.
- *onResume ()*, *opuse ()*: Abarcan la parte útil. Es visible y además tiene el foco de acción con el usuario.

En la Figura 11 se muestra el ciclo de vida de una Activity.

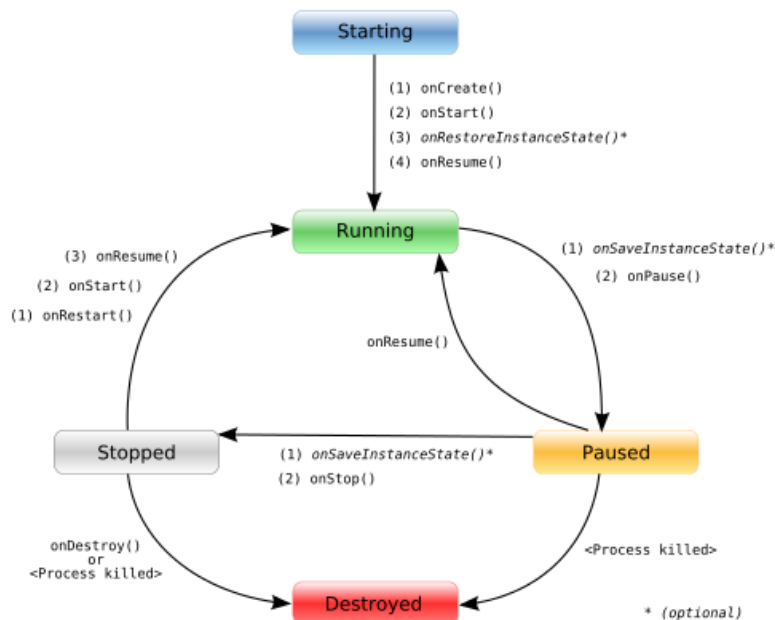


Figura 11. Ciclo de vida de una Activity





# Capítulo 3

## Descripción general del sistema

En el presente capítulo se va a llevar a cabo la descripción de las características fundamentales de la aplicación, se va a realizar también un amplio resumen de la plataforma elegida, así como las herramientas utilizadas para su desarrollo.

### 3.1 Presentación del sistema

El objetivo fundamental de este TFG es desarrollar una asistente multimodal que sea capaz de gestionar consultas relativas a un servicio de biblioteca digital, ya sea mediante el diálogo con él o mediante el uso de interfaces tradicionales como puede ser la pantalla y el teclado. La aplicación funcionará en dispositivos móviles con sistema operativo Android

En la Figura 12 se muestra el esquema general de la aplicación.

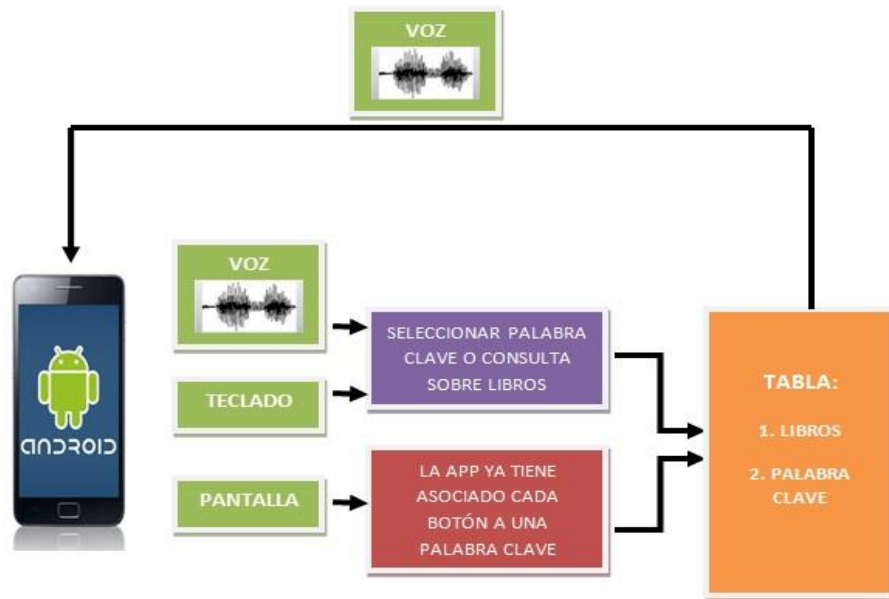


Figura 12. Esquema general de la aplicación

En dicho esquema, al ser una aplicación multimodal, pone de manifiesto que se puede obtener el contenido de la información mediante la voz, mediante el teclado, o pulsando botones.

El usuario cuenta con tres posibles maneras para realizar la consulta. Una de ellas es mediante el teclado, el usuario puede escribir cuál es la consulta que quiere realizar y la aplicación se encarga de procesar dicha información y proporcionarle al usuario la solución que más se adecue a la solicitud. Otra es acceder, mediante los distintos botones que posee la aplicación, a consultas específicas, sin necesidad de hablar o de escribir. Por ejemplo, existen botones que según se pulsan informan directamente de: el horario de la biblioteca, distribución de la biblioteca, periodo de exámenes, cómo enviar un correo a la biblioteca, cómo reservar un libro, dónde se encuentran los trabajos fin de grado, listado de los libros de la biblioteca y si están disponibles o no, etc.

Por último, la otra manera de acceder es mediante la voz. Gracias al reconocimiento automático del habla (RAH), la aplicación es capaz de procesar la información que le está solicitando el usuario, consultar la base de datos y proporcionarle la respuesta que más se acerque a la petición. Esto último se encarga de llevarlo a cabo el sintetizador de texto a voz (CTV).

A continuación se enumeran las funcionalidades que lleva a cabo la aplicación:

- Mostrar el listado de libros de la Biblioteca UC3M de Leganés, con todas sus características (Título, Autor, ISBN, Año, disponibilidad).
- Visualizar en un mapa la ubicación del libro solicitado.
- Mostrar la normativa para reservar libros.
- Acceso directo a la página Web de la universidad para reservar un libro o para enviar un correo a la biblioteca.
- Informar al usuario sobre:
  - Horario de la biblioteca.

- Distribución de la biblioteca.
- Periodo de exámenes.
- Donde se encuentran los trabajos fin de grado.

## 3.2 Herramientas usadas

### 3.2.1 Eclipse y SDK Android

El programa utilizado para implementar el código de la aplicación es Eclipse [ECLIPSE]. Se trata de un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma. Para poder desarrollar la aplicación en Android es necesario instalarse el SDK Android, kit de desarrollo de aplicaciones Android. Este software incluye paquetes con todas las APIs, documentaciones, ejemplos y versiones de Android, además de otras utilidades extras. Se encuentra disponible en [developer.android.com](http://developer.android.com). También es necesario añadir el plug-in ADT que incorpora un conjunto de herramientas y mejoras que aceleran el desarrollo de proyectos Android.

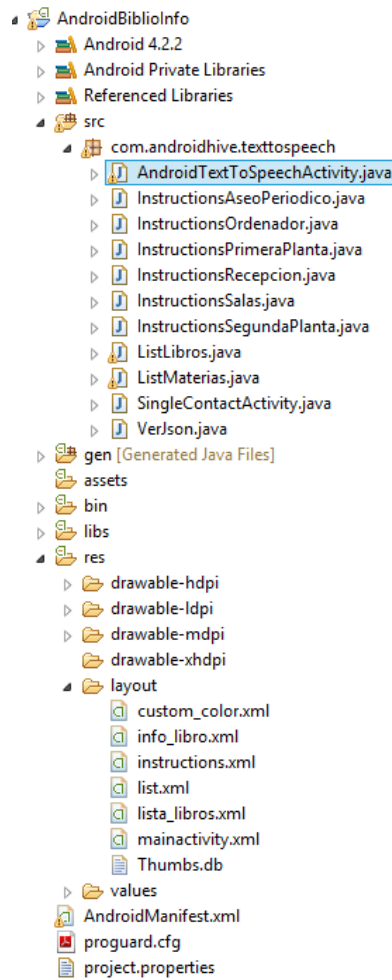


Figura 13. Esquema del TFG en Eclipse

En la Tabla 2 se detalla la principal funcionalidad de las clases mostradas anteriormente:

Clase	Funcionalidad
AndroidTextToSpeechActivity.java	Realiza las acciones descritas en el módulo de Inicio.
InstructionsAseoPeriodico.java	Muestra las indicaciones para llegar a los aseos o a los periódicos.
InstructionsOrdenador.java	Muestra las indicaciones para llegar a la sala de ordenadores.
InstructionsPrimeraPlanta.java	Muestra las indicaciones para llegar a la primera planta.
InstructionsRecepcion.java	Muestra las indicaciones para llegar a la recepción.
InstructionsSalas.java	Muestra las indicaciones para llegar a las salas de grupo.
InstructionsSegundaPlanta.java	Muestra las indicaciones para llegar a la segunda planta.
SingleContactActivity.java	Accediendo a la base de datos muestra las características detalladas de cada libro.
ListMaterias.java	Muestra las distintas materias.
ListLibros.java	Se conecta con la base de datos, recupera y muestra los libros contenidos en ella.
VerJson.java	Establece la conexión con la base de datos y recupera los datos en formato JSONArray.

Tabla 2. Activitys de la Aplicación

### 3.2.2 MySQL y PHP. Almacenamiento de datos en el módulo palabras clave y libros.

Se realiza a continuación una descripción detallada de las tecnologías utilizadas en los módulos de *búsqueda de palabras clave* y *búsqueda de libros*, cuya funcionalidad principal es consultar los datos almacenados en la base de datos MySQL. Esta base de datos consta de dos tablas:

1. Tabla *palabras clave*:

Columnas	Función
<b>Id</b>	Cada id está asociado a una palabra clave.
<b>Request.</b>	Se encuentran almacenadas las palabras clave que posteriormente son comparadas con las palabras dichas por el usuario.
<b>Response</b>	Texto asociado a cada palabra clave que se le proporciona al usuario mediante la voz, contestando a la consulta solicitada.

Tabla 3. Tabla palabras clave que contiene las posibles palabras dichas por el usuario y la respuesta asociada a cada una

2. Tabla *libros*:

Columnas	Función
<b>Id</b>	Cada libro tiene asociado un Id.
<b>Materia</b>	Materia a la que pertenece el libro.
<b>Código</b>	Códigos de cada libro.
<b>Título</b>	Título del libro.
<b>Autor</b>	Autor del mismo.
<b>Editorial</b>	Editorial a la que pertenece.
<b>ISBN</b>	Numero ISBN identificativo de cada libro.
<b>Disponibilidad</b>	Muestra si un libro está disponible actualmente en la biblioteca o se encuentra prestado. Si se encuentra, se indica exactamente su ubicación.

**Tabla 4. Tabla libros que contiene la información detallada de cada libro**

La razón por la cual se ha decidido usar una base de datos es por la simplicidad que proporciona a la hora de usar la aplicación para otras Bibliotecas distintas a la Biblioteca UC3M de Leganés. La funcionalidad primordial de la aplicación es hacer consultas a la base de datos, obteniendo la información de la tabla que se requiera en cada momento dependiendo de la consulta del usuario.

Actualmente el API de Android no dispone de ningún recurso que permita el acceso directamente a las bases de datos MySQL, por ello, en el presente Trabajo Fin de Grado se ha desarrollado un servicio web. La aplicación Android (cliente) envía una consulta mediante HTTP a un servidor web que accede a la base de datos MySQL y devuelve la información solicitada a la aplicación Android en formato JSON. Por esta razón los módulos de *búsqueda de palabras clave* y *búsqueda de libros* están formados por dos partes complementarias:

1. *Parte web*. Se ha utilizado Xampp [XAMPP], servidor web que incluye bases de datos MySQL, un servidor web Apache y el intérprete para lenguaje de script PHP. Además, proporciona acceso al administrador de base de datos: PhpMyAdmin.
  - 1.1 Lenguaje utilizado: PHP
  - 1.2 Base de datos: MySQL
  - 1.3 Servidor: Apache
2. *Parte Android*. En la parte del cliente se ejecuta una petición HTTP para conectarse a la parte web, obteniendo una respuesta en formato JSON.

### 3.2.2.1 Servidor web: Xampp

Como servidor web se ha utilizado Xampp para el almacenamiento de los ficheros PHP y de las bases de datos de la aplicación. XAMPP es el acrónimo de Cualquier Plataforma (X), Apache (A), MySQL (M), PHP (P) y Perl (P) [XAMPP1]. Se listan a continuación sus características más importantes:

- Para Windows existen dos versiones, una con instalador y otra portable para descomprimir y ejecutar.
- La licencia de esta aplicación es GNU (General PublicLicense), orientada a proteger la libre distribución, modificación y uso de software.
- La filosofía de XAMPP, como lo indican en su sitio web, es crear una distribución fácil de instalar.
- XAMPP solamente requiere descargar y ejecutar un archivo .zip.
- Sistema multiplataforma, es decir, existen versiones para diferentes sistemas operativos.

En la Tabla 5 se muestran las principales ventajas y desventajas del servidor web utilizado:

Ventajas	Desventajas
Herramienta muy práctica que nos permite instalar el entorno MySQL, Apache y PHP. trae otros servicios como servidor de correos y servidor FTP	No soporta MySQL desde la consola. Xampp trae PhpMyAdmin para administrar las bases de datos de MySQL, sin embargo para tareas más específicas es mejor utilizar la consola (línea de comandos) y Xampp no la soporta
Instalación es de lo más sencilla y configuraciones son mínimas o inexistentes	Dificultad para configurar aplicaciones de terceros.

**Tabla 5. Ventajas y desventajas de Xampp**

### **Utilización de Xampp en el módulo de búsqueda de palabras clave y módulo de búsqueda de libros**

Para desarrollar el *módulo Búsqueda de palabras clave* se utilizan las siguientes utilidades proporcionadas por el servidor Xampp:

- Servidor Apache/2.4.9 (Win32) OpenSSL/1.0.1g
- Base de datos MySQL (versión del cliente de base de datos: libmysql - mysqlnd 5.0.11-dev)
- PHP/5.5.11
- Administrador de base de datos phpMyAdmin 4.1.12
- ProFTPD: un servidor FTP.
- OpenSSL: para soporte a la capa de sockets segura.

Pasos llevados a cabos para su uso:

1. Activar el panel de control. El panel de control de XAMPP le da un control completo sobre todos los componentes de XAMPP instalados. Puede usar el

panel de control para iniciar o parar distintos módulos, lanzar una ventana de comandos UNIX, abrir el explorador de Windows, y ver todas las operaciones que se ejecutan en segundo plano.

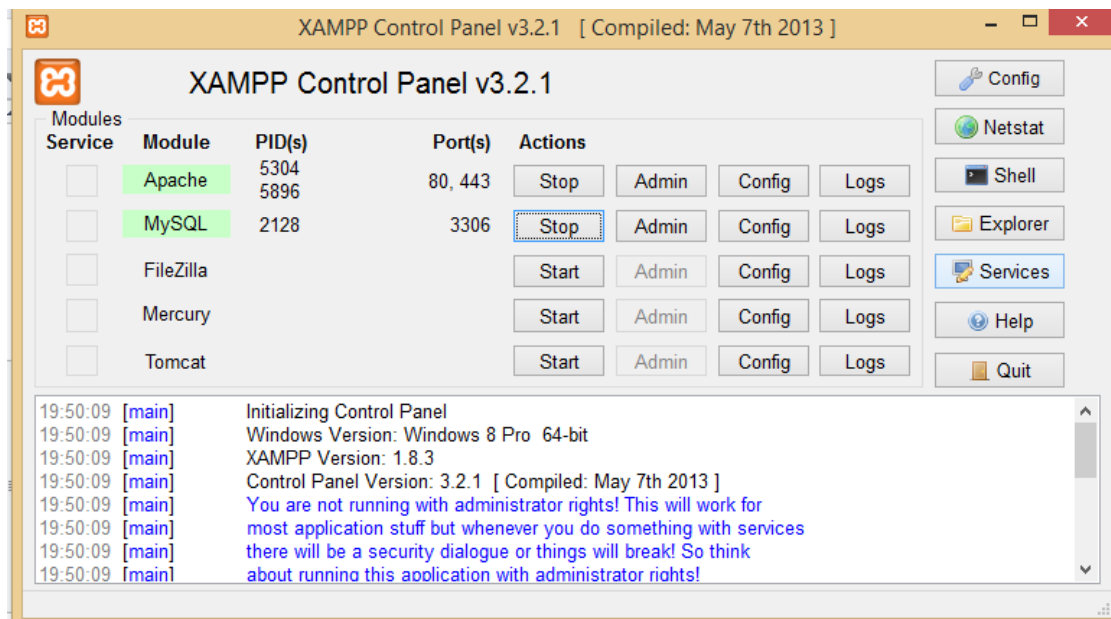


Figura 14. Panel de control de XAMPP

2. Activar Apache y MySQL.
3. Escribir en el navegador `http://localhost` o `127.0.0.1` seguido de la ruta donde se encuentre el archivo `.php`, en nuestro caso: `http://127.0.0.1/json/palabrasclave.php`
4. Se muestra el contenido de la base de datos palabras clave en formato JSON.

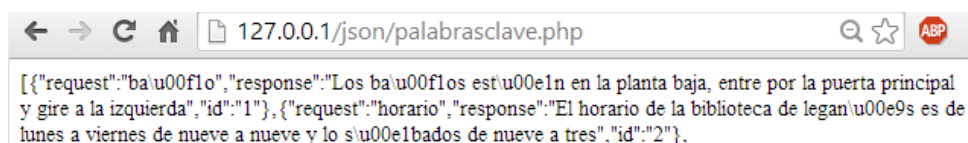


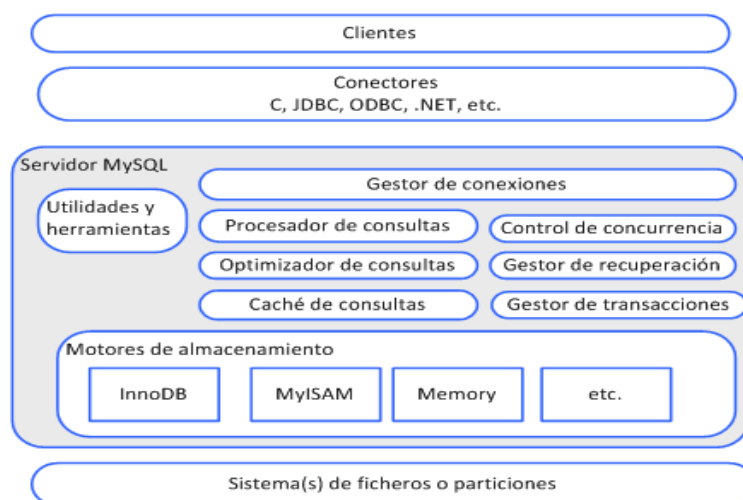
Figura 15. Representación en formato JSON de los valores obtenidos al realizar la petición.

### 3.2.2.2 MySQL

MySQL es un sistema de gestión de Base de Datos relacional, multiusuario y multiusuario con más de seis millones de instalaciones. [MySQL].

La arquitectura de MySQL tiene como característica más notable separar el motor de almacenamiento (que se encarga de los detalles de entrada-salida y representación de la información en memoria secundaria) del resto de los componentes de la arquitectura. Es decir, el diseño del gestor está preparado para que se pueda cambiar el gestor de almacenamiento. Esto permite incluso crear nuevos motores de almacenamiento especializados para ciertas tareas o tipos de aplicaciones [MySQLcaract].





**Figura 16. Arquitectura MySQL**

La Figura 16 muestra la división entre los componentes que conforman el servidor, las aplicaciones cliente que lo utilizan y las partes del sistema operativo en las que se basa el almacenamiento físico.

Las utilidades y herramientas de MySQL son los programas y aplicaciones que se incluyen con la distribución del gestor, o que pueden instalarse como aplicaciones adicionales. Estas incluyen las herramientas de backup, el navegador de consultas (QueryBrowser), las aplicaciones administrativas de interfaz gráfico y la herramienta de diseño MySQL Workbench, entre otras.

El elemento más notable de la arquitectura de MySQL es la denominada arquitectura de motores de almacenamiento reemplazables (pluggable storage engine architecture). La idea de esa arquitectura es hacer una interfaz abstracta con funciones comunes de gestión de datos en el nivel físico. De ese modo, el gestor de almacenamiento puede intercambiarse, e incluso un mismo servidor MySQL puede utilizar diferentes motores de almacenamiento para diferentes bases de datos o para diferentes tablas en la misma base de datos. Esto permite utilizar el motor de almacenamiento más adecuado para cada necesidad concreta. También permite que terceros puedan implementar motores de almacenamiento nuevos para necesidades específicas, o adaptar el código de los existentes a ciertos requisitos de almacenamiento.

De este modo, las interfaces definidas por MySQL aíslan el resto de los componentes de la arquitectura de las complejidades de la gestión física de datos, facilitando el mantenimiento de los motores de almacenamiento. También esto permite que ciertos motores de almacenamiento no implementen parte de los servicios, lo cual les hace inapropiados para algunas aplicaciones pero más eficientes para otras. Por ejemplo, un motor de almacenamiento que no implementa bloqueos en la base de datos no debe utilizarse en aplicaciones multi-usuario, pero la ausencia de sobrecarga de procesamiento en la gestión de los bloqueos para el acceso concurrente lo hará mucho más eficiente para una aplicación monousuario.

En consecuencia, una primera tarea de diseño físico en MySQL es la de decidir el motor de almacenamiento más apropiado. Los elementos que puede implementar un motor de almacenamiento son los siguientes:

- Concurrencia. Es responsabilidad del motor implementar una política de bloqueos (o no implementar ninguna). Una estrategia de bloqueos por fila permite una mayor concurrencia, pero también consume más tiempo de procesamiento en aplicaciones en las que la concurrencia no es realmente grande.
- Soporte de transacciones. No todas las aplicaciones necesitan soporte de transacciones.
- Comprobación de la integridad referencial, declarada como restricciones en el DDL de SQL.
- Almacenamiento físico, incluyendo todos los detalles de la representación en disco de la información.
- Soporte de índices. Dado que la forma y tipo de los índices depende mucho de los detalles del almacenamiento físico, cada motor de almacenamiento proporciona sus propios métodos de indexación (aunque algunos como los árboles B casi siempre se utilizan).
- Cachés de memoria. La eficiencia de los cachés de datos en memoria depende mucho de cómo procesan los datos las aplicaciones. MySQL implementa cachés comunes en el gestor de conexiones y la caché de consultas, pero algunos motores de almacenamiento pueden implementar cachés adicionales.
- Otros elementos para ayudar al rendimiento, como puede ser el uso de múltiples hilos para operaciones paralelas o mejoras de rendimiento para la inserción masiva.

Las ventajas y desventajas del MySQL se resumen en la Tabla 6:

VENTAJAS	DESVENTAJAS
Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento	Un gran porcentaje de las utilidades de MySQL no están documentadas
Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema	No es intuitivo, como otros programas (ACCESS)
Facilidad de configuración e instalación	
Soporta gran variedad de Sistemas Operativos	
Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.	
Conectividad y seguridad	

Tabla 6. Ventajas y desventajas del MySQL

### Utilización de MySQL en el módulo de búsqueda de palabras clave

Como se ha explicado anteriormente, Xampp incluye un sistema de gestión de base de datos MySQL. Para el desarrollo del *módulo de búsqueda de palabras clave* y *módulo de búsqueda de libros* se ha creado una base de datos MySQL llamada android cuya finalidad principal es la de almacenar en tablas la información referente a la biblioteca. Esta base de datos, Android, consta de dos tablas:

1. Tabla *palabras clave*:

Columnas	Función	Tipo	Cotejamiento
<b>Id</b>	Cada id está asociado a una palabra clave.	bigint(20)	
<b>Request.</b>	Se encuentran almacenadas las palabras clave que posteriormente son comparadas con las palabras dichas por el usuario.	varchar(50)	utf8_spanish_ci
<b>Response</b>	Texto asociado a cada palabra clave que se le proporciona al usuario mediante la voz, contestando a la consulta solicitada.	varchar(500)	utf8_spanish_ci

**Tabla 7.** Tabla palabras clave indicando el tipo y el cotejamiento usado

2. Tabla *libros*:

Columnas	Función	Tipo	Cotejamiento
<b>Id</b>	Cada libro tiene asociado un Id.	Bigint(20)	
<b>Código</b>	Códigos de cada libro.	Varchar(200)	latin1_swedish_ci
<b>Título</b>	Título del libro.	Varchar(100)	latin1_swedish_ci
<b>Autor</b>	Autor del mismo.	Varchar(100)	latin1_swedish_ci
<b>Editorial</b>	Editorial a la que pertenece.	Varchar(100)	latin1_swedish_ci
<b>ISBN</b>	Numero ISBN identificativo de cada libro.	Varchar(100)	latin1_swedish_ci
<b>Disponibilidad</b>	Muestra si un libro está actualmente en la biblioteca y su ubicación o si se encuentra prestado.	Varchar(200)	latin1_swedish_ci

**Tabla 8.** Tabla libros indicando el tipo y el cotejamiento usado

Para la creación de la base de datos android y de las tablas comentadas se ha utilizado la interfaz del administrador de base de datos PhpMyAdmin. Posteriormente, la manipulación de los datos se ha realizado desde los ficheros PHP almacenados en el servidor. En el punto 3.2.2.4 se explica y muestra como se ha llevado a cabo esta manipulación y que comandos se han utilizado para ello.

### 3.2.2.3 phpMyAdmin

phpMyAdmin es una herramienta escrita en PHP con la intención de manejar la administración de bases de datos MySQL a través de páginas web, utilizando Internet. Actualmente puede crear y eliminar Bases de Datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios, exportar datos en varios formatos y está disponible en 62 idiomas. Se encuentra disponible bajo la licencia GPL. En la Figura 17 se muestra la pantalla principal de phpMyAdmin.



Figura 17. phpMyAdmin

Las Figuras 18 y 19 muestran las tablas de la base de datos Android en phpMyAdmin comentadas en el apartado anterior.

id	codigo	titulo	autor	editorial	isbn	disponibilidad
1	L/S 621.3.01 SAV	Sistemas y circuitos	Savant C J	Pearson Educacion	2147483647	Se encuentra actualmente disponible en la bibliote...
2	L/S 517.95 HAB	Ecuaciones en derivadas parciales con series de Fo...	Haberman, Richard	Pearson-Prentice Hall	8420535346	Actualmente no se encuentra disponible en la bibli...
3	L/S 004.438 JAVA PRO	Programación en Java	Sánchez Allende, Jesús	McGraw Hill	9788448161071	Se encuentran disponibles 2 ejemplares en la planta uno y 1 ejemplar en acceso electrónico
4	L/S 53 ALO	Fisica	Alonso, Marcelo	Addison-Wesley Iberoamericana	0201002817	Actualmente todos los ejemplares están prestados
5	D 512 LOR	Algebra	Lorenz, Falko	Springer	9780387289304	Actualmente todos los ejemplares están prestados
6	L/S 531 JUA	Mecánica : problemas de exámenes resueltos	Juana Sardón, José María de	Paraninfo	8428320535	Están disponibles 4 libros. Planta 1
7	L/S 004.451.9	Android : guía para desarrolladores	Ableson, W. Frank	Anaya Multimedia	9788441526822	Existen solo dos manuales en la biblioteca y actua...
8	L/S 532 FLU	Fluidos : estática y dinámica	Abad Toribio, Laura	García-Maroto	9788415214892	Se encuentran disponibles 2 ejemplares en la plant...
9	L/S 004.438 C JOY	Abad Toribio, Laura	Joyanes Aguilar, Luis	McGraw Hill	8448198441	6 copias disponibles en UC3M Biblioteca Escuela Po

Figura 18. Presentación de la Tabla de libros

	request	response	id
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	baño	Los baños están en la planta baja, entre por la pu...	1
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	horario	Puede ver el horario detallado por días en el sigu...	2
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	gracias	A ti por la consulta	4
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	hola	hola ¿qué consulta quieres hacer?	6
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	distribución	la biblioteca tiene tres plantas, en la planta baj...	7
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	trabajo	Los trabajos fin de grado se encuentran en la segu...	8
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	periódico	los periódicos se encuentran según entra por la en...	9
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	grupo	las salas de grupo se encuentran en la tercera pla...	10
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	recepción	la recepción se encuentra en la planta baja, entre...	11
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	normativa	Se podrán reservar todos los documentos prestados,...	12
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	reservar	¿Quiere reservar un libro o quiere conocer la norm...	13
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	ordenador	Los ordenadores se encuentran en la tercera planta...	14
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	aseo	Los aseos están en la planta baja, entre por la pu...	15
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	renovar	Todos los documentos prestables se podrán renovar ...	17
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	video	a continuación se muestra un vídeo explicativo de ...	18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	sancion	En caso de no devolución, la Biblioteca requerirá ...	19
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	enlace	Para reservar un libro introduzca su NIA y su cont...	20
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	planos	a continuación se descargará en su dispositivo los...	21
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	circuitos	El libro de circuitos eléctricos se encuentra en l...	22
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	libro	Diga el nombre del libro que desea buscar. Si qui...	23
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	biblioteca	Se encuentra en la biblioteca de Leganés, campus d...	24

Figura 19. Presentación de la Tabla de palabras clave.

### 3.2.2.4 PHP (PHP Hypertext Pre-processo)

PHP es un lenguaje de programación de uso general de código del lado del servidor diseñado para el desarrollo web de contenido dinámico. Las características principales del lenguaje se resumen en la Tabla 9.

Características
Orientad al desarrollo web dinámico con acceso a información almacenada en una base de datos
Es considerado un lenguaje fácil de aprender, ya que en su desarrollo se simplificaron distintas especificaciones, como es el caso de la definición de las variables primitivas
El código fuente escrito en PHP es invisible al navegador web y al cliente, ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable
Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL
Capacidad de expandir su potencial utilizando módulos (llamados ext's o extensiones)
Posee una amplia documentación en su sitio web oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda
Es libre, por lo que se presenta como una alternativa de fácil acceso para todos
Permite aplicar técnicas de programación orientada a objetos
No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución
Tiene manejo de excepciones (desde PHP5)
Debido a su flexibilidad ha tenido una gran acogida como lenguaje base para las aplicaciones WEB de manejo de contenido, y es su uso principal

El programador puede aplicar en su trabajo cualquier técnica de programación o de desarrollo que le permita escribir código ordenado, estructurado y manejable

**Tabla 9. Características principales del lenguaje**

Actualmente, PHP está enfocado principalmente a la programación de scripts del lado del servidor, por lo que se puede hacer cualquier cosa que pueda hacer otro programa CGI, como recopilar datos de formularios, generar páginas con contenidos dinámicos, o enviar y recibir cookies. Aunque PHP puede hacer mucho más [PHP].

Existen principalmente tres campos principales donde se usan scripts de PHP, resumidos en la Tabla 10.

Campos	
<b>Scripts del lado del servidor</b>	Este es el campo más tradicional y el foco principal. Son necesarias tres cosas para que esto funcione. El analizador de PHP (módulo CGI o servidor), un servidor web y un navegador web. Es necesario ejecutar el servidor con una instalación de PHP conectada. Se puede acceder al resultado del programa de PHP con un navegador, viendo la página de PHP a través del servidor. Todo esto se puede ejecutar en su máquina si está experimentado con la programación de PHP.
<b>Scripts desde la línea de comandos</b>	Se puede crear un script de PHP y ejecutarlo sin necesidad de un servidor o navegador. Solamente es necesario el analizador de PHP para utilizarlo de esta manera. Este tipo de uso es ideal para scripts que se ejecuten con regularidad empleando cron (en *nix o Linux) o el Planificador de tareas (en Windows). Estos scripts también pueden usarse para tareas simples de procesamiento de texto.
<b>Escribir aplicaciones de escritorio</b>	Probablemente PHP no sea el lenguaje más apropiado para crear aplicaciones de escritorio con una interfaz gráfica de usuario, pero si se conoce bien PHP, y se quisiera utilizar algunas características avanzadas de PHP en aplicaciones del lado del cliente, se puede utilizar PHP-GTK para escribir dichos programas. También es posible de esta manera escribir aplicaciones independientes de una plataforma.

**Tabla 10. Campos principales donde se usan scripts de PHP**

Por último, la Tabla 11 resume los principales inconvenientes que presenta este lenguaje de programación.

Inconvenientes
Como es un lenguaje que se interpreta en ejecución, para ciertos usos puede resultar un inconveniente que el código fuente no pueda ser ocultado. La ofuscación es una técnica que puede dificultar la lectura del código pero no necesariamente impide que el código sea examinado.
Debido a que es un lenguaje interpretado, un script en PHP suele funcionar considerablemente más lento que su equivalente en un lenguaje de bajo nivel, sin embargo este inconveniente se puede minimizar con técnicas de caché tanto en archivos como en memoria.
Las variables al no ser tipificadas dificulta a los diferentes IDEs para ofrecer asistencias para el tipificado del código, aunque esto no es realmente un inconveniente del lenguaje en sí. Esto es solventado por Zend Studio añadiendo un comentario con el tipo a la declaración de la variable.

Tabla 11. Inconvenientes de PHP

### Utilización de PHP en el módulo Búsqueda de palabras clave y módulo de búsqueda de libros

Para el acceso a las tablas de la base de datos MySQL se han desarrollado ficheros PHP en el lado del servidor:

- **palabrasclave.php:** realiza la configuración necesaria para conectar a la base de datos, acceder a ella obtener los valores contenidos en la tabla palabrasclave y devolver el resultado en formato JSON.
- **libros.php:** al igual que el fichero anterior, realiza la configuración necesaria para conectar a la base de datos, acceder a ella obtener los valores contenidos en la tabla libros y devolver el resultado en formato JSON.

Los comandos usados para la implementación de dichos ficheros php son, se resumen en la Tabla 12.

Comando usado	Función que realiza
<code>mysql_connect('localhost:3306','root','')or die('no');</code>	Abre una conexión al servidor MySQL.
<code>mysql_query("SET NAMES 'utf8'");</code>	Enviar una consulta MySQL y establecer el charset de la base de datos.
<code>mysql_select_db('android');</code>	Seleccionar la base de datos MySQL llamada android.
<code>\$sql="SELECT * FROM palabrasclave";</code>	Obtener los datos contenidos en la tabla palabrasclave.
<code>\$resultado= mysql_query(\$sql);</code>	Almacenar los datos obtenidos tras realizar la consulta.



<code>while(\$row=mysql_fetch_assoc(\$resultado)){ \$arr[]=\$row;}</code>	Recuperar todas las filas de resultados como un array.
<code>\$json=json_encode(\$arr);</code>	Devolver la representación en JSON de cada valor del array.

**Tabla 12. Comandos usados en el fichero .php**

### 3.2.2.5 HTTP (Hypertext Transfer Protocol)

HTML, siglas de HyperText Markup Language («lenguaje de marcas de hipertexto»), es un lenguaje de formato de documentos para la elaboración de páginas web [HTML]. Es un estándar que sirve de referencia para la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, entre otros. Nace como evolución de SGML (sistema para definir lenguajes para dar formato a documentos “markup lenguajes”, amparado en la ISO 8879). Es un estándar a cargo de la W3C, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web, sobre todo en lo referente a su escritura e interpretación. Es el lenguaje con el que se definen las páginas web.

El lenguaje HTML basa su filosofía de desarrollo en la referenciación. Para añadir un elemento externo a la página (imagen, vídeo, script, entre otros.), este no se incrusta directamente en el código de la página, sino que se hace una referencia a la ubicación de dicho elemento mediante texto. De este modo, la página web contiene sólo texto mientras que recae en el navegador web (interpretador del código) la tarea de unir todos los elementos y visualizar la página final. Al ser un estándar, HTML busca ser un lenguaje que permita que cualquier página web escrita en una determinada versión, pueda ser interpretada de la misma forma (estándar) por cualquier navegador web actualizado.

#### Utilización de MySQL en el módulo Búsqueda de palabras clave y módulo de búsqueda de libros

Para el desarrollo de estos módulos ha sido necesario enviar peticiones HTTP al servidor Apache, para ello, desde el lado del cliente (aplicación Android) se han usado los comandos mostrados en la Tabla 13.

Comandos usados	Función que realiza
<code>HttpClient cliente = new DefaultHttpClient();</code>	Crear la conexión http
<code>HttpGet peticionGet = new HttpGet(url);</code>	Solicita el recurso al servidor url
<code>HttpResponse responseExecute = cliente.execute(peticionGet);</code>	Procesa la respuesta enviada por el servidor.

**Tabla 13. Comandos usados desde el lado del cliente para establecer la conexión HTTP.**

La implementación de estas clases se ha realizado en la actividad *VerJson.java*, para su correcto uso se ha utilizado la clase *AsyncTask* que permite ejecutar operaciones en segundo plano e ir publicando los resultados en un hilo de la interfaz gráfica sin ralentizar a la aplicación. Esta clase se explica detalladamente en la sección 3.3.



### 3.2.2.6 JSON (JavaScript Object Notation)

Formato popular para la representación de estructuras de datos en ficheros de texto que son fáciles de leer y escribir tanto por un ser humano como por un programa, tanto si está escrito en Javascript como si está escrito en Perl, PHP, o numerosos otros lenguajes de programación [JSON].

Está constituido por dos estructuras perfectamente válidas en cualquier lenguaje de programación:

1. Una colección de pares nombre / valor. Encerrado entre los caracteres “{” y “}”, separando la clave del valor por el símbolo “:”, y separando cada par del siguiente con el carácter “,”. En varios lenguajes esto es conocido como un objeto, registro, estructura, diccionario, tabla hash, lista de claves o un arreglo asociativo.
2. Una lista ordenada de valores. Encerrado entre los caracteres “[” y “]”, y separando cada valor del siguiente con el carácter “,”. En la mayoría de los lenguajes, esto se implementa como arreglos, vectores, listas o secuencias.

En JSON, se presentan estas estructuras de estas maneras los tipos resumidos en la Tabla 14.

<b>Object</b>	Un objeto es un conjunto desordenado de pares nombre/valor. Un objeto comienza con “{” y termine con “}”. Cada nombre es seguido por “:” y los pares nombre/valor están separados por “,”.
<b>Array</b>	Un arreglo es una colección de valores. Un arreglo comienza con “[” y termina con “]”. Los valores se separan por “,”.
<b>Value</b>	Un valor puede ser una cadena de caracteres con comillas dobles, o un número, o true o false o null, o un objeto o un arreglo. Estas estructuras pueden anidarse.
<b>String</b>	Una cadena de caracteres es una colección de cero o más caracteres Unicode, encerrados entre comillas dobles, usando barras divisorias invertidas como escape. Un carácter está representado por una cadena de caracteres de un único carácter. Una cadena de caracteres es parecida a una cadena de caracteres C o Java.
<b>Number</b>	Un número es similar a un número C o Java, excepto que no se usan los formatos octales y hexadecimales.

**Tabla 14. Estructuras de JSON**

La simplicidad de JSON ha dado lugar a la generalización de su uso, especialmente como alternativa a XML en AJAX. Una de las supuestas ventajas de JSON sobre XML como formato de intercambio de datos en este contexto es que es mucho más sencillo escribir un analizador sintáctico (parser) de JSON. En JavaScript, un texto JSON se puede analizar fácilmente usando la función eval(), lo cual ha sido fundamental para que JSON haya sido aceptado por parte de la comunidad de desarrolladores AJAX, debido a la ubicuidad de JavaScript en casi cualquier navegador web.

### Utilización de JSON en el módulo de búsqueda de palabras clave y módulo de búsqueda de libros

Para obtener los datos en formato JSON es necesario enviar la consulta al servidor Apache mediante HTTP y configurar el fichero .php con la función *json\_encode(\$arr)* la cual devuelve un String con la representación de los valores \$arr en formato JSON.

En la Figura 18 se muestra el contenido obtenido al realizar dicha petición a la tabla palabras clave de la base de datos android.

```
[{"request":"ba\u00f1o","response":"Los ba\u00f1os est\u00e1n en la planta baja, entre por la puerta principal y gire a la izquierda","id":"1"}, {"request":"horario","response":"El horario de la biblioteca de legan\u00e9s es de lunes a viernes de nueve a nueve y los s\u00e1bados de nueve a tres","id":"2"}, {"request":"gracias","response":"A ti por la consulta","id":"4"}, {"request":"hola","response":"hola \u00bfqu\u00e9 consulta quieres hacer?","id":"6"}, {"request":"distribuci\u00f3n","response":"la biblioteca tiene tres plantas, en la planta baja se encuentra la recepci\u00f3n, los aseos, ordenadores y mesas de grupo. en la primera planta se encuentran los libros de y mesas para estudiar. la planta dos es la del silencio y mas","id":"7"}, {"request":"trabajo","response":"Los trabajos fin de grado se encuentran en la segunda planta","id":"8"}, {"request":"peri\u00f3dico","response":"los peri\u00f3dicos se encuentran seg\u00fan entra por la entrada principal de la biblioteca a la izquierda","id":"9"}, {"request":"grupo","response":"las salas de grupo se encuentran en la tercera planta para reservarla previamente se necesita dejar el numero del carnet en recepci\u00f3n","id":"10"}, {"request":"recepci\u00f3n","response":"la recepci\u00f3n se encuentra en la planta baja, entre por la puerta principal, pase las alarmas y se la encuentra de frente","id":"11"}, {"request":"normativa","response":"Se podr\u00e1n reservar todos los documentos prestados, incluidos aquellos con el plazo de devoluci\u00f3n vencido. Cada usuario podr\u00e1 hacer un m\u00e1ximo de tres reservas. La Biblioteca avisar\u00e1 al usuario cuando el documento est\u00e9 disponible." ,"id":"12"},
```

**Figura 18. Representación en formato JSON de la respuesta enviada al servidor**

Una vez obtenido la respuesta en formato JSON, la aplicación Android extrae la información contenida para poder trabajar con ella, para ello utiliza las clases *org.json.JSONObject* y *org.json.JSONArray*. Los métodos usados de dichas clases de muestran en la Tabla 15.

Métodos usados	Función que realiza
JSONObject	Constructor que crea una instancia de la clase JSONObject a partir de la cadena de texto pasada como parámetro. Dicha cadena se obtiene de la respuesta del servidor y comienza con “{” y termina con “}”.
JSONArray	Constructor que crea una instancia de la clase JSONArray. Dicha cadena comienza con “[” y termina con “]”.
getJSONObject	Obtiene el objeto de la clase JSONObject asociado a la clave pasada como parámetro.

**Tabla 15. Métodos usados**

En la Figura 20 se muestra como se ha implementado el uso de algunos de los métodos comentados en la tabla anterior.

```
String url = "http://192.168.1.11/json/libros.php";
JSONArray arrayJson = VerJson.obtenerJsonUrl(url);

for (int i = 0; i < arrayJson.length(); i++) {

    JSONObject json = arrayJson.getJSONObject(i);
    Log.d("tfg", "jsondevuelto:" + json);
    titulo = json.getString("titulo");
    Log.d("tfg", "titulo:" + titulo);
    String autor = json.getString("autor");
    String editorial = json.getString("editorial");
    String place = json.getString("codigo");
    String isbn = json.getString("isbn");
    String disponibilidad = json.getString("disponibilidad");

}
```

Figura 20. Implemetación paseo del JSON de libros

### 3.3 Recursos necesarios para el sistema

Como se ha comentado, una de las maneras de acceder al contenido de la base de datos, anteriormente comentada, es a través del habla. A continuación se explica cómo se han implementado los módulos de Reconocimiento Automático del habla y de Síntesis de Texto a Voz.

#### 3.3.1 Reconocimiento automático del habla

El Reconocimiento automático del habla, ya mencionado en el Estado del Arte, proporciona las herramientas necesarias para transformar la voz en conceptos que después puedan utilizar las máquinas para emprender acciones. Este recurso ha sido implementado en el módulo asistente por voz, que únicamente permite la entrada y la salida de datos mediante la voz.

Para implementar el RAH en el presente proyecto se ha hecho uso de la clase *android.speech.RecognizerIntent* del SDK de Android.

```
@Override
public void onClick(View v) {
    Intent intent = new Intent(
        RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
        "SPANISH");
```

Figura 21. Implementación del reconocedor automatico del habla

Como se puede ver en la Figura 21, al hacer la llamada al RecognizerIntent se responde con la búsqueda por voz de Google, que ya está instalada en el dispositivo Android y que muestra al usuario la figura presentada a continuación. En cuanto a los parámetros pasados al reconocimiento de voz únicamente se ha especificado el modelo de lenguaje utilizado (EXTRA\_LINGUAJE\_MODEL) y el idioma en el que se realiza el reconocimiento a través de la clase EXTRA\_LINGUAJE, siendo el lenguaje de la aplicación SPANISH.

La voz emitida por los usuarios es transmitida a los servidores de Google que se encargan de realizar el reconocimiento. Finalmente, los resultados del reconocimiento de voz son devueltos a la aplicación para que puedan ser procesados.

Se usa el servicio de voz ofrecido por Google, ya que tiene una fácil integración en las aplicaciones y ofrece muy buenas prestaciones, pero es necesario que el dispositivo tenga acceso a Internet. Google Voice Recognizer se encuentra integrado en el sistema operativo Android desde el nivel de API 8.

### 3.3.2 Síntesis de texto a voz

El sintetizador de texto a voz conocido por las siglas TTS en Inglés o también por las siglas CTV en Español, se trata de un servicio que se encuentra disponible a partir de la versión 1.6 de Android el cual permite integrar de forma sencilla la síntesis de voz a texto en cualquier aplicación mediante la utilización del paquete android.speech.tts perteneciente al SDK de Android, en particular mediante la utilización de la librería android.speech.tts.TextToSpeech. Este módulo es el encargado de hacer que la aplicación interactúe con el usuario por eso es de los más utilizados ya que establece la “conversación” usuario-aplicación.

En la Figura 22 se muestra cómo se ha inicializado e implementado dicho servicio en el presente proyecto.

```
public TextToSpeech tts = new TextToSpeech(this, this);

private void convertTextToSpeech() {

    tts.speak(text, TextToSpeech.QUEUE_FLUSH, null);

}
```

Figura 22. Implementación de la síntesis de texto a voz

### 3.3.3 AsyncTask para el desarrollo de tareas en segundo plano

Como se ha comentado anteriormente, para un mejor funcionamiento de la aplicación es necesario ejecutar en un hilo distinto al principal las operaciones largas y costosas, evitando así la lentitud, el bloqueo o el mal funcionamiento de la aplicación. Para su implementación Android proporciona la clase AsyncTask, que permite ejecutar tareas en segundo plano.

Uno de los métodos pertenecientes a dicha clase es `doInBackground()`, éste se ejecuta en un hilo secundario y por lo tanto el código implementado en el no puede interactuar con con la interfaz, por el contrario los métodos `onPreExecute()`, `onPostExecute()` si se ejecutan en el hilo principal, por lo en ellos si se puede implementar código que intervenga con la interfaz. En la Figura 23 se muestra cómo se ha implementado este recurso en el TFG:

```
public static JSONArray obtenerJsonUrl(final String url) throws JSONException {  
  
    AsyncTask<Object, Integer, String> parkingsDownloader = new AsyncTask<Object, Integer, String>() {  
        @Override  
        protected String doInBackground(Object... params) {
```

**Figura 23. Uso del método `doInBackground`**



# Capítulo 4


## Descripción detallada de los módulos del sistema

El presente capítulo analiza de forma detallada cada uno de los módulos que componen la aplicación. En la siguiente figura se explica el funcionamiento general de la aplicación que posteriormente se desglosará en los módulos.

### 4.1 Módulo Inicio

#### 4.1.1 Funcionalidad

El *módulo de Inicio* hace referencia a la primera interfaz de la aplicación, esta pantalla permite el acceso a la mayoría de las funcionalidades de la aplicación. Este módulo implementa el primer diálogo que se proporciona al usuario y las opciones que puede elegir.

Para ejecutar la aplicación es necesario pulsar su icono . A continuación, se muestra al usuario la actividad principal con un `AlertDialog` que informa sobre las opciones que ofrece la aplicación tal y como se muestra en la Figura 24.



**Figura 24. Captura de pantalla del diálogo.**

Una vez presionado el botón aceptar, se muestra la primera actividad de la aplicación. La Figura 25 representa la pantalla del *módulo de Inicio* de la aplicación, en la que se muestra la distribución de las funcionalidades en la interfaz. Como se puede ver, se le permite al usuario hacer su consulta de tres maneras distintas, es por ello por lo que se trata de una aplicación multimodal.



**Figura 25. Captura de pantalla del módulo de Inicio.**



Las opciones que ofrece son:

- Entrada por voz. Activar el reconocedor automático del habla se activa presionando el botón de la lupa. Está descrito detalladamente en el *módulo Asistente por voz*.



Figura 26. Asistente por Voz.

- Entrada por teclado. Escribir la consulta en el de campo de texto, en vez de decírselo a la aplicación mediante la voz y pulsar el botón de buscar:

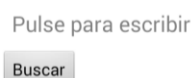







Figura 27. Entrada por teclado.

- Entrada por pantalla. Acceder a respuestas concretas, las más comunes, mediante el menú. Consiste en un GridView, menú deslizable, en el que se encuentran las opciones mostradas en la Tabla 16 :



Figura 28. Entrada por pantalla.

Imagen del menú	Función que realiza
	Texto informativo para el usuario sobre cómo se usa la aplicación.
	Muestra los libros existentes actualmente en la biblioteca UC3M de Leganés, ordenados por materias. Explicado detalladamente en el <i>módulo Búsqueda de libros</i> .
	Permite al usuario reservar un libro. Esta funcionalidad se detalla en el <i>módulo Consultar Pagina Web</i> .
	Informa al usuario de los correos de las distintas bibliotecas UC3M, por si el libro deseado no se encuentra en Leganés pero si en Colmenarejo o Getafe. Para ello accede al enlace la web de la universidad donde está publicada dicha información.
	Comunica el horario de la biblioteca de Leganés y muestra al usuario un calendario informando el horario de cada día del mes.






	Explica cómo está distribuida la biblioteca y descarga en el móvil el plano de la biblioteca.
	Informa, mediante la voz y un plano indicativo, donde se encuentran los Trabajos Fin de Grado y los Trabajos Fin de Carrera.
	Informa, mediante la voz y un plano indicativo, donde se encuentran las salas de grupo.
	Informa, mediante la voz y un plano indicativo, donde se encuentran los periódicos
	Informa, mediante la voz y un plano indicativo, donde se encuentra la recepción.

Tabla 16. Opciones del menú deslizable

### 4.1.2 Flujo de datos

Los pasos que se llevan a cabo para el flujo de datos del *módulo de Inicio* son:

1. La aplicación muestra la pantalla principal (Figura 25) en la que se presentan las tres maneras que existen para acceder al contenido de la base de datos: mediante voz, texto o el menú.
2. El usuario decide que opción quiere ejecutar para acceder a la información deseada. La entrada al sistema se puede dar mediante tres opciones:
  - a. Entrada por voz (ASR).
  - b. Entrada por teclado
  - c. Entrada por pantalla.
3. La aplicación interpreta la información requerida y la procesa con la base de datos para proporcionar al usuario una respuesta mediante dos salidas:
  - a. Salida por voz (TTS).
  - b. Salida por pantalla.

## 4.2 Módulo Asistente por voz

### 4.2.1 Funcionalidad

Este módulo utiliza únicamente el reconocedor automático del habla (ASR) de Google como entrada al sistema y la síntesis de texto a voz para la salida del mismo. Para

activarlo es necesario pulsar el *ImageButton* de la lupa

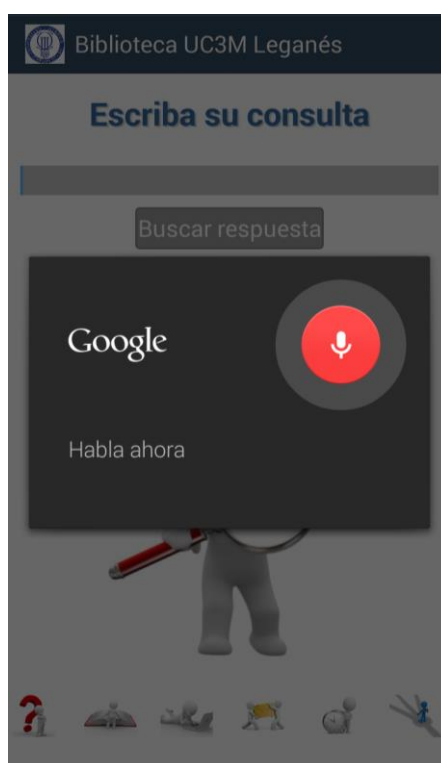


Figura 29. Reconocedor Automático del Habla

La funcionalidad de este módulo es reconocer correctamente las palabras del usuario para posteriormente compararlas con la base de datos y así, una vez consultada, contestar al usuario con la información requerida.

### 4.2.2 Flujo de datos

Los pasos implementados en el *módulo de Asistente por Voz* son:

1. Al activarse el RAH el usuario, mediante el sistema de diálogo, expone la consulta que desea hacer.

- Se analizan las palabras que ha introducido el usuario. Cuando éste habla internamente se toman una serie de posibles palabras que cree que puede haber dicho; por ejemplo, si se le dice “buenos días”, el RAH almacena las siguientes opciones. Para el desarrollo de la aplicación siempre se ha tomado la primera posición del array, pues suele ser la palabra que más se acerca a lo que ha dicho el usuario.

`text:[buenos días, buenos dias, Buenos Días]`

Se muestra en la Figura 30 cómo se ha implementado el código para esta funcionalidad.

```

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    switch (requestCode) {
        case RESULT_SPEECH: {
            if (resultCode == RESULT_OK && null != data) {

                ArrayList<String> text = data
                    .getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);

                txtText.setText(text.get(0));
                Log.d("marta", "text:" + text);
                posible = text.get(0).toLowerCase();
                Log.d("marta", "posible: " + posible);
            }
        }
    }
}
    
```

**Figura 30. Implementación módulo asistente por Voz**

- Una vez procesada la información y según las palabras dichas, se conduce al usuario al módulo correspondiente.

Palabras dichas por el usuario	Módulo
<b>Libro</b>	Búsqueda de Libros
<b>Cualquier otra palabra</b>	Búsqueda de Palabras Clave

**Tabla 17. Módulos utilizados**

Para el desarrollo de este módulo se ha usado la metodología descrita en el apartado 3.3.1 y 3.3.2 para implementar el conocimiento de voz y la síntesis de texto a voz.

En la Figura 31 se muestra el diálogo establecido entre el usuario y la aplicación.

**U:** Reservar libro

**S:** ¿Quiere reservar un libro o quiere conocer la normativa para llevar a cabo la reserva? Si quiere reservar un libro diga enlace, si lo que quiere es conocer la normativa, diga normativa.

**U:** Se podrán reservar todos los documentos prestados, incluidos aquellos con el plazo

de devolución vencido. Cada usuario podrá hacer un máximo de tres reservas. La Biblioteca avisará al usuario cuando el documento esté disponible.

**Figura 31. Ejemplo diálogo aplicación y usuario.**

## 4.3 Módulo Búsqueda de palabras clave

### 4.3.1 Funcionalidad

Al tratarse de una aplicación multimodal, se puede acceder a este módulo mediante diversos canales, ya sea usando el teclado, la pantalla o la voz.

La finalidad de este módulo, consiste en reconocer a qué información desea acceder el usuario y proporcionarle la respuesta correcta. Para detectar la palabra más relevante de la consulta realizada, se han llevado a cabo sucesivas pruebas con distintos usuarios para analizar las distintas preguntas que éstos hacen y cómo las hacen. En este análisis se han obtenido las palabras más importantes y frecuentes.

Y a partir de ahí, se ha ordenado la base de datos, comenzando por las palabras más frecuentes y finalizando con las menos. Así, por ejemplo, si el usuario dice: “¿Dónde está el baño de la Biblioteca?” primeramente se ha colocado en la base de datos la palabra “baño” y después “biblioteca”, y no al revés, para para que de la respuesta asociada a “baño” y no a “biblioteca”.

### 4.3.2 Flujo de datos

Una vez obtenida la palabra o frase que ha dicho el usuario, se recorre en busca de la una “palabra clave”. Estas palabras están almacenadas en la base de datos llamada “palabrasclave” mostrada en la Figura 32. Dicha tabla se compone de tres columnas como se muestra en la Tabla 18

Columnas	Función
<b>Id</b>	Cada id está asociado a una palabra clave.
<b>Request.</b>	Esta columna contiene todas las palabras claves relacionadas con la biblioteca que podría decir el usuario.
<b>Response</b>	La última columna hace referencia al texto como respuesta asociado a cada palabra clave

**Tabla 18. Explicación columnas tabla “palabras clave”**

	request	response	id
<input type="checkbox"/> Editar Copiar Borrar	baño	Los baños están en la planta baja, entre por la pu...	1
<input type="checkbox"/> Editar Copiar Borrar	horario	Puede ver el horario detallado por días en el sigu...	2
<input type="checkbox"/> Editar Copiar Borrar	gracias	A ti por la consulta	4
<input type="checkbox"/> Editar Copiar Borrar	hola	hola ¿qué consulta quieres hacer?	6
<input type="checkbox"/> Editar Copiar Borrar	distribución	la biblioteca tiene tres plantas, en la planta baj...	7
<input type="checkbox"/> Editar Copiar Borrar	trabajo	Los trabajos fin de grado se encuentran en la segu...	8
<input type="checkbox"/> Editar Copiar Borrar	periódico	los periódicos se encuentran según entra por la en...	9
<input type="checkbox"/> Editar Copiar Borrar	grupo	las salas de grupo se encuentran en la tercera pla...	10
<input type="checkbox"/> Editar Copiar Borrar	recepción	la recepción se encuentra en la planta baja, entre...	11
<input type="checkbox"/> Editar Copiar Borrar	normativa	Se podrán reservar todos los documentos prestados,...	12
<input type="checkbox"/> Editar Copiar Borrar	reservar	¿Quiere reservar un libro o quiere conocer la norm...	13
<input type="checkbox"/> Editar Copiar Borrar	ordenador	Los ordenadores se encuentran en la tercera planta...	14
<input type="checkbox"/> Editar Copiar Borrar	aseo	Los aseos están en la planta baja, entre por la pu...	15
<input type="checkbox"/> Editar Copiar Borrar	renovar	Todos los documentos prestables se podrán renovar ...	17
<input type="checkbox"/> Editar Copiar Borrar	video	a continuación se muestra un vídeo explicativo de ...	18
<input type="checkbox"/> Editar Copiar Borrar	sanción	En caso de no devolución, la Biblioteca requerirá ...	19
<input type="checkbox"/> Editar Copiar Borrar	enlace	Para reservar un libro introduzca su NIA y su cont...	20
<input type="checkbox"/> Editar Copiar Borrar	planos	a continuación se descargará en su dispositivo los...	21
<input type="checkbox"/> Editar Copiar Borrar	circuitos	El libro de circuitos eléctricos se encuentra en l...	22
<input type="checkbox"/> Editar Copiar Borrar	libro	Diga el nombre del libro que desea buscar. Si qui...	23
<input type="checkbox"/> Editar Copiar Borrar	biblioteca	Se encuentra en la biblioteca de Leganés, campus d...	24

Figura 32. Tabla palabras clave

### 4.3.2.1 Interacción multimodal

La Figura 33 muestra el flujo de datos del *módulo de Búsqueda de palabras clave* en interacción multimodal. El proceso llevado a cabo en interacción multimodal se resume en los siguientes pasos:

1. Al activarse el ASR, el usuario expone la información que desea conocer mediante el sistema de diálogo o escribiendo la consulta.
2. Una vez analizada la petición en el *módulo Asistente por voz*, se procede a la búsqueda de la palabra clave en la base de datos. Para ello se envía una petición Get mediante HTTP al servidor PHP.

```
HttpClient cliente = new DefaultHttpClient();
HttpGet peticionGet = new HttpGet(url);
```

3. El servidor PHP ejecuta un script para acceder a la base de datos MySQL.
4. El servidor PHP envía a la aplicación Android la información contenida en la tabla “palabras clave” que se encuentra en la base de datos. Dicha información se convierte en formato JSON mediante el script PHP.
5. La aplicación compara las palabras que ha dicho el usuario con las palabras de la columna request de la tabla mostrada en la Figura 32. Cuando éstas dos coinciden, se procede a leer la columna response.



Figura 33. Flujo de datos del módulo de Búsqueda de palabras clave interacción multimodal

### 4.3.2.2 Interacción por pantalla

La figura anterior muestra el flujo de datos del *módulo de Búsqueda de palabras clave* en interacción multimodal. El proceso llevado a cabo en interacción por pantalla se resume en los siguientes pasos:

1. El usuario selecciona el botón del menú que contiene la consulta que desea hacer.



Figura 34. Botón del menú seleccionado

2. Cada botón tiene asociado una palabra clave que representa una palabra de la columna request de la base de datos. Las distintas opciones del menú de la aplicación se muestran en la Tabla 9 del *módulo de Inicio*.
3. Se procede a la búsqueda de esa palabra en la base de datos. Para ello se envía una petición Get mediante HTTP al servidor PHP.

```
HttpClient cliente = new DefaultHttpClient();
HttpGet peticionGet = new HttpGet(url);
```

4. El servidor PHP ejecuta un script para acceder a la base de datos MySQL.
5. El servidor PHP envía a la aplicación Android la información contenida en la tabla “palabras clave” que se encuentra en la base de datos. Dicha información se convierte en formato JSON mediante el script PHP.

6. La aplicación compara las palabras que ha dicho el usuario con las palabras de la columna request de la tabla mostrada en la Figura 35. Cuando éstas dos coinciden, se procede a leer la columna response.

## 4.4 Módulo Búsqueda de libros

### 4.4.1 Funcionalidad

La finalidad de este módulo consiste en mostrar al usuario los libros de la biblioteca, organizados según su materia. Para ello, la aplicación preguntará al usuario qué libro desea buscar y ésta informará al usuario mediante la voz y mediante un plano si el libro se encuentra disponible y la ubicación en la biblioteca.

Al tratarse de una aplicación multimodal, se puede acceder a este módulo mediante varias vías, ya sea usando el teclado, la pantalla o la voz.

### 4.4.2 Flujo de datos

#### 4.4.2.1 Interacción multimodal

El proceso llevado a cabo se resume en los siguientes pasos:

1. La aplicación Android pide al usuario mediante la síntesis de texto a voz que indique el título del libro sobre el que desea conocer su ubicación. Éste puede hacerlo mediante el sistema de diálogo o escribiendo la consulta.
2. Una vez analizada la petición en el *módulo Asistente por voz*, se procede a la búsqueda del libro en la base de datos. Para ello se envía una petición Get mediante HTTP al servidor PHP.

```
HttpClient cliente = new DefaultHttpClient();  
HttpGet peticionGet = new HttpGet(url);
```

3. El servidor PHP ejecuta un script para acceder a la base de datos MySQL.
4. El servidor PHP envía a la aplicación Android la información contenida en la tabla “books” que se encuentra en la base de datos. Dicha información se convierte en formato JSON mediante el script PHP.
5. La aplicación compara el título del libro que ha dicho el usuario con los títulos que se encuentran en la base de datos, tabla mostrada en la Figura 35. Cuando éstas dos coinciden, se procede a leer la columna de disponibilidad.



materna	codigo	titulo	autor	editorial	isbn	disponibilidad
Análisis y diseño de circuitos	L/S 621.3.01 NIL	Circuitos eléctricos	Nilsson, James William	Pearson Prentice Hall	8420544582	Se encuentran disponibles 2 ejemplares en la plant...
Análisis y diseño de circuitos	L/D 621.3.01 ADB	Applied circuit theory, matrix and computer method...	Adby, P.R.	Ellis Horwood	0853121001	1 copia disponible en UC3M.Biblioteca Escuela Poli...
Análisis y diseño de circuitos	L/S 621.3.049 FAP	Circuits and systems : a modern approach	Papoulis, Athanasios	Holt, Rinehart and Winston	0030550977	1 copia disponible en UC3M.Biblioteca Escuela Poli...
Análisis y diseño de circuitos	L/D 621.3.01 SIE	Circuits, signals and systems	Siebert, William McC.	MIT Press	0262192292	8 copias disponibles en UC3M.Biblioteca Escuela Po...
Análisis y diseño de circuitos	L/S 621.38.049 AGA	Foundations of analog and digital electronic circu...	Agarwal, Anant	Morgan Kaufmann	9781558607354	2 copias disponibles en UC3M.Biblioteca Escuela Po...
Análisis y diseño de circuitos	L/D 621.3.01 TEM	Introduction to circuit synthesis and design	Temes, Gabor C. (1923- )	McGraw-Hill	0070634890	10 copias disponibles en UC3M.Biblioteca Escuela P...
Análisis y diseño de circuitos	L/S 621.3.01 TSE	Linear circuit analysis	Tse, Chi Kong	Addison-Wesley,	0201342960	4 copias disponibles en UC3M.Biblioteca Escuela Po...
Análisis y diseño de circuitos	L/D 621.3.01 DEC	Linear circuit analysis : time domain, phasor, and...	DeCarlo, Raymond A. (1950- )	Oxford University Press	0195136667	2 copias disponibles en UC3M.Biblioteca Escuela Po...
Análisis y diseño de circuitos	L/S 621.391 TAY	Principles of signals and systems	Taylor, Fred J. (1940- )	McGraw-Hill	0079111718	3 copias disponibles en UC3M.Biblioteca Escuela Po...
Programación	L/S 004.438 JAVA JIM	Aprende a programar con Java : un enfoque práctico...	Jiménez Marín, Alfonso	Paraninfo	9788497321303	2 copias disponibles en UC3M.Biblioteca Campus de ...
Programación	C/S 004.438 JAVA WIN	Developing Java software 3rd ed	Winder, Russel	John Wiley & Sons	9780470090251	5 copias disponibles en UC3M.Biblioteca Escuela Po...
Programación	C/S 004.438 JAVA WU	An introduction to object-oriented programming wit...	Wu, C. Thomas	McGraw-Hill Higher Education	9780071283687	3 copias disponibles en UC3M.Biblioteca Escuela Po...
Programación	C/S 004.438 JAVA ECK	Piensa en Java 2ª ed	Eckel, Bruce	Prentice Hall	8420531928	19 copias disponibles en UC3M.Biblioteca Escuela P...
Programación	L/S 004.438 JAVA PRO	Problemas resueltos de programación en lenguaje Ja...	Pérez Menor, José M <sup>2</sup>	Thomson	8497321766	12 copias disponibles en UC3M.Biblioteca Escuela P...
Algebra	D 512 LOR	Algebra	Lorenz, Falko	Springer	9780387289304	Actualmente todos los ejemplares estan prestados
Comunicaciones Móviles	L/S 621.396.4 COM	Comunicaciones móviles GSM	Hernando Rábanos, José María	Fundación Airtel	8493029823	3 copias disponibles en UC3M.Biblioteca Escuela Po...
Comunicaciones Móviles	L/D 621.396.4 GSM	GSM, GPRS and EDGE performance : evolution towards...	Halonen, Timo	John Wiley & Sons	470844574	2 copias disponibles en UC3M.Biblioteca Campus de ...

Figura 35. Presentación de la Tabla de libros

En la Tabla 19 se muestra cómo se desarrolla el código entre el usuario y la aplicación en modo oral cuando se quiere solicitar información sobre un libro de la biblioteca. En la Figura 36 se expone el plano del libro solicitado y las indicaciones de cómo llegar hasta él.

**S:** Diga el nombre del libro que desea buscar. Si quiere conocer la lista de libros disponibles actualmente en la biblioteca, puede consultar la base de datos pulsando el segundo botón del menú. Los encontrará ordenados por materias.

**U:** Programación en Java

**S:** Se encuentran disponibles 2 ejemplares en la planta dos (se muestra por pantalla un mapa)

Tabla 19. Diálogo establecido entre el usuario y la aplicación en el módulo Búsqueda de libros

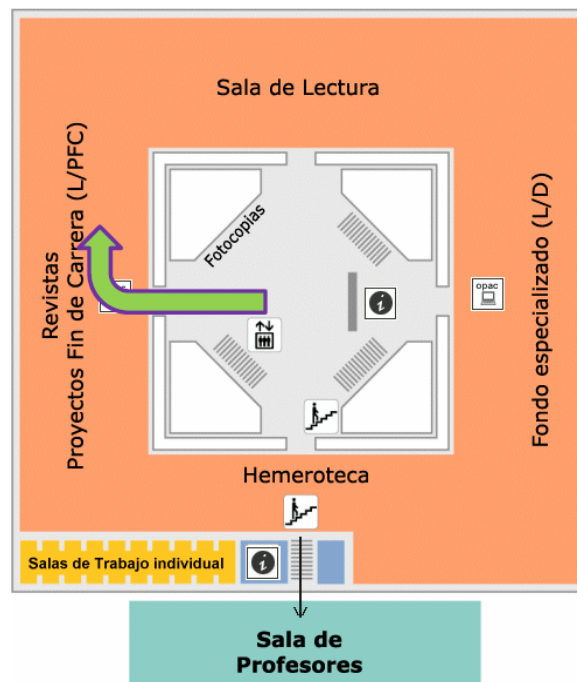



Figura 36. Plano libro Programación en Java

#### 4.4.2.2 Interacción por pantalla

El flujo de datos del *módulo de Búsqueda de libros* en interacción por pantalla se resume en los siguientes pasos:

1. El usuario pulsa el botón del menú 
2. Se muestra por pantalla al usuario la lista de materias, como se puede ver en la Figura 37.




**Figura 37. Lista de materias**

3. Una vez pulsada la materia deseada, se procede a la búsqueda de los libros que se encuentran en la biblioteca pertenecientes a dicha materia. Para ello se envía una petición Get mediante HTTP al servidor PHP.

```
HttpClient cliente = new DefaultHttpClient();  
HttpGet petitionGet = new HttpGet(url);
```

4. El servidor PHP ejecuta el script para acceder a la base de datos MySQL.
5. El servidor PHP envía a la aplicación Android la información contenida en la tabla “books” que se encuentra en la base de datos. Dicha información se convierte en formato JSON mediante el script PHP.
6. La aplicación muestra por pantalla los el título y el código de los libros pertenecientes a dicha materia (Figura 38).



Biblioteca UC3M Leganés

**Circuitos eléctricos**  
L/S 621.3.01 NIL

**Applied circuit theory, matrix and computer methods**  
L/D 621.3.01 ADB

**Circuits and systems : a modern approach**  
L/S 621.3.049 PAP

**Circuits, signals and systems**  
L/D 621.3.01 SIE

**Foundations of analog and digital electronic circuits**  
L/S 621.38.049 AGA

**Introduction to circuit synthesis and design**  
L/D 621.3.01 TEM

**Linear circuit analysis**  
L/S 621.3.01 TSE

**Linear circuit analysis : time domain, phasor,**

**Figura 38. Libros pertenecientes a la materia Análisis y diseño de circuitos.**

7. Si se pulsa el libro deseado se desglosa la información sobre dicho libro; Código, Título, Autor, Editorial, ISBN, Disponibilidad (Figura 39).



Biblioteca UC3M Leganés

**Circuitos eléctricos**

L/S 621.3.01 NIL  
Nilsson, James William  
Pearson Prentice Hall  
8420544582  
Se encuentran disponibles 2 ejemplares en la planta uno y 1 ejemplar

**Figura 39. Información detallada del libro seleccionado**

## 4.5 Módulo Consulta página web

### 4.5.1 Funcionalidad

Como se ha comentado en el *módulo de Inicio*, existen varias consultas en las que se necesita acceder directamente a la página web de la biblioteca de la universidad tales como reservar un libro, mandar un correo a la biblioteca o acceder al listado de correos de las distintas bibliotecas UC3M.

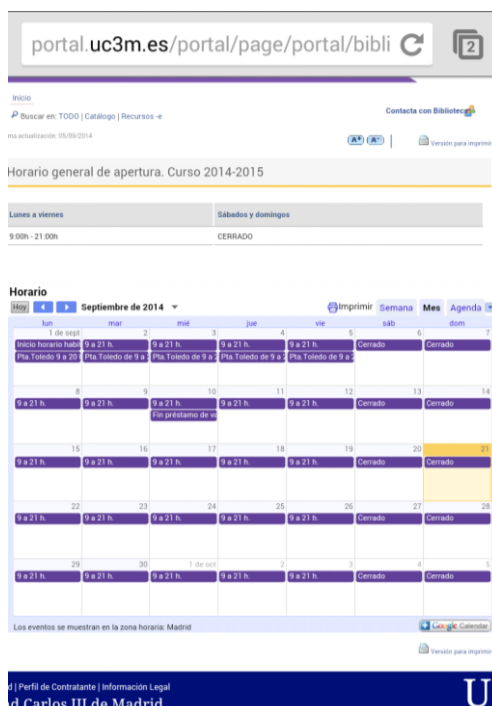


Figura 40. Acceso página web UC3M

### 4.5.2 Flujo de datos

Para implementar dicha funcionalidad se usa intents. Un intent es el elemento básico de comunicación entre los distintos componentes Android de una aplicación o entre distintas aplicaciones. Mediante un intent se puede mostrar una actividad desde cualquier otra, iniciar un servicio, enviar un mensaje broadcast, iniciar otra aplicación, etc.

La Figura 41 muestra el flujo de datos del *módulo de Consulta página web* en interacción multimodal y, a continuación, se detallan los pasos implementados para su desarrollo.

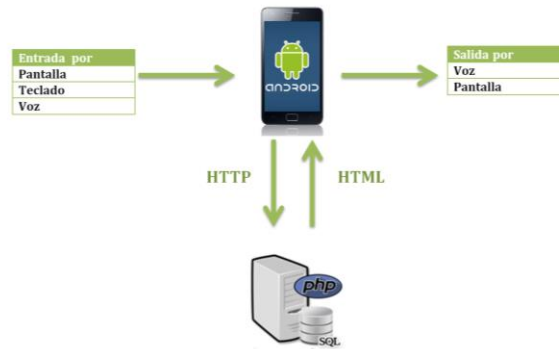


Figura 41. Flujo de datos módulo Consulta pagina web

1. El usuario comunica a la aplicación, a través del teclado, del reconocimiento de voz o del menú, que desea mandar un correo a la biblioteca de la universidad o reservar un libro. Cada opción lleva asociada una palabra clave y se gestiona como se ha comentado en el *módulo Búsqueda de Palabras Clave*.
2. Una vez contrastada la palabra con la de la base de datos se procede a mostrar al usuario la página web solicitada, mientras que se le explican mediante la voz, que pasos ha de seguir para llevar a cabo dicho procedimiento.
3. Para mostrar dicha página web la aplicación Android (cliente) envía una petición mediante HTTP al servidor del sitio web de la Biblioteca UC3M y se conecta a dicha página web.
4. El servidor del sitio web de la Biblioteca UC3M responde al cliente con el contenido HTML de la página web de la película solicitada.

## 4.6 Módulo Indicaciones-plano

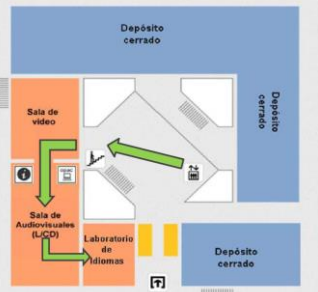
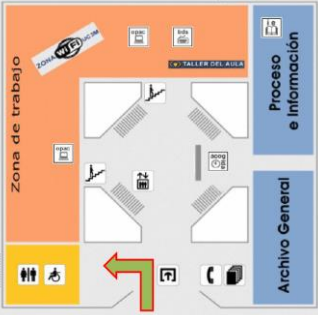
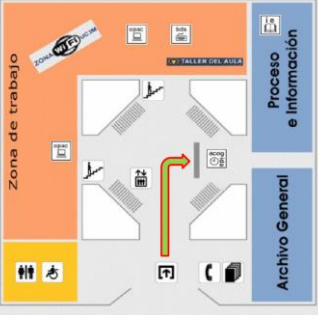
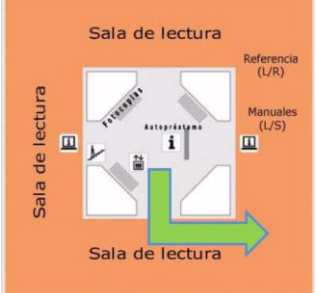

### 4.6.1 Funcionalidad

En este módulo se describe la funcionalidad del *módulo Indicaciones-plano*, en él se explica cómo se han implementado las indicaciones proporcionadas al usuario mediante la pantalla y que opciones existen.

### 4.6.2 Arquitectura

Para el desarrollo e implementación de este módulo también se ha recurrido al uso de los intets, como en el módulo anterior. Una vez reconocidas las palabras que ha dicho el usuario (*módulo de Asistente por Voz*), comparadas con la base de datos y obtenida la respuesta que se le va a proporcionar (*módulo Búsqueda de Palabras Clave* o *módulo de Búsqueda de libros*), se hace uso del presente módulo, el cual se encarga de mostrar al

usuario mediante un mapa las indicaciones para llegar al punto o libro solicitado. La Tabla 20 muestra las opciones que ofrece el menú deslizable:

Información solicitada	Planta	Plano indicativo
<b>Sala de audiovisuales y laboratorio de idiomas</b>	Sótano L/CD	
<b>Aseos o periódicos</b>	Planta baja	
<b>Recepción</b>	Planta baja	
<b>Libro: Programación en Java</b>	Primera planta Código:L/S	
<b>Libro: Ecuaciones en derivadas parciales con series de Fourier y problemas de contorno</b>	Primera planta Código:L/S	

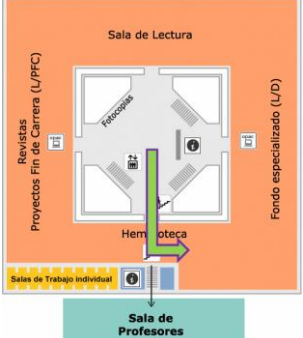
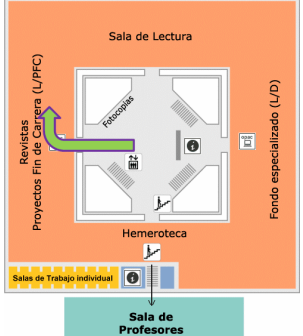
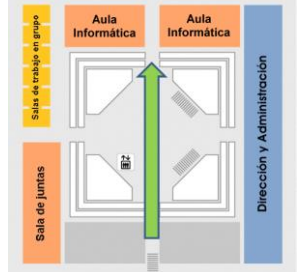
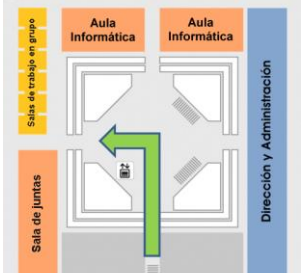
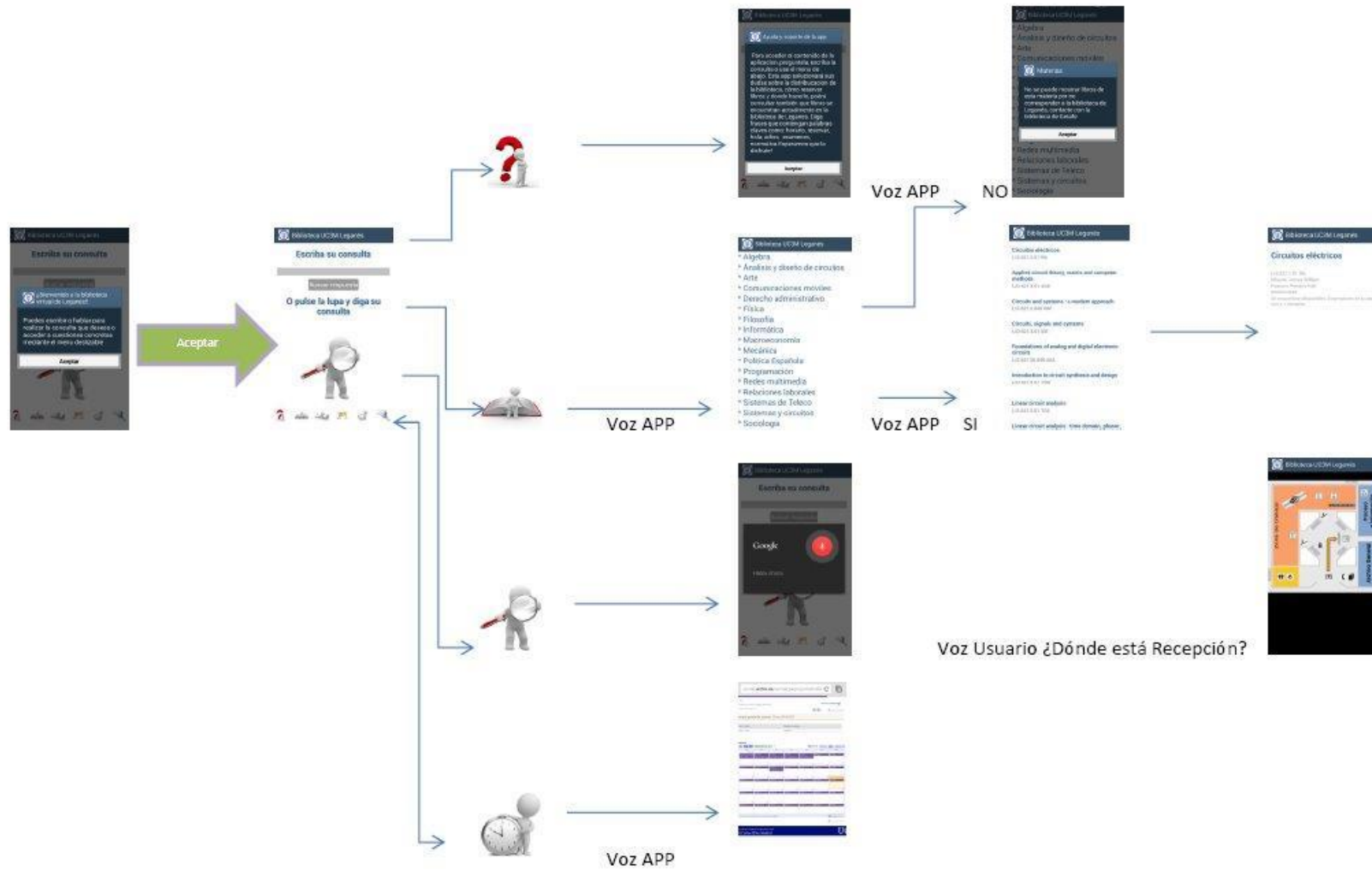
<p><b>Libro:</b> Circuitos eléctricos</p>	<p>Segunda planta Código L/D</p>	 <p>Este diagrama muestra un camino verde que comienza en la Sala de Profesores (situada en un recuadro azul debajo del mapa), se dirige hacia la Hemeroteca y luego hacia la Sala de Lectura. El mapa incluye áreas como 'Revistas', 'Proyectos Fin de Carrera (L/PFC)', 'Fotocopiadoras', 'Fondo especializado (L/D)' y 'Salas de Trabajo individual'.</p>
<p><b>Libro:</b> Mecánica: problemas de exámenes resueltos</p>	<p>Segunda planta Código L/D</p>	 <p>Este diagrama muestra un camino verde que comienza en la Sala de Profesores (situada en un recuadro azul debajo del mapa), se dirige hacia la Hemeroteca y luego hacia la Sala de Lectura. El mapa incluye áreas como 'Revistas', 'Proyectos Fin de Carrera (L/PFC)', 'Fotocopiadoras', 'Fondo especializado (L/D)' y 'Salas de Trabajo individual'.</p>
<p>Sala de ordenadores</p>	<p>Tercera planta</p>	 <p>Este diagrama muestra un camino verde que comienza en la Sala de Profesores (situada en un recuadro azul debajo del mapa), se dirige hacia las Salas de Trabajo en grupo y luego hacia las Salas de juntas. El mapa incluye áreas como 'Aula Informática', 'Dirección y Administración' y 'Sala de juntas'.</p>
<p>Salas de grupo</p>	<p>Tercera planta</p>	 <p>Este diagrama muestra un camino verde que comienza en la Sala de Profesores (situada en un recuadro azul debajo del mapa), se dirige hacia las Salas de Trabajo en grupo y luego hacia las Salas de juntas. El mapa incluye áreas como 'Aula Informática', 'Dirección y Administración' y 'Sala de juntas'.</p>

Tabla 20. Información solicitada

Las Figuras 42 y 43 muestran, de manera gráfica, las principales funcionalidades de la aplicación comentadas en los módulos anteriores.





RESTO OPCIONES (Comentadas a lo largo de la memoria)

Figura 42. Diagrama general de la Aplicación. Primeras opciones

### FINALIZAR SESIÓN



Figura 43. Diagrama general de la Aplicación. Salir



# Capítulo 5

## Evaluación de la aplicación

En este capítulo se recoge el método usado para la evaluación de la aplicación desarrollada para el Trabajo Fin de Grado y los resultados obtenidos a través de un cuestionario que recoge la opinión de los encuestados que han utilizado la aplicación. Los resultados se muestran en forma gráfica en este capítulo para una rápida y fácil comprensión.

### 5.1 Metodología de evaluación

La metodología usada para el desarrollo del presente TFG se ha llevado a cabo mediante un cuestionario en el que se recoge información sobre el grado de conocimiento del encuestado acerca de nuevas tecnologías, usos de sistemas de diálogo y de la aplicación

Para ello se han formulado 10 preguntas, a 10 personas de entre 23 y 60 años, en las que algunas de las preguntas son generales sobre el uso de los smartphones y el uso de los sistemas de diálogo, y por otro lado preguntas más específicas de la aplicación desarrollada.

Previamente al cuestionario los encuestados han probado la aplicación individualmente, teniendo plena libertad para descubrir las funcionalidades que ofrece la misma.

**Pregunta 1-¿Qué experiencia tiene usando Smartphone o tabletas?**

- Mucha
- Bastante
- Alguna
- Poca
- Ninguna

**Pregunta 2-¿Qué experiencia tiene usando interfaces orales?**

- Mucha
- Bastante
- Alguna
- Poca
- Ninguna

**Pregunta 3-Al interactuar con la aplicación, ¿ésta ha entendido bien sus preguntas?**

- Muy mal
- Mal
- Regular
- Bien
- Muy bien

**Pregunta 4-En su opinión la navegación por la aplicación fue....**

- Muy mala
- Mala
- Regular
- Buena
- Muy buena

**Pregunta 5-¿Ha contestado la aplicación correctamente a su consulta? Puntúe en una escala de 1 a 5 (1= “Bajo”, 5=“Alto”)**

- 1
- 2
- 3
- 4
- 5

**Pregunta 6-¿Cree que es útil esta aplicación? Puntúe en una escala de 1 a 5 (1= “Bajo”, 5=“Alto”)**

- 1
- 2
- 3
- 4
- 5

**Pregunta 7-¿Le ha sido fácil realizar las acciones que quería?**

- Muy difícil
- Difícil
- Normal
- Fácil
- Muy fácil

**Pregunta 8-Establezca el nivel de dificultad de la aplicación para usted.**

- Muy difícil
- Difícil
- Normal
- Fácil
- Muy fácil

**Pregunta 9-Puntúe en una escala de 1 a 5 cómo de intuitiva ha sido la aplicación para usted (1= “Bajo”, 5=“Alto”)**

- 1
- 2
- 3
- 4
- 5

**Pregunta 10-En términos generales está usted satisfecho con la aplicación**

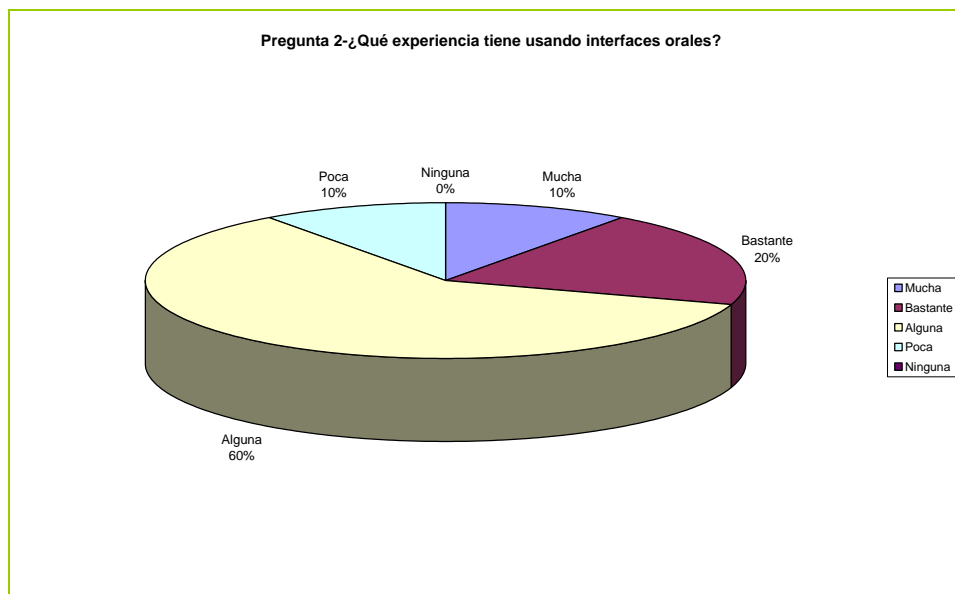
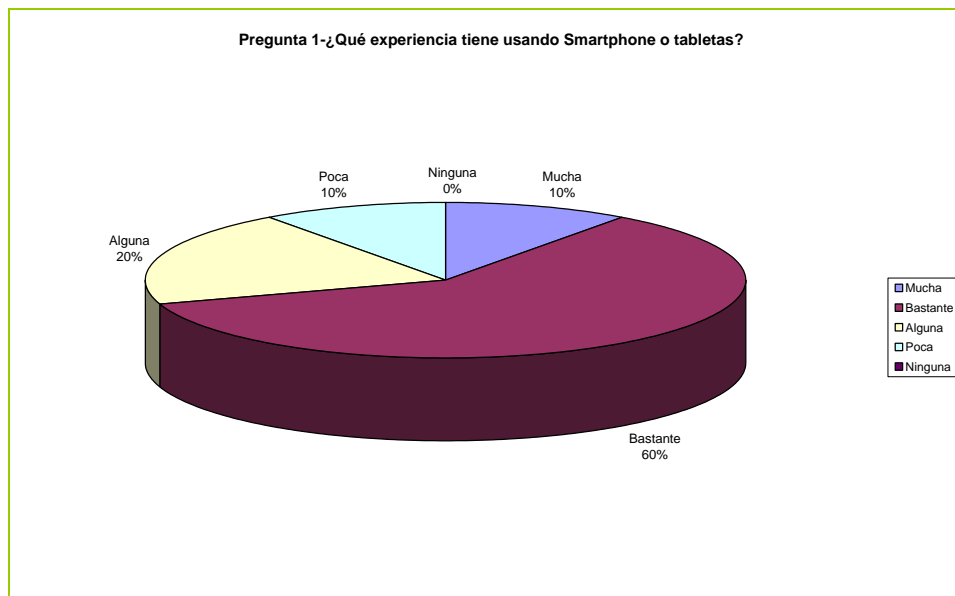
- Nada satisfecho
- Poco satisfecho
- Satisfecho
- Bastante satisfecho
- Muy satisfecho

**Figura 44. Encuesta**

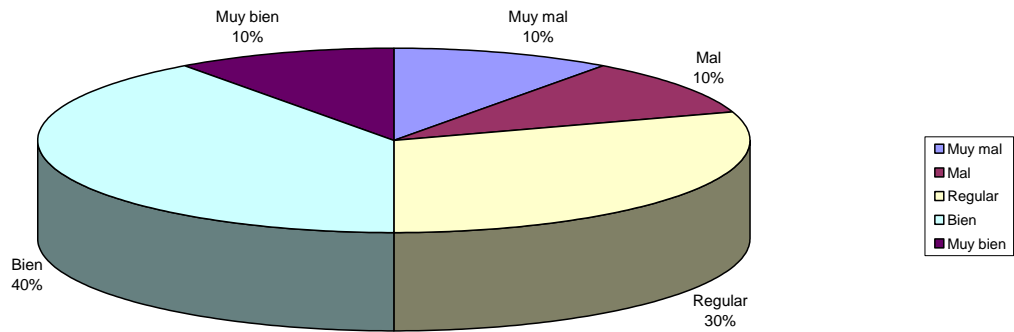
Una vez realizado este cuestionario y procesadas las respuestas de los 10 encuestados se procede a mostrar los resultados obtenidos a cada pregunta, mediante gráficas, en el siguiente apartado de la memoria.

## 5.2 Resultados de la Evaluación

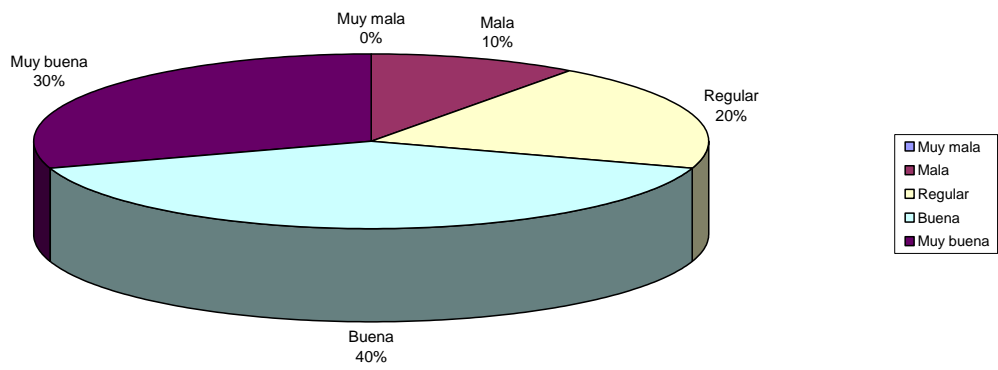
En las siguientes figuras se muestran las estadísticas de los resultados para cada una de las preguntas que se plantean en la encuesta de valoración de la aplicación.



Pregunta 3-Al interactuar con la aplicación, ¿ésta ha entendido bien sus preguntas?

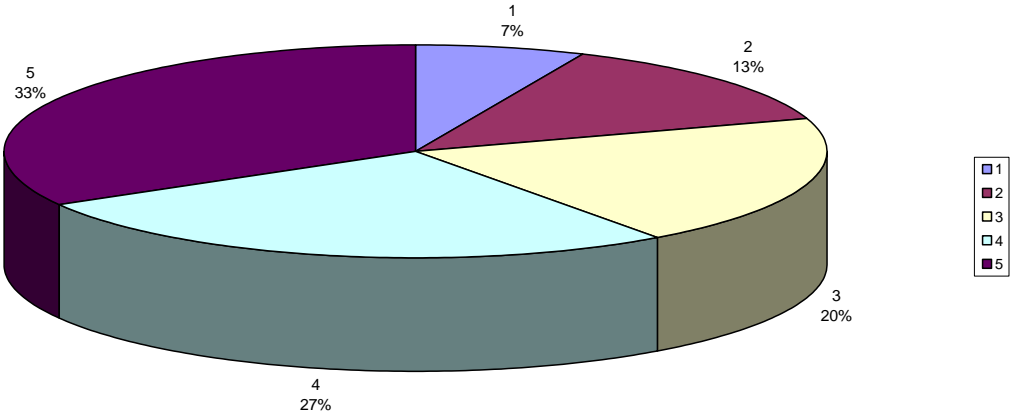


Pregunta 4-En su opinión la navegación por la aplicación fue....

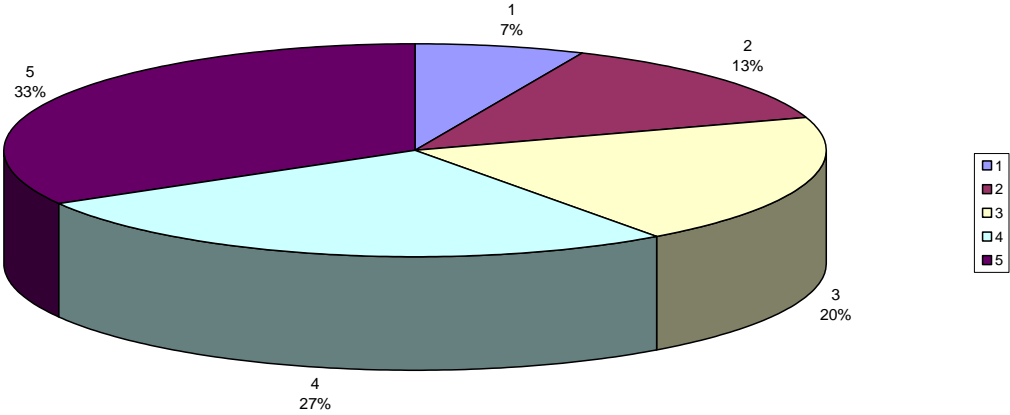




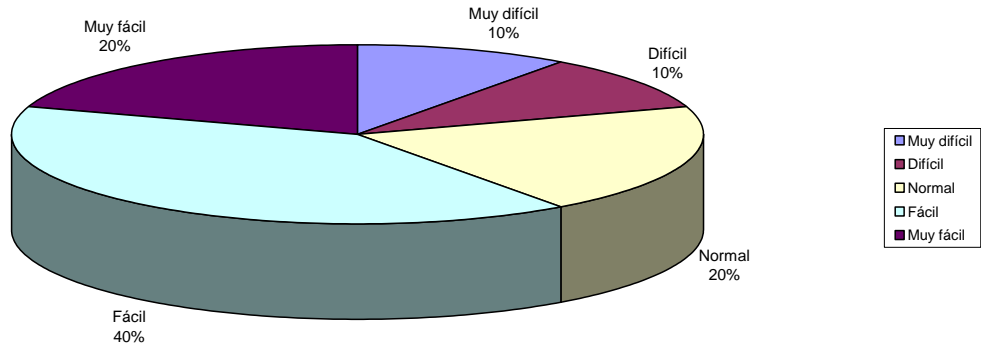
Pregunta 5-¿Ha contestado la aplicación correctamente a su consulta? Puntúe en una escala de 1 a 5 (1= "Bajo", 5="Alto")



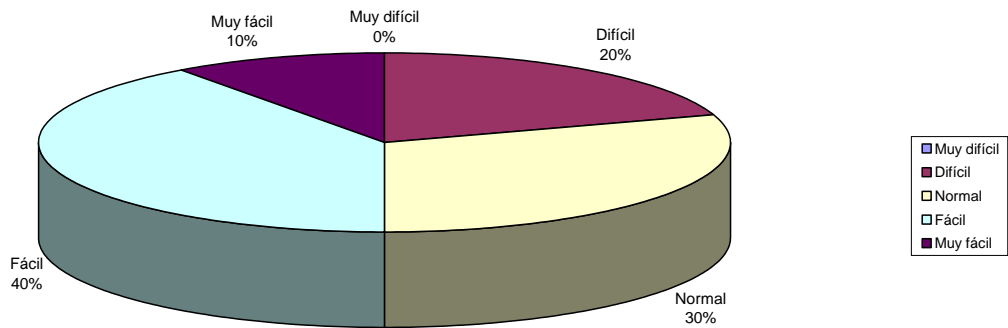
Pregunta 6-¿Cree que es útil esta aplicación? Puntúe en una escala de 1 a 5 (1= "Bajo", 5="Alto")

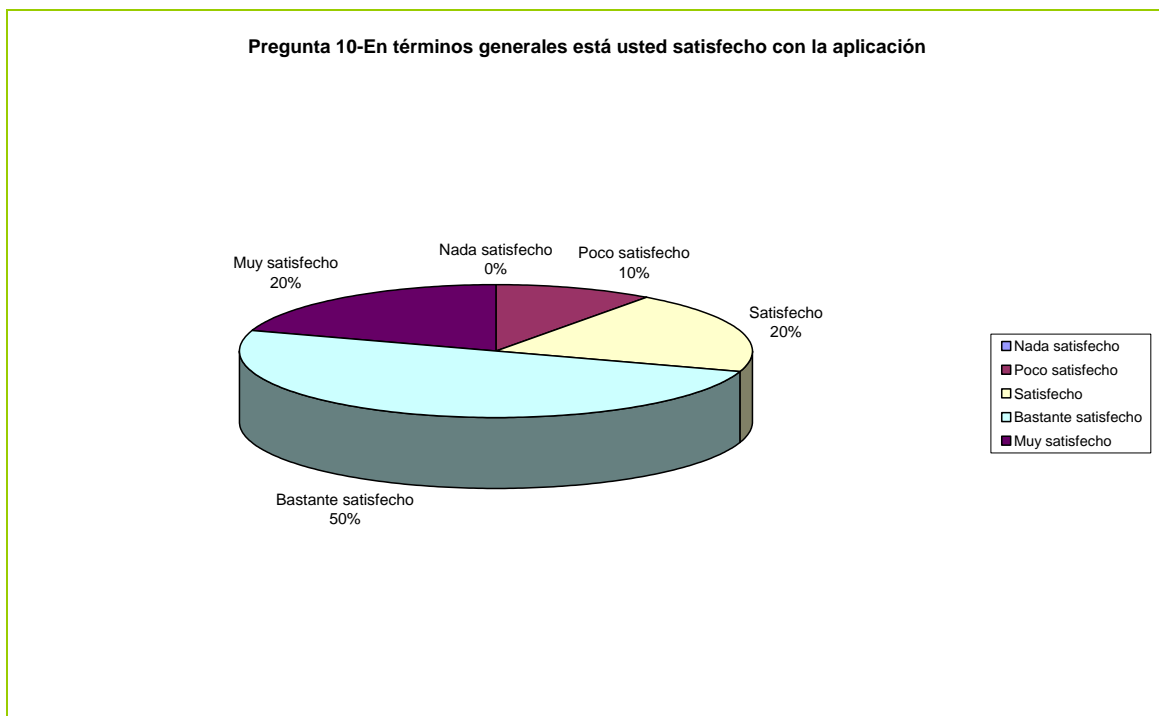
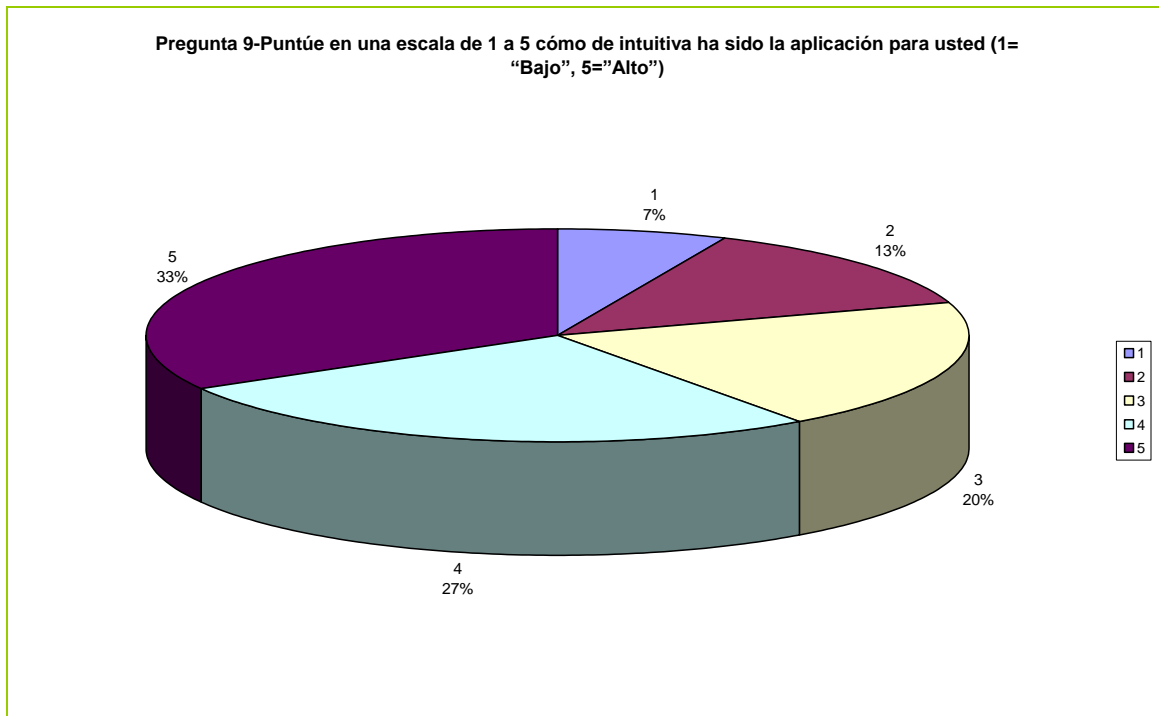


Pregunta 7-¿Le ha sido fácil realizar las acciones que quería?



Pregunta 8-Establezca el nivel de dificultad de la aplicación para usted.





**Figura 45. Estadísticas de los resultados para cada una de las preguntas que se plantean en la encuesta de valoración de la aplicación.**

De los resultados de la encuesta se obtiene que la mayoría de los encuestados tiene un grado de conociendo alto del uso de los Smartphones por lo que son usuarios competentes a la hora de probar la aplicación y poder compararla con otras. Así pues la mayoría de ellos está de acuerdo en que se trata de una aplicación bastante útil, que fue capaz de interactuar con el usuario de manera sencilla, estableciendo así que el nivel de dificultad de la aplicación se encuentra entre fácil y normal.

No todos los encuestados estaban familiarizados con las interfaces orales, aun así la mayoría de ellos considera que la navegación por la misma es buena, pero que ésta no siempre comprende bien la petición solicitada.

En general, de los resultados de la encuesta se obtiene que la gran mayoría de los usuarios estén contentos con la aplicación y más con su utilidad.



# Capítulo 6

## Conclusiones, trabajo futuro y presupuesto

En este capítulo se revisan los resultados obtenidos con el fin de extraer conclusiones y hacer balance del proyecto realizado en el presente Trabajo Fin de Grado. Seguidamente se exponen posibles líneas futuras que se podrían efectuar sobre la aplicación desarrollada con el fin de mejorar su eficiencia, así como un probable aumento de las funcionalidades que cubriría.

### 6.1 Conclusiones

En este Trabajo Fin de Grado se ha desarrollado una aplicación en Android basada en el reconocimiento de voz. Se trata de una aplicación que ofrece al usuario toda la información relativa a la Biblioteca UC3M de Leganés; satisface tanto preguntas generales relacionadas con la distribución de la biblioteca, como preguntas más concretas sobre la ubicación de cada libro, mostrando un mapa para completar dicha explicación. Es posible realizar todas éstas funciones mediante el habla, o bien escribiendo la consulta deseada mediante el teclado del dispositivo móvil o interactuando con la pantalla a través del menú deslizable.

Las funcionalidades que proporciona la aplicación se estructuran en módulos, que permiten al usuario descubrir las diferentes aportaciones que ofrece.

Comenzando por el *módulo de Inicio*, la aplicación ofrece la opción de usar la voz, el teclado o el menú para acceder a las distintas funcionalidades dependiendo de las preferencias o capacidades del usuario. Así pues si éste decide usar la voz será conducido al *módulo de Asistente por voz*. En éste se implementa el reconocedor automático del habla (ASR) de Google como entrada al sistema y la síntesis de texto a voz (TTS) como salida del mismo y su finalidad es reconocer correctamente las palabras del usuario para poder, posteriormente compararlas con la base de datos y así, una vez consultada, contestar al usuario con la información requerida.

Pero si por el contrario el usuario toma la opción de escribir la consulta o de acceder a preguntas concretas a través del menú, será conducido directamente a los *módulos Búsqueda de palabras clave* o *módulo de Búsqueda de Libros*. Estos dos módulos se encargan de consultar la base de datos y, según la petición obtenida, devolver la respuesta adecuada. El primero consulta la tabla *palabras clave* de la base de datos en la cual están almacenadas las posibles palabras que puede decir el usuario y que están relacionadas con la Biblioteca y la respuesta que se le proporciona al usuario. El segundo módulo se encarga de consultar la *tabla libros* la cual contiene la materia, el código, el título, el autor, la editorial, el ISBN y la disponibilidad relativa a cada libro. Una vez obtenida ésta información se muestra al usuario ordenada por materias.

Si estas respuestas necesitan la conexión a una página web, se conduce al *módulo de Consulta página web*, que consulta la base de datos y muestra el resultado por pantalla. Si por el contrario es necesario mostrar al usuario un mapa explicativo, se recurre al *módulo Indicaciones-plano*, que mediante el sistema de diálogo, explica al usuario la localización del recurso solicitado, apoyándose además en un plano.

Para poder implementar en la aplicación las funcionalidades de los sistemas de diálogo que incorpora el sistema operativo Android, en primer lugar, se ha tenido que realizar un estudio completo de dichos sistemas y de las versiones existentes. En el capítulo *Estado del arte* se incluyen sus funcionalidades, la arquitectura, los módulos que los conforman y ejemplos existentes en la actualidad.

Posteriormente, en el capítulo *Descripción general del sistema*, se detalla el uso de las distintas tecnologías y servicios implementados, tales como *AsyncTask*, *RecognizerIntent*, *JSONObject*, *JSONArray*, *Intent*, *GridView*, *AlertDialog*, *ArrayList<HashMap<String, String>>*, *HashMap<String, String>*, etc. del lado del cliente y del lado del servidor PHP, MySQL, HTTP y JSON.

Como conclusión final, podemos afirmar que los objetivos planteados en un principio se han cumplido. Con la realización de este Trabajo Fin de Grado se ha llevado a cabo un estudio completo de los distintos sistemas de diálogo, de los sistemas operativos móviles y del uso de bases de datos. Se ha hecho especial hincapié en Android, para poder así implementar todas las posibilidades que éste ofrece, con el fin de lograr una comunicación lo más sencilla y efectiva posible. Para su desarrollo también ha sido necesario el estudio de la organización de la Biblioteca UC3M de Leganés, sobre la que se ha desarrollado la aplicación. En cuanto al reconocimiento de los mensajes emitidos por el usuario, se ha estudiado de qué manera es posible gestionar la prioridad de las palabras que éste dice. Para ello, finalmente se ha organizado la base de datos que contiene las palabras clave, así pues, primeramente, se han colocado las palabras más frecuentes, y en último lugar las menos frecuentes. Se han realizado sucesivas pruebas

con distintos usuarios para analizar las distintas preguntas que estos hacen y cómo las hacen. Así por ejemplo, si el usuario dice: “¿Dónde está el baño de la Biblioteca?” primeramente se ha colocado en la base de datos la palabra baño y después biblioteca, para que de la respuesta asociada a “baño” y no a “biblioteca”.

Gracias a la aplicación desarrollada, cualquier usuario puede consultar información sobre la Biblioteca UC3M de Leganés sin necesidad de estar físicamente en ella, de una manera fácil y flexible ya que puede realizar la consulta tanto de forma oral como haciendo uso del teclado virtual del que dispone el dispositivo móvil Android. Así pues si el usuario se encuentra en un lugar en el que no se disponga de Internet, puede seguir haciendo uso completo de la aplicación mediante el teclado.

Otro de los objetivos cumplidos es el de realizar una aplicación que contenga, en su mayoría, toda la información referente a la biblioteca almacenada en una base de datos y que por lo tanto, la finalidad principal del código sea consultarla e implementar las funcionalidades correspondientes a la consulta solicitada. La finalidad de este objetivo es el de desarrollar una aplicación estándar que pueda servir para cualquier biblioteca, que sólo sea necesario actualizar la base de datos con la nueva información de la biblioteca, sin necesidad de hacer cambios en el código.

Finalmente, también se ha cumplido con uno de los objetivos más importantes, el de facilitar la accesibilidad a los servicios proporcionados por esta aplicación a usuarios inexpertos en el uso de un ordenador o de Internet, así como a personas con discapacidades auditivas y visuales.

En la actualidad, la mayoría de las aplicaciones orales disponibles en el mercado permiten únicamente realizar una serie de acciones por medio de la voz, sin existir la reciprocidad de que la aplicación se comunique con el usuario también de forma oral.

La complejidad de la aplicación es elevada debido a la diversidad de las funciones implementadas, destacando la implementación del sistema de diálogo y el uso de las bases de datos.

Concluyendo, este Trabajo Final de Grado desarrolla e innova un portal de voz bibliotecario pionero en ofrecer una amplia oferta de servicios relacionados con la Biblioteca UC3M de Leganés, sin necesidad de estar físicamente en ella. Gracias a ella, los usuarios disponen de un sistema útil y accesible, que proporciona de forma fácil y eficaz el acceso a las funcionalidades básicas relacionadas con la biblioteca.



## 6.2 Trabajo futuro

Como líneas de trabajo futuras, se proponen los siguientes puntos:

- Conseguir que el servicio de reconocimiento de voz trabaje también cuando la aplicación no tenga acceso a Internet. De esta manera, el servicio estaría disponible en todo momento, independientemente de si el usuario dispone o no de tráfico de datos.
- Implementar el mecanismo de “Siempre escuchando” de forma que la aplicación siempre esté a la espera de que el usuario diga una palabra claves para comenzar el diálogo.
- Incluir la posibilidad de implementar la aplicación en modo multilingüe, permitiendo al usuario usarla en el lenguaje deseado, ampliando considerablemente las fronteras de difusión de la aplicación.
- Ampliar el contenido de la aplicación y las bases de datos incluyendo información relativa a la Biblioteca UC3M de Getafe y de Colmenarejo.
- Añadir nuevas funcionalidades y consultas relacionadas con la Biblioteca UC3M de Leganés.
- Portar la aplicación a otros sistemas operativos, como Apple y Windows Phone, consiguiendo así más popularidad y difusión.

## 6.3 Presupuesto

El presente presupuesto es consecuencia de los datos aportados en la Figura 4, apartado 1.3, en la que se presenta en un Diagrama de Gantt la planificación temporal de este TFG. Como modelo de ayuda para confeccionarle se utiliza la plantilla que proporciona la universidad (UC3M, 2014).

### Costes de Personal

A partir de la fórmula:  $\text{Coste} = \text{duración horas} * \text{coste hora/hombre de un ingeniero}$ . Se calcula el coste de personal, teniendo en cuenta que para el presente proyecto se han trabajado un total de 592 y se fija el coste hora/ingeniero a 30€/hora. Dando como resultado unos gastos de personal de 17.760 €.

## Costes de Equipo

Para la elaboración del proyecto ha sido necesario adquirir:

- Recursos Hardware
  - Ordenador portátil HP. 500€
  - Ordenador portátil MacBookAir. 800€
  - Terminal móvil Samsung Galaxy. 250€
  - Cable USB. 10€
  
- Recursos Software
  - Entorno de desarrollo integrado de código abierto multiplataforma Eclipse.
  - SDK (Software Development Kit) de Android: kit de desarrollo de software para Android.
  - JDK (Java Development kit): kit de desarrollo de Java.
  - Plug-in ADT (Android Development Tools) para Eclipse: permite desarrollar aplicaciones Android utilizando Eclipse.
  - Dropbox.
  - Paquete Microsoft Office 2010.
  - Adobe Reader.

En el coste de los equipos solo se considera el uso de los elementos Hardware, ya que los de software son de coste 0€ y se calcula de una forma simplificada el 15% del valor de estos para la realización de este TFG teniendo en cuenta el uso dado y la amortización de los mismos. Total 234€

Finalmente, la siguiente tabla muestra el resumen de costes del presente proyecto.

Costes	Horas	Coste (€)	Total
<b>Personal</b>	592	30	17760
<b>Equipos</b>		1560	
<b>Amortización Equipo</b>	15%		234
<b>Total Costes</b>			17994
<b>IVA</b>	21%		3778,74
<b>TOTAL</b>			21772,74

**Figura 46. Presupuesto de la Aplicación**

El presupuesto total de este proyecto asciende a la cantidad de VEINTE UN MIL SETECIENTOS SESENTA Y DOS CON SETENTA Y CUATRO CENTIMOS.

Madrid a 22 de setiembre 2014

Fdo. Marta Castro Llorente



# Glosario

**Android** es un sistema operativo basado en Linux, diseñado principalmente para dispositivos móviles con pantalla táctil como teléfonos inteligentes o tabletas, inicialmente desarrollado por Android Inc., que Google respaldó económicamente y más tarde compró en 2005.

**API (Application Programming Interface)**: conjunto de llamadas que ofrecen acceso a funciones y procedimientos, representando una capa de abstracción para el desarrollador.

**ASR (Automatic Speech Recognition)**: módulo de un sistema de diálogo que se encarga de reconocer la señal de voz emitida por el usuario y devolver la(s) secuencia(s) de palabras reconocida(s) más probable mediante la aplicación de técnicas de proceso de señal de voz. Las siglas en español son RAH (Reconocimiento Automático del habla).

**APK (Application Package File)**: paquete para el sistema operativo Android.

**Background**: representa un proceso que se ejecuta en segundo plano.

**Dalvik**: nombre de la máquina virtual utilizada por el sistema operativo Android. Dalvik está específicamente adaptada a las características de rendimiento de un dispositivo móvil y trabaja con ficheros de extensión “.dex”, obtenidos desde el bytecode de Java.

**E/S**: abreviatura de Entrada/Salida. Un elemento de E/S es aquél que permite la comunicación entre un sistema de procesamiento de datos y una entidad externa a él (un usuario humano u otro sistema de procesamiento). Entrada se considera toda aquella información que es recibida por un sistema, mientras que salida es aquella información que es enviada por el mismo.

**FTP (File Transfer Protocol)**: protocolo de red para la transferencia de ficheros entre sistemas conectados a una red TCP (*Transmission Control Protocol*) basado en la arquitectura cliente servidor.

**GPL (General Public License):** licencia que permite utilizar el software MySQL de forma gratuita.

**GPS (Global Positioning System):** sistema global de navegación que, mediante satélites, permite ubicar un objeto en la superficie terrestre con una precisión que va desde varios metros a centímetros.

**HTML (HyperText Markup Languaje):** lenguaje de marcado para la elaboración de páginas web.

**HTTP (Hypertext Transfer Protocol):** desarrollado por W3C e IETF, es un protocolo de nivel de aplicación de la web que sigue el esquema petición-respuesta entre un cliente y un servidor.

**IDE (Integrated Development environment) de Eclipse:** entorno de desarrollo integrado de código abierto multiplataforma.

**IETF (Internet Engineering Task Force):** organización internacional abierta de normalización, que tiene como objetivos el contribuir a la ingeniería de Internet, actuando en diversas áreas, como transporte, encaminamiento y seguridad.

**Intent:** elemento básico de comunicación entre los distintos componentes Android de una aplicación o entre distintas aplicaciones. Mediante un intent se puede mostrar una actividad desde cualquier otra, iniciar un servicio, enviar un mensaje broadcast, iniciar otra aplicación, etc.

**Java** es un lenguaje de programación desarrollado en el año 1995 por James Gosling de Sun Microsystems específicamente para entornos de red como Internet. Se trata de un lenguaje orientado a objetos, lo que significa que sus programas se construyen con módulos de código. A diferencia de otros lenguajes de programación, Java es un lenguaje interpretado, y por lo tanto, tiene más tiempo de ejecución.

**JDK (Java Development Kit):** kit de desarrollo de Java.

**JSON (JavaScript Object Notation):** formato ligero de intercambio de datos. El formato de texto JSON es completamente independiente del lenguaje con el que se está programando pero utiliza convenciones ampliamente utilizadas en distintos lenguajes de programación. Por lo tanto, JSON es un lenguaje ideal para el intercambio de datos.

**MySQL:** MySQL es un sistema de gestión de bases de datos relacionales desarrollado y distribuido por la compañía comercial MySQL AB. El software de base de datos MySQL proporciona un servidor de base de datos SQL muy rápido, multihilo, multiusuario y robusto. Alimenta desde websites de aficionados hasta plataformas profesionales como WordPress.

**OpenGL** es una especificación estándar que define una API multilenguaje y multiplataforma para escribir aplicaciones que produzca gráficos 2D y 3D. La interfaz consiste en más de 250 funciones diferentes que pueden usarse para dibujar escenas tridimensionales complejas a partir de primitivas geométricas simples, tales como puntos, líneas y triángulos

**PDO (PHP Data Objects):** define una interfaz ligera para poder acceder a bases de datos en PHP. No se puede realizar ninguna de las funciones de la base de datos utilizando la extensión PDO por sí misma sino que se debe utilizar un controlador de PDO específico de la base de datos para tener acceso a un servidor de bases de datos.

**PHP (Hypertext Preprocessor):** lenguaje de código abierto utilizado en el desarrollo web y que puede ser incrustado en HTML. Principalmente, PHP está enfocado a la programación de scripts en el lado del servidor.

**PhpMyAdmin:** administrador de bases de datos MySQL que se utiliza vía web.

**plug-in ADT (Android Development Tools):** plug-in para Eclipse que permite desarrollar aplicaciones Android utilizando Eclipse

**RAH (Reconocimiento Automático del habla):** módulo de un sistema de diálogo que se encarga de reconocer la señal de voz emitida por el usuario y devolver la(s) secuencia(s) de palabras reconocida(s) más probable mediante la aplicación de técnicas de proceso de señal de voz. Las siglas en inglés son ASR (*Automatic Speech Recognition*).

**RecognizerIntent:** clase del paquete android.speech que define las constantes necesarias para integrar el reconocimiento de voz iniciado desde un *intent*.

**SDK (Software Development Kit):** constituye un conjunto de herramientas que permiten a un desarrollador crear aplicaciones para una determinada plataforma o lenguaje.

**Service:** componente de una aplicación para Android que representa una aplicación ejecutada sin interfaz de usuario, generalmente en segundo plano, mientras otras aplicaciones (con interfaz) están activas en la pantalla del dispositivo. Este componente se implementa a través de la clase Service.

**Sistema de diálogo:** programas informáticos cuya finalidad es interactuar con los usuarios oralmente o de forma multimodal para proporcionar diversos servicios.

**Smartphone:** dispositivo móvil que representa una evolución de los teléfonos móviles, con la inclusión de pantalla táctil, teclado, conexión Wi-Fi, aplicaciones de usuario como navegador web o cliente de correo, entre otros.

**TextToSpeech:** clase del paquete android.speech.tts que se encarga de la síntesis de texto a voz.

**TTS (Text-to-Speech):** módulo de un sistema de diálogo que se encarga de transformar la respuesta que recibe del sistema como texto en lenguaje natural, en la señal de audio correspondiente que recibirá el usuario.

**W3C (World Wide Web Consortium):** asociación internacional formada por organizaciones miembros del consorcio, personal y el público en general, que trabajan conjuntamente para desarrollar estándares web. W3C pretende guiar la web hacia su máximo potencial a través del desarrollo de protocolos y pautas que aseguren el crecimiento futuro de la web.

**XAMPP:** servidor independiente de plataforma de código libre para el almacenamiento de los ficheros PHP y de las bases de datos de la aplicación. XAMPP es el acrónimo de Cualquier Plataforma (X), Apache (A), MySQL (M), PHP (P) y Perl (P).

**XML (Extensible Markup Language):** representa un lenguaje estándar que, mediante el uso de etiquetas y atributos, permite expresar e intercambiar fácilmente estructuras de datos.



# Bibliografía

[ANDROID] <https://sites.google.com/site/swcuc3m/home/android/generalidades/2-2-arquitectura-de-android>

[ANDROIDver] <http://www.androidcurso.com/index.php/tutoriales-android/31-unidad-1-vision-general-y-entorno-de-desarrollo/146-las-versiones-de-android-y-niveles-de-api>

[ANDROIDEstruc] <http://www.androidcurso.com/index.php/tutoriales-android/31-unidad-1-vision-general-y-entorno-de-desarrollo/149-componentes-de-una-aplicacion>

[ANDROIDvida] <http://www.androidcurso.com/index.php/tutoriales-android/37-unidad-6-multimedia-y-ciclo-de-vida/158-ciclo-de-vida-de-una-aplicacion>

[BNE] <http://www.bne.es/es/Inicio/index.html>

[CEREPROC] <https://www.cereproc.com/>

[CTV] [http://es.wikipedia.org/wiki/Conversor\\_texto-voz](http://es.wikipedia.org/wiki/Conversor_texto-voz)

[DIHANA] Luís Hurtado, Fernando Blat, S. Grau, David Griol, Emilio Sanchis, Encarna Segarra, F. Torres Sistema de diálogo para el Proyecto DIHANA Procesamiento del lenguaje natural, ISSN 1135-5948, N°. 35, 2005, págs. 453-454



[ECLIPSE] [http://es.wikipedia.org/wiki/Eclipse\\_\(software\)](http://es.wikipedia.org/wiki/Eclipse_(software))

[EMEZETA] EMEZETABLOG Hernández José Ramón Ing Tec. Informática ULL Tenerife <http://www.emezeta.com/articulos/10-sintetizadores-de-voz-tts-para-android>

[Epistemowikia] [http://cala.unex.es/cala/epistemowikia/index.php?title=Reconocimiento\\_Autom%C3%A1tico\\_del\\_Habla#El\\_Problema\\_del\\_Reconocimiento\\_Autom.C3.A1tico\\_del\\_Habla](http://cala.unex.es/cala/epistemowikia/index.php?title=Reconocimiento_Autom%C3%A1tico_del_Habla#El_Problema_del_Reconocimiento_Autom.C3.A1tico_del_Habla)

[ESPEAK] <http://espeak.sourceforge.net/>

[Gantt] [http://es.wikipedia.org/wiki/Diagrama\\_de\\_Gantt](http://es.wikipedia.org/wiki/Diagrama_de_Gantt)

[Griol2007] Desarrollo y Evaluación de Diferentes Metodologías para la Gestión Automática del Diálogo (Tesis Doctoral). Valencia, España: Universidad Politécnica de Valencia.

[HTML] <http://es.wikipedia.org/wiki/HTML>

[HTS] <http://hts.sp.nitech.ac.jp/>

<http://www.ivona.com/>

[JSON] <http://es.wikipedia.org/wiki/JSON>

[Kioskea] <http://es.kioskea.net/faq/7819-las-aplicaciones-de-reconocimiento-de-voz-para-ios-y-android#cual-es-la-utilidad-de-las-aplicaciones-de-reconocimiento-de-voz>

[Lebai2013] Lebai Lutfi, Syaheerah Binti 2013 User-centric Need-driven Affect Modeling for Spoken Conversational Agents: Design and Evaluation Tesis Doctoral ETSIT (UPM)

[López\_Cózar2014] Sistemas de diálogo hablado y multimodal. [http://www.ugr.es/~rlopezc/sistemas\\_dialogo.htm](http://www.ugr.es/~rlopezc/sistemas_dialogo.htm)

[MySQL] <http://es.wikipedia.org/wiki/MySQL>

[MySQLcaract] <http://dev.mysql.com/doc/refman/5.0/es/features.html>  
<http://www.gridmorelos.uaem.mx/~mcruz//cursos/miic/MySQL.pdf>  
<http://cnx.org/contents/fa33ef02-f882-4b8d-9f67-4be268dd6d03@1>

[PHP] <http://es.wikipedia.org/wiki/PHP>  
<http://php.net/manual/es/intro-what-is.php>  
<http://php.net/manual/es/intro-what-cando.php>

[Quijote] <http://cervantesvirtual.com/servlet/CuentaBibsMundo?portal=0&codigo=28&url=http%3A%2F%2Fwww.kcl.ac.uk%2Fhumanities%2Fcch%2F>

[RAH] [http://es.wikipedia.org/wiki/Reconocimiento\\_del\\_habla](http://es.wikipedia.org/wiki/Reconocimiento_del_habla)

[SD\_TEAM] Procesamiento del Lenguaje Natural, Revista nº 45, septiembre 2010, pp 331-332 ISSN 1135-5948 © 2010 Sociedad Española para el Procesamiento del Lenguaje Natural

[SEPLN] Sociedad Española para el Procesamiento del Lenguaje Natural  
<http://www.sepln.org/>

[SIRI] <https://www.apple.com/es/ios/siri/>

[Smartphone] <http://www.gartner.com/newsroom/id/2623415>

[SPAESP]  
<https://play.google.com/store/apps/details?id=com.svox.pico.voice.spa.esp>

[SOMercado] <http://www.marketing-movil-sms.com/noticias/que-sistema-operativo-movil-me-interesa-para-mi-app/>

[SOMobiles]  
[http://es.wikipedia.org/wiki/Sistema\\_operativo\\_m%C3%B3vil#Sistemas\\_Operativos\\_M.C3.B3viles](http://es.wikipedia.org/wiki/Sistema_operativo_m%C3%B3vil#Sistemas_Operativos_M.C3.B3viles)

[SOUcados] <http://blog.uchceu.es/informatica/ranking-de-sistemas-operativos-mas-usados-para-2014/>

[SVOX] <http://en.wikipedia.org/wiki/SVOX>

[Tomei2013] Tomei, Lawrence (Robert Morris University, USA) Advances in Educational Technologies and Instructional Design (AETID) Book Series 2013 ISSN: 2326-8905 Technologies for Inclusive Education: Beyond Traditional Integration Approaches Chapter 1 Towards the Use of Dialog Systems to Facilitate Inclusive Education <http://www.safaribooksonline.com/library/view/technologies-for-inclusive/9781466625303/Text/978-1-4666-2530-3.ch001.xhtml>

[UCHCEU] <http://blog.uchceu.es/informatica/ranking-de-sistemas-operativos-mas-usados-para-2014/>

[Vlingo] <http://www.vlingo.com>

[Vocalia] <http://www.creaceed.com/es/vocalia>

[W3C] <http://www.w3c.es/>

[XAMPP] <https://www.apachefriends.org/es/index.html>

[XAMPP1] <http://myu-charly.blogspot.com.es/>