

# Plataforma middleware para recomendadores contextuales en plataformas móviles

Víctor Manuel Martínez Coello

MÁSTER EN INVESTIGACIÓN EN INFORMÁTICA. FACULTAD DE INFORMÁTICA  
UNIVERSIDAD COMPLUTENSE DE MADRID

---



Trabajo Fin Máster en Sistemas Inteligentes

Septiembre 2013

Director/es y/o colaborador:

Díaz Agudo, María Belén  
Recio García, Juan Antonio

# Autorización de difusión

Víctor Manuel Martínez Coello

Septiembre 2013

El/la abajo firmante, matriculado/a en el Máster en Investigación en Informática de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Máster: “Plataforma middleware para recomendadores contextuales en plataformas móviles”, realizado durante el curso académico 2012-2013 bajo la dirección de María Belén Díaz Agudo y Juan A. Recio García en el Departamento de ISIA, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en Internet y garantizar su preservación y acceso a largo plazo.

# Resumen en castellano

Como contribución en este trabajo se presenta un estudio sobre las tecnologías móviles existentes en la actualidad, para la obtención de información contextual a través de los dispositivos móviles, con el objetivo de su explotación en sistemas recomendadores dependientes del contexto. Para poder obtener la información estudiada en este trabajo a través de los dispositivos móviles se presenta la librería MobileContext para la implementación de sistemas recomendadores contextuales en distintas plataformas móviles.

## Palabras clave

Sistemas de recomendación, contexto, plataformas móviles, dispositivos móviles, Phone-Gap.

# Abstract

As a contribution, this work showcases a research about current mobile technologies and their uses, focusing in their exploitation in context-aware recommender systems. For obtaining the studied information from the mobile devices, the library MobileContext is presented, which is available for being be used in different implementations of context-aware recommender systems in several mobile platforms.

## Keywords

Recommender systems, context, mobile platforms, mobile devices, PhoneGap.

# Índice general

<b>Índice</b>	<b>I</b>
<b>Lista de Figuras</b>	<b>III</b>
<b>Lista de Tablas</b>	<b>IV</b>
<b>Agradecimientos</b>	<b>V</b>
<b>1. Introducción</b>	<b>2</b>
1.1. Objetivos . . . . .	4
1.2. Estructura de la memoria . . . . .	5
<b>2. Trabajo relacionado</b>	<b>8</b>
2.1. Sistemas recomendadores . . . . .	8
2.1.1. Sistemas de recomendación aplicados al comercio electrónico . . . . .	12
2.2. Contexto . . . . .	13
2.3. Plataformas móviles . . . . .	15
2.4. Aplicaciones contextuales para dispositivos móviles . . . . .	21
2.4.1. Geolocalización . . . . .	22
2.4.2. Comunicación . . . . .	23
2.4.3. Social . . . . .	25
2.5. Herramientas para el desarrollo de aplicaciones contextuales en dispositivos móviles . . . . .	26
2.5.1. Conclusiones . . . . .	29
<b>3. Categorización del conocimiento contextual aplicable a aplicaciones de recomendación en dispositivos móviles</b>	<b>32</b>
3.1. Dimensión 1: Tipología del contexto móvil . . . . .	34
3.1.1. Contexto social . . . . .	34
3.1.2. Contexto geotemporal . . . . .	36
3.1.3. Contexto tecnológico . . . . .	36
3.1.4. Contexto personal . . . . .	37
3.2. Dimensión 2: Dinamismo . . . . .	38
3.3. Dimensión 3: Interacción con el usuario . . . . .	43
<b>4. Librería para el desarrollo de aplicaciones de recomendación contextual en dispositivos móviles</b>	<b>50</b>
4.1. Tecnologías móviles para la obtención del conocimiento contextual . . . . .	50

4.2. MobileContext . . . . .	54
4.2.1. Base tecnológica . . . . .	55
4.2.2. Descripción funcional de MobileContext . . . . .	62
4.2.3. Demostrador . . . . .	71
4.3. Caso de uso: MobiMall . . . . .	78
<b>5. Conclusiones y trabajo futuro</b>	<b>80</b>
5.1. Trabajo futuro . . . . .	82
<b>Bibliografía</b>	<b>84</b>

# Lista de Figuras

1.1. Relación tecnología-contexto-marketing . . . . .	3
2.1. Estudio Morgan Stanley incremento ventas smartphones . . . . .	15
2.2. Arquitectura sistema operativo Android . . . . .	17
2.3. Arquitectura sistema operativo iOS . . . . .	19
2.4. Mercado plataformas móviles . . . . .	21
2.5. Evolución mercado móvil . . . . .	27
4.1. Arquitectura App con PhoneGap . . . . .	55
4.2. Arquitectura desarrollo plugin . . . . .	57
4.3. Arquitectura extensión funcionalidades de PhoneGap . . . . .	58
4.4. Arquitectura plataforma propuesta . . . . .	59
4.5. Geolocalización latitud-longitud . . . . .	72
4.6. Geolocalización por descripción . . . . .	72
4.7. Clima en la localización del dispositivo . . . . .	73
4.8. Información sobre dispositivo móvil . . . . .	74
4.9. Conexiones disponibles/estados . . . . .	74
4.10. Estado conexión Wifi . . . . .	75
4.11. Búsqueda de contacto en agenda . . . . .	75
4.12. Formulario login en Facebook . . . . .	76
4.13. Información desde perfil de Facebook . . . . .	76
4.14. Captura fotografía . . . . .	77
4.15. Escaneo código de barras/QR . . . . .	77
4.16. Esquema aplicación MobiMall . . . . .	78

# Lista de Tablas

3.1. Dinamismo atributos tipología social . . . . .	39
3.2. Dinamismo atributos tipología geotemporal . . . . .	40
3.3. Dinamismo atributos tipología tecnológica . . . . .	41
3.4. Dinamismo atributos tipología personal . . . . .	42
3.5. Obtención contexto social . . . . .	44
3.6. Obtención contexto geotemporal . . . . .	46
3.7. Obtención contexto tecnológico . . . . .	47
3.8. Obtención contexto personal . . . . .	48

# Agradecimientos

En primer lugar quiero agradecer a Belén Díaz Agudo y Juan A. Recio García por haberme permitido realizar este proyecto, así como por su ayuda y trabajo a lo largo del curso. También agradecer a mi familia y amigos por haberme ayudado y apoyado a lo largo de este camino una vez más, sin ellos no habría sido posible.



# Capítulo 1

## Introducción

En pocos años el uso de los teléfonos móviles ha cambiado drásticamente, pasando de ser utilizados casi únicamente para realizar llamadas telefónicas y envío de mensajes, tanto cortos como multimedia, a ser utilizados para acceder a contenidos web, como navegadores GPS, como cámara de fotos o como medio de conexión y difusión en redes sociales. Además, el éxito de las redes sociales desde la aparición de la web 2.0 y las características de los denominados teléfonos inteligentes o *smartphones* han contribuido a una mayor participación en estas redes y una mayor publicación de contenidos gracias a la posibilidad de estar conectado desde cualquier lugar. Esta combinación entre redes sociales y los smartphones ofrece una gran cantidad de oportunidades, principalmente para las empresas, las cuales pueden obtener una mayor cantidad de información de los usuarios y llevar a cabo campañas de marketing online aprovechando estos medios. Por ejemplo, conocer la localización exacta de un usuario en un momento determinado y obtener información de éste a través de las redes sociales permitiría llevar a cabo ofertas personalizadas o adecuadas a dicha situación, pudiendo realizar dichas ofertas a través del dispositivo móvil del usuario. Este tipo de campañas resultan más atractivas que las convencionales, en las que se realizan ofertas genéricas a través de correo electrónico o propaganda. Además las ofertas personalizadas están limitadas a técnicas de marketing como la fidelización a través de programas de puntos. Aproximaciones como *SOLOMO*<sup>1</sup>, acrónimo de Social + Local + Mobile ofrecen

---

<sup>1</sup><http://solomotechnology.com/products/>

herramientas para el marketing basado en localización y redes sociales considerándolo una apuesta segura de futuro. Por ello en [Esteban, 2013] se estudia la posibilidad de aplicar este tipo de técnicas de marketing mediante la utilización de sistemas de recomendación contextuales 2.1. Para ello se estudian diversas técnicas de marketing y se adaptan a la tendencia comentada anteriormente del marketing online. En dicho trabajo se define la relación entre las técnicas de marketing y los sistemas de recomendación dependientes del contexto. Esta relación viene dada por la información contextual del usuario que será necesaria para aplicar cada una de dichas técnicas.

La mayor dificultad para poder aplicar estas técnicas es la obtención de la información contextual. Este trabajo se centra en ese aspecto, estudiar cómo se pueden obtener los atributos contextuales definidos en [Esteban, 2013] con las tecnologías móviles actuales. En la Figura 1.1 puede verse la colaboración entre los dos trabajos, la relación entre el contexto y las técnicas de marketing en [Esteban, 2013] y la de la tecnología con el contexto en este.

Para alcanzar este objetivo se ha realizado un estudio sobre el uso de las tecnologías móviles

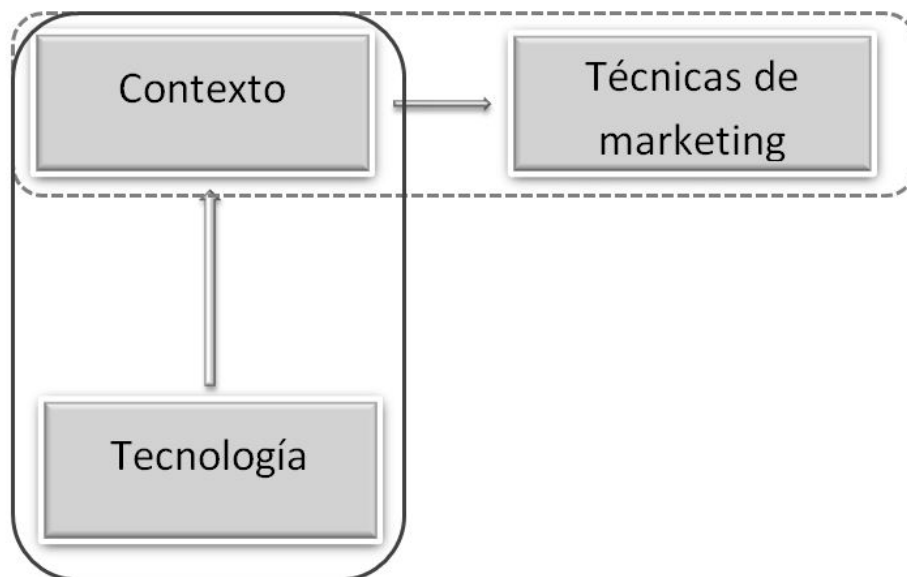


Figura 1.1: Relación tecnología-contexto-marketing

en la actualidad, así como de las distintas plataformas disponibles. Además de la evolución

en el mercado de los dispositivos móviles en los últimos años y sus posibilidades tecnológicas. Se pretende estudiar qué información puede ser obtenida mediante las tecnologías móviles y cómo aplicarse a los sistemas de recomendación. Concretamente se analizan las posibilidades de las tecnologías móviles a los sistemas de recomendación basados en contexto. De esta forma este trabajo se enmarca en la clasificación y obtención de la información contextual a través de dispositivos móviles para posibilitar el desarrollo de sistemas de recomendación contextuales, dejando fuera la relación de estos con las técnicas de marketing.

## 1.1. Objetivos

El objetivo principal es ofrecer una solución tecnológica para la implementación de sistemas de recomendación basados en contexto mediante plataformas móviles. Para llevar a cabo esta tarea se ha de identificar las tecnologías móviles disponibles en la actualidad, las distintas plataformas existentes en el mercado y las características de cada una de ellas con el objetivo de estudiar qué posibilidades ofrecen para la obtención de información contextual de un usuario a través de su dispositivo móvil. Una vez identificadas las principales plataformas móviles y las tecnologías disponibles en la actualidad para la obtención de información contextual se debe estudiar los métodos de obtención de dicha información contextual a través de las oportunidades que ofrecen las distintas plataformas como sus APIs o tecnologías disponibles tanto en los dispositivos como en servicios vía web. Esto incluye la categorización de la información contextual obtenible a través de plataformas móviles. De esta forma se ha desarrollado una clasificación de acuerdo a tres categorías: tipología, dinamismo e interacción con el usuario.

Como resultado de este trabajo se presenta una librería -MobileContext- que permite la obtención de la información contextual de forma independiente a la plataforma móvil utilizada. Como demostración del trabajo realizado se mostrará una serie de casos en los que se obtiene información contextual a través del dispositivo móvil, así como un caso de uso real de una aplicación que utiliza la librería propuesta para obtener atributos contextuales

y aplicar con ellos técnicas de marketing.

## 1.2. Estructura de la memoria

Esta memoria está estructurada de la manera que se describe a continuación. En primer lugar se expone el marco teórico de los sistemas recomendadores, así como el estudio del contexto y su utilidad en sistemas recomendadores. Además se muestra un estudio sobre las plataformas móviles más populares existentes en la actualidad y el uso del contexto en aplicaciones móviles. Una vez estudiado el trabajo relacionado y las tecnologías móviles existentes en la actualidad se presenta una categorización del contexto móvil definido en este trabajo y en [Esteban, 2013]. Por último, una vez realizada la categorización del contexto y su relación con las tecnologías móviles se propone una librería que permite la obtención de atributos contextuales de manera automática para su uso en sistemas de recomendación contextual. Además de la librería propuesta se presenta un demostrador en el que se obtienen atributos contextuales a través del dispositivo móvil, así como un caso práctico en el cual se utiliza dicha librería para obtener determinados atributos contextuales.

**Capítulo 2. Trabajo relacionado.** En este capítulo se ofrece una introducción a los sistemas de recomendación, dentro de los cuales se distinguen los distintos tipos de sistemas existentes así como las diferentes técnicas que se utilizan para su diseño e implementación, entre los cuales se destaca los sistemas que se valen de información adicional para mejorar el proceso de recomendación, estos son los recomendadores dependientes del contexto. Una vez estudiados los distintos tipos de sistemas existentes se realiza un estudio sobre las distintas tecnologías y plataformas móviles, identificándose distintas aplicaciones existentes que utilizan y/u obtienen información contextual en su funcionamiento. Por último se muestran las distintas herramientas disponibles para el desarrollo de sistemas de recomendación en dispositivos móviles.

**Capítulo 3. Categorización del conocimiento contextual aplicable a aplicaciones de recomendación en dispositivos móviles.** En el tercer capítulo se presenta un

modelo teórico del contexto móvil propuesto en este trabajo. Para ello se categorizan los atributos contextuales partiendo de la revisión del estado del arte. La categorización realizada consiste en 3 dimensiones: tipología contextual, dinamismo e interacción con el usuario. Para la primera dimensión se han agrupado los atributos contextuales según sus características o naturaleza identificándose 4 tipologías del contexto: social, personal, tecnológico y geotemporal. En cuanto a la segunda dimensión -dinamismo- se estudia la propiedad dinámica de cada atributo definido en el contexto móvil, diferenciándose 3 niveles de dinamismo: bajo, medio y alto. Como última categoría -interacción con usuario- se estudia la posibilidad de obtener atributos contextuales de manera automática sin la necesidad de ser introducidos explícitamente por un usuario.

**Capítulo 4. Librería para el desarrollo de aplicaciones de recomendación contextual en dispositivos móviles.** En el cuarto capítulo se presenta el desarrollo de una librería que permite la obtención de información contextual en dispositivos móviles. Dicha librería ha sido desarrollada según el modelo teórico expuesto en el capítulo anterior. Se propone la utilización del estándar multiplataforma propuesto por PhoneGap, para ello se muestra la arquitectura requerida en este estándar utilizando tecnologías web (Javascript, HTML, CSS) e implementaciones nativas. El desarrollo de la librería se ha desarrollado mediante la utilización de la API proporcionada por Phonegap, servicios web e implementaciones nativas en la plataforma móvil Android. Posteriormente se expone un demostrador formado por una serie de casos prácticos en los que se obtiene información contextual desde un dispositivo smartphone y, por último, un caso de uso de la librería propuesta para su aplicación a técnicas de marketing con sistemas de recomendación.

**Capítulo 5. Conclusiones y trabajo futuro.** En este capítulo se exponen las conclusiones del trabajo realizado. La primera de ellas corresponde a la identificación de las tecnologías móviles y el uso de las redes sociales a través de los smartphones, así como las posibilidades que la combinación de ambas ofrecen. En segundo lugar se destaca la propuesta de un modelo teórico del contexto móvil, definido a partir de la revisión del estado del

arte realizada en este trabajo. En tercer lugar, el desarrollo de la librería MobileContext que permite la obtención de información contextual de manera automática en smartphones siguiendo el estándar multiplataforma de Phonegap y el modelo teórico del contexto móvil definido en el anterior capítulo, así como una serie de casos prácticos en los que se obtiene dicha información. Por último se ha expuesto un caso de uso real en el que se utiliza la librería propuesta para obtener información contextual necesaria para la aplicación de técnicas de marketing mediante sistemas de recomendación.

Posteriormente se exponen las posibles líneas futuras de trabajo. Dentro de ellas se destaca una mayor integración con redes sociales (Facebook y Twitter) dadas las limitaciones encontradas en el desarrollo del trabajo; además de la necesidad de realizar implementaciones nativas para determinados atributos contextuales que no ha sido posible su obtención mediante servicios web o a través de la interfaz proporcionada por Phonegap, por lo cual su desarrollo ha sido realizado únicamente en la plataforma Android. En cuanto al modelo teórico propuesto en este trabajo se propone el estudio de otras dimensiones para el diseño e implementación de sistemas recomendadores concretos, ya que la propuesta realizada es un modelo general y puede no ser suficiente para sistemas con características más específicas. Por último resulta conveniente la utilización de la librería propuesta en distintos sistemas de recomendación y plataformas móviles para una mayor evaluación.

# Capítulo 2

## Trabajo relacionado

En este capítulo se presenta el marco teórico de los sistemas recomendadores en general y los distintos tipos de técnicas existentes para realizar recomendaciones tanto individuales como grupales. Además se incluye una introducción al concepto de contexto y su utilidad en sistemas de recomendación. Posteriormente, se estudian las distintas plataformas móviles existentes en la actualidad y las posibilidades tecnológicas que ofrecen para ser utilizadas en sistemas de recomendación contextual y así establecer la relación entre la tecnología móvil y los sistemas de recomendación .

### 2.1. Sistemas recomendadores

Los sistemas de recomendación surgen de la necesidad de facilitar la búsqueda y elección de productos o servicios a usuarios en dominios amplios o muy variados. Los sistemas de recomendación son una clase de sistemas de filtrado de información en los que se recomienda items que pueden interesar a un usuario en base a sus características, como pueden ser sus gustos o situación.

Los sistemas de recomendación se pueden clasificar según las técnicas utilizadas para filtrar la información (colaborativos, basados en contenido o aproximaciones híbridas); según el usuario objetivo (recomendadores individuales o grupales) y, por último, los recomendadores convencionales o aquellos que se valen de información adicional para mejorar el proceso de recomendación (recomendadores contextuales) [[Jannach et al., 2010](#)].

Existen diversas técnicas para realizar el filtrado de información en sistemas recomendadores, distinguiéndose dos principales. En primer lugar están los sistemas colaborativos, que utilizan las valoraciones de los productos que realizan los usuarios y, para un usuario concreto, se le recomiendan items que sus usuarios “parecidos” han valorado positivamente, para determinar esos usuarios parecidos se pueden utilizar distintos algoritmos como el coeficiente de correlación de *Pearson*. Este tipo de técnica resulta adecuada y eficaz cuando se dispone de muchas valoraciones por parte de usuarios. Productos como libros o música pueden ofrecer una gran cantidad de valoraciones ya que su precio no suele ser alto y los usuarios adquieren mayor cantidad de este tipo de productos. En cambio cuando se trata de productos o items cuya adquisición resulta más costosa o menos periódica (p.e. electrodomésticos) no se dispone de tanta información sobre valoraciones de los usuarios.

En situaciones en las que no se pueda disponer de una cantidad de valoraciones adecuada se pueden utilizar técnicas basadas en contenido. En este tipo de sistemas la recomendación se realiza mediante la búsqueda y filtrado de productos similares (en cuanto a características) a otros productos comprados por el usuario o que ha visitado en el caso de catálogos online. Para ello se utilizan técnicas como el *K-NN* (los K vecinos más próximos). Para la aplicación de técnicas basadas en contenido es necesario disponer de información categorizada de los productos para poder filtrar los productos de manera correcta.

Ambas técnicas tienen sus pros y sus contras, en el caso de la técnica de filtrado colaborativo se encuentra el problema de la falta de valoraciones o *cold start*, el cual se puede solucionar realizando estimaciones a partir de otras o asignándoles un valor neutro. Para los sistemas basados en contenido la falta de información de características de determinados productos puede ser un problema también. Otro problema presente en los sistemas de recomendación es el de la diversidad, por ejemplo, si un usuario visita un producto en un catálogo online y se le muestran a continuación productos similares nada garantiza que ese producto tenga las características que él busca.

Para solucionar estas debilidades también existen técnicas híbridas que utilizan filtrado cola-

borativo y basado en contenido en el proceso de recomendación [Burke, 2002]. Estas técnicas híbridas utilizan la técnica más adecuada en cada fase del proceso de recomendación dependiendo del dominio en que se aplique, de esta manera se logra reducir las debilidades de las técnicas mencionadas anteriormente y mejorar así la eficacia del sistema.

Para realizar las conversaciones en la recomendación destacan dos métodos, el de navegación por propuesta, en el que el vendedor propone productos al usuario y este da su opinión o deseo, y el de navegación mediante preguntas, en el que el vendedor realiza preguntas al usuario para determinar qué es lo que busca o necesita. Pero estas técnicas tienen sus inconvenientes como, por ejemplo, que el usuario no sepa describir bien lo que quiere o que el vendedor no realice las preguntas adecuadas (navegación mediante preguntas) y que el vendedor no muestre los productos correctos a las necesidades del usuario (navegación mediante propuesta).

En cuanto al usuario final del sistema de recomendación existen sistemas que no realizan recomendaciones individuales si no grupales. Estos últimos resultan útiles para la recomendación de actividades que se realicen en grupo como puede ser ir al cine o un viaje turístico. Un sistema de recomendación realiza un modelado del usuario, en el caso de sistemas grupales es necesario realizar una recomendación para todo el grupo y no sólo para un determinado individuo. Para ello existen dos métodos destacados, el primero consiste en, a partir de los perfiles individuales de cada usuario, obtener un usuario modelo de todo el grupo y realizar la recomendación a dicho perfil modelado; otra alternativa consiste en realizar recomendaciones para cada usuario que forma el grupo, obteniéndose un ítem de recomendación por cada usuario y, posteriormente, escoger (según los factores sociales del grupo) el ítem que más se adecue al grupo [Quijano Sánchez, 2010]. Para ello se utilizan técnicas como las funciones de agregación. Estas funciones permiten escoger el ítem más adecuado centrándose en dos aspectos: satisfacción y miseria (descontento de un usuario con un ítem). En [Gartrell et al., 2010] se diseña un sistema de recomendación grupal para asistir al cine. Para ello en este sistema se realizan recomendaciones individuales basadas en

el perfil de cada usuario y posteriormente se escoge la función de agregación más adecuada para dicho grupo. En este estudio se realizó un experimento para determinar qué función de agregación resulta mas adecuada según el factor social del grupo. Para realizarlo distinguen 4 situaciones sociales: grupo de parejas, grupo de amigos, grupo de conocidos y grupo de recién conocidos.

Además de estas técnicas existe otro tipo de sistemas recomendadores que utilizan información adicional del usuario para mejorar la calidad de la recomendación, los sistemas de recomendación dependientes del contexto [[Woerndl and Schlichter, 2007](#)]. En ellos se utiliza información extra que permite modelar mejor al usuario, esta información puede ser de cualquier tipo y depende del ámbito del sistema, un atributo contextual puede ser por ejemplo la localización de un usuario, hora del día o información del entorno.

Otro aspecto diferenciador entre sistemas recomendadores consiste en la proactividad o reactividad de los mismos, un sistema proactivo realiza recomendaciones a un usuario sin la necesidad de que éste lo solicite o realice una búsqueda, mientras que un sistema reactivo responde ante la realización de una petición. El uso de un tipo de sistema u otro depende del dominio del problema en que se aplique. Existen muchos campos en los que se pueden aplicar técnicas de recomendación, uno de ellos es el del comercio electrónico, con el fin de facilitar al usuario la búsqueda de productos y por ende aumentar la cantidad de ventas. Este tipo de sistemas es utilizado actualmente por muchos comercios, como por ejemplo y el más conocido Amazon<sup>1</sup>.

Otro campo en el que se puede aplicar técnicas de recomendación es el e-learning. Trabajos como [[Verbert et al., 2012](#)] estudian el uso de sistemas recomendadores y la importancia del contexto del estudiante para mejorar la calidad del aprendizaje online mediante la propuesta de medios y contenidos; así como [[Ruiz-Iniesta et al., 2010](#)] en el que se aplican técnicas de recomendación para la enseñanza mediante un repositorio de objetos de aprendizaje que utiliza técnicas de recomendación para mostrar recursos y contenidos al estudiante en las

---

<sup>1</sup>[www.amazon.com](http://www.amazon.com)

materias de estudio.

### 2.1.1. Sistemas de recomendación aplicados al comercio electrónico

El comercio electrónico es un campo que ha crecido de manera extraordinaria en los últimos años y que actualmente es utilizado por la mayoría de los grandes comercios, gracias a las posibilidades que permite internet y las técnicas de marketing online. Con él los usuarios pueden consultar los productos que se encuentran en las tiendas y además comprar desde cualquier lugar. Incluso en algunos casos adquirir productos a menor precio debido al abaratamiento de los costes por no tener una tienda física que mantener.

Como se comentó anteriormente Amazon utiliza técnicas de recomendación en su catálogo online. Cuando un usuario consulta el catálogo online de productos de Amazon y selecciona un ítem se le muestra como propuesta ítems parecidos, que son aquellos que compraron clientes que además adquirieron el artículo seleccionado. Utilizando esta misma idea se aplicó a la recomendación por autor en la venta de libros, es decir, al usuario se le mostraban libros que otros usuarios habían comprado frecuentemente y que además habían comprado libros del autor del ítem seleccionado [Schafer et al., 1999].

Para la aplicación de sistemas de recomendación en el comercio electrónico se proponen diversas técnicas para categorizar los ítems como las taxonomías, en [Kim, 2013] se propone y evalúa la utilización de taxonomías para clasificar de una manera mas eficiente y correcta los ítems disponibles. Además de poder realizar la recomendación en función a esa clasificación. Otros trabajos como [Li et al., 2005] proponen soluciones híbridas para los procesos de recomendación de múltiple contenido y múltiples intereses, para ello proponen un algoritmo híbrido que combina el filtrado colaborativo basado en usuario y el basado en ítem.

Además de las recomendaciones individuales en el campo del comercio electrónico, las recomendaciones grupales también han sido estudiadas y aplicadas. Gracias a las redes sociales los portales de venta pueden obtener información sobre la red del usuario y realizar recomen-

daciones utilizando diversas técnicas. Trabajos como [Li et al., 2013] estudian y proponen mecanismos de recomendación en comercio electrónico incluyendo factores sociales.

Otro sistema de comercio electrónico que utiliza técnicas de recomendación es eBay<sup>2</sup>. El sistema de funcionamiento de este popular portal se basa en la confianza de los usuarios en los vendedores. A través del sistema de *feedback* del que dispone eBay permite a un comprador valorar al vendedor según ciertos parámetros como rapidez, comunicación o calidad del producto. Esto permite a otros usuarios ver las valoraciones realizadas antes de comprar el artículo y también recibir recomendaciones de ítems según las valoraciones de éstos [Schafer et al., 1999].

Además las nuevas tecnologías en el campo de los dispositivos móviles, como los Smartphone, permiten ampliar el campo del comercio electrónico no solo en ordenadores sino también en dispositivos móviles. Así mismo las técnicas de marketing utilizadas pueden aplicarse al comercio electrónico para dispositivos móviles, gracias a la conectividad de datos permiten al usuario estar conectados desde cualquier lugar, y junto con la gran dimensión de la pantalla y la navegación táctil, acceder a contenido web y realizar compras desde dichos dispositivos.

Una vez vistas las características de los sistemas de recomendación, así como los distintos tipos existentes, técnicas utilizadas para cada uno de ellos y los distintos campos en los que pueden ser aplicados nos centraremos en los sistemas recomendadores contextuales. A continuación se verá una introducción al contexto, así como a su categorización y utilización.

## 2.2. Contexto

Aunque no existe una definición de contexto unificada y aplicable a todos los ámbitos, podemos dar una definición de contexto en el ámbito lingüístico: *entorno lingüístico del cual depende el sentido y el valor de una palabra, frase o fragmento considerados* o una más abs-

---

<sup>2</sup><http://www.ebay.com/>

tracta entorno físico o de situación, ya sea político, histórico, cultural o de cualquier otra índole, en el cual se considera un hecho. Pero existen muchas definiciones de contexto en muchas áreas de estudio, ya sea psicología, historia, computación y demás áreas de investigación. En [Bazire and Brézillon, 2005] se hace un estudio sobre un corpus de definiciones de contexto de distintas áreas en estudiando la posibilidad de encontrar una definición unánime de contexto. La conclusión a la que se llega en este estudio es que no se puede dar una definición unánime de contexto aplicable a cualquier ámbito de estudio, siendo similares las definiciones de éste dentro de un mismo campo. Aunque no se logra el objetivo de dar una definición genérica de contexto que pueda utilizarse en cualquier campo, identifican ciertos factores determinantes a la hora de dar una definición de contexto: *entidad afectada por el contexto, su foco de atención, su actividad, su situación, su entorno y eventualmente, un observador*; siendo la definición de contexto ligada al área de conocimiento en el que se aplique.

La importancia del contexto radica en que aumenta el conocimiento del sistema sobre las características del usuario. Por lo general, a la hora de hacer una recomendación a un usuario ésta será mejor o más completa cuanto más información se tenga de él. Por ejemplo, un sistema recomendador de actividades turísticas puede tener en cuenta las preferencias que el usuario indique o sean inferidas a partir de otros datos, como gustos de cultura, deporte o restauración. Pero información extra como el clima que hay en la zona en la que se encuentra puede mejorar el proceso de recomendación, evitando por ejemplo en el caso de que el clima sea lluvioso recomendar actividades al aire libre. Otra información contextual que podría obtenerse para mejorar la recomendación es conocer con quién se encuentra, si son amigos, familiares o conocidos y poder realizar así una recomendación más adecuada. Toda esta información extra puede mejorar el proceso de recomendación en muchos sistemas. En este trabajo se ha centrado el foco de atención en el campo de los dispositivos móviles junto con las técnicas de marketing aplicadas a recomendadores por lo que lo que de aquí en adelante consideraremos contexto a toda aquella información que concierne a este ámbito, ese

decir, información del usuario obtenible con el dispositivo móvil y que posibilita la aplicación de técnicas de recomendación contextual. A continuación revisamos las distintas plataformas móviles para presentar las posibilidades respecto a la recomendación contextual.

Como se ha expuesto en esta sección el uso de información adicional sobre un usuario permite mejorar el proceso de recomendación. A continuación se presenta un estudio sobre las distintas plataformas móviles existentes en la actualidad y cómo las aplicaciones obtienen y/o utilizan el contexto en dichas plataformas.

### 2.3. Plataformas móviles

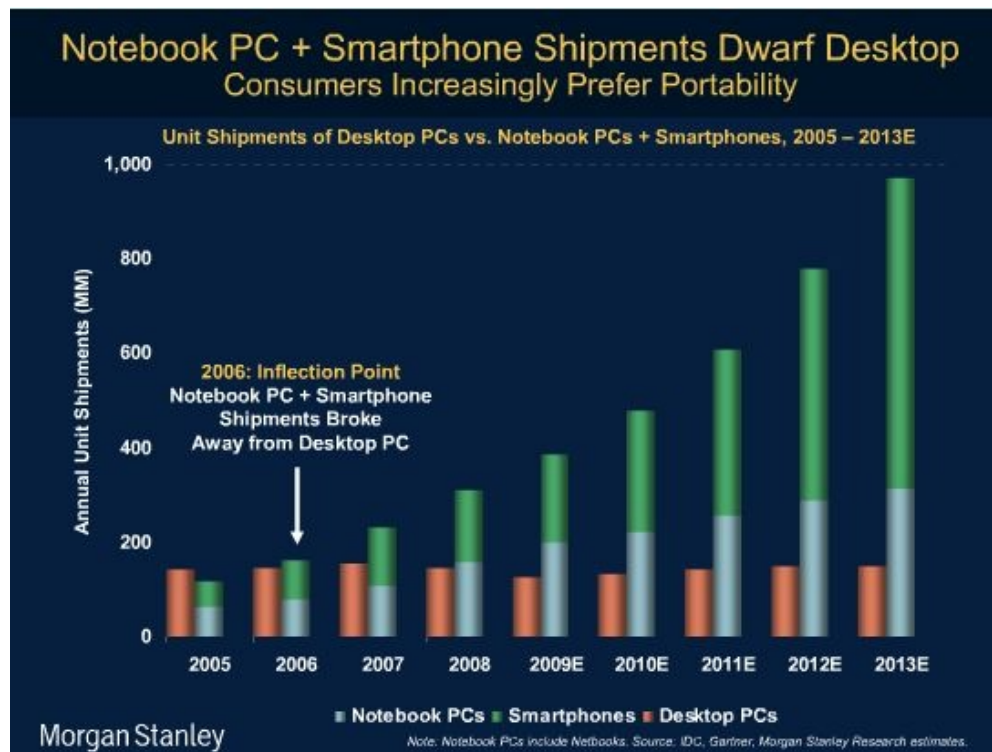


Figura 2.1: Estudio Morgan Stanley incremento ventas smartphones

En la actualidad las tecnologías móviles están cada vez más presentes en nuestras vidas, en especial en los últimos años en los que con la entrada de los teléfonos inteligentes o *smartphones* se ha producido un cambio radical en el sector. Su producción y venta han

incrementado de manera significativa en los últimos años como puede verse en la Figura 2.1. Un estudio<sup>3</sup> sobre la venta de ordenadores de escritorio, portátiles y smartphones confirma el gran aumento de producción de los smartphones en los últimos años en comparación al de ordenadores de escritorio y portátiles.

Este tipo de dispositivos ofrecen funcionalidades muy atractivas como el acceso a contenido web ya sea vía WiFi o a través de datos móviles y las populares apps, pequeñas aplicaciones para dispositivos móviles en las que encontramos una gran variedad de tipos, desde ocio y juegos hasta herramientas de productividad, negocios, comunicación o venta online.

En la actualidad el uso de los smartphones está muy extendido y la gran mayoría de los usuarios de móviles disponen de uno, las plataformas más utilizadas y que copan la mayor parte del mercado son Apple con su iPhone y su sistema operativo iOS y los dispositivos que utilizan el sistema operativo Android propiedad de Google y que se comercializa con una gran variedad de fabricantes en un rango de precios/prestaciones muy grande. Estas dos plataformas copan la mayor parte del mercado móvil en la actualidad, aún así existen otras alternativas en el mercado aunque con una cuota significativamente menor, entre ellas las más destacadas son Blackberry de RIM y Windows Phone de Microsoft.

Android es un sistema operativo destinado principalmente a dispositivos móviles táctiles, aunque en sus inicios también se incluyó como sistema alternativo en netbooks, ordenadores portátiles de bajo rendimiento y gran portabilidad (alrededor de 10" de tamaño). Android es un sistema operativo de código abierto basado en Linux y desarrollado por *Android, Inc.* bajo la financiación de Google que acabó adquiriéndolo en 2005, el primer dispositivo móvil en incluir este sistema apareció en 2008 de la mano de HTC. La arquitectura de este sistema operativo está basada en el kernel de Linux, el desarrollo de aplicaciones para esta plataforma se realiza en el lenguaje de programación Java, incluyendo esta plataforma una máquina virtual denominada *Dalvik*, similar a la máquina virtual de java y modificada para

---

<sup>3</sup>Consultora Morgan Stanley: <http://www.morganstanley.com>



Figura 2.2: Arquitectura sistema operativo Android

dispositivos móviles. Android proporciona un framework para el desarrollo de aplicaciones en su plataforma. En la Figura 2.2 puede verse la arquitectura descrita sobre esta plataforma, el desarrollo de aplicaciones se apoya en un framework que proporciona un conjunto de *managers* para la gestión de funcionalidades como las notificaciones, el teléfono, actividad de la aplicación o gestión de recursos entre otros. A su vez este framework se encuentra soportado por una API o conjunto de librerías como SQLite para la gestión de bases de datos en el mismo dispositivo, WebKit para la visualización en formato web u OpenGL para entornos gráficos, además de otras librerías. A su vez la capa de framework de aplicación también se apoya en un entorno de ejecución (Android Runtime) que permite a las aplicaciones interactuar con el entorno en el que se está ejecutando, en este módulo se encuentra la máquina virtual Dalvik mencionada anteriormente. El último nivel corresponde al kernel de Linux que como ya se comentó es la base del sistema operativo Android, este nivel ofrece soporte a la capa de librerías para el manejo del hardware del dispositivo como los controladores de teclado, cámara, energía, memoria y demás hardware disponible en el terminal.

La comercialización de dispositivos móviles con el sistema operativo Android no pertenece a una única marca si no que existe una gran cantidad de fabricantes que lo utilizan, entre los más populares encontramos Samsung, HTC y en el mercado de las tabletas Asus y Sony también han centrado su apuesta en este sistema; dada esta variedad de marcas en el mercado se pueden adquirir dispositivos móviles con el sistema operativo Android en un rango de precios muy amplio, desde dispositivos muy económicos hasta terminales de alta gama, este es sin duda uno de los factores que ha impulsado la popularidad de este sistema.

El desarrollo en Android se realiza en el lenguaje de programación Java, para ello se proporciona un conjunto de librerías que permite la implementación y compilación de aplicaciones, el Android SDK <sup>4</sup>, en él además se incluye ejemplos de proyectos con el código fuente, así como herramientas de desarrollo y un emulador para la plataforma. Para la implementación de aplicaciones en la plataforma Android se puede utilizar el Android Developer Tools (ADT), un plugin proporcionado para el entorno de desarrollo Eclipse con el que además se permite la depuración de la aplicación tanto en un emulador de terminal como en un dispositivo real a través del Android Debug Bridge (ADB).

Sin embargo el mercado de los sistemas operativos móviles no está dominado completamente por Android, la marca Apple goza también de gran popularidad en el mercado desde la aparición de los smartphones con su sistema operativo iOS. Este sistema operativo móvil se comercializa, al contrario que Android, únicamente por los dispositivos fabricados por la propia marca Apple, lo cual le da un aura de exclusividad que también lo hizo muy popular desde el lanzamiento del primer terminal en 2007, el iPhone. Otra diferencia significativa con Android es que la plataforma iOS no es de código abierto aunque el uso y manejo de estos dos sistemas es bastante similar por la navegación táctil y el uso de las pequeñas aplicaciones móviles denominadas Apps.

La arquitectura del sistema operativo iOS puede verse en la Figura 2.3 y está formada

---

<sup>4</sup><http://developer.android.com/sdk/index.html>

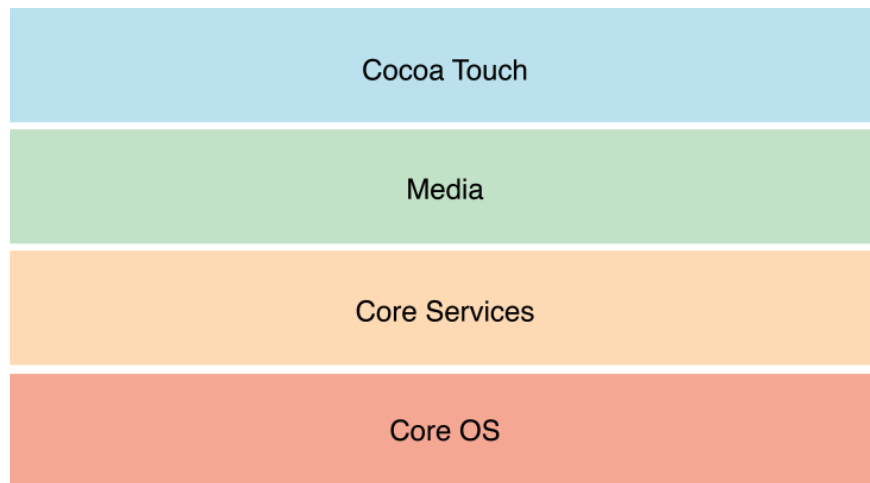


Figura 2.3: Arquitectura sistema operativo iOS <sup>5</sup>

por 4 capas organizadas de manera jerárquica. La primera de ellas corresponde a la capa *Cocoa Touch* en la que se encuentra los servicios y funciones de más alto nivel para el desarrollo de aplicaciones en la plataforma como la entrada táctil, multitarea, notificaciones y demás funciones de alto nivel. En el siguiente nivel se encuentra la capa *Media* o capa de medios, esta capa ofrece al desarrollador funciones de audio, imagen y vídeo para su aplicación. Como tecnologías gráficas incluye diversos frameworks para animación y 3D, así como de manejo de texto. Bajo esta última podemos encontrar la capa *Core Services*, la cual ofrece los servicios fundamentales que todas las aplicaciones usan. Dentro de estos servicios podemos encontrar soporte para la difusión de archivos, manejo de base de datos SQLite o XML. La última capa que forma la arquitectura de iOS corresponde a la *Core OS*, esta capa da soporte para las características de más bajo nivel proporcionando una serie de frameworks con los que interactuar con hardware o funciones del sistema.

El desarrollo de aplicaciones para estos dispositivos ha aumentado considerablemente dada su popularidad; desarrollándose aplicaciones móviles desde grandes empresas hasta cualquier persona con los conocimientos suficientes, desarrollándose tanto aplicaciones de pago como gratuitas. Para ello las plataformas móviles ofrecen herramientas y APIs mediante las cuales se puede desarrollar por completo una aplicación móvil, también existen

herramientas que permiten la creación de aplicaciones de manera sencilla, rápida y sin la necesidad de tener conocimientos de programación. Estas herramientas como *Shoutem*<sup>6</sup> ofrecen funcionalidades como noticias, medios e interacción con redes sociales entre otras características, además de poder utilizar plantillas ya definidas lo cual permite que cualquier persona pueda desarrollar su propia aplicación móvil sin tener conocimientos de programación, en cambio estas herramientas pueden resultar insuficientes para un desarrollo más avanzado en el que se necesiten características no incluidas en estas dado que al ser herramientas de construcción sin desarrollo de software por parte del usuario no puede ofrecer la misma versatilidad que el desarrollo puramente nativo en cada plataforma.

En el caso de la plataforma móvil iOS el lenguaje de programación requerido para el desarrollo de aplicaciones es Objective-C [Kochan, 2011]. Apple proporciona la herramienta X-Code<sup>7</sup>, la cual proporciona el SDK de iOS necesario para acceder a los frameworks y APIs que ofrece la plataforma. Además, dicha herramienta proporciona un simulador mediante el cual se pueden testear las aplicaciones desarrolladas. Sin embargo, para poder testear las aplicaciones en dispositivos reales será necesario adquirir una licencia de desarrollador. Dicha licencia también es necesaria para la publicación de aplicaciones en el App Store<sup>8</sup>.

En esta sección se ha estudiado la evolución del mercado móvil en los últimos años, así como el cambio radical producido por la aparición de los *smartphones* gracias a todas las posibilidades que ofrecen. También se ha podido ver como las plataformas móviles Android e iOS copan la mayor parte del mercado en estos dispositivos, así como las arquitecturas de éstas plataformas y las herramientas que ofrecen para el desarrollo de aplicaciones. En la siguiente sección se estudia cómo estas aplicaciones desarrolladas para las plataformas más populares hacen uso del contexto del usuario o pueden permitir su obtención.

---

<sup>6</sup><http://www.shoutem.com>

<sup>7</sup><https://developer.apple.com/technologies/tools/>

<sup>8</sup><http://www.apple.com/osx/apps/app-store.html>

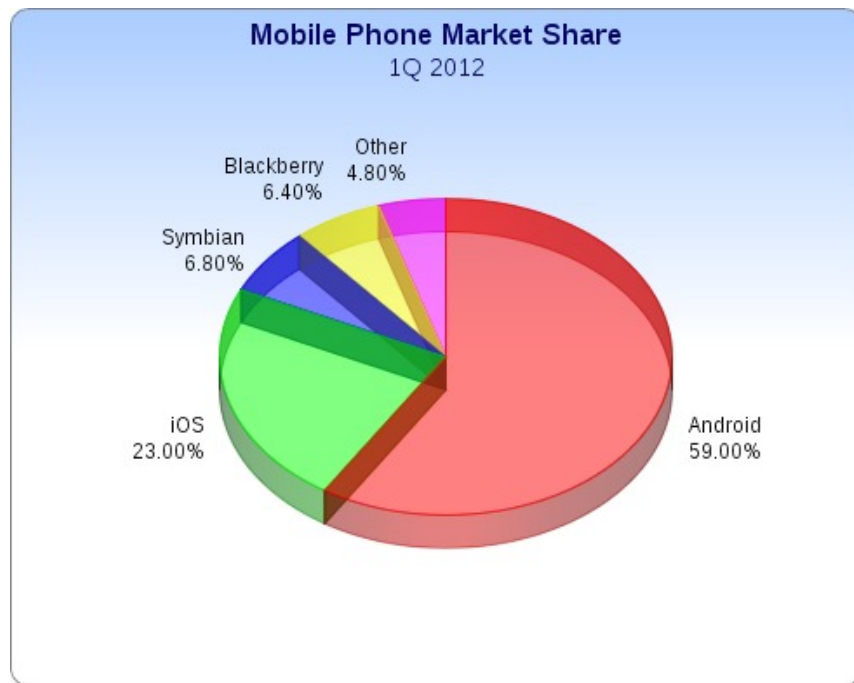


Figura 2.4: Mercado plataformas móviles

## 2.4. Aplicaciones contextuales para dispositivos móviles

Las nuevas tecnologías móviles nos permiten obtener una gran variedad de información necesaria para los sistemas recomendadores contextuales gracias a sus características como la conexión a internet, interfaz usuario-dispositivo sencilla, sensores, etc.

Existen muchas apps que obtienen o utilizan esta información contextual, desde mapas de geolocalización como Google Maps<sup>9</sup> hasta aplicaciones que permiten al usuario chatear con otros usuarios que se encuentren en un perímetro cercano. Esta información contextual puede distinguirse en varias categorías, según el tipo de información o su ámbito de uso. Para realizar este trabajo se han estudiado distintas aplicaciones que utilizan información contextual con el fin de estudiar qué información podemos obtener con las tecnologías móviles y cómo puede categorizarse. A continuación se muestran distintas aplicaciones móviles que se valen de información contextual en su funcionamiento. Dentro de las aplicaciones estudiadas

<sup>9</sup><http://www.google.com/mobile/maps/>

podemos encontrar 3 grupos diferenciados según la naturaleza de la información contextual que utilizan, en concreto podemos considerar las categorías de *geolocalización*, *comunicación* y *social*.

### 2.4.1. Geolocalización

Dentro de la categoría de geolocalización existen una gran variedad de aplicaciones que obtienen y utilizan de distintas maneras información sobre la localización física del usuario:

- Cell-Tracker <sup>10</sup>: Esta aplicación permite al usuario conocer su localización, en concreto se utiliza una técnica basada en las células de antenas GSM, sin la utilización del sensor GPS.
- Foursquare <sup>11</sup>: Foursquare es una aplicación que permite al usuario obtener las entradas en los lugares que visita, como museos, restaurantes, tiendas, etc. Con la posibilidad de poner opiniones y valoraciones de estos lugares y que amigos u otros usuarios puedan verlas. Además incluye la recomendación de lugares en base a gustos propios y personas con gustos parecidos. Ya que esta aplicación permite a los usuarios publicar los lugares que visita, pudiendo votarlos y comentarlos, puede ser útil para para obtener información sobre los gustos de lugares, comercios, ocio, etc. y utilizar esta información en recomendadores contextuales.
- GPS Tracking-Pro <sup>12</sup>: App que permite la localización amigos y familiares mediante GPS en tiempo real de una manera sencilla y sin la necesidad de realizar ajustes. La localización se determina utilizando la triangulación de satélites GPS y en el caso de que no se disponga de cobertura por estos se realiza mediante la localización por células móvil.

---

<sup>10</sup>[https://play.google.com/store/apps/details?id=com.local\\_cell\\_tracker&hl=en](https://play.google.com/store/apps/details?id=com.local_cell_tracker&hl=en)

<sup>11</sup><https://es.foursquare.com/download/>

<sup>12</sup><https://play.google.com/store/apps/details?id=com.fsp.android.c&hl=en>

- Latitude <sup>13</sup>: Similar al anterior Google Latitude es una app del framework de Google con la que el usuario puede compartir su localización con los usuarios que quiera, así como controlar el nivel de detalle con el que cada uno ve dicha localización, como puede ser ver la localización exacta o únicamente la ciudad en la que se encuentra. Así como obtener las entradas de lugares visitados y que sus amigos puedan verlo, al igual que en Foursquare.
- Family Tracker <sup>14</sup>: Muy similar a las anteriores ofrece la posibilidad de consultar la localización de amigos o familiares así como rastrear un terminal vía web o compartir la ubicación en redes sociales.

## 2.4.2. Comunicación

La siguiente categoría, comunicación, resulta muy importante ya que es uno de las características que han hecho tan populares a los smartphones. Con las aplicaciones de comunicación los usuarios pueden comunicarse entre ellos mediante mensajes escritos (incluso por audio en algunos casos) con sistemas basados en chats, pero con la versatilidad de poder hacerlo desde cualquier lugar con su dispositivo sin necesidad de utilizar un ordenador. A través de una conexión a internet, este tipo de aplicaciones han desbancado a los antiguos SMS. Algunas de estas aplicaciones utilizan información contextual como la localización del usuario para establecer chats con desconocidos que se encuentren cerca, otras en cambio simplemente utilizan la agenda telefónica del usuario para mostrar contactos con los que puedan chatear. Aunque no todas las aplicaciones de comunicación utilicen información contextual resulta conveniente su estudio ya que a partir de ellas puede obtenerse una gran cantidad de información como las relaciones sociales.

Entre las más destacadas se encuentran Whatsapp<sup>15</sup> y Google Talk<sup>16</sup> que recientemente

---

<sup>13</sup>[http://www.google.es/intl/en\\_my/mobile/latitude/](http://www.google.es/intl/en_my/mobile/latitude/)

<sup>14</sup><https://play.google.com/store/apps/details?id=com.fibercode.familytracker&hl=en>

<sup>15</sup><http://www.whatsapp.com/>

<sup>16</sup><http://www.google.com/hangouts/>

ha pasado a llamarse Hangouts, esta última perteneciente también al framework de Google. Aunque este tipo de aplicaciones suelen tener las mismas funcionalidades (conversación en chat por mensajes, conversaciones de grupo, envío de imágenes y audio..) el método de “enlace” entre usuarios no es igual en todas, en Whatsapp por ejemplo los usuarios están registrados con su número de teléfono y se utiliza la agenda de contactos existente en el móvil para conectarlos, Google Talk en cambio utiliza las cuentas de correo de Google para el registro y mostrar las personas con las que se quiere chatear. Además existen diferencias entre las distintas aplicaciones en cuanto a los dispositivos desde los que se puede utilizar, por ejemplo Whatsapp solo permite su uso en un único dispositivo impidiendo poder utilizar la misma cuenta en un ordenador o desde otro terminal, en cambio Google Talk al estar asociado a la cuenta de correo de Google puede utilizarse desde el navegador a través de Gmail o con una aplicación propia para ordenadores. Otra funcionalidad interesante de las apps de comunicación para este estudio es la posibilidad de crear grupos de chat, con ellos múltiples usuarios pueden mantener una conversación siendo este un componente social muy importante también.

Otra aplicación de este estilo muy popular es *LINE*<sup>17</sup>, aplicación de origen japonés, por lo que está más extendido su uso en el país oriental; esta aplicación sigue el mismo funcionamiento de registro y comunicación que Whatsapp, pero con la novedad de poder utilizar la misma cuenta tanto en el móvil como en ordenadores y la posibilidad de realizar llamadas a través del protocolo VoIP.

En cuanto a las aplicaciones de comunicación que utilizan información contextual como parte importante de su funcionalidad encontramos SpotBros<sup>18</sup> y Circle<sup>19</sup>, ambas permiten abrir chats o grupos con personas desconocidas que se encuentre en una localización cercana a la del usuario, sin la necesidad de “agregar” a otros usuarios.

Otro medio de comunicación muy extendido y popular desde la aparición de la web 2.0 son

---

<sup>17</sup><http://line.naver.jp/en/>

<sup>18</sup><http://www.spotbros.com/>

<sup>19</sup><http://discovercircle.com/>

los foros, sitios web en los que los usuarios pueden comunicarse y hablar sobre cualquier tipo de tema con el sistema de hilos y posts. La tecnología móvil no se ha quedado atrás en este campo y se han desarrollado aplicaciones para poder comunicarse en foros con un dispositivo móvil desde cualquier lado, entre ellas una de las más populares es *Tapatalk*<sup>20</sup>.

### 2.4.3. Social

Como ya se dijo anteriormente algunas aplicaciones de comunicación tienen unos fuertes componentes de relación social y a partir de estas también se podría obtener información contextual, pero además de eso existen también aplicaciones exclusivamente destinadas a las redes sociales más populares y que serán útiles en el desarrollo de este trabajo.

Dentro de las aplicaciones sociales destacamos aquellas que permiten la conexión desde el dispositivo móvil a las redes sociales más populares en la actualidad, estas aplicaciones permiten un uso casi idéntico de estas redes al que se realiza mediante un ordenador, permitiendo al usuario por ejemplo difundir contenido o comunicarse con amigos o contactos enlazados en su cuenta. Facebook dispone de su propia aplicación para terminales<sup>21</sup> desde la cual se pueden subir fotos y visualizar existentes, escribir comentarios o agregar contactos entre otras funcionalidades. Otra red social muy popular en la actualidad es Twitter, la cual también dispone de aplicación para terminales ofreciendo la misma funcionalidad que en su uso mediante navegador web. Ambas aplicaciones ofrecen herramientas a los desarrolladores para interactuar en la cuenta del usuario y obtener información a través de ella, lo cual resultará importante en este estudio.

Se han mencionado las aplicaciones sociales de las redes que se consideran más populares en la actualidad, pero otras como LinkedIn<sup>22</sup>, Instagram<sup>23</sup> o Google+<sup>24</sup> también disponen de aplicación móvil y pueden ser conveniente su estudio para la obtención de información

---

<sup>20</sup><http://www.tapatalk.com/>

<sup>21</sup><https://www.facebook.com/mobile/>

<sup>22</sup><http://www.linkedin.com/mobile>

<sup>23</sup><http://instagram.com/>

<sup>24</sup><http://www.google.com/mobile/+/>

contextual en el ámbito social.

Como hemos visto los diversos tipos de aplicaciones móviles estudiados tienen características de interacción social, aún siendo de categorías distintas de la social los dos primeros. En el Capítulo 1 se presentó la motivación de este estudio que era el cada vez mayor uso de los dispositivos móviles dadas sus características y con la ayuda de las redes sociales su uso y difusión de información es cada vez mayor lo cual puede verse en cómo aplicaciones de geolocalización o comunicación utilizan estas relaciones sociales o contribuyen a su creación.

A lo largo de esta sección se ha podido ver una serie de aplicaciones móviles de distintos tipos que obtienen y/o utilizan información contextual en diversas plataformas móviles. A continuación se exponen una serie de herramientas y técnicas para el desarrollo de aplicaciones contextuales en dispositivos móviles, así como una arquitectura propuesta para el desarrollo de aplicaciones multiplataforma.

## 2.5. Herramientas para el desarrollo de aplicaciones contextuales en dispositivos móviles

Existen distintas librerías que facilitan el desarrollo de aplicaciones de recomendación contextual en dispositivos móviles. Podemos encontrar *ContextPhone* [Raento et al., 2005], una plataforma para el desarrollo de aplicaciones contextuales en dispositivos móviles. Este trabajo se desarrolló en 2005 y está enfocado a teléfonos de la plataforma Symbian y los dispositivos Nokia Series60, lo cual supone una limitación para su uso. La motivación de ContextPhone es similar a la de este trabajo: investigar y desarrollar una librería que permita el futuro desarrollo de aplicaciones contextuales de manera más sencilla y con más posibilidades explotando todas las características de los dispositivos móviles. En concreto en [Raento et al., 2005] se desarrollan distintos módulos para controlar el dispositivo y obtener información contextual, podemos encontrar cuatro módulos:

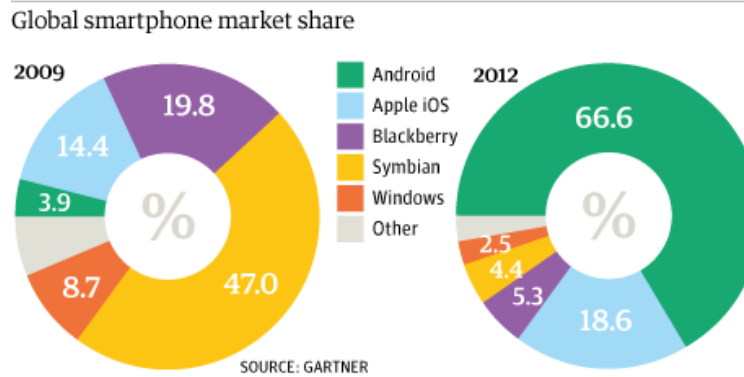


Figura 2.5: Evolución mercado móvil

- *Sensors*: El módulo de sensores obtiene información como la localización por GPS o identificación por células, así como los registros de llamadas, mensajes SMS/MMS, nivel de batería o actividad del usuario. Este módulo es quizás el mas enfocado a la búsqueda de información contextual.
- *Communications*: Módulo que permite la comunicación exterior a través de distintos medios, como GPRS, SMS, MMS o Bluetooth. Notese que este módulo no se centra en la obtención de información contextual, pero para el desarrollo de aplicaciones contextuales ofrece la posibilidad de conexión a internet a través de GPRS o de comunicación a través de medios como mensajes cortos, multimedia o bluetooth.
- *Customizable applications*: El módulo de personalización de aplicaciones permite sustituir o aumentar las características de las aplicaciones integradas en el dispositivo.
- *System services*: Módulo de servicios del sistema que permite a las aplicaciones acceder a la recuperación de errores y el log, además de poder lanzar servicios en segundo plano.

En resumen, ContextPhone permite la obtención de información contextual desde un teléfono móvil para el desarrollo de aplicaciones contextuales. Dentro de esta información contextual que se puede obtener podemos encontrar la geolocalización por GPS o Cell ID, los contactos del usuario, nivel de batería, actividad del usuario en el dispositivo y acceso a contenidos como mensajes cortos y multimedia así como al registro de llamadas. Esta

plataforma permite el desarrollo de aplicaciones contextuales pudiendo utilizar estas características de manera más sencilla y rápida sin la necesidad de una implementación nativa en el dispositivo. El principal inconveniente es que su uso está restringido a dispositivos de la plataforma Symbian y Nokia Series60, lo cual en la fecha de desarrollo del trabajo pudo ser una decisión acertada ya que por aquel entonces Symbian y Nokia copaban el mercado de dispositivos móviles. Sin embargo en la actualidad y en pocos años (debido a la aparición de los smartphones) este mercado ha sido dominado por las plataformas Android e iOS. Podemos ver en la Figura 2.5 como en 2009 el 47% de dispositivos eran de la plataforma Symbian, pero la actualidad es bien distinta y la mayoría de aplicaciones se desarrollan para las plataformas más populares (Android e iOS actualmente).

Debido a la dualidad Android e iOS en el monopolio de los smartphones han surgido iniciativas para “estandarizar” el desarrollo de aplicaciones. *PhoneGap* [Wargo, 2012], es una plataforma de desarrollo de aplicaciones móviles mediante tecnologías web (HTML [Graham, 1995], CSS [Meyer, 2009] y Javascript [Goodman et al., 2007]) que permite desarrollar aplicaciones en este formato alejándose del desarrollo nativo para cada dispositivo. De esta forma se posibilita desarrollar una misma aplicación para distintas plataformas sin la necesidad de hacer grandes cambios al cambiar de una a otra. *PhoneGap* se creó en 2008 por *Nitobi Software* con la intención de profundizar en el uso de lenguajes web (Javascript) para el desarrollo móvil. Su propuesta se fundamentó en dos principales creencias, la primera es que el formato web puede dar solución a la portabilidad entre plataformas, la segunda, que toda tecnología acaba volviéndose obsoleta. La primera creencia se basa en que los lenguajes web HTML, CSS y Javascript, aún con sus limitaciones y debilidades han conseguido llegar a una masa crítica, gracias a que *cualquier persona puede publicar cualquier cosa desde cualquier lugar*. La segunda creencia, la obsolescencia de la tecnología, es algo que no necesita demostración, es algo observable a lo largo de la historia, aunque haya tecnologías que perduran décadas lo que hoy es tendencia mañana puede caer en el olvido.

El ejemplo más claro para el campo de los dispositivos móviles se comentó anteriormente, hace no muchos años Nokia era líder con diferencia en el sector móvil desde la aparición de los primeros dispositivos destinados al público general. Posteriormente la plataforma móvil Symbian tuvo gran éxito, pero con la aparición de los smartphones no pudo competir con la popularidad que obtuvieron Android e iOS cambiando drásticamente la situación del mercado móvil en favor de estos últimos en poco tiempo. Por ello aunque hoy en día parece clara la hegemonía de estas dos plataformas nada garantiza que esto se mantenga así en el futuro más inmediato. Por ejemplo Microsoft está intentando aumentar cuota de mercado con Windows 8. De esta manera resulta conveniente no cerrar los desarrollos de software para móviles a determinadas plataformas. Ésto es lo que pretende PhoneGap con su propuesta de utilizar el formato web para el desarrollo de aplicaciones. Para ello ofrecen un framework gratuito y de código libre que permite hacer de puente entre las aplicaciones web y los dispositivos móviles. PhoneGap contribuye en la *Apache Software Foundation* (AFS) bajo el nombre de *Apache Cordova*. En la Sección 4 se verá más detalladamente el funcionamiento de esta tecnología y su importancia para el desarrollo de este trabajo.

### 2.5.1. Conclusiones

En este capítulo se ha dado una introducción a los sistemas de recomendación y las distintas técnicas que existen en la actualidad para realizar recomendaciones, así como diversos campos en los que se ha estudiado o realizado su aplicación. Dentro de los sistemas de recomendación encontramos los sistemas dependientes del contexto, los cuales pueden mejorar el proceso de recomendación con la aportación de información adicional sobre el usuario. Ya que no existe una definición unánime de contexto, en este trabajo se ha definido como aquella información sobre el usuario que puede obtenerse desde su dispositivo móvil para ser utilizada en un sistema recomendador contextual. Para identificar qué información contextual se puede obtener a través de los dispositivos móviles se han estudiado las distintas plataformas móviles existentes en la actualidad así como la tendencia que ha

sufrido el mercado en este campo, observándose un cambio drástico desde la aparición de los smartphones. Anteriormente a la aparición de estos el mercado móvil estaba copado por Nokia y la plataforma Symbian, pero tras esta revolución tecnológica nos encontramos con una dualidad en la que las plataformas iOS y Android abarcan la mayoría de este mercado. Esto lleva a la necesidad de una estandarización en el desarrollo de aplicaciones móviles que puede lograrse utilizando tecnologías web. Además una vez estudiadas las plataformas móviles más utilizadas en la actualidad se han identificado aplicaciones que obtienen o utilizan información contextual en su funcionamiento, esta información contextual que ya se utiliza resulta útil para el desarrollo de este trabajo cuyo principal objetivo es el desarrollo de una librería que permita la obtención de información contextual en sistemas de recomendación a través de los dispositivos móviles.

Se han encontrado librerías similares, en concreto se ha estudiado ContextPhone, la cual se desarrolló para la plataforma Symbian y ofrece una gran cantidad de información para el desarrollo de aplicaciones en dicha plataforma. Sin embargo como hemos visto el mercado de los dispositivos móviles puede cambiar rápidamente con la aparición de nuevas tecnologías o plataformas novedosas. Por ello en este trabajo se ha estudiado utilizar los estándares de tecnología web para el desarrollo en dispositivos móviles con el fin de que puedan ser reutilizados en un futuro sin la necesidad de desarrollar completamente una nueva librería para otras plataformas. En el siguiente capítulo explica la contribución realizada en este trabajo.



## Capítulo 3

# Categorización del conocimiento contextual aplicable a aplicaciones de recomendación en dispositivos móviles

En el Capítulo 2 se destacó la ventaja que aporta utilizar información contextual en los sistemas de recomendación. Añadir información extra a los sistemas de recomendación no tiene por qué consistir únicamente en añadir toda esa información al sistema sin más. Para mejorar el proceso de recomendación resulta conveniente estudiar las cualidades y propiedades de todos estos atributos contextuales para poder utilizarse de una manera más correcta. Por ello en este trabajo se presenta un modelo teórico del contexto móvil propuesto. En este modelo teórico se ha categorizado el contexto móvil de manera jerárquica en: *dimensiones* divididas en *categorías* que a su vez se componen de *atributos*.

A partir del estudio realizado sobre las distintas tecnologías móviles existentes en la actualidad y a los atributos contextuales identificados en la revisión del estado del arte [Esteban, 2013],[Verbert et al., 2012],[Raento et al., 2005], el primer paso de este trabajo ha consistido en categorizar la información contextual en distintas dimensiones para su explotación en aplicaciones de recomendación contextual. Utilizaremos el término “contexto móvil” para referirnos a la información contextual accesible a través de dispositivos móviles. En concreto se han identificado 3 dimensiones en las que será necesario clasificar la información contextual. Estas 3 dimensiones se corresponden con la categorización del contexto según

su *tipología*; el *carácter temporal* o dinámico de la información a obtener; y la *interacción con el usuario* necesaria para su obtención.

La primera dimensión de la información que se identifica es la de clasificación tipológica, algo básico y común a la hora de diseñar un sistema recomendador contextual. Consiste en dividir los atributos contextuales según su “tipo”, agrupando los que contienen características o cualidades similares.

La segunda dimensión en la que se ha clasificado el contexto consiste en el carácter temporal o dinámico de los atributos contextuales. Esta clasificación resulta conveniente dado el objetivo de obtener esta información a través de dispositivos móviles y utilizarla en sistemas de recomendación. Ya que para la realización de recomendaciones contextuales será necesario consultar dicha información, siendo el carácter estático o dinámico de cada atributo un factor a tener en cuenta para obtener y actualizar la información contextual de una manera correcta.

En último lugar se ha identificado una tercera dimensión que resulta de gran relevancia en este trabajo y es el método de obtención de los atributos contextuales. Esta dimensión distingue entre la interacción implícita o explícita para obtener el contexto móvil. De cara al usuario una de los principales motivaciones de este estudio es poder obtener de manera automática la mayor cantidad de información posible a través de los dispositivos móviles evitando así problemas como saturar al usuario con preguntas o la posibilidad de que el usuario no introduzca la información de manera correcta. La obtención de la información contextual de manera automática mejoraría el proceso de recomendación, por ello resulta de gran importancia estudiar bien qué atributos pueden obtenerse de manera automática, así como detectar aquellos que deben ser estrictamente introducidos o indicados por un usuario. En sistemas como [Verbert et al., 2012], para determinar qué material académico se le propone al alumno en su sistema e-learning toman ésta última categoría como un factor clave.

A continuación describiremos cada uno de los 3 ejes en los que se basa nuestro modelo de

categorización del contexto móvil.

### 3.1. Dimensión 1: Tipología del contexto móvil

Existen distintas categorizaciones tipológicas del contexto. Por ejemplo en el trabajo [Adomavicius and Tuzhilin, 2011] identifican las categorías tipológicas *social*, *físico*, *personal* y *semántico*.

En [Verbert et al., 2012] para el diseño del sistema de recomendación de recursos en e-learning identifican una mayor cantidad de tipologías contextuales, en concreto: *Informático*, *Localización*, *Tiempo*, *Actividad*, *Condiciones físicas*, *Recursos*, *Usuario* y *Relación social*.

Esta categorización tipológica del contexto es aparentemente muy específica, siendo la mayoría de las categorías simples atributos contextuales. En este trabajo se propone una categorización tipológica del contexto mediante la agrupación de atributos con características similares. Esta categorización se ha realizado basándonos en los trabajos previos estudiados como [Verbert et al., 2012], [Adomavicius and Tuzhilin, 2011] y el estudio realizado en el Capítulo 2 sobre las tecnologías móviles y su relación con el contexto.

Así, a partir de la revisión del estado del arte distinguiremos la información de esta dimensión del contexto móvil en las siguientes tipologías: **contexto social**, **geotemporal**, **tecnológico** y **personal**.

A continuación veremos qué atributos contextuales estarían incluidos en cada categoría y a qué información concreta corresponde cada uno de ellos.

#### 3.1.1. Contexto social

En cuanto al contexto social entendemos toda aquella información que trate sobre las relaciones de un individuo con otros, como pueden ser amistades, familiares, etc. Dentro de esta categoría encontramos atributos como si el usuario se encuentra sólo o en grupo (2 personas o más). Además también podría ser necesario obtener información sobre los amigos o familiares que le acompañan para aplicar técnicas de recomendación contextuales,

esta información puede ir desde la afinidad del usuario con ellos hasta información sobre gustos o perfil de éstos. Otro atributo importante del contexto social es la identificación de líderes de opinión, sobre todo para la aplicación de técnicas de marketing, como por ejemplo la técnica de marketing viral [[Leskovec et al., 2007](#)].

También incluimos en esta categoría información relativa a las actividades que le gusta realizar al usuario en su tiempo libre ya sea en grupo o individualmente, así como la situación laboral en la que se encuentra, para modelar mejor el perfil de usuario en factores como precio de productos, y su estado familiar (soltero, casado, con hijos, etc.) que podría ser utilizado a la hora de recomendar items no solamente indicados para él sino además también para sus familiares. En resumen en el contexto social podemos encontrar los siguientes atributos:

- S1 Grupo/Individual: Situación del individuo en un instante concreto, si esta sólo o acompañado.
- S2 Amigos: Información relativa a los amigos, necesaria para la aplicación de técnicas de recomendación grupales.
- S3 Afinidad: Relación del usuario con amigos (conocidos, amigos, pareja, etc), dentro de las técnicas de recomendación grupales existen trabajos como [[Gartrell et al., 2010](#)] que demuestran la diferencia de resultados según la afinidad entre las personas que forman el grupo al que se le realiza la recomendación, obteniéndose mejor feedback escogiendo técnicas distintas según la afinidad del grupo [[Gartrell et al., 2010](#)] .
- S4 Liderazgo: Determina el líder de un grupo, usuario con mayor influencia sobre todos los individuos que lo forman.
- S5 Aficiones: Actividades de ocio que realiza el individuo en su tiempo libre, puede resultar útil para la recomendación de ítems que concuerden con esas actividades (recomendadores basados en contenido).
- S6 Situación laboral: Estudiante, empleo, desempleado, etc.

S7 Estado familiar: Soltero, casado, con hijos, etc.

### 3.1.2. Contexto geotemporal

La siguiente categoría identificada corresponde al contexto geotemporal, en el cual hemos considerado información relativa a localizaciones, clima y tiempo. Dentro de la localización incluimos dos atributos correspondientes a la localización del usuario, es decir, dónde se encuentra geográficamente localizado (en coordenadas latitud y longitud), para determinar si se encuentra en un centro comercial, en su casa o en otros lugares o si se encuentra próximo a alguna tienda; así como su localización dada por país, ciudad, dirección, etc.

Respecto al clima será necesario obtener información relativa al estado climático en la localización del usuario, información sobre si llueve, hace calor, frío u otros será de ayuda a la hora de realizar una recomendación como se explicó en la Sección 2.1. Por último la característica temporal consiste en obtener información sobre el momento del día en el que se encuentra. En resumen, para el contexto de tipo geotemporal tenemos:

G1 Localización: Localización física dada por latitud-longitud o país, ciudad, dirección, etc.

G2 Clima: Estado climático en la localización del usuario, este atributo es dependiente del de localización.

G3 Momento del día.

### 3.1.3. Contexto tecnológico

El contexto tecnológico está formado por la tecnología móvil disponible por el usuario. Entre ellas destacamos la plataforma móvil y versión utilizada, la disponibilidad de conexión a internet, en la que distinguiremos las conexiones mediante datos móviles y mediante conexión WiFi a un punto de acceso. La información sobre la disponibilidad de conexión GPS también es relevante en este caso de estudio, así como tecnologías de comunicación como la disponibilidad de conexión por Bluetooth punto a punto o mediante WiFi (Wifi P2P o

Wifi Ad-hoc). El último atributo del contexto tecnológico consiste en la información sobre la agenda del dispositivo, que podría englobarse también en las de comunicación aunque a otro nivel.

T1 Dispositivo: Información sobre plataforma y versión del dispositivo.

T2 Datos móviles: Disponibilidad de conexión a internet mediante datos móviles.

T3 Conexión a internet WiFi: Información sobre conexión a puntos de acceso Wifi.

T4 GPS. Disponibilidad y estado del sensor GPS.

T5 Bluetooth: Información de dispositivos enlazados o disponibles por Bluetooth.

T6 WiFi ad-hoc: Redes y grupos punto a punto.

T7 Agenda: Información de la agenda telefónica en el dispositivo como teléfonos, emails, favoritos, etc.

### **3.1.4. Contexto personal**

La última categoría que destacamos es la personal, la cual engloba aspectos que definen al usuario. Como contexto personal distinguimos atributos como la edad y el sexo, considerados atributos importantes a la hora de realizar recomendaciones. Otros atributos que hemos destacado dentro del contexto personal para el uso del marketing en recomendadores son el historial de compra (información sobre las compras realizadas), necesidades del usuario, por ejemplo si necesita ropa o artículos de sport o para el trabajo; información del calendario relativa a eventos que tenga el usuario como pueden ser bodas, fiestas o cumpleaños y eventos de tipo general como nochevieja, navidades, verano, etc. Por último otro atributo personal necesario será el estado de humor que consideramos importante a la hora de realizar recomendaciones.

P1 Edad

P2 Historial de compra

P3 Necesidades

P4 Calendario

P5 Humor

## 3.2. Dimensión 2: Dinamismo

Además de la categorización tipológica del contexto también debe considerarse el carácter dinámico de esta información. Así observamos que cierta información puede ser más dinámica que otra, es decir, la relativa frecuencia con la que puede cambiar en el tiempo la información obtenida. Encontraremos desde atributos que se mantienen constantes en un cierto periodo de tiempo hasta otros que toman valores en un instante de tiempo concreto y es necesario consultar su estado con mucha mayor frecuencia.

Por ello la segunda dimensión en la que se ha categorizado el contexto consiste en el dinamismo de la información contextual, es decir, estudiar la constancia en el tiempo de los atributos. En el desarrollo de este trabajo se ha considerado importante esta dimensión dado el objetivo final de este estudio, que es la utilización de las tecnologías móviles para obtener información contextual. Puesto que los atributos contextuales definidos anteriormente pueden ser más o menos constantes en el tiempo se debe tener este factor en cuenta a la hora de utilizar estas tecnologías en sistemas recomendadores.

Existe una gran diferencia en el dinamismo de datos como pueden ser la información del dispositivo utilizado, relativamente estático, frente a la geolocalización o compañía del usuario, que puede variar en periodos de tiempo no controlables. Por ello los atributos contextuales más estáticos requerirán una menor frecuencia de actualización u obtención, frente a aquellos más dinámicos que será necesario obtenerlos cuando se requiera una recomendación en sistemas reactivos o actualizarlos automáticamente cada cierto periodo para sistemas proactivos.

	Social
Alto	Grupo/Individual Afinidad Liderazgo Amigos
Medio	
Bajo	Aficiones Situación laboral Estado familiar

Cuadro 3.1: Dinamismo atributos tipología social

El anterior ejemplo corresponde a atributos que se pueden obtener automáticamente sin necesidad de una introducción manual de parte del usuario, pero no ocurre lo mismo con datos que se necesitan obtener de manera explícita. En esta situación es conveniente tener en cuenta el dinamismo, ya que estos datos requieren realizar una petición de información al usuario. Un ejemplo claro puede ser el estado de humor, que no puede obtenerse automáticamente. La discusión sobre la interacción con el usuario se presentará a continuación en la Sección 3.3. Para estudiar el dinamismo del contexto móvil no podemos dar una escala general ya que cada variable utiliza escala distintas y para gran parte de los atributos contextuales definidos resulta extremadamente complejo conocer el dinamismo exacto. Por ello se realiza una clasificación del dinamismo de estos atributos en 3 categorías: dinamismo bajo, medio y alto; definiendo los atributos de nivel dinámico bajo aquellos que pueden variar una o ninguna vez a lo largo del día, medio para aquellos que pueden hacerlo entre 2-4 veces y alto para los de mayor variabilidad.

A continuación clasificaremos los atributos de la primera dimensión, tipología, de acuerdo a las categorías de esta segunda dimensión dentro de nuestro marco de trabajo.

**Tipología social** En la categoría de contexto social encontramos los atributos grupo/individual, amigos, afinidad, liderazgo, aficiones, situación laboral y estado familiar. El primero de ellos, grupo/individual se ha considerado un atributo de nivel dinámico alto ya que a lo largo del día un usuario puede estar sólo o en compañía numerosas veces. Además, establecer una

	Geotemporal
Alto	Clima
Medio	Localización Momento del día
Bajo	

Cuadro 3.2: Dinamismo atributos tipología geotemporal

relación entre el dinamismo de estos atributos no resulta sencillo dada la dependencia que existe entre unos y otros. Por ejemplo el atributo afinidad (que recordemos corresponde a la relación de un usuario con los demás que le acompañan en una determinada situación) lo hace dependiente del atributo grupo/individual. Podemos decir entonces que el dinamismo de el primero es igual o mayor que el del segundo (nivel alto).

La información relativa a los amigos puede tener dos interpretaciones distintas. La primera es considerarlo un atributo estático siendo esta información de amigos por ejemplo en redes sociales. Pero si lo consideramos como la información sobre amigos con los que se encuentra en un determinado momento está también ligado al atributo grupo/individual (al igual que el de afinidad) por lo que se debería considerar igual de dinámico que éste.

Otra relación existente entre varios atributos es la del liderazgo, dado que la identificación de un líder en un grupo depende de los individuos que lo formen. Podemos considerar este atributo dependiente del atributo grupo/individual, siendo su dinamismo de igual valor al del atributo anterior. Para los tres atributos restantes (aficiones, situación laboral y estado familiar) no se observa ninguna dependencia significativa con los demás, además se pueden considerar atributos relativamente poco dinámicos y muy similares en este aspecto.

**Tipología geotemporal** El contexto geotemporal está compuesto por los atributos localización, clima y momento del día, podemos ver que existe una relación clara entre los dos primeros atributos. Estando así condicionado el clima por la localización del usuario, sin embargo dicho atributo no es totalmente dependiente de la localización dado que en una misma localización el clima puede variar sin que ésta cambie, por ello establecemos la

	Tecnológico
Alto	Datos móviles Internet@WiFi
Medio	
Bajo	Dispositivo GPS Bluetooth WiFi-P2P Agenda

Cuadro 3.3: Dinamismo atributos tipología tecnológica

relación del dinamismo del clima estrictamente mayor que el de la localización, quedando el clima encuadrado en el nivel de dinamismo alto. En cuanto al dinamismo propio de la localización si consideramos dicha propiedad como las coordenadas exactas en las que se encuentra un usuario es evidente que es completamente dinámico, pero si queremos referirnos a ello como una localización determinada (en casa, en el trabajo, en un centro comercial, en una ciudad concreta, etc.) podemos decir que es un atributo de dinamismo medio. El atributo momento del día sigue el mismo comportamiento, si se toma como un valor numérico continuo se puede considerar completamente dinámico, pero si lo tomamos como valores discretos como mañana, tarde y noche no se puede considerar completamente dinámico, por tanto en cuanto a la relación de dinamismo con los demás atributos consideramos su dinamismo equivalente al del atributo de localización (nivel medio).

**Tipología tecnológica** Como vimos anteriormente el contexto tecnológico está formado por los atributos de información del dispositivo móvil, disponibilidad de conexión de datos móviles, conexión a internet WiFi, sensor GPS, Bluetooth, Wifi Ad-hoc (WifiP2P) e información de la agenda. Puede verse que el primer atributo (dispositivo) es de nivel dinámico bajo ya que el cambio de plataforma móvil o versión ocurre al cambiar de dispositivo o mediante actualizaciones de software, siendo estos casos puntuales. Además, la única relación observable entre estos atributos puede ser la de datos móviles y WiFi, siendo inusual en dispositivos móviles estar conectado a internet mediante las dos tecnologías a la vez, por

	Personal
Alto	
Medio	Humor Necesidades Historial de compra
Bajo	Edad Calendario

Cuadro 3.4: Dinamismo atributos tipología personal

ello dada la alternancia entre estas conexiones a lo largo del día en el dispositivo móvil de un usuario le asignamos un valor de dinamismo alto. Dado que los únicos atributos referentes a las conexiones que pueden variar en un mismo terminal son estos dos, establecemos que su dinamismo es mayor que el de los demás. La disponibilidad de conexión por Bluetooth, GPS o WifiP2P está determinada por las características del dispositivo, por lo cual se consideran atributos con un dinamismo bajo. Si, en cambio, se enfocasen dichos atributos como conexión activada o desactivada en el dispositivo podrían considerarse más dinámicos. La información sobre la agenda no guarda relación con los demás atributos y por tanto su dinamismo se debe estudiar en función del periodo de actualización de este registro. Dado que no es muy usual para un usuario añadir numerosos contactos a su agenda en periodos cortos de tiempo consideraremos este atributo poco dinámico (nivel bajo). La relación de atributos y dinamismo de la tipología contextual tecnológica puede verse en la Tabla 3.2.

**Tipología personal** En la categoría del contexto personal tenemos los atributos edad, historial de compra, necesidades, calendario y humor. En esta categoría no se han encontrado dependencias significativas por lo que estudiaremos el dinamismo de estos atributos de manera independiente. En primer lugar tenemos el atributo edad. Al igual que el momento del día este tiene varias posibles interpretaciones. En la definición exacta de este atributo se puede considerar dinámico, pero en el ámbito de aplicación lo consideramos un atributo estático en relación a los demás, por tanto le asignamos el valor de dinamismo bajo. Al igual que en el anterior el atributo calendario puede variar a lo largo del tiempo teniendo

propiedades dinámicas, pero no con una gran frecuencia. Por ello le asignamos también el valor de dinamismo bajo. El siguiente atributo, historial de compra, depende mucho del perfil de usuario. Puede haber individuos que realicen compras muy a menudo o a diario y en cambio otros que lo hagan cada mucho tiempo. Ocurre algo similar con los atributos necesidades y humor, atributos que pueden considerarse muy aleatorios, por lo tanto a estos tres últimos les asignamos un valor de dinamismo medio, mayor que el de los anteriores atributos.

### **3.3. Dimensión 3: Interacción con el usuario**

La tercera dimensión en la que se ha estudiado el contexto en este trabajo corresponde a la forma en la que se obtiene la información contextual. A la hora de implementar un sistema recomendador contextual utilizando tecnologías móviles resulta importante tener en cuenta este aspecto, ya que no toda la información puede ser introducida por el usuario, como por ejemplo, la geolocalización por coordenadas (latitud, longitud) y otras pueden introducirse, pero no se asegura su veracidad. En el caso de la disponibilidad de conexión a internet el usuario puede indicar que dispone de ella, pero no ser cierto. Además de esos dos factores existe otro no menos importante, este es la obtención automática de la información. Toda la información que pueda ser obtenida automáticamente sin que el usuario deba indicarla agilizaría el proceso de recomendación y lo haría menos pesado para éste (véase el problema de navegación mediante propuesta en el que demasiadas preguntas agotan al usuario). Teniendo estos tres factores en cuenta podemos observar que resulta conveniente obtener automáticamente la mayor cantidad de información posible. Esto permitirá obtener atributos contextuales veraces, mas ágilmente y sin el problema de agotar al usuario o que no tenga los conocimientos necesarios para hacerlo.

A continuación se estudian qué atributos pueden obtenerse de manera automática utilizando tecnologías móviles y cuales sería estrictamente necesario ser introducidos de manera

explícita. El estudio sobre la obtención automática del contexto está basado en las tecnologías identificadas en la Sección 2.3 y en las aplicaciones analizadas en la Sección 2.4.

Atributo	Automática	Explícita
Grupo / Individual	Si	No
Amigos	Si	No
Afinidad	Si	No
Liderazgo	Si	No
Aficiones	No	Si
Situación laboral	No	Si
Estado familiar	No	Si

Cuadro 3.5: Obtención contexto social

**Tipología social.** La obtención de los atributos correspondientes a la tipología social puede verse en la Tabla 3.5. A continuación vemos la discusión de la obtención de los atributos uno por uno.

- Grupo/Individual: La obtención de este atributo puede realizarse de diversas maneras. La primera de ellas puede ser mediante la geolocalización de los usuarios. Para una determinada aplicación si se obtiene la localización de los usuarios y su relación (conectados en la aplicación, agendas telefónicas, etc.) puede determinarse si se encuentra sólo o en compañía. Otra posible solución puede ser utilizando tecnologías de conexión existentes en los dispositivos móviles como Bluetooth o WifiP2P (las cuales permiten ver qué dispositivos se encuentran en un rango cercano) y, al igual que en el anterior método, obteniendo la relación determinar el atributo contextual. Por ello se clasifica como atributo de obtención automática. Además, el usuario puede introducir de manera explícita con quién se encuentra, pero dependiendo del sistema puede no resultar conveniente que el usuario introduzca datos no veraces.
- Amigos: Como ya se vio en la Sección 2.4 las redes sociales permiten la obtención de gran información sobre usuarios, y gracias a las conexiones de estas redes puede

obtenerse información sobre sus amigos. Al igual que el atributo anterior clasificamos éste como de obtención automática y no recomendable de introducción explícita.

- **Afinidad:** La información sobre la afinidad de un usuario con otros (amigos, conocidos, pareja, etc.) puede determinarse analizando datos provenientes de las aplicaciones de comunicación estudiadas en la Sección 2.4 (frecuencia de comunicación, número de mensajes, grupos de conversación comunes, etc.) o a través de las redes sociales, con técnicas como la utilizada por Facebook para determinar los “amigos cercanos”, cómo son las fotos en común, grupos, eventos o amigos en común. Dados estos métodos de obtención clasificamos el atributo afinidad como de obtención automática.
- **Liderazgo:** Al igual que en el anterior la influencia de un usuario sobre los demás puede determinarse mediante las aplicaciones de comunicación o sistemas de opinión y deliberación (eventos en Facebook). Esto enmarca el atributo liderazgo como de obtención automática. Para la introducción explícita cualquier usuario podría indicar que él es líder de un grupo y no ser veraz, por lo que no es recomendable utilizarla.
- **Aficiones:** La información sobre las aficiones del usuario se ha clasificado como de introducción explícita ya que no se han encontrado tecnologías o redes que permitan su obtención automática.
- **Situación laboral:** La situación laboral resulta conveniente ser introducida por el propio usuario ya que, en la actualidad, el uso de redes laborales en internet es muy reciente y puede no disponerse de suficiente información.
- **Estado familiar:** Similar al caso anterior, usuarios de redes pueden introducir información sobre su estado familiar en sus perfiles, pero ya que no es una práctica suficientemente común es recomendable introducirse de manera explícita, por ejemplo al crear su cuenta en la aplicación y rellenar su perfil.

Atributo	Automática	Explícita
Localización	Si	No
Clima	Si	Si
Momento del día	Si	No

Cuadro 3.6: Obtención contexto geotemporal

**Tipología geotemporal.** La obtención de los atributos correspondientes a la tipología geotemporal puede verse en la Tabla 3.6. A continuación vemos la discusión de la obtención de los atributos uno por uno.

- Localización: atributo correspondiente a la geolocalización de usuario puede obtenerse de manera automática utilizando las tecnologías móviles vistas en la Sección 2.4, como son la geolocalización por sensor GPS o mediante redes móviles. Por ello y, dado que introducir manualmente coordenadas de geolocalización clasificamos este atributo como de obtención automática exclusivamente.
- Clima: El estado climático en la localización de usuario puede ser obtenido a través de servicios web existentes que recopilan información de estaciones meteorológicas, para ello será necesario dar las coordenadas físicas en las que se encuentra el usuario. Otra forma de obtenerlo puede ser la introducción explícita por parte del usuario mediante la selección de una lista de estados por ejemplo (soleado, nublado, lluvioso, etc.). El atributo clima queda entonces clasificado como de obtención automática y explícita.
- Momento del día: El momento del día resulta fácil de obtener en cualquier sistema, ya que es una información a determinar a partir de el valor horario en un momento concreto, quedando encuadrado en obtención únicamente automática.

**Tipología tecnológica.** La obtención de los atributos correspondientes a la tipología tecnológica puede verse en la Tabla 3.7. A continuación vemos la discusión de la obtención de los atributos uno por uno.

Atributo	Automática	Explícita
Dispositivo	Si	No
Datos móviles	Si	No
Conexión Wifi	Si	No
GPS	Si	No
Bluetooth	Si	No
WifiP2P	Si	No
Agenda	Si	No

Cuadro 3.7: Obtención contexto tecnológico

- Dispositivo: La información relativa a la plataforma móvil del dispositivo del usuario puede obtenerse a través del propio sistema operativo, por lo cual resulta conveniente obtenerlo de manera automática evitando que el usuario introduzca datos erróneos.
- Datos móviles: La disponibilidad de datos móviles puede ser obtenida también a través de la propia plataforma, por lo cual se clasifica como de obtención automática, siendo no recomendable una introducción explícita.
- Wifi: Igual al atributo anterior, para consultar si la conexión a internet se realiza mediante Wifi las plataformas móviles permiten su obtención de manera sencilla. Por lo tanto se clasifica como de obtención únicamente automática.
- GPS: La disponibilidad o no de sensor GPS puede consultarse de nuevo a través de las interfaces de programación que ofrecen las plataformas móviles, por tanto lo clasificamos de igual manera que los atributos anteriores (únicamente obtención automática).
- Bluetooth: La tecnología Bluetooth puede ser consultada también a través de la plataforma móvil. Pudiendo ver si se encuentra habilitada o no y los dispositivos que se encuentren cerca con esta conexión activada. Por tanto este atributo al igual que los anteriores se clasifica como de obtención automática estrictamente.
- WifiP2P: Para la conexión punto a punto mediante Wifi ocurre lo mismo que en los anteriores atributos de la tipología tecnológica, puede ser obtenido automáticamente

a través de la plataforma móvil y resulta conveniente evitar la introducción explícita.

- **Agenda:** Recordemos que el atributo *agenda* corresponde a la información de contactos que el usuario tiene almacenados en su dispositivo móvil, como pueden ser números de teléfono, emails, etc. El método para obtener esta información es a través de la plataforma móvil de nuevo, evitando la introducción explícita.

Atributo	Automática	Explícita
Edad	Si	Si
Historial compra	Si	Si
Necesidades	No	Si
Calendario	Si	Si
Humor	No	Si

Cuadro 3.8: Obtención contexto personal

**Tipología personal.** La obtención de los atributos correspondientes a la tipología personal puede verse en la Tabla 3.8. A continuación vemos la discusión de la obtención de los atributos uno por uno.

- **Edad:** La edad del usuario puede obtenerse a través de perfiles en redes sociales en las que suele ser común tener que introducir este dato, por tanto puede clasificarse como de obtención automática. En cuanto a la introducción explícita, ya que mediante el método automático comentado no resulta posible asegurar la veracidad de la información (la verificación de la edad de un usuario no es una práctica habitual en redes sociales), puede ser obtenido de manera explícita también.
- **Historial compra:** El historial de compra puede obtenerse a través de cuentas de tiendas, pero siendo éstas "privadas", ese historial sería accesible hipotéticamente para la aplicación correspondiente a la tienda. Otro método puede ser el de escaneo de códigos de barra y QR, almacenando datos sobre las compras realizadas según se realizan por

el usuario. Dados estos métodos podemos decir que el atributo se clasifica como de obtención automática y explícita.

- **Calendario:** Como se comentó en la sección 3.1.4 el atributo calendario corresponde a información de eventos generales o del usuario como por ejemplo cumpleaños o bodas, los cuales han tenido que ser introducidos previamente por el usuario aunque después se puedan obtener de manera automática a través de la plataforma móvil. Por esa razón se ha de clasificar como de obtención automática y explícita.
- **Humor:** El estado de humor, en la actualidad no es posible obtenerse de manera automática de manera sencilla, por tanto una posible solución es que sea introducido de manera explícita por el usuario seleccionando una serie de estados predefinidos.

# Capítulo 4

## Librería para el desarrollo de aplicaciones de recomendación contextual en dispositivos móviles

En este capítulo se presenta la librería desarrollada para la obtención de información contextual definida en el capítulo anterior a través de los smartphones. Para ello se estudiará que tecnologías existen en la actualidad en las plataformas móviles y en la web a un nivel más técnico y detallado, posteriormente se presentará el framework base del que se ha partido así como el diseño y arquitectura de la librería propuesta. Por último como demostrador se presentan ejemplos reales de obtención de los atributos contextuales definidos en diversos dispositivos móviles.

### 4.1. Tecnologías móviles para la obtención del conocimiento contextual

Una vez hemos estudiado la categorización del contexto, su dinamismo y sus posibilidades de obtención automática analizaremos qué oportunidades ofrecen tecnologías móviles y plataformas para obtener esta información a un nivel más técnico. Esto nos servirá de base para nuestra librería.

Para poder obtener la información contextual en smartphones, las tecnologías actuales ofrecen una gran variedad de soluciones. A continuación se exponen ejemplos de tecnologías que

permiten obtener atributos contextuales de manera automática con estas tecnologías.

- Geolocalización: Como ya se habló en apartados anteriores la geolocalización de un usuario es en la actualidad una característica muy utilizada por las aplicaciones móviles y resulta aún más importante para aplicar las técnicas de marketing especificadas en [Esteban, 2013]. La mayor parte de los dispositivos smartphone en la actualidad incluyen sensores GPS que permiten obtener la localización física del usuario en coordenadas de latitud y longitud, teniendo soporte nativo en las plataformas móviles como la de Android<sup>1</sup> o iOS<sup>2</sup>. Además de la utilización del sensor GPS puede utilizarse la técnica de localización por identificación de células de antenas móvil [Huber, ](CellID) lo cual nos permite obtener la localización de un individuo a través de su conexión móvil si su dispositivo no dispone de sensor GPS o no se encuentra disponible. Otra opción para obtener la localización física consiste en obtenerla a través de la IP de una conexión a un punto de acceso WiFi, aunque esto no se puede realizar con todos los puntos de acceso resulta una buena alternativa en caso de que la localización por GPS y CellID no funcione.

Una cuestión a plantear es si basta con conocer las coordenadas físicas de un usuario. Para calcular distancias o situar al usuario en un mapa este método es idóneo, pero si se quiere desarrollar una aplicación contextual que quiera conocer la ciudad, país o dirección en la que se encuentre será necesario obtener esa localización a partir de las coordenadas. La API de Google Maps<sup>3</sup> ofrece soporte para ello. Con una consulta especificando la latitud o longitud podemos obtener toda esa información, desde el país, ciudad hasta la dirección y código postal exacto.

- Clima: Otro atributo contextual que hemos destacado en este trabajo y en [Esteban, 2013] es el estado climático en la localización del usuario. En la actualidad se están desarro-

---

<sup>1</sup><http://developer.android.com/guide/topics/location/index.html>

<sup>2</sup><https://developer.apple.com/library/ios/navigation/#section=Frameworks&topic=CoreLocation>

<sup>3</sup><https://developers.google.com/maps/documentation/webservices/>

llando sensores climáticos incluidos en los propios dispositivos, pero los dispositivos actuales no los incorporan, por lo que para conocer el estado climático en una localización concreta se debe consultar a estaciones meteorológicas. Existen APIs web que permiten obtener información de estas consultando por ciudad o coordenadas de localización como *Wunderground*<sup>4</sup> u *OpenWeatherMap*<sup>5</sup>.

- Conexiones: En el apartado de comprobar las conexiones disponibles y obtener información de ellas como dispositivos cercanos por Bluetooth o Wifi ad-hoc, así como puntos de acceso Wifi o disponibilidad de conexión a internet las APIs nativas de las plataformas móviles ofrecen soporte para ello<sup>6 7</sup>.
- Social: Las relaciones sociales entre individuos también se considera un aspecto importante en este trabajo dada su importancia en los sistemas recomendadores grupales y el éxito de las redes sociales y plataformas de comunicación en los últimos años. A través de las redes sociales como Facebook o Twitter podemos obtener una gran cantidad de información sobre relaciones entre usuarios como su afinidad, amigos en común o interacción entre ellos. Para poder obtener esta información Facebook y Twitter ofrecen APIs para desarrolladores de aplicaciones<sup>8 9</sup>.

En el apartado de comunicaciones ya destacamos anteriormente las aplicaciones móviles de chat, las cuales son muy populares y se utilizan con gran frecuencia. Aplicaciones como Whatsapp nos pueden permitir obtener información de afinidad entre dos usuarios, por ejemplo comprobando la frecuencia con la que hablan o la cantidad de grupos en común, así como estimar, en un grupo de amigos, quién es el líder o la persona mas influyente, información importante para aplicar algunas técnicas de marketing como

---

<sup>4</sup><http://www.wunderground.com/weather/api/>

<sup>5</sup><http://openweathermap.org/api>

<sup>6</sup><http://developer.android.com/guide/topics/connectivity/index.html>

<sup>7</sup><https://developer.apple.com/library/ios/navigation/#section=Topics&topic=Networking%20%26amp%3B%20Internet>

<sup>8</sup><http://developers.facebook.com/>

<sup>9</sup><https://dev.twitter.com/>

la viral [Leskovec et al., 2007]. Otras aplicaciones de comunicación como GoogleTalk pueden resultar útiles para obtener información sobre contactos del usuario.

- Contactos: Al igual que en [Raento et al., 2005] el acceso a contactos de la agenda del usuario e información sobre ellos puede obtenerse automáticamente desde las plataformas nativas obteniéndose información como números de teléfono o cuentas de correo electrónico para relacionar usuarios en una hipotética aplicación contextual <sup>10</sup> <sup>11</sup>. Además las funciones adicionales de las agendas telefónicas como favoritos o registros de llamadas pueden aportarnos información adicional sobre las relaciones entre usuarios.
- Tecnología disponible: La información sobre la tecnología que utiliza el dispositivo del usuario puede servir a la hora de desarrollar una aplicación contextual para - por ejemplo- personalizar la interfaz, o difundir. Este aspecto resulta importante ya que la tecnología evoluciona con gran rapidez y los dispositivos móviles incorporan nuevas funcionalidades, tanto en hardware como software. Por este motivo no todas las aplicaciones que se desarrollan pueden utilizarse o aprovechar todo su potencial en dispositivos más antiguos. Por ejemplo, en el sistema operativo Android a lo largo de la evolución del sistema se van incorporando nuevas funcionalidades que pueden no ser compatibles con versiones anteriores <sup>12</sup>. Podemos obtener esta información de manera sencilla a través de las APIs nativas del dispositivo.
- Momento del día: En la sección 2.1 se comentó la importancia de este atributo para la aplicación en recomendadores contextuales, pudiéndose obtener de manera sencilla a través del propio sistema operativo móvil y su API <sup>13</sup>.
- Calendario: La información sobre los eventos próximos de un usuario podemos obtenerla a través de los gestores de calendario que proporcionan las plataformas móviles,

---

<sup>10</sup><http://developer.android.com/reference/android/provider/Contacts.html>

<sup>11</sup><https://developer.apple.com/library/ios/navigation/#section=Frameworks&topic=AddressBook>

<sup>12</sup><http://developer.android.com/guide/topics/manifest/uses-sdk-element.html>

<sup>13</sup><http://developer.android.com/reference/android/text/format/Time.html>

pudiendo así obtener fechas de cumpleaños, bodas y demás eventos que puedan ser utilizados por sistemas recomendadores,<sup>14 15</sup>.

- Lugares visitados y valoraciones. Como ya mencionamos anteriormente en algunos tipos de sistemas recomendadores las opiniones o valoraciones de un usuario son muy importantes a la hora de realizar una recomendación. Aplicado a localizaciones los lugares que se han visitado ya sean ciudades, museos, restaurantes, tiendas y demás, junto con las valoraciones pueden obtenerse a partir de redes sociales como Facebook, Twitter y otras destinadas a esta funcionalidad como Foursquare<sup>16</sup> y Google Latitude. La obtención de esta información podrá obtenerse utilizando las APIs que estas aplicaciones ofrecen a los programadores.

## 4.2. MobileContext

En este trabajo se ha hecho una revisión de las tecnologías móviles y su utilidad en la actualidad, así como la importancia de los smartphones y sus atractivas funcionalidades que, junto a las redes sociales, permiten la aplicación de técnicas de marketing en estos dispositivos así como a sistemas recomendadores.

Una vez estudiados los atributos contextuales que pueden obtenerse mediante las tecnologías móviles se ha categorizado el contexto en 3 dimensiones, según su esencia, dinamismo y posibilidad de obtención. Para esta última dimensión ha sido necesario estudiar más a fondo y de manera más técnica como pueden obtenerse la información contextual de un usuario de manera automática para facilitar la labor en sistemas recomendadores y así disponer de mayor información lo más veraz posible. Como ya se ha discutido en la sección anterior estos factores pueden hacer que el proceso de recomendación sea más completo y efectivo.

Por último en este trabajo se ha propuesto el desarrollo de una librería que permita la

---

<sup>14</sup><http://developer.android.com/guide/topics/providers/calendar-provider.html>

<sup>15</sup><https://developer.apple.com/library/ios/documentation/DataManagement/Conceptual/EventKitProgGuide/Introduction/Introduction.html>

<sup>16</sup><https://developer.foursquare.com/>

obtención de información contextual en plataformas móviles basándonos en los estudios anteriormente realizados.

Para ello y dado el problema planteado en la Sección 2.5 para el desarrollo de aplicaciones en dispositivos móviles sobre la necesidad de un “estándar” para el desarrollo en plataformas móviles dados los cambios de tendencia poco predecibles en el mercado móvil, para el desarrollo de esta librería se ha utilizado el framework **PhoneGap** siguiendo el estándar multiplataforma propuesto [Firtman, 2013] que permite el desarrollo de aplicaciones móviles en formato web sin la necesidad de realizar implementaciones nativas salvo que se necesiten funcionalidades extra que no incluya la plataforma.

#### 4.2.1. Base tecnológica

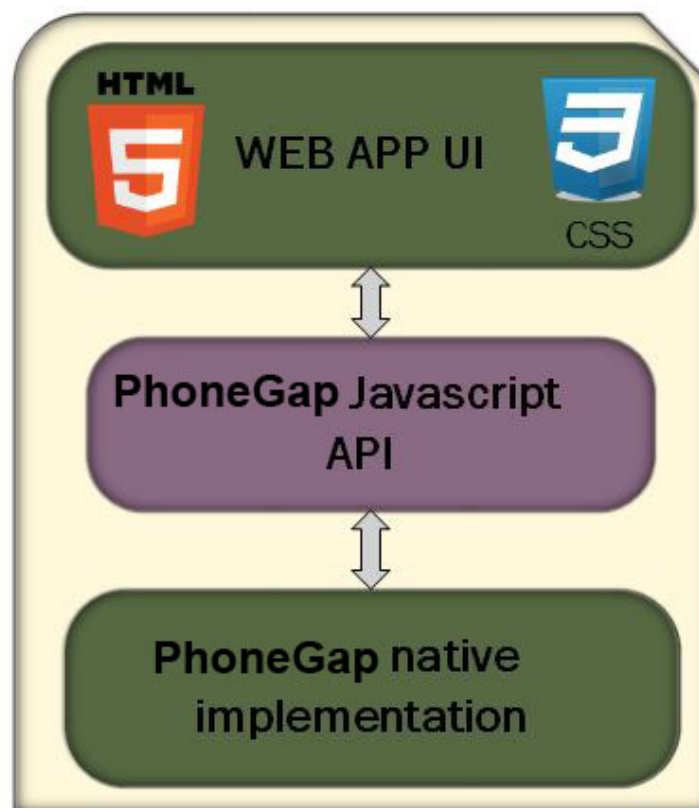


Figura 4.1: Arquitectura App con PhoneGap

Como hemos dicho anteriormente PhoneGap es una plataforma para el desarrollo de aplicaciones en distintas plataformas móviles siguiendo ciertos estándares logrando con ello poder portar la aplicación de una plataforma a otra sin realizar cambios significativos.

La arquitectura para el desarrollo de una aplicación móvil en esta plataforma puede verse en la Figura 4.1. Consiste en desarrollar la interfaz de usuario y funcionalidades en formato web utilizando *HTML5*, *CSS3* y *Javascript* pudiéndose utilizar también librerías web como **JQerymobile** [Reid, 2011]. Para acceder a funcionalidades del dispositivo como cámara, almacenamiento, contactos y demás PhoneGap proporciona una API <sup>17</sup> en Javascript con la que se pueden utilizar dichas funcionalidades, cabe mencionar que las funcionalidades disponibles no incluyen todas las disponibles en las plataformas móviles ya que el framework continua en desarrollo. Más adelante se explicará como extender estas funcionalidades.

Como hemos visto la arquitectura de Cordova esta formada por un nivel superior en el que se desarrolla la interfaz de usuario en formato web, la cual esta enlazada al sistema a través del *WebView engine*, en un nivel más inferior encontramos la API Javascript que permite utilizar las funcionalidades de un dispositivo móvil a través de las implementaciones nativas de cada funcionalidad para cada plataforma móvil; en concreto el framework incluye implementaciones nativas para, Android, BlackBerry, iOS, Symbian, WebOS, Windows Phone 7, Windows Phone 8, Windows 8, Bada y Tizen.

## Desarrollo en PhoneGap

Algunas de las funcionalidades que ofrece la API en Javascript de PhoneGap resultan útiles para ciertos atributos contextuales que deseamos obtener. Por ejemplo la geolocalización, el acceso a la cámara del dispositivo y a los contactos. Aún así muchas de estas funcionalidades se han modificado para proporcionar una librería con un uso más sencillo o ampliarlas. Para ello se han establecido configuraciones por defecto en algunos métodos, ya que algunas proporcionadas por PhoneGap necesitan una configuración como entrada. De

---

<sup>17</sup><http://docs.phonegap.com/en/2.5.0/index.html>

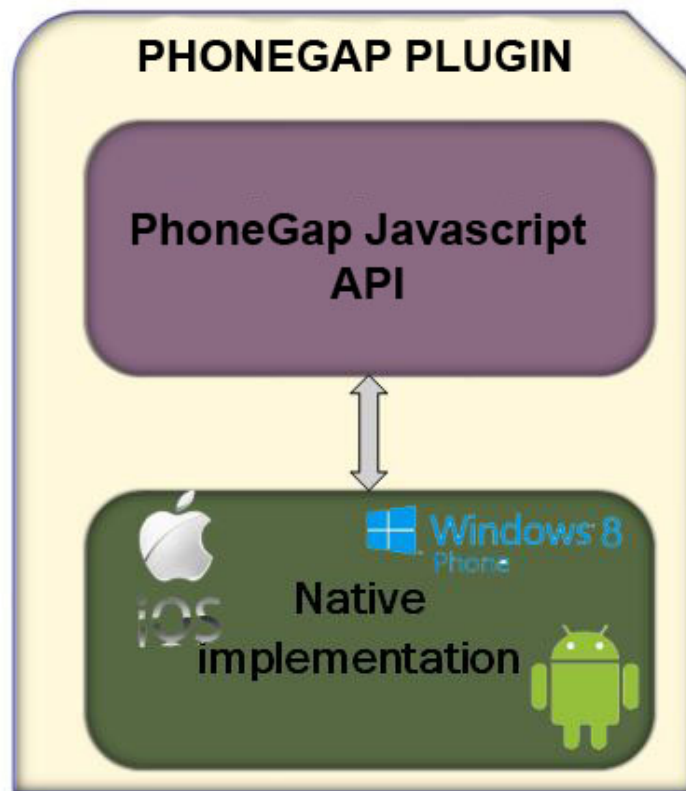


Figura 4.2: Arquitectura desarrollo plugin

este modo se proporciona un mayor número de métodos abarcando diversas posibilidades de configuración. Además de eso en algunos casos se ha modificado el formato de los datos de salida para facilitar su uso. Un ejemplo de esta simplificación es el acceso a la cámara del dispositivo móvil, PhoneGap proporciona un único método para realizar fotos a través de su API, al cual es necesario introducirle una configuración determinada si se requieren acciones muy específicas como el origen de la foto (cámara o galería), la calidad de esta, tipo de medio (foto/vídeo), cámara frontal o posterior y demás opciones. Para ello se simplifica el uso y se implementan una gran variedad de configuraciones por defecto que permitirán su uso de manera más sencilla.

Ya que no todos los atributos contextuales especificados en este trabajo se han podido obtener a través de la API de PhoneGap se han desarrollado nuevas funcionalidades

siguiendo el esquema que proporciona el framework. Este esquema puede observarse en la Figura 4.2 en el que podemos ver la arquitectura de los denominados *plugins* de PhoneGap, estos plugins permiten añadir funcionalidades extra en el desarrollo de una aplicación móvil utilizando el framework. El funcionamiento de los *plugins* es sencillo, consiste en realizar una implementación de la funcionalidad deseada en el lenguaje nativo del sistema operativo móvil (en la Sección 2.3 se vió como realizarlo) al que esté destinado e interconectarlo con la aplicación web a través de una API Javascript. Se puede observar que este funcionamiento es idéntico a la conexión entre los niveles 2 y 3 de la arquitectura general de PhoneGap en la Figura 4.1, siendo este el esquema que se utiliza en el framework para el desarrollo de todas sus funcionalidades. Puede verse en más detalle el desarrollo de un plugin en PhoneGap en la documentación proporcionada <sup>18</sup>.

En el inicio del desarrollo de este proyecto se ha comenzado por la implementación de las

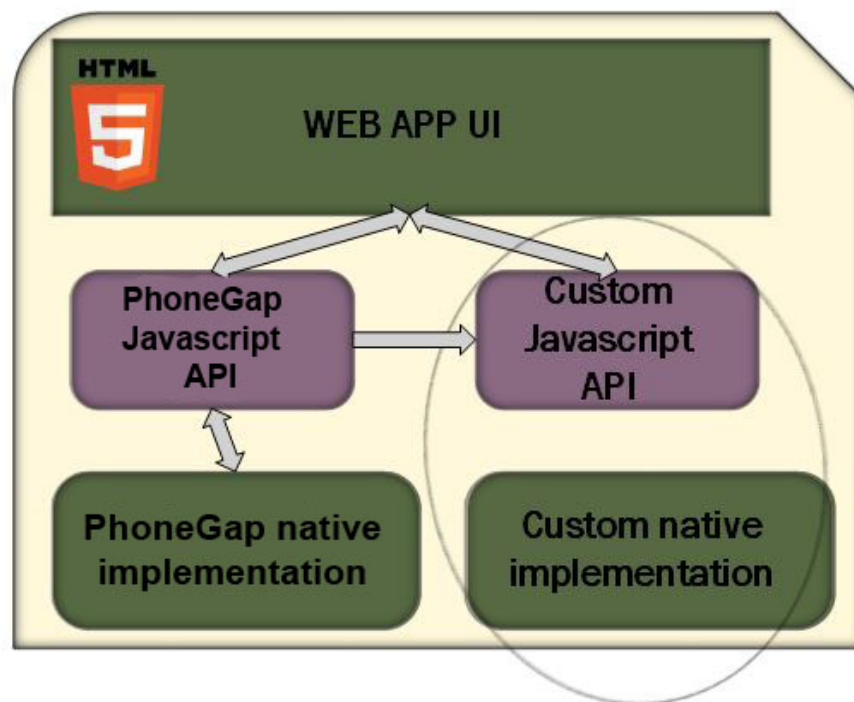


Figura 4.3: Arquitectura extensión funcionalidades de PhoneGap

<sup>18</sup>[http://docs.phonegap.com/en/2.5.0/guide\\_plugin-development\\_index.md.html](http://docs.phonegap.com/en/2.5.0/guide_plugin-development_index.md.html)

funcionalidades extra en la plataforma Android para posteriormente continuar con su desarrollo en iOS. La arquitectura final propuesta con las funcionalidades extra puede verse en la Figura 4.3, en la que tenemos la arquitectura básica de una aplicación en PhoneGap con la interfaz gráfica en formato web interconectada con la librería Javascript de PhoneGap. En el mismo nivel encontramos la API desarrollada extendiendo y simplificando funcionalidades de la propia librería de PhoneGap y realizándose un desarrollo nativo en la plataforma móvil Android.

### Arquitectura de MobileContext

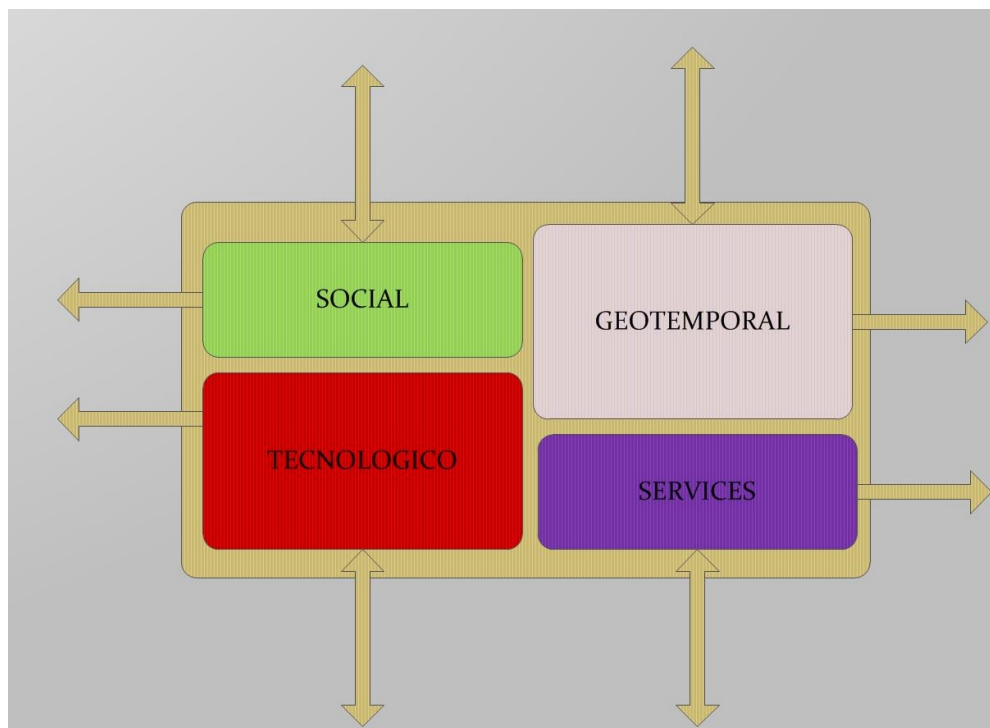


Figura 4.4: Arquitectura plataforma propuesta

La librería desarrollada en este trabajo esta formada por 4 módulos principales que corresponden a 3 de las 4 tipologías del contexto móvil descritas en la Sección 3.1, la tipología personal no se ha incluido como módulo dado que como se vio en la Sección 3.3 puede obtenerse de manera automática muy poca información. Como puede verse en la Figura

4.3, tenemos los módulos *Geotemporal*, *Tecnológico*, *Social* y *Services*. Este último módulo ha sido añadido para ofrecer una serie de servicios extra que pueden ser útiles a la hora de desarrollar un sistema recomendador en plataformas móviles.

- **Geotemporal:** Este módulo permite al acceso a la información clasificada como geotemporal en la Sección 3.1. A través de él se puede obtener la geolocalización del usuario del dispositivo móvil ya sea a través de GPS o redes móviles, esta geolocalización vendrá dada por las coordenadas de latitud y longitud, además también se puede obtener la localización dada por país, ciudad o código postal, así como la dirección exacta en la que se encuentra. Con este módulo se puede acceder también a la hora del sistema, así como al estado climatológico en la localización del usuario, este estado vendrá dado a partir de la geolocalización física del usuario y contendrá atributos como el estado general (soleado, nublado, etc.), la temperatura actual, temperaturas máximas y mínimas, viento, presión, etc.
- **Tecnológico:** El módulo tecnológico ofrece tanto información del dispositivo como de las conexiones disponibles. A través de este módulo se puede acceder a toda la información relativa al dispositivo, como la plataforma móvil utilizada, así como su versión, el nivel de batería, obtener información sobre todos los contactos disponibles en la agenda o buscar un contacto determinado. En el apartado de comunicaciones completa la información contextual tecnológica permitiendo no solo obtener el estado de la red si no también pudiendo activar y enlazar conexiones con el dispositivo. En concreto se tiene acceso a la información del estado de conexión a internet, así como la fuente de dicha conexión ya sea 2G, 3G, 4G, Ethernet o mediante un punto de acceso Wifi. Además se ha desarrollado la consulta y modificación de estado para las conexiones Wifi, Bluetooth y WifiP2P, pudiéndose consultar si se encuentra conectado con algunas de estas tecnologías, información sobre dispositivos o puntos de accesos disponibles, configuraciones almacenadas, habilitar o deshabilitar dichas conexiones y conectar a puntos de acceso o dispositivos.

Los dos últimos módulos desarrollados *Social* y *Services* proporcionan información del usuario proveniente de las redes sociales así como algunas funcionalidades extra que se han añadido independientemente de la obtención de información contextual.

- **Social:** El módulo social permite acceder a información relativa a las redes sociales del usuario que como vimos en la Sección 3.1 es un campo muy importante en la actualidad. Las funcionalidades principales desarrolladas en este módulo están centradas en la red social Facebook y la app de comunicaciones Whatsapp, para la primera a través del login del usuario se obtiene información de este mediante el API proporcionado por Facebook, información como el número de amigos, relación y distancia entre usuarios, fotos compartidas, etc. La app de comunicación whatsapp nos permite obtener información sobre la relación del usuario del dispositivo móvil con otros como el número de mensajes escritos, cantidad de grupos compartidos, frecuencia de mensajes, etc.
- **Services:** Este último módulo lleva incorporado un plugin que permite el escaneo de códigos de barras y QR <sup>19</sup> utilizando la cámara del dispositivo, esta funcionalidad permitiría en un recomendador contextual almacenar el historial de compra de manera sencilla para el usuario escaneando etiquetas o códigos. Como se dijo anteriormente se han añadido funcionalidades que no consisten en la obtención de información contextual, pero que podrían ser utilizadas para el desarrollo de sistemas recomendadores en sistemas móviles, en concreto está disponible un servicio de subida de archivos, desde archivos generales hasta subida directa desde la cámara del dispositivo, a su vez también se dispone de otro servicio de descarga de archivos al dispositivo.

En la Sección 3.2 se expuso la necesidad de estudiar el dinamismo de los atributos contextuales para su correcta utilización. Por ello en la librería se ha incluido dicha característica, cada atributo tiene asociado el momento en el que ha sido actualizado dicho atributo, pudiéndose consultar en todo momento. Este valor será actualizado cada vez que se realice una petición de actualización en el atributo contextual.

---

<sup>19</sup><https://github.com/phonegap/phonegap-plugins/tree/master/Android/BarcodeScanner>

## 4.2.2. Descripción funcional de MobileContext

En esta sección se especificarán las funciones disponibles en la librería propuesta. Puede verse la especificación para los módulos Geo, Device, Communications, Social y Services [4.3](#). Dentro de cada módulo se detalla para cada función los parámetros de entrada, salida y los permisos necesarios en las plataformas para poder utilizarlas.

### Geotemporal

- *getGeoLocation(success,fail)*: Recibe como parámetros las funciones de éxito y fallo. En caso de éxito devuelve en la función *success* un objeto en formato JSON con los atributos altitud y longitud que definen la geolocalización del usuario:

```
geo:{  
  latitud,  
  longitud,  
  timestamp }
```

- *getLocation(success,fail)*: Recibe como parámetros las funciones de éxito y fallo. En caso de éxito devuelve en la función *success* un objeto en formato JSON con la localización del dispositivo especificada con los atributos *país, comunidad, ciudad, localidad, dirección* y *código postal*:

```
geolex:{  
  país,  
  comunidad,  
  localidad,  
  direccion,  
  zip,  
  timestamp }
```

- *getWeather(success,fail)*: Recibe como parámetros las funciones de éxito y fallo. En caso de éxito devuelve en la función *success* un objeto en formato JSON con informa-

ción sobre el clima en la localización actual del dispositivo:

```
geolox:{  
estado,  
temp,  
tempmax,  
tempmin,  
humedad,  
lluvia,  
viento,  
nubes,  
timestamp }
```

- *getDate(success,fail)*: Para la obtención del atributo contextual *Momento del día* este método devuelve a través de la función *success* el valor actualizado en formato JSON al igual que en los demás atributos:

```
geolox:{  
date,  
timestamp }
```

## Tecnológico

- *getDeviceInfo(success,fail)*: Recibe como parámetros las funciones de éxito y fallo. En caso de éxito devuelve en la función *success* un objeto en formato JSON con información sobre el dispositivo del usuario:

```
internet_state:{  
platform,  
version,  
name,  
UUID,
```

```
batterylevel,  
timestamp }
```

- *getInternetState(success,fail)*: Recibe como parámetros las funciones de éxito y fallo. En caso de éxito devuelve en la función *success* un objeto en formato JSON con información sobre la disponibilidad de conexión a internet y el modo de conexión (WiFi, 2G, 3G, 4G o Ethernet):

```
internet_state:{  
state,  
source,  
timestamp }
```

- *getWifitState(success,fail)*: Recibe como parámetros las funciones de éxito y fallo. En caso de éxito devuelve en la función *success* un objeto en formato JSON con información sobre el estado de la conexión WiFi del dispositivo, en caso de estar conectado a través de esta tecnología se obtiene información sobre el punto de acceso:

```
wifi_state:{  
state,  
conected,  
BSSID,  
SSID,  
signal,  
timestamp }
```

- *scanWifitNets(success,fail)*: Recibe como parámetros las funciones de éxito y fallo. En caso de éxito devuelve en la función *success* un objeto en formato JSON con información sobre los puntos de acceso disponibles, así como información sobre cada uno de ellos:

```
wifi_scan:{
```

*nets*[],

*timestamp* } El atributo *nets* contiene una lista sobre redes disponibles, de cada elemento de esta lista se encuentra la siguiente información:

*net*:{

*conected*,

*BSSID*,

*SSID*,

*signal*,

*security* }

- *EnableWifi(success,fail)*: Función para activar la funcionalidad Wifi en el dispositivo. Devuelve la confirmación de éxito si ha sido posible habilitarla o de fallo en caso contrario.
- *DisableWifi(success,fail)*: Similar a la anterior esta función permite desactivar la funcionalidad Wifi en el dispositivo. Devuelve la confirmación de éxito si ha sido posible deshabilitarla o de fallo en caso contrario.
- *getBluetoothState(success,fail)*: Recibe como parámetros las funciones de éxito y fallo. En caso de éxito devuelve en la función *success* un objeto en formato JSON con información sobre el estado de la conexión Bluetooth del dispositivo, en caso de estar conectado a través de esta tecnología se obtiene información sobre el dispositivo al que se encuentra conectado:

*BT\_state*:{

*state*,

*conected*,

*DeviceID*,

*timestamp* }

- *EnableBT(success,fail)*: Función para activar la funcionalidad Bluetooth en el dispositi-

tivo. Devuelve la confirmación de éxito si ha sido posible habilitarla o de fallo en caso contrario.

- *DisableBT(success,fail)*: Similar a la anterior esta función permite desactivar la funcionalidad Bluetooth en el dispositivo. Devuelve la confirmación de éxito si ha sido posible deshabilitarla o de fallo en caso contrario.
- *ConnectBT(DeviceID,success,fail)*: Esta función permite conectar el dispositivo Bluetooth a otro que se encuentre disponible. Recibe como parámetro el ID del dispositivo al que se desea conectar y las funciones de éxito y fallo correspondientes que recogerán la información devuelta por la función (éxito en caso de conexión y fallo en caso de no ser posible realizar la conexión).
- *scanWifiP2PNets(success,fail)*: Recibe como parámetros las funciones de éxito y fallo. En caso de éxito devuelve en la función *success* un objeto en formato JSON con información sobre los puntos de acceso WiFiP2P disponibles, así como información sobre cada uno de ellos:

```
wifip2p_scan:{
```

```
  devs[],
```

```
  timestamp } El atributo devs contiene una lista sobre dispositivos con WiFiP2P disponibles, de cada elemento de esta lista se encuentra la siguiente información:
```

```
  dev:{
```

```
    conected,
```

```
    DEVID,
```

```
    signal }
```

## Social

- *getFacebookLogin(success,fail)*: Al realizar la llamada a esta función aparecerá en la aplicación el formulario de login de Facebook para que el usuario introduzca su cuenta

y contraseña, en caso de éxito en el login se devolverá en formato JSON información de la cuenta del usuario: *fb\_login*:

```
{  
  UID,  
  age,  
  friends,  
  closest_friends[], IDs de los amigos mas cercanos  
  timestamp }
```

Las siguientes funcionalidades ofrecidas a través de la aplicación móvil Whatsapp únicamente están disponibles con permisos de superusuario en el terminal Android.

- *getWhatsappGroups(success,fail)*: Al realizar la llamada a esta función se consulta el número de grupos de conversación en los que esta incluido el usuario en la aplicación Whatsapp, así como los integrantes de cada grupo. *wappgroups*:

```
{  
  count,  
  groups[],  
  timestamp }
```

El atributo *groups* contendrá una lista de grupos de Whatsapp en la que cada elemento dispone de la siguiente información:

```
group:  
{  
  name,  
  admin, Número de teléfono del administrador de grupo  
  numbers[], Lista con los números de teléfono que forman parte del grupo  
  timestamp }
```

- *getLastMessage(telf,success,fail)*: Al realizar la llamada a esta función se consulta la fecha del último mensaje escrito con el usuario correspondiente al número de teléfono recibido por parámetro: *last\_message*:

*date,*  
*timestamp }*

- *getNumberMessages(telf,success,fail)*: Al realizar la llamada a esta función se consulta el número de mensajes escritos con el usuario correspondiente al número de teléfono recibido por parámetro: *last\_message*:{

*number,*  
*timestamp }*

- *getNumberGroups(telf,success,fail)*: Al realizar la llamada a esta función se consulta el número de grupos que el usuario del dispositivo comparte con el correspondiente al número de teléfono recibido por parámetro: *last\_message*:{

*number,*  
*timestamp }*

## Services

- *getPictureLowQualityRear(success,fail)*: Con esta función se lanzará la funcionalidad de cámara en el dispositivo con la configuración de baja calidad y de la cámara posterior. En caso de que se realice la fotografía se devolverá por parámetro la ruta donde se ha almacenado la imagen:

*pic\_low\_rear*:{  
*path,*  
*timestamp }*

- *getPictureLowQualityFront(success,fail)*: Al igual que con la anterior esta función se lanzará la funcionalidad de cámara en el dispositivo con la configuración de baja calidad y de la cámara frontal en este caso. En caso de que se realice la fotografía se devolverá por parámetro la ruta donde se ha almacenado la imagen:

*pic\_low\_front*:{

*path,*  
*timestamp }*

Análogas a las anteriores se tienen las funciones para calidades media y máxima:

- *getPictureMediumQualityRear(success,fail):*

*pic\_med\_rear:{*  
*path,*  
*timestamp }*

- *getPictureMediumQualityFront(success,fail):*

*pic\_med\_front:{*  
*path,*  
*timestamp }*

- *getPictureFullQualityRear(success,fail):*

*pic\_full\_rear:{*  
*path,*  
*timestamp }*

- *getPictureFullQualityFront(success,fail):*

*pic\_full\_front:{*  
*path,*  
*timestamp }*

También se dispone de una función que permite introducir por parámetro el porcentaje de calidad deseada:

- *getPictureCustomQualityRear(quality,success,fail):*

*pic\_custom\_rear:{*  
*path,*  
*timestamp }*

- *getPictureCustomQualityFront(quality,success,fail)*:

```
pic_custom_front:{  
  path,  
  timestamp }
```

Además podemos obtener la fotografía directamente desde la galería de imágenes que el usuario dispone en su dispositivo:

- *getPictureFromAlbum(success,fail)*: Al llamar a esta función el usuario podrá explorar su galería y seleccionar una imagen de ella. Al seleccionarse dicha imagen se devolverá por parámetro la ruta en la que se encuentra almacenada dicha imagen.

```
pic_album:{  
  path,  
  timestamp }
```

Otro servicio que se ha añadido a la librería consiste en la subida y descarga de imágenes y ficheros:

- *uploadPhoto(server,success,fail)*: La llamada a esta función lanzará la funcionalidad de cámara de nuevo y una vez se realice la fotografía será enviada a la dirección especificada por el parámetro *server*.
- *uploadFile(server,file\_path,success,fail)*: Más genérica, esta función permite la subida del archivo especificado por su ruta en el parámetro *file\_path* a la dirección *server*.
- *downloadPhoto(server,target\_path,success,fail)*: Con esta función se puede descargar un archivo desde la dirección *server*, la ruta de almacenamiento del archivo debe ser especificada en el argumento *target\_path*.

Por último, en el módulo de servicios se ha añadido una funcionalidad ya desarrollada correspondiente a la lectura de códigos de barra y QR que puede ser de utilidad para la identificación de productos en el desarrollo de un sistema de recomendación:

- *Scan(success,fail)*: Al realizar la llamada al método *Scan* se activará la cámara del dispositivo para escanear un código de barras o QR. Si el código es identificado correctamente se devuelve la información sobre el código en un objeto JSON con el siguiente formato:

```
scan:{  
  type:QR/Barcode,  
  code,  
  timestamp }
```

### 4.2.3. Demostrador

En esta sección se pueden ver ejemplos de la librería propuesta en este trabajo. Estas pruebas se han realizado mediante el desarrollo de una aplicación móvil sencilla en HTML siguiendo el esquema de desarrollo propuesto por PhoneGap 4.1 para la plataforma Android, utilizándose para estas dos dispositivos distintos, el primero de ellos un smartphone con la versión 2.3.6 de Android y el segundo una tablet de la misma plataforma con la versión 4.1 del sistema operativo. Como características de hardware el dispositivo móvil dispone de conexión 2G, 3G, Wifi, Bluetooth y GPS, el segundo dispositivo dispone de la tecnología WifiP2P y conexión a punto de acceso Wifi. A continuación se muestran algunos ejemplos de uso de los distintos módulos disponibles en la librería.

#### Geotemporal

Para el módulo geotemporal se mostrarán los ejemplos de geolocalización en coordenadas, geolocalización con descripción y por último el estado climático en dicha localización.

- Localización latitud-longitud:

En la Figura 4.5 podemos ver el ejemplo de obtención de coordenadas de localización de un usuario. Utilizando la librería propuesta obtenemos las coordenadas en latitud y longitud donde se encuentra el dispositivo móvil.

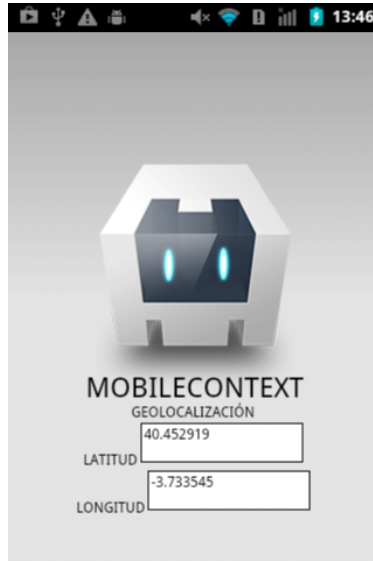


Figura 4.5: Geolocalización latitud-longitud

- Localización descriptiva:



Figura 4.6: Geolocalización por descripción

En la figura 4.6 podemos ver la obtención de la localización descriptiva a partir de las coordenadas latitud-longitud del ejemplo de la Figura 4.5.

- Clima: A partir de las coordenadas de latitud y longitud que vimos en la Figura 4.5

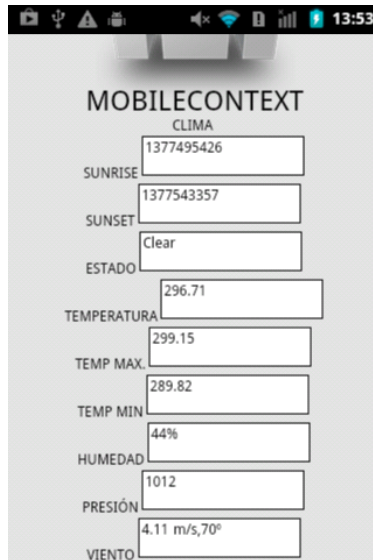


Figura 4.7: Clima en la localización del dispositivo

se ha obtenido el estado climático para esa localización. En la Figura 4.7 podemos ver los valores obtenidos.

## Tecnológico

Para el módulo tecnológico mostraremos los ejemplos:

- **Dispositivo:** Utilizando la librería propuesta y realizando una consulta a los métodos correspondientes podemos obtener información sobre el dispositivo móvil. Como podemos ver en la Figura 4.8 se ha obtenido la plataforma móvil utilizada, así como la versión del sistema operativo, el ID del dispositivo y el modelo. Además también podemos ver el porcentaje de batería de que dispone el terminal.
- **Conexiones disponibles:**

Realizando llamadas a los métodos disponibles en el módulo tecnológico se obtiene el estado y/o disponibilidad de conexiones. En la Figura 4.9 podemos ver los valores obtenidos para la consulta de disponibilidad de conexión a internet y redes móviles, así como el estado de las conexiones Wifi, WifiP2P y Bluetooth.



Figura 4.8: Información sobre dispositivo móvil

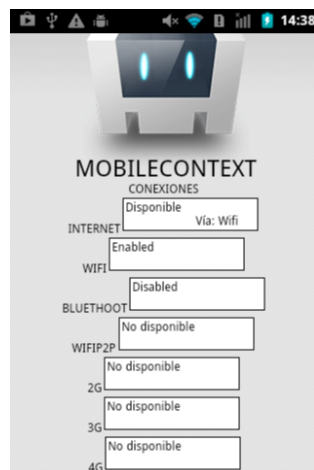


Figura 4.9: Conexiones disponibles/estados

- Estado conexión Wifi:

Otra funcionalidad disponible en este módulo es la de obtener información sobre la conexión Wifi. Realizando una llamada al método correspondiente podemos obtener si el dispositivo se encuentra conectado o no a un punto de acceso y, en el caso de estar conectado, obtener información sobre dicho punto de acceso y la conexión como el ID de éste, la IP proporcionada, el nombre público del punto de acceso, así como la



Figura 4.10: Estado conexión Wifi

dirección física (MAC) y la intensidad de la señal recibida. Todos estos datos podemos verlos en la Figura 4.10

- **Búsqueda contacto:** Además el módulo tecnológico permite obtener información de todos los contactos que el usuario tiene en su agenda, se ha incluido una funcionalidad de búsqueda por nombre que puede verse en la Figura 4.11.



Figura 4.11: Búsqueda de contacto en agenda

## Social

Dentro del módulo social se muestra un ejemplo de la integración con Facebook.

- Login Facebook:

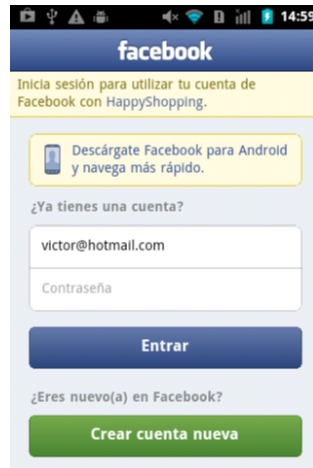


Figura 4.12: Formulario login en Facebook

Realizando una llamada al método correspondiente del módulo social aparecerá en la pantalla del usuario el formulario de inicio de sesión de Facebook que puede verse en la Figura 4.12. Una vez el usuario inicie sesión se podrá obtener información de él a través de la aplicación desarrollada. En la Figura 4.13 se puede ver un ejemplo de información obtenida.



Figura 4.13: Información desde perfil de Facebook

## Services

Para el módulo de servicios podemos ver ejemplos de obtención de una fotografía con la cámara del dispositivo y de escaneo de código de barras o QR.

- Cámara: Realizando la llamada a la función de la librería correspondiente se activa en el dispositivo la función de cámara fotográfica, en la Figura 4.14 puede verse este ejemplo.

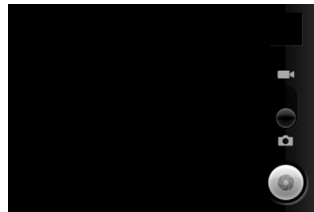


Figura 4.14: Captura fotografía

- QR/Barcode: La funcionalidad extra añadida para el escaneo de códigos de barras o QR puede verse en la Figura 4.15, colocando en el recuadro el código a escanear podrá obtenerse el valor correspondiente a dicho código.

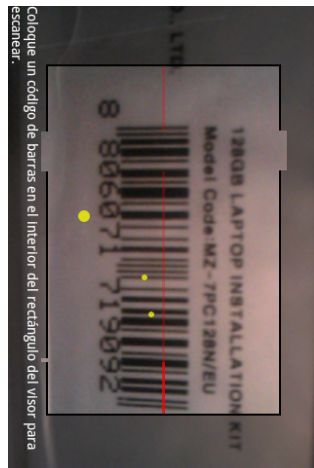


Figura 4.15: Escaneo código de barras/QR

### 4.3. Caso de uso: MobiMall

En la Sección 4.2.3 se presentó un demostrador utilizando la librería propuesta para obtener información contextual en la plataforma móvil Android. Además de este demostrador esta librería ha sido utilizada también en un caso de uso, en concreto, en [Esteban, 2013] se ha desarrollado una aplicación móvil utilizando tecnologías web siguiendo la arquitectura de PhoneGap, véase Figura 4.1. Dicha aplicación implementa una serie de casos de uso propuestos en el trabajo, cada caso de uso corresponde con una técnica de marketing aplicada a sistemas de recomendación. Para poder establecer esa relación entre los sistemas de recomendación y las técnicas de marketing es necesaria la obtención de información contextual para poder aplicar estas técnicas de manera correcta. Por tanto resulta necesario obtener esa información contextual a través de los dispositivos móviles, es ahí donde el trabajo aquí propuesto entra en la relación entre las técnicas de marketing y los sistemas de recomendación. En la Figura 4.16 puede verse el esquema de la relación entre ambos trabajos.

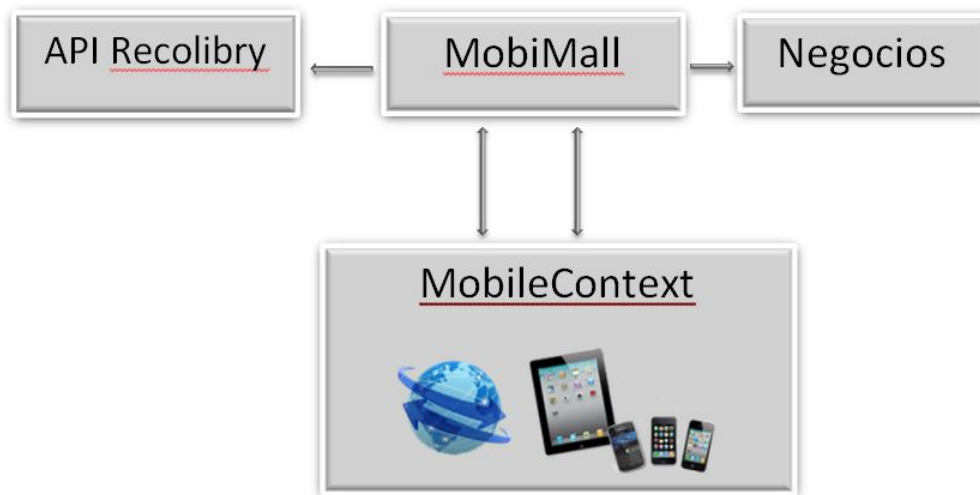


Figura 4.16: Esquema aplicación MobiMall

Concretamente, en el desarrollo de los casos de uso se han utilizado las siguientes funcionalidades de la librería MobileContext:

- Subida de imágenes: Se ha utilizado el método disponible en el módulo de servicios para permitir al usuario de la aplicación subir fotos de productos en los que tenga interés y pedir opinión a otros usuarios.
- Usuarios cercanos: Una característica de la aplicación MobiMall es la posibilidad de ofrecer ofertas a grupos de usuarios desconocidos que se encuentren en una localización cercana, como un centro comercial. Para ello se ha utilizado la funcionalidad de geolocalización que ofrece MobileContext.
- Contactos: También se ha utilizado la búsqueda de contactos disponible en la librería para que el usuario de la aplicación pueda seleccionar aquellos contactos a los que desea mandar la imagen de un producto y pedir su opinión.
- ID usuario en Facebook: Una de las características principales del sistema MobiMall es la de la recomendación. Para poder realizar recomendaciones de prendas a usuarios de la aplicación se ha utilizado una API proporcionada por una aplicación integrada en Facebook, en la cual el usuario debe estar registrado. Por ello el método de login de Facebook y la obtención de su identificador de usuario con MobileContext ha sido también utilizada

# Capítulo 5

## Conclusiones y trabajo futuro

En este trabajo se ha expuesto la creciente tendencia en la actualidad en el uso de los smartphones, dadas sus características como el acceso a contenido web, medios y localización cada vez son utilizados por un mayor número de usuarios. Esto se ha visto reflejado en un aumento en sus ventas en comparación con portátiles u ordenadores de escritorio. Estas características, además se han visto potenciadas con el uso de las redes sociales en estos dispositivos con el desarrollo de aplicaciones dedicadas a ellas. Las redes sociales también han mostrado una gran evolución desde su aparición y cada vez cuentan con mayor número de usuarios. Por esta razón disponen de mayor información que puede ser obtenida para su aplicación a diversos campos como el de los sistemas de recomendación.

En concreto en este trabajo se ha estudiado el uso de los dispositivos móviles para la obtención de información contextual de manera dinámica y poder extender estos sistemas no solo a los ordenadores de uso común sino también a los smartphones, pudiendo así realizar recomendaciones a un usuario utilizando su dispositivo móvil. Para llevar a cabo esta tarea se han identificado las tecnologías móviles existentes y estudiado qué posibilidades ofrecen para su aplicación en sistemas de recomendación. Dentro de estas tecnologías se han visto las distintas plataformas existentes en la actualidad y cómo las aplicaciones de dichas plataformas hacen uso y/u obtienen

información contextual del usuario. Además se ha comprobado que las tendencias en la cuota de mercado de las plataformas móviles puede variar de manera muy rápido en cuestión de pocos años; como se ha visto en la Sección 2.3 con el ejemplo del dominio del mercado por parte de Symbian y el cambio drástico que ha llevado en poco a Android y iPhone a ocupar ese puesto gracias a la aparición y la apuesta por los dispositivos smartphones. Por esa razón en el desarrollo de la librería propuesta en este trabajo se ha optado por el uso de estándares multiplataforma, en concreto PhoneGap[Lunny, 2011][Myer, 2011][Ghatol and Patel, 2012].

Como contribución en este trabajo en primer lugar se han identificado los atributos contextuales que pueden ser útiles para su aplicación en sistemas de recomendación. Estos atributos contextuales se han definido partiendo de la revisión del estado del arte y de la identificación de las tecnologías y plataformas móviles utilizadas en la actualidad. Posteriormente se ha visto la necesidad de categorizar estos atributos contextuales con el fin de estudiarlos en mayor profundidad y para su posterior obtención y utilización. En concreto se ha descrito un modelo teórico del contexto móvil que engloba 3 dimensiones: la dimensión tipológica, el dinamismo de los atributos y la interacción con el usuario. En la primera de estas dimensiones -tipológica- se han agrupado atributos según su naturaleza, perteneciendo atributos a una misma tipología contextual aquellos que tienen características similares. Para esta dimensión se han definido 4 tipologías: tipología social, personal, tecnológica y geotemporal. La segunda dimensión -dinamismo- corresponde al estudio de cómo pueden variar los valores de los atributos contextuales en función del tiempo, así se ha clasificado cada atributo de las 4 tipologías contextuales en 3 niveles: dinamismo bajo, medio y alto. Con esta clasificación se puede hacer un mejor uso de los atributos definidos evitando consultas innecesarias para atributos estáticos y valores incorrectos para aquellos con un nivel de dinamismo alto. Como última dimensión se ha propuesto la interacción con el usuario, estudiando la posibilidad de obtener automáticamente atributos del contexto móvil o,

si por el contrario es necesario una introducción explícita por parte del usuario. Así se han definido aquellos atributos que pueden o deben obtenerse de manera implícita además de cómo hacerlo.

Una vez definido el modelo teórico del contexto móvil se ha desarrollado una librería que permite la obtención de información contextual de manera implícita basándose en dicho modelo. En esta librería se han incluido aquellos atributos que han sido clasificados en la tercera dimensión como posibles valores implícitos. Además dicha librería ha sido dividida en módulos como reflejo de la clasificación de atributos según su tipología, pudiendo incluir módulos de manera independiente según las necesidades. La segunda dimensión -dinamismo- puede verse reflejada en esta librería con la inclusión de un valor temporal en cada atributo, para poder consultar el instante de tiempo en el que se actualizó por última vez dicho atributo y valorar la posibilidad de actualizarlo. Para el desarrollo de la librería propuesta se ha utilizado el estándar multiplataforma propuesto por Phonegap, así como los recursos ofrecidos por dicha plataforma; pudiéndose utilizar la librería en distintas plataformas móviles.

En último lugar, como caso práctico se ha presentado un demostrador en el cual puede verse la obtención de diversos atributos contextuales en la plataforma Android, así como un caso de uso en el que se ha utilizado la librería propuesta para desarrollar una aplicación -MobiMall- que requiere la obtención de información contextual para aplicar técnicas de marketing con sistemas de recomendación.

## **5.1. Trabajo futuro**

En la finalización de este trabajo se han identificado una serie de trabajos futuros que pueden ser necesarios para mejorar la contribución aquí propuesta. En primer lugar puede estudiarse la inclusión nuevos atributos en el modelo de contexto móvil ya que en este trabajo se ha definido para su aplicación en sistemas de recomendación en

general y para un sistema de recomendación concreto pueden requerirse atributos que no se han incluido en el modelo propuesto. Al igual que con los atributos definidos, las dimensiones propuestas forman un modelo teórico general para estos sistemas, por lo que para sistemas específicos se debe estudiar la necesidad de incluir otras dimensiones necesarias según las particularidades del sistema.

Como se vio en la Sección 4 la obtención de los atributos contextuales ha sido realizada de 3 maneras: a través de la API proporcionada por Phoneygap, a través de servicios web y con implementaciones nativas. Aquellos atributos obtenidos con los dos primeros métodos pueden ser utilizados en diversas plataformas, pero no así con aquellos que requieren implementaciones nativas ya que se ha realizado el desarrollo nativo en la plataforma móvil Android. Por ello se propone extender la obtención de estos atributos realizando las implementaciones nativas en plataformas como iOS o Windows8.

Otro punto a desarrollar puede ser la integración en las APIs de las redes sociales, ya que no se ha podido explotar todo el potencial de la API proporcionada por Facebook; además la eliminación de la API en Javascript de Twitter ha impedido el acceso a información del usuario en dicha red. Por tanto se propone una mayor integración en Facebook y Twitter en el caso de que vuelva a proporcionar una interfaz en dicho lenguaje, así como la posibilidad de interactuar de manera directa con las aplicaciones que estas redes sociales ofrecen en las plataformas móviles.

Por último resulta conveniente el uso de la librería propuesta en distintas plataformas móviles ya que aunque se ha realizado el desarrollo utilizando un estándar multiplataforma las pruebas han sido realizadas con aplicaciones Android, así como su uso en distintos tipos de sistemas de recomendación reales que puedan ser evaluados por distintos usuarios.

# Bibliografía

- [Adomavicius and Tuzhilin, 2011] Adomavicius, G. and Tuzhilin, A. (2011). Context-aware recommender systems. In *Recommender systems handbook*, pages 217–253. Springer.
- [Bazire and Brézillon, 2005] Bazire, M. and Brézillon, P. (2005). Understanding context before using it. In *Modeling and using context*, pages 29–40. Springer.
- [Burke, 2002] Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370.
- [Esteban, 2013] Esteban, A. G. (2013). Uso del contexto social en estrategias de marketing para sistemas recomendadores.
- [Firtman, 2013] Firtman, M. (2013). *Programming the mobile web*. O’Reilly.
- [Gartrell et al., 2010] Gartrell, M., Xing, X., Lv, Q., Beach, A., Han, R., Mishra, S., and Seada, K. (2010). Enhancing group recommendation by incorporating social relationship interactions. In *Proceedings of the 16th ACM international conference on Supporting group work*, pages 97–106. ACM.
- [Ghatol and Patel, 2012] Ghatol, R. and Patel, Y. (2012). *Beginning PhoneGap: Mobile Web Framework for JavaScript and HTML5*. Apress.
- [Goodman et al., 2007] Goodman, D., Morrison, M., and Eich, B. (2007). *Javascript® bible*. John Wiley & Sons, Inc.
- [Graham, 1995] Graham, I. S. (1995). *The HTML sourcebook*. John Wiley & Sons, Inc.
- [Huber, ] Huber, D. Background positioning for mobile devices-android vs. iphone.

- [Jannach et al., 2010] Jannach, D., Zanker, M., Felfernig, A., and Friedrich, G. (2010). *Recommender systems: an introduction*. Cambridge University Press.
- [Kim, 2013] Kim, Y. S. (2013). Recommender system based on product taxonomy in e-commerce sites. *Journal of Information Science and Engineering*, 29(1):63–78.
- [Kochan, 2011] Kochan, S. G. (2011). *Programming in objective-C*. Addison-Wesley Professional.
- [Leskovec et al., 2007] Leskovec, J., Adamic, L. A., and Huberman, B. A. (2007). The dynamics of viral marketing. *ACM Transactions on the Web (TWEB)*, 1(1):5.
- [Li et al., 2005] Li, Y., Lu, L., and Xuefeng, L. (2005). A hybrid collaborative filtering method for multiple-interests and multiple-content recommendation in e-commerce. *Expert Systems with Applications*, 28(1):67 – 77.
- [Li et al., 2013] Li, Y.-M., Wu, C.-T., and Lai, C.-Y. (2013). A social recommender mechanism for e-commerce: Combining similarity, trust, and relationship. *Decision Support Systems*, 55(3):740 – 752.
- [Lunny, 2011] Lunny, A. (2011). *PhoneGap Beginner’s Guide*. Packt Publishing.
- [Meyer, 2009] Meyer, E. A. (2009). *CSS: the definitive guide*. O’Reilly Media, Inc.
- [Myer, 2011] Myer, T. (2011). *Beginning PhoneGap*. John Wiley & Sons.
- [Quijano Sánchez, 2010] Quijano Sánchez, L. (2010). *Impacto de los factores y organizaciones sociales en los procesos de recomendación para grupos*. PhD thesis.
- [Raento et al., 2005] Raento, M., Oulasvirta, A., Petit, R., and Toivonen, H. (2005). Contextphone: A prototyping platform for context-aware mobile applications. *Pervasive Computing, IEEE*, 4(2):51–59.
- [Reid, 2011] Reid, J. (2011). *jQuery Mobile*. O’Reilly Media, Inc.
- [Ruiz-Iniesta et al., 2010] Ruiz-Iniesta, A., Jiménez-Díaz, G., and Gómez-Albarrán, M. (2010). Personalización en recomendadores basados en contenido y su aplicación a repositorios de objetos de aprendizaje. *IEEE-RITA*, 5(1):31–38.

- [Schafer et al., 1999] Schafer, J. B., Konstan, J., and Riedi, J. (1999). Recommender systems in e-commerce. In *Proceedings of the 1st ACM conference on Electronic commerce*, pages 158–166. ACM.
- [Verbert et al., 2012] Verbert, K., Manouselis, N., Ochoa, X., Wolpers, M., Drachsler, H., Bosnic, I., and Duval, E. (2012). Context-aware recommender systems for learning: A survey and future challenges. *Learning Technologies, IEEE Transactions on*, 5(4):318–335.
- [Wargo, 2012] Wargo, J. M. (2012). *PhoneGap Essentials: Building Cross-platform Mobile Apps*. Addison-Wesley Professional.
- [Woerndl and Schlichter, 2007] Woerndl, W. and Schlichter, J. (2007). Introducing context into recommender systems. In *Proceedings of AAAI Workshop on Recommender Systems in E-Commerce*, pages 138–140.