



TESINA DE GRADO

Universidad Nacional de La Plata

Título: Aplicaciones de Hipermedia Física en el ámbito académico

Autor: Tomás E. Córdoba

Director: Gustavo Rossi

Codirector: Javier Díaz

Carrera: Licenciatura en Sistemas

Resumen

En primera instancia se proveerá la funcionalidad de una plataforma de hipermedia física con objetivo de aplicarla en un ámbito académico. Se desarrollará como aplicación Web y se utilizará como metodología de sensado los códigos de barra quick response, analizando el rendimiento de estos para este tipo de sistemas.

La segunda contribución, es integrar el concepto de aplicación de hipermedia física con las metodologías de comunicación que utilizan las redes sociales actuales.

Líneas de Investigación

Investigación aplicada.

Trabajos realizados

Estudio y desarrollo de plataformas de hipermedia física.

Conclusiones

Se desarrolló la funcionalidad de una aplicación de hipermedia física; esto dio como resultado el modelo de implementación para una plataforma de hipermedia física sobre Web, logrando una clara separación entre el modelo de navegación y los contenidos a visualizar.

Se logró determinar el rendimiento de las distintas tecnologías disponibles para generar este tipo de plataformas.

Trabajos futuros

Incorporar caminos para recorrer, como lista de ubicaciones a visitar.

Realizar los cambios necesarios para que la aplicación sea cien por ciento accesible.

Realizar múltiples pruebas en diferentes sistemas operativos móviles para garantizar un funcionamiento homogéneo en todos los navegadores.

Fecha de presentación: Octubre de 2010

Índice general

Capítulo 1. Introducción	
1.1 Introducción a la tesis	3
1.2 Contribuciones	4
Capítulo 2. Introducción al tema	
2.1 Hipermedia	5
2.2 Hipermedia física	5
Capítulo 3. Redes sociales como métodos de comunicación	
3.1 Actualidad	10
3.2 Comunicación en las redes sociales	11
3.3 Hipermedia y redes sociales	12
Capítulo 4. Desarrollo de una plataforma de hipermedia física	
4.1 Estructura	13
4.2 Teoría de sensado	14
4.3 Administrador de ubicaciones	16
4.4 Aplicación para dispositivos móviles	17
4.5 Teoría de grafos para búsqueda de caminos entre ubicaciones	18
4.6 Definición de las ubicaciones en el mundo real	19
4.7 Actualidad de aplicaciones para hipermedia física	20
Capítulo 5. Realización de la plataforma de hipermedia	
5.1 Google Maps como metodología de ubicación geográfica	21
5.2 QR Codes como metodología de identificación de lugares	28
5.3 Plataforma hipermedia sobre Web	31
5.3.1 Tecnologías utilizadas en el caso de prueba	32
5.3.2 Descripción del caso de prueba	37
5.4 Desarrollo en un ámbito académico	57
5.5 Otros contextos de aplicabilidad	58
5.6 Actualidad Web, browsers y accesibilidad	60
Capítulo 6. Conclusiones	
6.1 Aplicabilidad de la plataforma de hipermedia física	68
6.2 Códigos QR como metodología de sensado	69
6.3 Desarrollo del caso de prueba Croover	69
Capítulo 7. Bibliografía	71

Capítulo 1. Introducción

1.1 Introducción a la tesis

El crecimiento exponencial de los dispositivos móviles se debe, en gran parte, a la necesidad de las personas de estar comunicados en todo momento. Las aplicaciones y sistemas operativos que estos ejecutan, deben acompañar este crecimiento implementando nuevas funcionalidades.

Esta evolución de los sistemas y las tecnologías han introducido, entre otras funciones importantes, el acceso a la Web en los dispositivos móviles. Esto permite hacer llegar más información a las manos de las personas, en el momento y en el lugar que lo necesitan.

El ambiente que rodea a la persona que está usando el dispositivo móvil influye en las inquietudes que motivan a la navegación. Esto nos obliga a considerar la gran importancia e influencia que tiene el mundo que nos rodea en la utilización de los dispositivos móviles. La incorporación del espacio físico en las aplicaciones de artefactos móviles, implica tener en cuenta la ubicación de la persona, para poder brindarle navegación e información acorde a lo que la persona está viendo. La necesidad de agregar información física a la Hipermedia Tradicional, que sólo abarca las relaciones entre los elementos digitales, resultó en la Hipermedia Física. Como resumen de los avances en esta área, cabe destacar que todo comienza con la combinación del ambiente real con el virtual, denominado "Realidad aumentada": en lugar de sumergir a las personas en un mundo virtual artificialmente creado, la meta es aumentar los objetos del mundo físico reforzándolos con una riqueza de información digital y capacidades de comunicación. [4]

El problema de "registrar" los objetos del mundo real y machearlos con la información electrónica correspondiente, es un área activa de investigación en la realidad aumentada. De este concepto surgen investigaciones como lo son Hiperreal [26] y Augmented Reality [18].

En la actualidad, las redes sociales produjeron cambios significativos en las metodologías de comunicación y en las interfaces, para operar con ellas por medio de los exploradores Web. La mayoría de las personas jóvenes, que se pueden encontrar en un contexto académico, hacen uso de este tipo de redes sociales.

La motivación de esta tesina esta impulsada por la necesidad de una plataforma de hipermedia física que permita mejorar la comunicación en un ámbito académico, analizando la integración entre una plataforma de hipermedia física y las estrategias de comunicación de una red social.

Una cuestión clave al diseñar aplicaciones sensibles al contexto es cómo modelar el contexto de una manera que pueda apoyar la evolución de la aplicación y la adición de nuevas características sin tener que rediseñar toda la aplicación [5]. Por esto, es muy importante la escalabilidad del sistema ante nuevos requerimientos o funcionalidades especiales que deba soportar la plataforma. Se debe considerar la utilización de frameworks MVC (Model-View-Controller). Una estructura MVC mantiene ordenado el código fuente, permitiendo un mantenimiento mucho más eficiente y la eficaz escalabilidad del sistema en un futuro. El uso de frameworks MVC para desarrollos ágiles como lo es Symfony, suelen ser una buena opción para desarrollar sistemas robustos.

En esta tesina se busca desarrollar un caso de prueba que permita analizar el rendimiento de una plataforma de hipermedia física implementada sobre la Web. Para determinar si los tiempos de espera de carga para los dispositivos móviles es tolerable; dado que los usuarios necesitan la información en tiempo real cuando se encuentran parados en frente de los objetos.

1.2 Contribuciones

Se busca ofrecer un análisis de la estructura que debería tener una plataforma de administración de ubicaciones y de una aplicación de navegación para los agentes móviles. De manera de facilitar las futuras implementaciones y detallar puntos específicos a tener en cuenta a la hora de definir los requerimientos.

Se realizará una prueba de conceptos que constará de la implementación de una plataforma de hipermedia física, analizando el uso de la misma en el ámbito académico y mostrando la aplicabilidad a otros contextos.

Se analizará el uso de códigos QR como técnica de sensado. Si bien hoy en día existen muchas tecnologías para este proceso, como por ejemplo bluetooth o procesamiento digital de imágenes, los códigos de barra QR son una forma muy cómoda y económica de identificar las ubicaciones.

Capítulo 2. Introducción al tema

2.1 Hipermedia

La hipermedia surge como resultado de la fusión de dos tecnologías, el hipertexto y la multimedia. El hipertexto es la organización de una determinada información en diferentes nodos, conectados entre sí a través de enlaces. Los nodos pueden contener sub-elementos con entidad propia. Un hiperdocumento estaría formado por un conjunto de nodos conectados y relacionados temática y estructuralmente. Por otro lado, se denomina tecnología multimedia a aquella que permite integrar diferentes medios en una misma presentación. [21]

La hipermedia, por tanto, es la tecnología que nos permite estructurar la información de una manera no-secuencial, a través de nodos interconectados por enlaces. La información presentada en estos nodos podrá integrar diferentes medios (texto, sonido, gráficos, etc.). Podemos decir entonces que el hipertexto provee una estructura de navegación a través de los datos textuales, mientras que el multimedia nos ofrece no sólo esta estructura de navegación a través de datos textuales, sino a través de una gran variedad de tipos de datos de diferentes morfologías. Si en hipertexto se habla de datos, en multimedia es corriente hablar de componentes. Los documentos multimedia constan de una colección completa de componentes cada uno de los cuales puede estar compuesto, a su vez, de otros componentes o bien de elementos de datos, llamados también entidades. [21]

Estos conceptos (hipermedia, hipertexto y multimedia) suelen ser confundidos entre sí, debido principalmente a su estrecha relación semántica. Por ello, es normal encontrar literatura en la que se utilice alguno de estos términos para referirse a cualquiera de los otros dos. [21]

A continuación listaremos algunas características típicas de la Hipermedia:

- Se puede utilizar la Hipermedia tanto para leer o escribir información.
- Su información incluye estructuras no secuenciales que pueden seguirse de diferentes maneras.
- Las estructuras de información deben seguir asociaciones naturales.
- La información debe ser estructurada jerárquicamente.
- Cada unidad de información debe ser presentada en una “pantalla diferente”.
- Debe ser posible compartir parte de la información entre diferentes usuarios.

La Hipermedia nos permite comunicarnos de una manera más efectiva, ya que es relacional y multimedial y, por lo tanto, es más cercana a nuestro modo habitual de expresión y pensamiento. [4]

La Hipermedia es un medio que utiliza y relaciona varias áreas del conocimiento humano tales como Ciencias de la comunicación, Ciencias cognitivas, Ergonomía y factores humanos, Sistemas, Informática, Psicología, y otros.

Algunas definiciones de Hipermedia son:

- “Hipermedia simplemente extiende la noción de texto en el Hipertexto incluyendo información visual, sonora y otros tipos de datos...”. [23]

- La *Hiper-media* son presentaciones que se ramifican, en respuesta a las acciones del usuario, o sistemas de palabras o imágenes preorganizadas que pueden explorarse libremente o consultarse a través de estilos específicos. Ellas no serán “programadas” sino diseñadas, escritas, dibujadas y editadas... Como la prosa y cuadros comunes serán *media* y debido a que en algún modo también son “multi-dimensionales” podemos llamarlas *hiper-media*, siguiendo el uso matemático de la palabra “hiper-”(...).[\[23\]](#)
- “Hipermedia es una extensión de la idea de *Hipertexto* que incorpora otros componentes tales como: video, ilustraciones, diagramas, voz y animación, así como imágenes generadas por computadora. Generalmente un autor crea las *ligas o links* entre los distintos medios; texto, gráficos, diagramas, fotografías, video, música, películas u otros medios (...)”. [\[15\]](#)
- Hipermedia es “una extensión de Hipertexto, un concepto que designa narrativa altamente interconectada o información vinculada”. [\[24\]](#)

Se pueden mencionar las siguientes ventajas de la Hipermedia:

- mayor facilidad de acceso a datos esparcidos, con menor retardo temporal,
- puede enlazar múltiples datos en red,
- menos espacio físico de almacenamiento,
- puede compartirse por más de un usuario,
- lectura orientada al usuario,
- en potencia podría reunir toda la literatura universal,
- estructura/relacional semántica de los datos orientada al usuario,
- utiliza en una sola estructura datos de diversa índole: texto libre (datos no estructurados), redes semánticas (semiestructurados) y tablas (datos estructurados),
- mínimos requerimientos de destrezas de programación, para construir complejas estructuras.

Y con respecto a las desventajas lo siguiente:

- supone aproximadamente un treinta por ciento de retardo en la lectura,
- menor transportabilidad,
- es necesario cierto aprendizaje de manejo de computadoras,
- no existe una única interfaz estándar,
- posibilidad de estructura en spaghetti,
- no hay una definición central de la estructura de datos, ni un camino sencillo para realizar acciones generales o cálculos específicos sobre los datos. [\[3\]](#)

2.2 Hipermedia física

El paradigma de Hipermedia Física (PH) se diferencia del paradigma de hipermedia permitiendo la navegación a través de objetos físicos y digitales en una red hipermedia. En PH, los objetos físicos se han mejorado con la información digital, permitiendo al usuario móvil conseguir esa información a través de un navegador Web mientras explora el mundo real (físico). En de estos sistemas se supone una red de sensores mínima, de forma que el sistema es notificado cuando el usuario se encuentra frente a un objeto digital mejorado. Como respuesta a tal notificación, el sistema presenta al usuario información sobre ese objeto físico y un conjunto de

enlaces (por ejemplo en forma de URL) a otros objetos. Estos vínculos pueden ser físicos o digitales, cada uno con su semántica de navegación. Los enlaces digitales se comportan al igual que los vínculos de hipermedia convencionales y por lo tanto la navegación es atómica (es decir, el acto de clic en un vínculo mediante la expedición de una petición http y navegar a una página Web sin pasos intermedios). Por otra parte, al hacer clic en un enlace físico expresa la intención del usuario de caminar físicamente del objeto físico actual al vínculo de destino. Sin embargo, no hay garantía de que el usuario efectivamente complete esta tarea, lo que implica tomar la decisión de navegar y luego "caminar en el enlace" hacia el objeto de destino. Sólo cuando el usuario llega a la meta, podemos decir que ha efectivamente navegado el enlace físico. [6]

El principal problema en los sistemas PH es que la navegación física no es atómica, dado que una gran parte depende de la actividad y la voluntad del usuario. Él puede perderse en su camino, se puede ver distraído por otros lugares de interés y dar un rodeo, o incluso cancelar el viaje al objetivo inicial y elegir un nuevo destino. [6]

Es de suma importancia asegurar la modularidad del sistema. A pesar de que esto debería ser una regla de oro para cualquier desarrollo de software, una arquitectura escalable es crucial en este caso, ya que se espera agregar nuevas funcionalidades y mejoras constantes. Como una ejemplo, considerar la adición de nuevas características de contexto (por ejemplo, los medios de transporte que el usuario utiliza para recorrer un enlace físico), añadiendo nuevos tipos o comportamientos a los objetos físicos, como por ejemplo una alerta o servicio de alerta cuando suceda algún evento inesperado (por ejemplo, un camino se encuentra bloqueado debido a un accidente).

Como ya se comentó en la introducción, en los últimos años se ha intentado la integración entre el mundo real y el mundo físico real, teniendo como objetivo aumentar los objetos del mundo físico reforzándolos con información digital. La información a menudo aparece como demasiado desconectada del mundo físico. Por lo general utilizamos los objetos físicos en un ámbito, y la información electrónica relacionada en otro totalmente separado.

El paradigma de PH permite construir relaciones significativas entre el mundo real y digital usando el paradigma de hipermedia. El término hipermedia física ha sido utilizado para describir cualquier sistema de hipermedia que trata con las propiedades del mundo físico.

Se puede aumentar la realidad a partir de estas tres estrategias:

- Aumentar al usuario: El usuario lleva un dispositivo, normalmente en las manos o en la cabeza, para obtener información sobre los objetos físicos. Ejemplos de esto pueden ser dispositivos del estilo Palms que muestren información o también cascos de realidad virtual.
- Aumentar los objetos físicos: En este caso los objetos físicos cambian según lo que esté delante de ellos. Generalmente implica agregar sistemas de sensores o GPS, y un ejemplo de las aplicaciones que se relaciona con esto son aquellas que implican posicionamiento.
- Aumentar el ambiente que rodea al usuario y a los objetos: Ni el usuario, ni el objeto son directamente afectados. En cambio, los dispositivos proporcionan y coleccionan información del ambiente circundante, desplegando la información hacia los objetos y capturando la información sobre las interacciones del usuario con ellos.

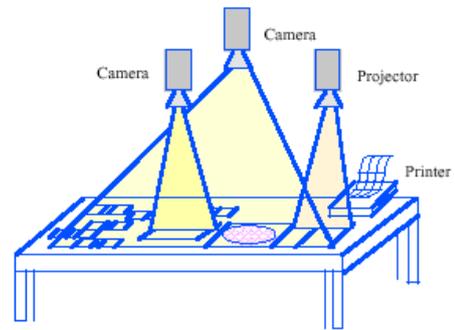
Ejemplo de estos son cámaras que capturan las actividades que realiza una persona y dependiendo de ello generan cambios en algún dispositivo.[4]



Aumentar al usuario [2]



Aumentar los objetos físicos [9]



Aumentar el ambiente que rodea al usuario y a los objetos [18]

Figura 2.1 – Ejemplos de realidad aumentada

La realidad aumentada (Augmented Reality) es una variación de los ambientes virtuales (Virtual Environments), o realidad virtual como se llama más comúnmente. Las tecnologías de ambiente virtual sumergen totalmente al usuario dentro de un ambiente sintético. Mientras que está sumergido, el usuario no puede ver el mundo real que lo rodea. En contraste, la Realidad Aumentada permite que el usuario vea el mundo real, con los objetos virtuales sobrepuestos sobre o compuestos con el mundo verdadero. Por lo tanto suplementa la realidad, en vez de sustituirla completamente. [18]

En contraste con la búsqueda Web tradicional, los sistemas basados en ubicación permiten a los usuarios buscar información en áreas geográficas específicas. Esto se realiza mediante un mapa donde el usuario puede indicar el área de interés y entonces puede ejecutar la búsqueda. Cuando los usuarios se mueven a través del espacio físico, el sistema debe ordenar y ver la información de su alrededor, y sólo mostrar información cuando alcanza un cierto umbral.

Existen también otros sistemas de información basados en ubicación, donde la idea básica es conectar partes de información digital a una coordenada de latitud-longitud específica vía algún dispositivo móvil. Luego, los usuarios con algún dispositivo móvil, acceden a esta información. De esta manera, los usuarios conseguirán la impresión de que la información digital realmente se halla en cierto modo en ese lugar.

La Mixed Reality fue definida por Paul Milgram como la "combinación de mundos reales y virtuales en alguna parte a lo largo de la 'serie continua de la virtualidad' que conecta ambientes totalmente verdaderos con los totalmente virtuales". [20]

La hipermedia se basa y se implementa bajo el paradigma orientado a objetos, permitiendo construir aplicaciones de *Mixed Reality*. La hipermedia juega un papel central en este tipo de aplicación como una manera de integrar los diferentes medios de comunicación y estructurar la información que se manipula por las aplicaciones. También constituye una forma natural de guardar las asociaciones entre los diferentes materiales en una exploración del mundo real.

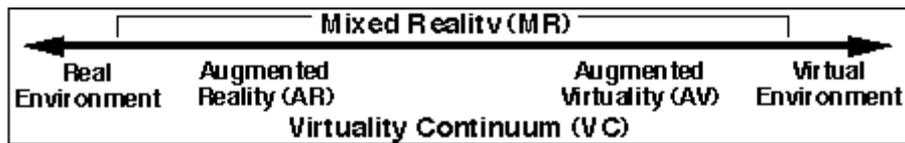


Figura 2.2: Explicación de Paul Milgram sobre MR [20].

El modo de navegación difiere entre la hipermedia tradicional y la hipermedia física. Si nos referimos al momento de activación, queda claro que en la hipermedia tradicional esto ocurre al navegar un link que nos lleva a ese nodo. En cambio en la hipermedia física esto ocurre de manera implícita cuando el usuario se encuentra parado frente a un objeto físico determinado. Es decir por algún mecanismo de detección de ubicación, se determina que el usuario está parado frente a un “objeto aumentado” y se le proporciona cierta información. Aquí, dependiendo de la naturaleza de la aplicación deben analizarse algunas otras cuestiones. Por ejemplo, cómo será el mecanismo de feedback para la notificación de la llegada del usuario a un nuevo objeto aumentado; qué información proveerle en ese momento, si se mantiene o no información de contexto sobre los lugares ya visitados, etc.

Siguiendo esta última idea es importante aclarar que en los links de hipermedia tradicional, el link contiene información que especifica el objeto fuente, que a menudo es, por ejemplo una URI. Cuando se permite a los links tener “anclas físicas” (es decir, que lo que denota al link es un objeto físico), su resolución puede volverse mucho más compleja. Por ejemplo: podrían usarse sistemas de ubicación para descubrir una persona frente a un “ancla física”, o una interfaz explícita de activación (tal como un lector de barras) que sea activada en la localización física del ancla (al ser explorada); o el ambiente del usuario podría ser implícitamente supervisado y luego, analizando dicha información, comparar contra marcadores simbólicos.

Dentro de los ambientes físicos, los links físicos deben indicarse vía alguna forma de señal. Es decir, el usuario tendría que poder percibir que está frente a un objeto que puede brindarle más información de lo que simplemente está observando. Las señales podrían ser direccionadas digitalmente, como por ejemplo con pantallas para la representación visual o altavoces para representación sonora que informen al usuario. Una estrategia alternativa es proporcionar una representación digital del mundo físico (tal como un mapa), en el cual los links físicos pueden ser colocados de forma digital. También se podría usar Realidad Virtual, o Realidad Aumentada. Uno de los problemas que aparecen aquí es que la notificación de la presencia de un ancla desvía el foco de la tarea de un usuario. Una modalidad orientada a la pregunta-explicita aparece así más atractiva que la exhibición universal, a menos que las anclas estén garantizadas por algún criterio, como podría ser la personalización de una determinada aplicación.

Cuando un link existe en el mundo digital, la idea de transversal, es a menudo bastante transparente para usuario, pero en el mundo físico, la carga puede estar en el usuario que tiene que realizar lo transversal. Por ejemplo, se podría tener un link en un monumento histórico a otro en la misma ciudad: una vez que el sistema tiene resuelto el ancla de la fuente, para seguir el link, el usuario debe moverse al destino.

Las aplicaciones sensibles al contexto se refieren a sistemas de software que puedan adaptar su comportamiento, interfaz y estructuras de acuerdo al contexto del usuario. La información sobre el contexto pueden ser puestos a disposición del sistema, ya sea explícitamente por el usuario (por ejemplo, introduciendo un nombre de usuario) o implícitamente asociados a través de los sistemas sensoriales. [12]

Queda claro entonces, que en las aplicaciones de Hipermedia Física los “objetos aumentados” deben caracterizarse bajo algún criterio que permita individualizarlos. Será este el que permita a la aplicación detectar cuando una persona se encuentra delante de ellos. Deberá hacerse hincapié en el comportamiento esperado de la aplicación cuando el usuario se posiciona frente a un objeto o cuando desea obtener información de cómo llegar a este desde la posición actual. [4]

Capítulo 3. Redes sociales como métodos de comunicación

3.1 Actualidad

Las redes sociales son un fenómeno que ha revolucionado las relaciones entre las personas en la sociedad actual. Una red social es una estructura social compuesta de objetos del mundo real (personas u organizaciones u otras entidades), las cuales están conectadas por uno o varios tipos de relaciones, tales como amistad, parentesco, intereses comunes, intercambios económicos, relaciones sexuales, o que comparten creencias, conocimiento o prestigio.

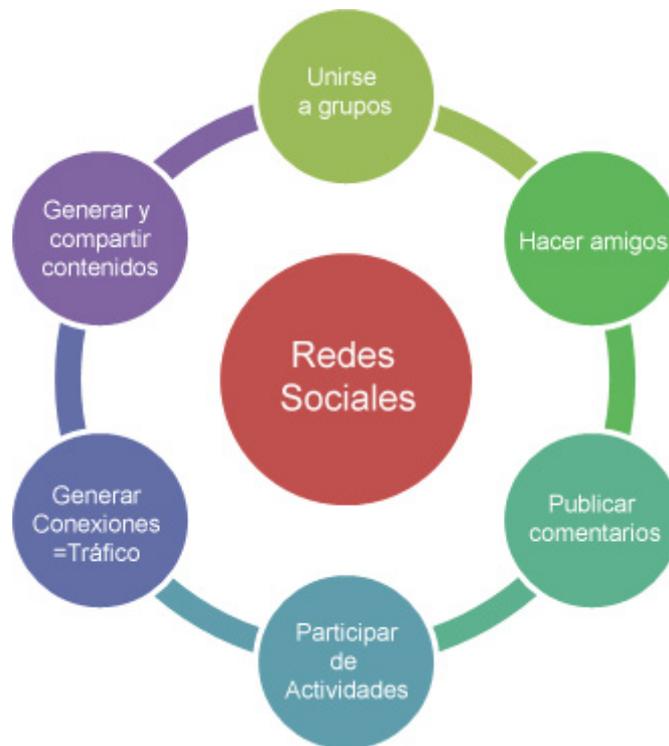


Figura 3.1 Ciclo de interacción en una red social

Una red social es una estructura social que se puede representar en forma de uno o varios grafos en el cual los nodos representan individuos (a veces denominados actores, o en nuestro caso, ubicaciones geográficas) y las aristas relaciones entre ellos. Las relaciones pueden ser de distinto tipo, como intercambios financieros, amistad, relaciones sexuales, o rutas aéreas. También es el medio de interacción de distintas personas como por ejemplo juegos en línea, chats, foros, spaces, etc.

Las redes sociales en Internet son agrupaciones que componen una comunidad o comunidades en torno a una o varias temáticas. Estas comunidades están conectadas por temas de interés común como hobbies, música, deportes, negocios etc.

Una red social dispone de una plataforma tecnológica que permite a los

usuarios interactuar, conversar y compartir en torno a temáticas de su interés. Estos espacios usualmente permiten comentar, postear artículos, subir fotos, videos y finalmente compartir todo esto con amigos dentro y fuera de la misma comunidad.

Las redes sociales pueden ser divididas en categorías. En la categoría temática se podría mencionar Flickr. Una red social para los interesados en la fotografía que se ha convertido en el lugar de encuentro para los aficionados a esta actividad. Además de subir sus fotos pueden comentar y participar de comunidades.

Otro caso exitoso es LinkedIn. Una red social en torno a la oferta laboral donde los usuarios pueden relacionarse, conectarse o consultar por profesionales de su mismo rubro u otros. También se ha posicionado como la plataforma de consulta al momento de querer establecer nuevos negocios y encontrar la gente idónea para esto. LinkedIn se ha convertido en el reemplazo virtual de la tarjeta de presentación. Otros ejemplos son las que agrupan aficiones como los deportes.

En la categoría general se agrupan todas las redes sociales masivas como Facebook, Orkut, Myspace o Hi5. No son necesarias de describir ya que la mayoría de nosotros las conocemos. Sin embargo es importante destacar que no todas son grandes a nivel global ya que, dependiendo del continente, podemos darnos cuenta que unas son más utilizadas que otras.

El análisis de redes sociales estudia esta estructura social aplicando la teoría de grafos e identificando las entidades como "nodos" o "vértices" y las relaciones como "enlaces" o "aristas". La estructura del grafo resultante es a menudo muy compleja. Como se ha dicho, puede haber muchos tipos de lazos entre los nodos. La investigación multidisciplinar ha mostrado que las redes sociales operan en muchos niveles, desde las relaciones de parentesco hasta las relaciones de organizaciones a nivel estatal (se habla en este caso de Redes políticas), desempeñando un papel crítico en la determinación de la agenda política y el grado en el cual los individuos o las organizaciones alcanzan sus objetivos o reciben influencias.

En su forma más simple, una red social es un mapa de todos los lazos relevantes entre todos los nodos estudiados. Se habla en este caso de redes "sociocéntricas" o "completas". Otra opción es identificar la red que envuelve a una persona u objeto (en los diferentes contextos sociales en los que interactúa); en este caso se habla de "red personal".

La red social también puede ser utilizada para medir el capital social (es decir, el valor que un individuo obtiene de los recursos accesibles a través de su red social). Estos conceptos se muestran, a menudo, en un diagrama donde los nodos son puntos y los lazos, líneas.

Actualmente no solo individuos hacen uso de las redes sociales, las organizaciones participan con distintos objetivos, como medir el capital social, calcular estadísticas en base a opinión de los clientes, captar nuevos mercados, mejorar la comunicación empresa-cliente y también para publicar nuevos lanzamientos y promociones. De modo que podemos decir que hoy las redes sociales son herramientas de comunicación, tanto personal como organizacional y no podemos imaginar una vida sin ellas.

Todas las redes sociales actuales ofrecen distintos métodos de comunicación para unificar nuestras aplicaciones automáticamente con ellas. Por ejemplo, Twitter, tiene una API para poder operar con su base de datos desde el "exterior". De esta

forma podemos publicar mensajes desde nuestra aplicación, editarlos, ver los seguidores del usuario, y los mensajes que este recibe. Facebook, en cambio, permite desarrollar aplicaciones, que mediante previa autorización del usuario al cual se le está permitiendo el acceso, permite editar los contenidos de la red social. Esta unificación es necesaria para hacer una plataforma robusta y que además sea aceptada por la gente, ya que nadie quiere andar publicando un mismo mensaje en todas las redes sociales a las que pertenece, es preferible hacerlo de forma automática.

```
protected function apiCall($twitter_method, $http_method, $format,
$options, $require_credentials = true)
{
    $curl_handle = curl_init();

    $api_url = sprintf('http://twitter.com/%s.%s', $twitter_method,
    $format);

    if (($http_method == 'get') && (count($options) > 0))
    {
        $api_url .= '?' . http_build_query($options);
    }

    curl_setopt($curl_handle, CURLOPT_URL, $api_url);

    if ($require_credentials)
    {
        curl_setopt($curl_handle, CURLOPT_USERPWD, $this->credentials);
    }

    if ($http_method == 'post')
    {
        curl_setopt($curl_handle, CURLOPT_POST, true);
        curl_setopt($curl_handle, CURLOPT_POSTFIELDS,
        http_build_query($options));
    }

    curl_setopt($curl_handle, CURLOPT_RETURNTRANSFER, TRUE);
    curl_setopt($curl_handle, CURLOPT_HTTPHEADER, array('Expect:'));
    $twitter_data = curl_exec($curl_handle);
    $this->http_status = curl_getinfo($curl_handle, CURLINFO_HTTP_CODE);
    $this->last_api_call = $api_url;
    curl_close($curl_handle);
    return $twitter_data;
}
```

Ejemplo de interacción con API de Twitter con PHP

Muchas de las APIs ofrecidas por las distintas redes sociales que existen hoy en día en la Web, utilizan un protocolo de comunicación denominado OAuth. Se trata de un protocolo abierto, propuesto por Blaine Cook y Chris Messina, que permite la autorización segura de una API de modo estándar y simple para aplicaciones de escritorio, móviles y Web.

Para los desarrolladores de consumidores, OAuth no es más que un método para interactuar con datos protegidos. Para desarrolladores de proveedores de servicio, OAuth proporciona a los usuarios un acceso a sus datos al mismo tiempo que

protege las credenciales de su cuenta. En otras palabras, OAuth permite a un usuario del sitio A compartir su información en el sitio A (proveedor de servicio) con el sitio B (llamado consumidor) sin compartir toda su identidad. Veamos un ejemplo:

Juan (propietario de los recursos) ha subido unas fotos de unas vacaciones privadas (recursos protegidos) a su sitio "photos.example.net" (servidor) para compartir fotos. A él le gusta usar el "printer.example.com" web (cliente) para imprimir una de estas fotos. Típicamente, Juan accede en "photos.example.net" usando su nombre de usuario y contraseña. [11]

Sin embargo, Juan no desea compartir su nombre de usuario y contraseña con la página web "printer.example.com", que necesita tener acceso a la foto para imprimirla. Con el fin de ofrecer a sus usuarios una mejora de servicios "printer.example.com" ha registrado a un conjunto de credenciales del cliente "photos.example.net" antes de tiempo:

```
Identificador de cliente
    dpf43f3p214k3103

Clave del Cliente secreta compartida
    kd94hf93k423kf44
```

La web "printer.example.com" también ha configurado su aplicación para utilizar los criterios de valoración que figuran en la documentación de la API del protocolo "photos.example.net", que utilizan el método de firma "HMAC-SHA1". [11]

```
Temporary Credential Request
    https://photos.example.net/initiate

Resource Owner Authorization URI:
    https://photos.example.net/authorize

Token Request URI:
    https://photos.example.net/token
```

Antes de que "printer.example.com" pueda pedir a Juan que le diera acceso a las fotos, primero debe establecer un conjunto de credenciales temporales con "photos.example.net" para identificar la solicitud de delegación. Para ello, el cliente envía el siguiente HTTPS [RFC2818] solicitud al servidor:

```
POST /initiate HTTP/1.1
Host: photos.example.net
Authorization: OAuth realm="Photos",
    oauth_consumer_key="dpf43f3p214k3103",
    oauth_signature_method="HMAC-SHA1",
    oauth_timestamp="137131200",
    oauth_nonce="wIjqoS",
    oauth_callback="http%3A%2F%2Fprinter.example.com%2Fready",
    oauth_signature="74KNZJeDHnMBp0EMJ9Zht%2FXKycU%3D"
```

El servidor valida la solicitud y responde con una serie de credenciales temporales en el cuerpo de la respuesta HTTP:

```
HTTP/1.1 200 OK
Content-Type: application/x-www-form-urlencoded
oauth_token=hh5s93j4hdidpola&oauth_token_secret=hdhd0244k9j7ao03&
oauth_callback_confirmed=true
```

El cliente redirecciona al agente de usuario de Juan al Resource Owner Authorization del servidor para obtener la aprobación de Juan para acceder a sus fotos privadas:

```
https: // / photos.example.net / autorice? oauth_token =
hh5s93j4hdidpola
```

El servidor solicita a Juan ingresar utilizando su nombre de usuario y contraseña y si tiene éxito, le pide que se apruebe la concesión de acceso para "printer.example.com" a sus fotos privadas. Juan aprueba la solicitud y su agente de usuario es redirigido a la devolución de llamada URI proporcionado por el cliente en la solicitud anterior:

```
http://printer.example.com/ready?
  oauth_token hh5s93j4hdidpola = & = oauth_verifier
hfdp7dh39dks9884
```

La solicitud de devolución de llamada informa al cliente que Juan ha completado el proceso de autorización. El cliente solicita una nueva serie de credenciales utilizando sus credenciales temporales (sobre un canal de transporte seguro, Layer Security (TLS)):

```
POST /token HTTP/1.1
Host: photos.example.net
Authorization: OAuth realm="Photos",
  oauth_consumer_key="dpf43f3p214k3103",
  oauth_token="hh5s93j4hdidpola",
  oauth_signature_method="HMAC-SHA1",
  oauth_timestamp="137131201",
  oauth_nonce="walatlh",
  oauth_verifier="hfdp7dh39dks9884",
  oauth_signature="gKgrFCywp7r00XSjdot%2FIHF7IU%3D"
```

El servidor valida la solicitud y responde con un conjunto de credenciales en el cuerpo de la respuesta HTTP:

```
HTTP/1.1 200 OK
Content-Type: application / x-www-form-urlencoded

  oauth_token nnch734d00sl2jdk = & = oauth_token_secret
pfkkdhi9sl3r4s00
```

Con este conjunto de tokens de credenciales, el cliente ya está listo para solicitar la foto privada:

```
GET /photos?file=vacation.jpg&size=original HTTP/1.1
Host: photos.example.net
Authorization: OAuth realm="Photos",
  oauth_consumer_key="dpf43f3p214k3103",
  oauth_token="nnch734d00sl2jdk",
  oauth_signature_method="HMAC-SHA1",
  oauth_timestamp="137131202",
  oauth_nonce="chapoH",
  oauth_signature="MdpQcU8iPSUjWoN%2FUdMsK2sui9I%3D"
```

El servidor "photos.example.net" valida la solicitud y responde con la foto solicitada. "printer.example.com" es capaz de continuar el acceso a fotos privadas de Juan con el mismo conjunto de tokens de credenciales mientras esté en vigencia la

autorización de Juan, o hasta que Juan revoque el acceso. [11]

3.2 Comunicación en las redes sociales

Las distintas metodologías que tienen las redes sociales para comunicar son objeto de estudio de muchos analistas, es por eso que, a la hora de crear nuevas plataformas o aplicaciones, es recomendable que se preste atención a lo que ya está en uso por millones de personas alrededor del mundo. Si bien la mayor parte de las redes sociales es utilizada por un público joven, existe un gran porcentaje de personas adultas que hacen uso de este tipo de aplicaciones. Si lo que deseamos implementar debe ser usado por jóvenes, no podemos ignorar las metodologías de comunicación que usan las redes actualmente y la disposición de los elementos para la navegabilidad. De esta manera los usuarios comprenderán mejor la aplicación si es lo que ellos esperan o lo que están acostumbrados a usar.

El concepto de mensaje ha sido desvirtuado a lo largo del tiempo, siendo utilizado para múltiples acciones, como mensajes cortos de teléfonos móviles, email, mensajes por medio de redes sociales, etc. Esta generalización del término “mensaje” permite hacer uso de mensajería sin necesidad de arriesgarse a implementar un nuevo concepto, que podría ser rechazado por los usuarios. De modo que cualquier forma de mensaje que desarrollemos, será entendida a la brevedad por más sofisticada que sea la interfaz de usuario.

Existen tantas metodologías de comunicación como redes sociales, pero vamos a analizar las características principales de las dos más utilizadas mundialmente Facebook y Twitter.

Twitter se basa en la idea de SMS, surgió como una utilidad de enviar mensajes a un grupo de amigos. En 140 caracteres uno tiene que publicar el ya conocido “¿Qué pasa?”, en el cual las personas cuentan lo que están haciendo en ese momento, lo que les paso hace instantes o simplemente compartir un link interesante o algo que se les cruzo por la mente. El mensaje aparece públicamente, esto quiere decir que cualquiera puede verlo. Existe entonces la posibilidad de administrar los mensajes que un usuario desea que se visualicen en la pantalla principal. Las personas agregan a las personas que les interesa escuchar y se convierten en “seguidores” de estas. Una vez que un usuario es seguidor de otro, el primero “escucha” todo lo que el segundo publica.

Facebook, en cambio, es más flexible. A la hora de realizar las publicaciones no posee un límite tan estricto de 140 caracteres como Twitter y las publicaciones pueden ser comentadas por los demás usuarios como así también marcarlas como “Me gusta”. Además de las funcionalidades extras que posee como publicación de videos, o álbum de fotos, en cambio Twitter sólo permite la publicación de un link externo para compartir este tipo de contenidos. Haciendo más practica la plataforma, sin sobrecargar el servidor con datos que ya se encuentran publicados en otro lugar de la Web.

3.3 Hipermedia y redes sociales

En la actualidad existen muchas redes sociales, la mayoría surgen por una necesidad específica de los usuarios, como por ejemplo compartir fotos, mensajes, etc. Pero ninguna de ellas está enfocada en ubicaciones físicas del mundo real, sino a personas específicas. Si bien muchos lugares (comercios, centros turísticos, organizaciones, etc.) tienen su página Facebook o usuario Twitter para estar en contacto con la sociedad y promocionar su organización, ninguna red social de hoy en día manifiesta un interés general por las ubicaciones del mundo real. Por lo tanto, las redes sociales actuales satisfacen las necesidades básicas de comunicación pero muchas veces se puede observar la falta de algún agregado para darle más utilidad a la red social.

El objetivo principal de las redes sociales es comunicar a las personas, todas satisfacen esta necesidad. Pero al tratarse de ubicaciones físicas suelen faltar algunas acciones que podrían ser de mucha ayuda para las personas que visitan estos lugares. Algunos de los agregados que suelen hacer falta en las redes sociales para las ubicaciones físicas son:

- La necesidad de saber en qué parte del mundo está la ubicación. Muchas aplicaciones lo solucionan con una pequeña foto de la localidad donde se encuentra, o un link externo a Google Maps, aunque no parece suficiente.
- Ubicaciones amigas, ubicaciones cercanas, y sucursales (en caso de que sean empresas). Como llegar hasta ellas (calcular caminos) y donde se encuentran.
- Encuestas, para saber que les parece a las personas una ubicación, pedir opinión pública, o simplemente para hacer alguna estadística.
- Estadísticas de visita a una ubicación dada. Saber cuántos usuarios navegan diariamente por la ubicación.
- Consejos respecto al lugar donde se encuentra el usuario.
- Distintas formas de llegar a la ubicación, en auto, bicicleta, o a pie.

Muchas de estas cuestiones se abordan en el caso de prueba denominado Croover que explicaremos en el capítulo 5, donde se aprecia la utilidad de estos agregados.

Capítulo 4. Desarrollo de una plataforma de hipermedia física

4.1 Estructura

Una plataforma de hipermedia física se basa en un simple concepto: “¿Dónde me encuentro parado?”. Es la simple necesidad de obtener información acerca del lugar donde me encuentro actualmente. Una vez que la persona accede a la información disponible para la ubicación donde se encuentra, se le puede recomendar otras ubicaciones similares, un mapa para ubicarse globalmente, otras ubicaciones cercanas, etc.

Una de las tareas a realizar cuando se desea diseñar una plataforma de hipermedia física, es establecer la navegabilidad de las ubicaciones. Si bien el usuario se encuentra parado frente a la ubicación y desea obtener información, se debe elegir la metodología de identificación o sensado del lugar. Si el usuario toma una fotografía panorámica del elemento o se detecta al usuario por acercamiento vía bluetooth o bien por un código de barras disponible en el lugar. Hablaremos sobre estos métodos en la próxima sección.

Describiendo a los usuarios de la plataforma notaremos que podemos distinguir claramente dos usuarios o personas que harán uso del sistema: el **agente móvil**, persona que se está moviendo y desea obtener información acerca de los lugares que va visitando, y el **administrador** que define las ubicaciones y los contenidos publicados en cada una. Por lo tanto una estructura simple podría definirse en dos partes: una parte administrativa para la definición de nuevas ubicaciones y publicación de contenido, y otra parte móvil, que es la aplicación que los usuarios móviles utilizan para acceder a los contenidos publicados en cada ubicación.

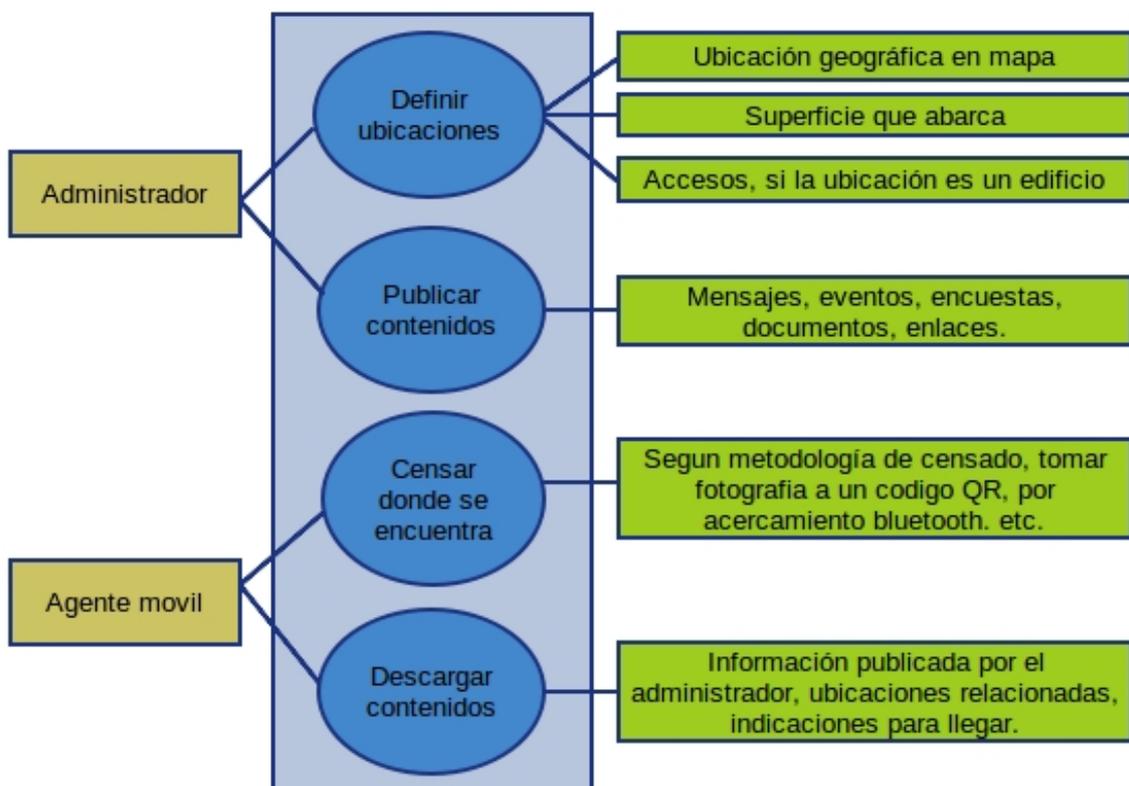


Figura 4.1.1. Simplificación de las actividades típicas de los usuarios de una plataforma de hipermedia física.

4.2 Teoría de sensado

Como vimos anteriormente el usuario debe relacionarse con la ubicación en la cual se encuentra parado. Para identificar dicha ubicación y establecer el enlace entre el objeto ante el cual la persona se encuentra parada y los contenidos almacenados en la base de datos se utilizan diferentes mecanismos de sensado. Dos de los más conocidos y nombrados suelen ser los dispositivos bluetooth y los códigos de barra.

En el caso de bluetooth el servicio es más eficiente ya que no necesita interacción del usuario para acceder al contenido, simplemente por acercamiento de la persona al lugar se decodifica donde se encuentra y se muestra el contenido correcto. En cambio, para las técnicas de códigos de barras la persona debe tomar la fotografía del mismo para poder acceder a la información. Lo cual puede producir retrasos, ya sea por mala lectura del código por deterioro del mismo, o simplemente por ausencia del código de barras que identifica la ubicación. El código de barra más utilizado es el "Quick Response" denominados también QR, en el próximo capítulo hablamos sobre sus características.



Figura 4.2.1 Esquema de funcionamiento para sensado por Bluetooth

Si observamos en detalle ambas metodologías podemos darnos cuenta que, en cuanto a aspectos de costos, resulta más económico implementar una plataforma con sensado por códigos QR ya que solo se necesita que la aplicación administradora de ubicaciones permita la impresión del código QR correspondiente para poner en la ubicación. Para el caso del sensado por Bluetooth se necesita de un dispositivo que

permita conectar con los dispositivos móviles en cada ubicación. Esto trae inconvenientes, si bien la tecnología Bluetooth es relativamente económica, se necesita un técnico que configure el dispositivo y quede listo para funcionar sin fallas.



Figura 4.2.2 Funcionamiento de sensado por códigos de barra quick response

Otra opción es por procesamiento de imágenes: a partir de una foto panorámica de la ubicación se analiza la imagen y se detectan los patrones que permitan identificar la ubicación de manera unívoca. O al menos se realiza una búsqueda lo más exacta posible del lugar donde se encuentra parada la persona. De esta forma ya no se necesita ningún tipo de accesorio anexo, como en el caso del código de barras y el dispositivo Bluetooth; solo con tomar una foto del contexto el sistema puede detectar dónde se encuentra el usuario. El costo que se debe pagar es el de mantener la base de datos con las fotos actualizadas, para evitar la no identificación, en el caso de que haya habido cambios en el panorama.

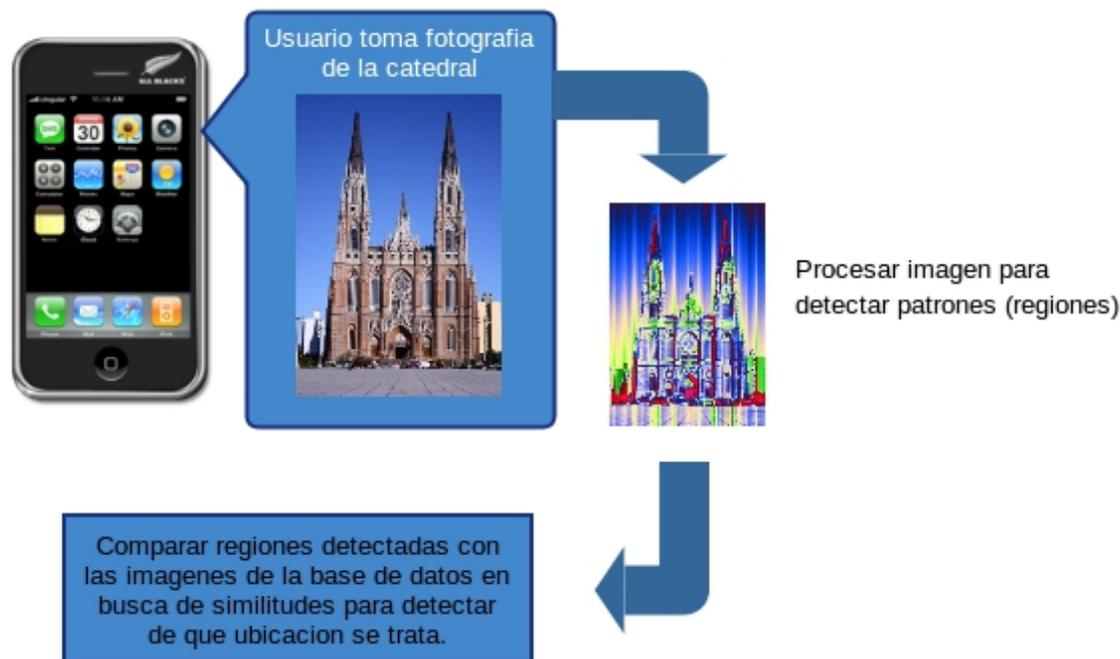


Figura 4.2.3 Esquema de funcionamiento para sensor por reconocimiento de imágenes

4.3 Administrador de ubicaciones

Un usuario administrador debe tener la posibilidad de administrar y gestionar los contenidos de todas sus ubicaciones. La administración de las ubicaciones por lo tanto es compleja, por un lado se debe definir qué tipo de contenidos se desean publicar, es decir, que información queremos que el usuario móvil pueda ver; teniendo en cuenta que el agente móvil tiene un dispositivo con tamaño de pantalla limitado y controles básicos de navegabilidad. Y por otro lado, la especificación geográfica de la ubicación, donde se encuentra esa ubicación en el mundo y como se puede llegar hasta ella.

Se le debe ofrecer a los usuarios la posibilidad de observar un mapa, esto puede darse con un simple imagen que represente donde se encuentra el usuario o utilizando algún sistema de posicionamiento geográfico o servicio de cartografía digital. Si se utiliza el método de la imagen creada por el usuario administrador, es muy simple y dependerá del usuario la creación del mapa manualmente. Esta posibilidad de navegar un mapa, detalla otros aspectos a tener en cuenta. Es importante definir cómo se va a realizar la identificación geográfica de las ubicaciones, punto de acceso, y también superficie que abarca dicha ubicación, en este caso se deben almacenar varios pares ordenados de latitudes y longitudes que representan los vértices del perímetro de la ubicación.

Como ya hemos dicho, el administrador de ubicaciones puede verse como la aplicación para crear, editar y borrar ubicaciones. Si bien en algunas instituciones puede ser una única persona la encargada de administrar las ubicaciones, también puede existir más de un administrador. Este concepto es incorporado por el análisis de redes sociales, donde múltiples usuarios definen ubicaciones y editan el contenido de las mismas. De esta forma todas las personas en el mundo somos administradores de

ubicaciones y agentes móviles al mismo tiempo.

4.4 Aplicación para dispositivos móviles

La cantidad de dispositivos móviles y la falta de estándares para crear aplicaciones hacen tedioso el desarrollo de aplicaciones. Pero existe algo en común hoy en día entre estos dispositivos, y es que todos incluyen un navegador Web (browser) propio. Esto simplifica las implementaciones, ya que podemos pensar en la aplicación móvil como una aplicación Web que se opera desde cualquier browser.

Los teléfonos móviles vienen con un sistema operativo instalado que les permite realizar diferentes funciones, incluidas la navegación Web por medio de exploradores móviles que adaptan los contenidos para que sean visualizados en pantallas de tamaño pequeño. Algunos de los sistemas operativos más conocidos son: Windows Mobile, Symbian, Android, iPhone OS y RIM. Algunos de estos sistemas operativos solo permiten la descarga de algunos de los navegadores disponibles, no como las computadoras de escritorio que nos permiten descargar y utilizar cualquier navegador Web.

Los teléfonos celulares tienen una pequeña resolución de pantalla en comparación a una computadora de escritorio. Los navegadores móviles optimizan el contenido Web para que sea más sencillo mostrarlo en la resolución del display que poseen. Dependiendo del contenido de una página Web la carga en un dispositivo móvil suele tomar más tiempo que en un explorador de PC.

La mayoría de los navegadores móviles soportan los lenguajes comúnmente usados HTML, CSS y JavaScript, aunque no todos soportan la reproducción de contenidos multimediales de tipo Flash. Aunque con la llegada de HTML5 ya no es problema, dado que la última versión soporta visualización de video sin la necesidad de tener ningún tipo de plugin instalado.

Si bien existe gran variedad de navegadores Web móviles, los más conocidos son:

- Opera Mini
- Skyfire Safari
- Google Android
- Microsoft IE for Mobile
- Firefox Mobile
- Bolt
- Teashark
- Blazer

De este modo, el mayor problema que se puede presentar a la hora de desarrollar una plataforma de hipermedia móvil basada en Web para agentes móviles, es lograr que todos los estilos CSS se vean iguales en todos los exploradores. Pero este es un problema ya muy conocido para navegadores desktop también.

Para la parte móvil se debe poner mucho énfasis en la disposición de los elementos que componen la aplicación, debe ser fácil de usar y se deben aprovechar los espacios de manera eficiente, ya que las pantallas de los dispositivos móviles son medianamente pequeñas.

4.5 Teoría de grafos para búsqueda de caminos entre ubicaciones

A medida que la plataforma de hipermedia física va creciendo, las ubicaciones se van relacionando unas con otras y termina siendo una gran red de nodos interconectados, lo que ya conocemos como “grafo”.

Los usuarios móviles necesitan moverse ubicación a ubicación con algún tipo de guía, que los ayude a desplazarse por el mundo. Por lo tanto, el cálculo de caminos está relacionado con la denominada teoría de grafos.

Un grafo es un conjunto, no vacío, de objetos llamados vértices (o nodos) y una selección de pares de vértices, llamados aristas (*edges* en inglés) que pueden ser orientados o no. Típicamente, un grafo se representa mediante una serie de puntos (los vértices) conectados por líneas (las aristas). Existen diferentes formas de almacenar grafos en una computadora. La estructura de datos usada depende de las características del grafo y el algoritmo usado para manipularlo. Entre las estructuras más sencillas y usadas se encuentran las listas y las matrices, aunque frecuentemente se usa una combinación de ambas. Las listas son preferidas en grafos dispersos porque tienen un eficiente uso de la memoria. Por otro lado, las matrices proveen acceso rápido, pero pueden consumir grandes cantidades de memoria.

- **Lista de incidencia** - Las aristas son representadas con un vector de pares (ordenados, si el grafo es dirigido), donde cada par representa una de las aristas.

- **Lista de adyacencia** - Cada vértice tiene una lista de vértices los cuales son adyacentes a él. Esto causa redundancia en un grafo no dirigido (ya que A existe en la lista de adyacencia de B y viceversa), pero las búsquedas son más rápidas, al costo de almacenamiento extra.

En esta estructura de datos la idea es asociar a cada vértice i del grafo una lista que contenga todos aquellos vértices j que sean adyacentes a él. De esta forma sólo reservará memoria para los arcos adyacentes a i y no para todos los posibles arcos que pudieran tener como origen i . El grafo, por tanto, se representa por medio de un vector de n componentes (si $|V|=n$) donde cada componente va a ser una lista de adyacencia correspondiente a cada uno de los vértices del grafo. Cada elemento de la lista consta de un campo indicando el vértice adyacente. En caso de que el grafo sea etiquetado, habrá que añadir un segundo campo para mostrar el valor de la etiqueta.

También se puede definir un grafo dentro de una estructura matricial:

- **Matriz de incidencia** - El grafo está representado por una matriz de A (aristas) por V (vértices), donde [arista, vértice] contiene la información de la arista (1 - conectado, 0 - no conectado)

- **Matriz de adyacencia** - El grafo está representado por una matriz cuadrada M de tamaño n^2 , donde n es el número de vértices. Si hay una arista entre un vértice x y un vértice y , entonces el elemento $m_{x,y}$ es 1, de lo contrario, es 0.

Los vértices constituyen uno de los dos elementos que forman un grafo. Como ocurre con el resto de las ramas de las matemáticas, a la Teoría de Grafos no le interesa saber *qué* son los vértices. En nuestro caso se asocian directamente a

ubicaciones físicas del mundo real.

Por lo tanto, para el cálculo de caminos entre ubicaciones se puede utilizar cualquier algoritmo de búsqueda en grafos. Algunos de los más conocidos son:

- Algoritmo de búsqueda en anchura (BFS)
- Algoritmo de búsqueda en profundidad (DFS)
- Algoritmo del vecino más cercano
- Algoritmo de Dijkstra

Ahora bien, como habíamos mencionado, los vértices de nuestro grafo son nuestras ubicaciones. ¿Qué pasa si queremos calcular el camino entre una ubicación y otra? Cuando calculamos caminos punto a punto entre ubicaciones debemos tener en cuenta que la persona no quiere ir caminando por sobre los edificios de una ciudad o atravesar paredes dentro de un edificio. Por lo cual la plataforma debe proveer una manera de definir los caminos entre ellas. O al menos una manera de llegar a la ubicación si esta se encuentra dentro de un edificio, si hay que subir escaleras/ascensores, pasillos, puertas que atravesar, etc.

Google Maps ofrece hoy lo que ellos denominan Driving Directions que permite calcular el cómo llegar por medio de calles de una ubicación a otra. Es de gran utilidad para no tener que definir todo el procesamiento de grafos y demás cálculos de caminos. Lo único que no está incluido es el cálculo de caminos dentro de los edificios, si se tratase de una oficina o aula, como llegar hasta ella.

4.6 Definición de las ubicaciones en el mundo real

Las ubicaciones de una plataforma física deben estar relacionadas a su posición geográfica real. Cuando hablamos de posición geográfica nos referimos obviamente a su latitud y longitud en el planeta tierra. Hoy en día es común hablar de GPS y el servicio de cartografía online de Google: Google Maps. Ambas son buenas elecciones a la hora de definir la posición física real de un objeto en el mundo. Pero debemos tener en cuenta que para el caso de GPS se necesita de una interface que permita cargar automáticamente una posición en el mapa, de lo contrario sería muy engorroso para el usuario especificar a mano la latitud y la longitud del lugar. Y para el caso de Google Maps necesitamos una conexión a Internet si quisiéramos tener más imagen de la que queda cacheada en el explorador.

Toda ubicación ocupa una determinada superficie, por más mínima que sea, entonces ya no hablamos de un único punto de ubicación sino de múltiples puntos que definen su superficie y una puerta de entrada, si fuese un edificio. Por lo tanto debemos mantener un conjunto de puntos que unidos definen la superficie de una ubicación y un punto de acceso. Esto nos determina parte del modelo de una plataforma de hipermedia física, ya que deberíamos almacenar en una tabla los pares latitud y longitud de cada punto que unidos determinan la superficie de la ubicación.

Debemos definir entonces que tan completa es la plataforma que intentamos implementar, si necesitamos de la superficie como un dato obligatorio o si podemos prescindir de ella. Si se va a utilizar una tecnología cartográfica del tipo Google Maps o si se va a contar con interfaces para GPS.

4.7 Actualidad de aplicaciones para hipermedia física

La hipermedia física es un concepto en crecimiento constante, siempre se agregan nuevos servicios a los ya existentes, por ejemplo Google Maps provee localización de lugares y cálculo de caminos de un lugar a otro sin problemas. Pero no existen aplicaciones que implementen concretamente el concepto de hipermedia física, o hipermedia móvil como ya hemos nombrado. Por ejemplo, no se tiene en cuenta quién es la persona que se está moviendo entre los objetos físicos, para recomendar otras ubicaciones que visitar. Es decir no es lo mismo una persona que está de turista recorriendo una ciudad que un historiador que se encuentra recorriendo museos. Ese tipo de concepto de identificar a la persona y según su perfil guiarlo no está desarrollado en las aplicaciones que actualmente existen en el mercado. También es importante tener en cuenta desde dónde viene el usuario para poder analizar y encontrar un patrón sobre las ubicaciones que visita y de esta manera ofrecer una guía más completa.

Otro concepto faltante en las aplicaciones de la actualidad es el de relacionar una ubicación con otra, ubicaciones amigas, lo cual es de gran ayuda a la hora de recomendar otros lugares para visitar. Por ejemplo para los usuarios que tienen empresas con distintas sucursales y desean relacionar unas a otras. Por otra parte también está la definición de superficie que abarca una ubicación, que no está contemplado en ninguna aplicación actual. Lo más cercano a esa definición es lo que introdujo Google con edificaciones 3D. Que permiten dibujar un edificio de cualquier tamaño y aspecto, pero que no contempla ubicaciones específicas dentro de ellos, como oficinas o aulas, aunque seguramente no se demoren mucho en agregarlo.

Estos agregados que hacen a una plataforma de hipermedia física completa son el motivo de que hoy no exista una aplicación lo suficientemente robusta como para soportar el concepto mismo de hipermedia física. Si bien como hemos nombrado Google, está a un paso de convertir Google Maps en una verdadera herramienta de hipermedia física.

A modo de ejemplo podemos citar el sistema Topos que se desarrolló en el proyecto de WorkSPACE. Su objetivo fue desarrollar un sistema de Hipermedia Física para dar a los usuarios, un ambiente familiar al mundo real en el que trabajan. Organizando las mezclas entre materiales digitales y físicos, mediante colecciones de objetos. El concepto central en Topos es el *workspace* que es el medio principal para agrupar y organizar los materiales y objetos en el espacio 3D. Otro aspecto interesante que provee es que permite agrupar, mezclar y conectar los espacios de trabajo de varias maneras diferentes. [4]

Otro proyecto interesante sobre Hipermedia Física es el framework HyCon para sistemas de Hipermedia context-aware. Su arquitectura da soporte para realizar anotaciones, manejo de links y tours guiados asociando ubicaciones y objetos con dispositivos RFID o Bluetooth con mapas, páginas Web y colecciones de recursos. Introduce el uso de XLinks y un nuevo sistema de búsqueda para la Web llamado Geo-Based Search (GBS). Lo interesante de este sistema es que soporta los mecanismos clásicos de Hipermedia para navegar, buscar, hacer notaciones y tours guiados en el mundo físico, permitiendo a los usuarios realmente unir objetos digitales y/o físicos. Cabe destacar también que esta arquitectura propuesta, incluye interfaces para incluir la capa de sensores, que encapsulan el GPS y otros sistemas de posicionamiento que se pueden utilizar. [4]

El sistema GeoNotes intenta unir áreas bastante dispares de investigación dentro de la interacción hombre-máquina. Intenta borrar el límite con el espacio físico y

al mismo tiempo reforzar el espacio digital socialmente.

Basado en tecnología de posicionamiento permite añadir notas virtuales a través de algún dispositivo móvil a ubicaciones del mundo real. Cuando otra persona pasa por este sitio le llegará a su dispositivo móvil una notificación informándole de la existencia de información, y luego la persona podrá leerla.

Los usuarios, de esta manera, podrán observar y participar en los modelos sociales del espacio geográfico. GeoNotes crea un awareness social que incentiva la expresión y formación de la identidad personal.

GeoNotes utiliza GPS y DGPS, esto es útil solo para áreas físicas muy grandes. El problema que tienen estas tecnologías es que no son exactas, poseen un margen de error en el cálculo de la posición y raramente se utilizan en edificios o ambientes interiores. [4]

HyCon es un framework que extiende el paradigma de Hipermedia con el mundo físico. Sirve para sistemas de Hipermedia context-aware. Lo interesante de este sistema es que soporta los mecanismos clásicos de Hipermedia para navegar, buscar, hacer anotaciones y tours guiados en el mundo físico, permitiendo a los usuarios realmente unir objetos digitales y/o físicos. Cabe destacar también que esta arquitectura propuesta, incluye interfaces para incluir la capa de sensores, que encapsulan el GPS y otros sistemas de posicionamiento que se pueden utilizar. Introduce el uso de XLinks y un nuevo sistema de búsqueda para la Web llamado Geo-Based Search (GBS).

Los objetivos de HyCon son: extender Hipermedia con el mundo físico, soportar colecciones automáticas de información contextual e información social, y funcionar con dispositivos móviles heterogéneos. [4]

HyperReal es un framework orientado a objetos que permite construir aplicaciones context-aware y mixed reality. Para lograr dicho framework se establece un modelo genérico que integra conceptos ya introducidos en esta área, basándose en los conceptos de Dexter establecidos para el Hipertexto. Además para incorporar los diferentes medios, utiliza los conceptos de Hipermedia adaptativa y espacial. Integra en el mismo framework documentos virtuales, ambientes 3D y el mundo físico para construir aplicaciones de mixed reality.

El modelo genérico especifica una base estructural para establecer la relación entre los espacios reales o virtuales y los soportes de mecanismos contextuales. La representación del espacio virtual y físico está integrada en el modelo, permitiendo modelar las relaciones entre los diferentes tipos de elementos que están ubicados en el espacio.

HyperReal puede manejar diferentes elementos, y define un esquema de presentación que abstrae los conceptos de navegación relevantes, incluyendo el awareness de los links. El awareness de los links provee una manera de representar la información de navegación. En los ambientes de mixed reality, esto es particularmente importante para tener este mecanismo en el nivel del modelo, dando un dinamismo natural a la representación del link. [4]

El proyecto proXimity es un sistema que busca aumentar la realidad dando al Hipertexto, una presencia física en el mundo verdadero. Está basado en trabajos

sobre Hipermedia y movilidad en el mundo real e intenta extender la metáfora del link de Hipermedia en el espacio físico.

En este proyecto surge el concepto de “walk the Links” o “*caminar los links*”. Este trabajo establece fuertes indicios para suponer que en el futuro, el Hipertexto y el mundo real van a unirse y permitirán que el usuario en efecto “camine los links”.

El principal aporte de proXimity es establecer tres componentes para el Hipertexto en la Web: un componente espacial, uno temporal y otro semántico. El componente semántico permitirá agrupar los links bajo algún criterio. El temporal dará la información de la ubicación tanto de la persona como de los objetos del mundo real, y el espacial resolverá las cuestiones de como atravesar las distancias.

ProXimity es eficaz en el uso de nuevas tecnologías sobre el conocimiento, ambientes y dominios del usuario (semántica Web, IR, Bluetooth, GPRS), esto lo convierte en un sistema único y diferente. [4]

TOPOS es un prototipo que permite la manipulación y mantenimiento de relaciones espaciales entre materiales en un ambiente tridimensional. Se integra con aplicaciones para soportar colaboración en tiempo real. Esta es una manera de aumentar la realidad multidimensionalmente.

El objetivo fue desarrollar un sistema de Hipermedia Física para dar a los usuarios, un ambiente familiar al mundo real en el que trabajan. Organizando las mezclas entre materiales digitales y físicos, mediante colecciones de objetos. El concepto central en Topos es el *workspace* que es el medio principal para agrupar y organizar los materiales y objetos en el espacio 3D. Otro aspecto interesante que provee es que permite agrupar, mezclar y conectar los espacios de trabajo de varias maneras diferentes.

Se puede utilizar tanto para sistemas de Hipermedia espacial abstractos con espacios abiertos o concretos usando por ejemplo un edificio como un “background” para la colocación relativa de materiales.

Se integra un sistema de tag RFID para soportar la creación, manipulación y mantenimiento de las relaciones en el ambiente interactivo. RFID se utiliza para registrar material físico y generar un ID digital.

Otro proyecto interesante, llevado a cabo en una ciudad de Dinamarca, es el denominado “*GAMA – ON THE TRAIL OF UNKNOWN LAND*” (en la ruta de la tierra desconocida). Este proyecto fue implementado como juego en la ciudad de Horsens durante todo el verano de 2008.

La obra consta de 14 escenas (etiquetas) y la duración es de 1 h. y 20 m. La historia es sobre un estudiante de medicina, cristiano, que está en camino al funeral de su hermana. En su camino a la iglesia, recibe una misteriosa llamada telefónica de una mujer que dice tener un mensaje de su hermana, Johanne. La mujer afirma ser Anna Christina Bering, la esposa muerta hace mucho tiempo de un famoso explorador, Vitus Bering, que era descendiente de Horsens. El usuario es el protagonista de una búsqueda del tesoro psicológica, y él siempre está provisto con hechos históricos de la ciudad a través de su búsqueda de mensajes de su difunta hermana. [17]

La estructura conceptual general sigue el modelo clásico de Hollywood, que es un curso lineal donde los hechos de drama aumentan hasta un clímax al final de la trama. El usuario necesita encontrar pistas, por ejemplo, viejas cartas en los cajones y

otros objetos en su búsqueda. En GAMA varias escenas tienen lugar en contextos cerrados, cubiertos, y por lo tanto hace un mejor uso de la entorno. [17]

Los lugares en GAMA se fijaron antes de que el drama fuese escrito. Esto significaba que las secuencias de audio independientes, fueron grabadas de acuerdo a la distancia aproximada a la siguiente ubicación, que realiza con menos pausas en la historia, y por lo tanto, no hay necesidad de llenar lugares vacíos. La duración del clip de audio más largo en GAMA fue de once minutos, y no fue demasiado tiempo, incluso para aquellos usuarios que caminan rápidamente. Además, el juego podría ser más específico en cuanto a las ubicaciones al tener determinados los lugares físicos antes del proceso de escritura. Esto hace que la experiencia del usuario sea más completa, sin embargo, más difícil para los escritores para componer con más restricciones que cuando se trata de dramas teatrales clásicos. [17]

Las experiencias generales y las lecciones aprendidas sobre los dramas urbanos incluyen no hacer el curso a través de la ciudad demasiado complejo. No debe ser demasiado difícil encontrar la manera del siguiente tag ya que esto causaría una mayor atención en el mapa cuando se navega a través de la ciudad, y menos énfasis en el drama y la propia ciudad. Además, las etiquetas debe ser cuidadosamente colocadas y protegidas. [17]

Klamm Dan es el Coordinador de Extensión y Marketing de la Universidad de Syracuse, él opina que con más estudiantes que vienen al campus universitario con Smartphones, es importante crear oportunidades para ellos en un medio que ellos entienden. La cultura móvil de los colegios es el escenario perfecto para los servicios basados en localización. Considere ir a los lugares donde se congregan los estudiantes y la capacitación. Si bien muchos estudiantes tienen la tecnología, no todos son conscientes de las aplicaciones que pueden utilizar en el campus, y más importante aún, qué hay en él para ellos. [16]

Las aplicaciones basadas en la localización geográfica no son sólo para promociones y descuentos. La geolocalización puede tener un efecto real en la educación a nivel universitario, ayudando a crear relaciones con los futuros estudiantes y sus familias, comprometiendo a los estudiantes con los materiales de su curso, y a fortalecer los lazos con antiguos alumnos. [16]

Las universidades están siempre buscando maneras de fortalecer los lazos dentro de sus comunidades y muchas instituciones de educación superior ya han aplicado planes de medios de comunicación social para ayudarles a llevar a cabo este fin. Los servicios de localización son el siguiente paso en la creación de relaciones significativas con los futuros estudiantes, los estudiantes actuales y ex-alumnos. [16]

A continuación se analizan algunas formas en que las universidades pueden aprovechar los servicios basados en la localización con consejos, asesoramiento, y algunas instituciones que ya lo han aplicado con éxito. [16]

Crear una experiencia de visita especial (Universidad de Harvard)

Los futuros estudiantes a menudo citan su gira del campus como un factor importante en la elección de las universidades, ¿por qué no hacer de su primera visita a la institución, una verdadera experiencia digital? La utilización de un servicio basado en la localización como el Foursquare (Cuadrangular), puede dar consejos en todo el campus para que el visitante tenga una experiencia rica y vibrante. [16]

Señalar tradiciones y hechos poco conocidos que están ligados a los lugares. Publicar información sobre edificios, monumentos y estatuas. Direcciones, preguntas más frecuentes acerca de la seguridad, peatones, y navegación en el campus. Para futuros estudiantes y sus padres, estos consejos rápidos son inmensamente útiles para aprender sobre su entorno y descubrir la cultura del campus, si está caminando con un grupo de visita guiada o explorando por su cuenta. [16]

Del mismo modo, las instituciones pueden dejar consejos que pertenecen a los ex alumnos. Como pequeños recuerdos que puedan ser refrescantes acerca de las cosas que han cambiado todo el campus desde su última visita. Al ver un edificio nuevo en el sitio de un antiguo campo de fútbol puede ser discordante, pero un consejo de Foresquare determinando cuándo se produjo el cambio puede proporcionar un contexto dentro del respeto de la historia del lugar. [16]

Harvard, en la vanguardia del uso de Foursquare entre las universidades, ofrece una mezcla de información histórica y cosas que hacer en todo el campus en su página de Foursquare. [16]

Foursquare consiste en un sitio web donde los usuarios pueden registrarse y ver los distintos spots de la ciudad y luego recibir consejos y detalles sobre cada ubicación. Como usuario Foursquare, el valor es inmediatamente evidente: leer sobre algo que quieres hacer en la web, un click para guardar la elección y luego tener acceso a esta actividad o lugar dentro de la aplicación móvil. Además, cuando usted está usando la aplicación, Foursquare le alerta acerca de ubicaciones momentáneamente cercanas a donde el usuario se encuentra y que había guardado en la lista TO-DO (En la figura 4.7.2 se puede observar la pantalla TO-DO list de la aplicación móvil de Foursquare). [16]

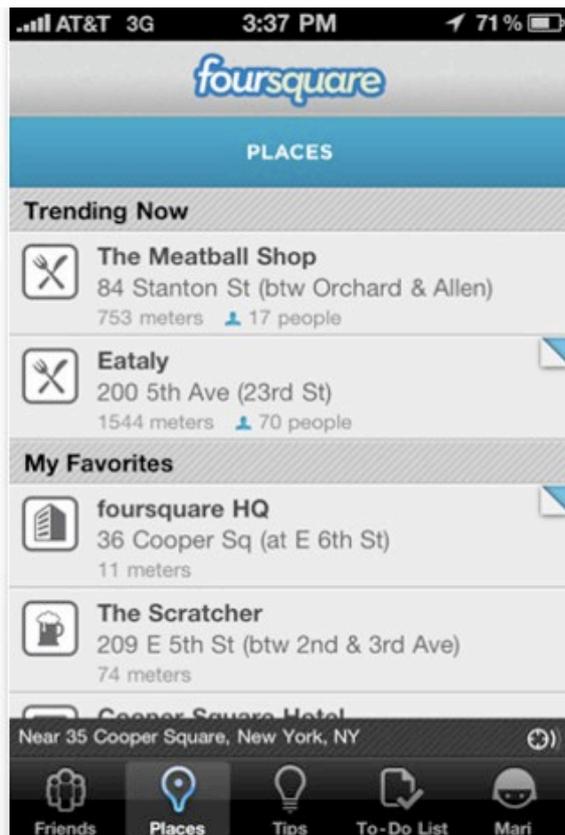
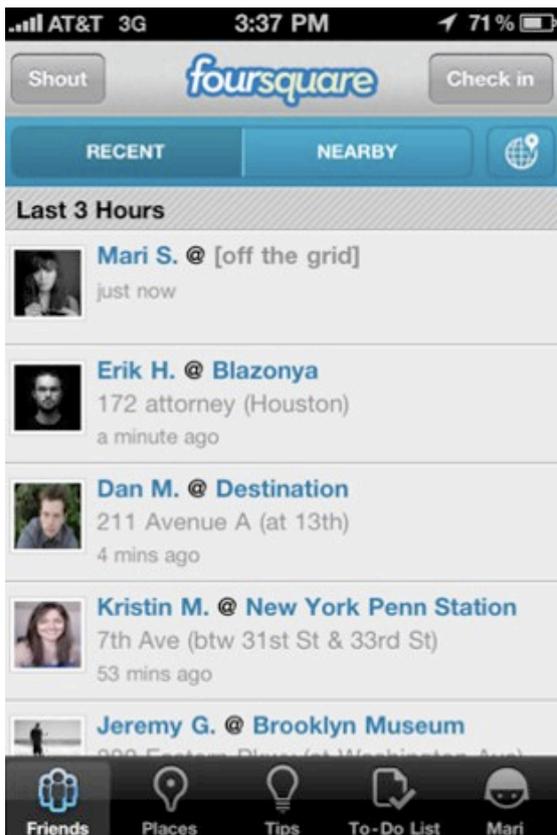


Figura 4.7.1 Última versión de Foursquare para iPhones

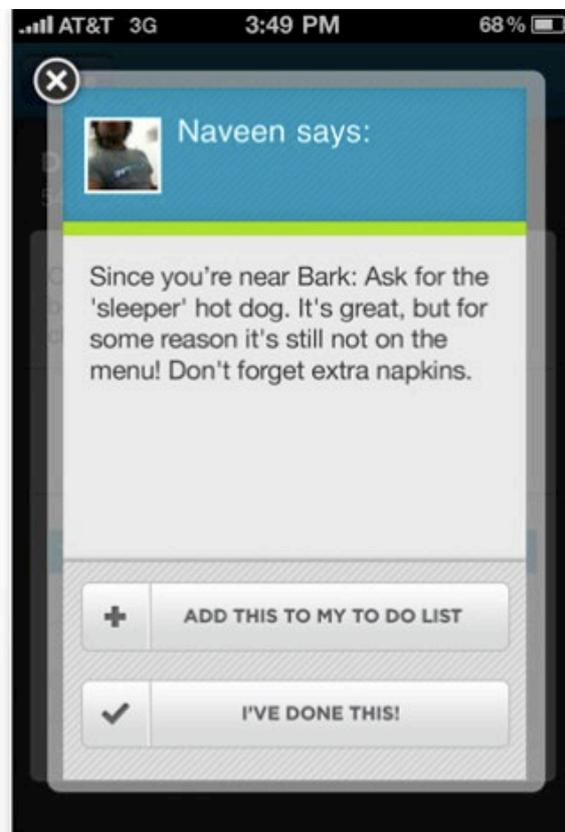


Figura 4.7.2 Pantalla TO-DO List de Foursquare

Las universidades más innovadoras pueden animar a los estudiantes a explorar sus campus y participar en eventos, ofreciendo premios cuando los usuarios entren a un conjunto determinado de lugares. A través de Foursquare o Gowalla, las instituciones pueden ofrecer premios o distintivos que los estudiantes desbloquean visitando lugares establecidos - aulas, el patio, el estadio deportivo, la librería del campus, o cualquier lugar con una historia detrás de él. [16]

Esto puede ser particularmente beneficioso en involucrar a los estudiantes que son nuevos en el campus y los expone a las tradiciones de la escuela. En agosto de 2010, la Universidad de Oregon utilizó Foursquare como parte de su Semana de Bienvenida. [16]

Mejorar la experiencia en eventos

Desde primer día de visita a la universidad por parte de los alumnos, hasta la graduación (para no mencionar todos los conciertos, juegos deportivos y conferencias en el medio) las instituciones están repletas de eventos. Para añadir otra capa de participación a estos eventos, las universidades pueden aprovechar las tecnologías basadas en la localización como Whrrl. Esta red permite a los usuarios no sólo su entrada a un lugar específico, sino que además permite subir fotos y mensajes a ese lugar para que puedan compartir la experiencia con todas las personas que los rodean. [16]

En mayo de 2010, la Universidad de St. Edwards ha utilizado Whrrl para conmemorar su ceremonia de graduación. [16]

Mejorar los ingresos con descuentos y promociones

Las universidades pueden ofrecer ofertas especiales, promociones y descuentos en tiendas como librerías y cafés del campus. Los servicios de localización se están convirtiendo rápidamente en la tarjeta de fidelización del nuevo mundo móvil. Gowalla, Foursquare y Loopt se han asociado con los minoristas en todo el país para ofrecer descuentos a los consumidores que facturen en sus ubicaciones. Las universidades pueden hacer lo mismo. Al ofrecer pequeños premios o gratificaciones a los que convertirse en alcalde de una ubicación, puede incentivar o sólo dejar caer a curiosear. [16]

La lealtad del cliente también se puede desarrollar a través de incentivos para la acumulación de ingresos a un lugar. Si hay varias opciones de lugar para almorzar, por ejemplo, un alumno podría estar tentado a visitar el mismo lugar con regularidad, sabiendo que van a recibir un refresco o postre gratis cada 10 visitas. [16]

Los servicios de localización también pueden ser beneficiosos para ayudar a generar el compromiso con las universidades fuera del campus. Las instituciones que deseen impulsar a sus estudiantes a explorar la comunidad en general pueden ofrecer recompensas para aquellos que se aventuran más allá de los límites del campus y colaborar con las empresas locales. También pueden dejar sugerencias en torno a la región geográfica para que los estudiantes reciban notas de su institución, ya que exploran nuevos lugares. [16]

Los estudiantes interesados en una ciudad grande y desconocida pueden recibir consejos de ex alumnos de su institución, con respecto a los gastos del semestre en el extranjero y pueden leer las sugerencias del conocimiento del programa académico. Estos son sólo algunos ejemplos de cómo las escuelas pueden

difundir su cultura a través de servicios basados en localización. [16]

Mientras más estudiantes hacen uso de los servicios basados en la localización, más valioso se vuelve a la institución en materia de colección de datos. Las universidades pueden ver datos demográficos, determinar si ciertos eventos son más populares que otros, tomar decisiones basadas en sugerencias (buenas y malas) de lugares del campus, ver las tendencias en el uso de varias instalaciones, como centros de salud y más. Con el tiempo, el uso de datos de ubicaciones pasará a formar parte de la cultura de la recopilación de datos que ya existe en la educación superior. Las instituciones que hagan uso temprano de este tipo de información, estarán en la vanguardia de la ola de tecnología de localización, que está a punto de surgir, y están diseñados para aprovechar sus muchos beneficios. [16]

Capítulo 5. Realización de la plataforma de hipermedia

5.1 Google Maps como metodología de ubicación geográfica

En la actualidad la mejor forma de implementar cartografía en un sitio Web es por medio del servicio ofrecido por Google: Google Maps. Es un servicio completo para operar con mapas y ofrece un conjunto extenso de API's para interactuar con el servicio:

Maps JavaScript API

Esta API sirve para integrar un mapa Google en un sitio Web a través de JavaScript, permitiendo manipular y agregar contenido dinámicamente.

Maps API for Flash

Diseñado especialmente para ser usado en sitios o aplicaciones basados en Flash. Esta API permite a los desarrolladores Flex integrar Google Maps en aplicaciones Flash. Al igual que la versión de JavaScript, esta API de ActionScript proporciona una serie de utilidades (métodos) para manipular y añadir contenido a los mapas a través de una variedad de servicios, lo que le permite integrar aplicaciones robustas y mapas interactivos en un sitio Web.

Google Earth API

Esta API sirve para integrar un mundo 3D digital en un sitio Web, permitiendo a sus usuarios viajar a cualquier parte del planeta, incluso debajo del océano sin necesidad de abandonar el sitio.

Static Maps API

Esta API permite la carga rápida de un mapa estático y simple en formato de imagen en la Web, sin necesidad de usar JavaScript u otro mecanismo de carga dinámica.

Web Services

Google permite usar URLs para realizar peticiones especiales, como Geocoding, direcciones, elevaciones y lugares con información de aplicaciones clientes y manipular los resultados en JSON o XML.

Maps Data API

Esta API permite mostrar, almacenar y actualizar, información almacenada por Google en los mapas, como por ejemplo markers, líneas y formas. [10]

La API de Google Maps es un servicio gratuito, disponible para cualquier sitio Web que no cobre por el acceso a sus servicios. Las empresas que imponen costos para el acceso deben utilizar la API Premier: PremierGoogle API de Google Maps, que ofrece características mejoradas, soporte técnico y un acuerdo de nivel de servicio.

Los servicios que provee Google para cartografía son únicos a nivel mundial. El uso de la API GoogleMaps JavaScript es la mejor para un tipo de plataforma de hipermedia física como la que se intenta desarrollar en esta tesina como prueba de conceptos. Esta API proporciona una serie de utilidades para manipular mapas (del mismo modo que lo utiliza el propio Google en <http://maps.google.com>) y crear contenidos para el mapa a través de una variedad de servicios, lo que le permite crear

aplicaciones robustas con el uso de mapas en un sitio Web.

La API de Google Maps V3 ha sido diseñada para cargar rápidamente y funcionar bien en dispositivos móviles. En particular, se concentraron en el desarrollo para dispositivos móviles avanzados como el iPhone y teléfonos con sistema operativo Android. Los dispositivos móviles poseen tamaños de pantalla más pequeños que los navegadores que corren en computadoras de escritorio, es por eso que algunas cuestiones deben ser analizadas según cada dispositivo. Por ejemplo, a menudo cada dispositivo tiene un comportamiento específico, que los diferencia del resto, como "pellizco para zoom" en el iPhone. Si usted desea que su aplicación funcione correctamente en dispositivos móviles, se recomienda lo siguiente:

Google recomienda en su sección para desarrolladores, configurar el tag <div> que contiene el mapa con atributos width y height con valores del 100%. Pero teniendo en cuenta que algunos exploradores más viejos no se muestran bien con este tipo de valores. Para poder detectar si el usuario está utilizando un dispositivo iPhone o Android, se puede realizar mediante la inspección la propiedad navigator.userAgent con el uso de DOM:

```
function detectBrowser()
{
    var useragent = navigator.userAgent;
    var mapdiv = document.getElementById("map_canvas");
    if (useragent.indexOf('iPhone') != -1 ||
useragent.indexOf('Android') != -1 )
    {
        mapdiv.style.width = '100%';
        mapdiv.style.height = '100%';
    } else {
        mapdiv.style.width = '600px';
        mapdiv.style.height = '800px';
    }
}
```

Esto nos permite alterar el layout para que se muestre en un dispositivo particular o en un conjunto de dispositivos de la misma manera.

Para los dispositivos iPhone se utiliza el siguiente <meta> tag:

```
<meta name="viewport" content="initial-scale=1.0, user-
scalable=no" />
```

Esto establece que el mapa debe mostrarse en full-screen y el usuario no debería poder cambiarlo de tamaño. Este tipo de parámetros también es soportado por los dispositivos con sistemas operativos Android en su versión 1.5. Para el caso de iPhone que utiliza el browser Safari, es necesario incluir el tag <meta> en la sección <head>.

Los servicios que ofrece la API de Google son:

- **Geocoding**, se refiere a la identificación de la localización geográfica de un usuario o un dispositivo a través de diversos mecanismos colección de datos. Típicamente, la mayoría de los servicios de geolocalización utiliza la red de enrutamiento o dispositivos GPS internos para determinar esta ubicación. Pero

se debe tener en cuenta que es un mecanismo específico de la API, por lo tanto algunos dispositivos, exploradores lo aceptan y otro no, por lo que no podemos asumir que siempre es posible la geolocalización para una aplicación web.

En la actualidad, existen varias maneras de detectar la ubicación del usuario a través de un navegador. Ninguno de estos métodos es parte de la API de Google Maps, sino que son las normas comunes de la industria.

Los nuevos navegadores están empezando a apoyar el estándar del W3C de geolocalización. Esta norma es parte de HTML5 y probablemente se convertirá en el estándar en el futuro. Todas las aplicaciones que desean utilizar la geolocalización deberían apoyar este estándar. Algunos navegadores utilizan direcciones IP para la detección de ubicación de un usuario, aunque esto sólo proporciona una estimación muy aproximada. Como la dirección IP de un usuario sólo puede proporcionar una estimación aproximada de la ubicación, no se recomienda utilizar este enfoque para la geolocalización. El enfoque propuesto por la W3C es el más fácil y totalmente compatible, por lo que se debe dar prioridad sobre otros métodos.

```
// Try W3C Geolocation (Preferred)
if(navigator.geolocation) {
    browserSupportFlag = true;

    navigator.geolocation.getCurrentPosition(function(position) {
        initialLocation = new
        google.maps.LatLng(position.coords.latitude,position.
        coords.longitude);
        map.setCenter(initialLocation);
        , function() {
            handleNoGeolocation(browserSupportFlag);
        });
    });

// Try Google Gears Geolocation
} else if (google.gears) {
    browserSupportFlag = true;
    var geo =
google.gears.factory.create('beta.geolocation');
    geo.getCurrentPosition(function(position) {
        initialLocation = new
google.maps.LatLng(position.latitude,position.longitude);
        map.setCenter(initialLocation);
    }, function() {
        handleNoGeoLocation(browserSupportFlag);
    });

// Browser doesn't support Geolocation
} else {
    browserSupportFlag = false;
    handleNoGeolocation(browserSupportFlag);
}
```

- **Directions**, permite calcular las direcciones (usando una variedad de métodos de transporte: bicicleta, a pie, en auto) mediante el objeto DirectionsService.

Este se comunica con el servicio de direcciones de Google Maps API quien recibe las peticiones y devuelve el conjunto de direcciones de los resultados calculados. Se puede manejar estos resultados de direcciones por el programador directamente mediante el objeto DirectionsRenderer.

Se deben especificar los orígenes y destinos, ya sea como cadenas de texto (por ejemplo, "Chicago, IL" o "Darwin, NSW, Australia") o como objetos LatLng que expresan un punto en el mapa como una latitud y una longitud. El servicio puede devolver las direcciones de varias partes utilizando una serie de waypoints. Las direcciones se muestran como una polilínea que dibujan la ruta en un mapa, o, además de una serie de descripción textual dentro de un elemento <div> (por ejemplo, "Gire a la derecha en la rampa de Williamsburg Bridge").

Para utilizar las direcciones en la versión 3 de la API, se debe crear un objeto DirectionsService y llamar a DirectionsService.route() para iniciar una solicitud al servicio, pasándole un objeto DirectionsRequest y el literal que contiene los términos de entrada y un método callback para ejecutar a la recepción de la respuesta.

- **Elevation**, el servicio proporciona datos de elevación para las ubicaciones en la superficie de la tierra, incluyendo la ubicación de profundidad en el suelo marino (que devuelven valores negativos). En los casos en que Google no posea las medidas exactas de elevación en la localización que se solicita, el servicio interpola y devuelve un valor promedio de los cuatro lugares más cercanos.

El objeto ElevationService ofrece una interfaz simple para consultar lugares por datos de elevación. Además, usted puede solicitar muestras de datos de elevación a lo largo de caminos, lo que permite calcular los cambios de elevación equidistantes a lo largo de las rutas. El objeto ElevationService se comunica con la API de Google Maps Elevation que recibe las peticiones de elevación y devuelve los datos de dicha elevación.

Tenga en cuenta que la velocidad de estas solicitudes está limitada por el abuso del servicio.

Con el servicio de elevación, se puede desarrollar aplicaciones para senderismo y ciclismo, aplicaciones móviles de posicionamiento, o de baja resolución de topografía.

- **Street View**. Google Street View ofrece vistas panorámicas de 360 grados de carreteras en toda su área de cobertura. La lista de ciudades en la actualidad que Google Maps ofrece soporte está disponible en el centro de asistencia de Google Maps.

Google Maps API JavaScript proporciona un servicio Street View para la obtención y manipulación de las imágenes usadas en Google Maps Street View. A diferencia de la API V2, el servicio Street View en la V3 es compatible de forma nativa en el navegador.

Street View es más útil cuando lo indica una ubicación en el mapa. Para agregar esta opción a un mapa, sólo tendrá que agregar el control de Street View. Para agregar este control, dentro de MapOptions del mapa, se debe especificar el parámetro streetViewControl en true. El mapa aparecerá con un

control de Street View Pegman. [10]

Además de los servicios que provee, la API Javascript de Google Maps, contiene un importante conjunto de objetos que forman la estructura que ofrece la API. Estos objetos definen los elementos con los cuales vamos a interactuar para manipular y operar los mapas. Pero antes veamos algunos conceptos utilizados por la API para comprender mejor la forma de trabajar con ella:

Overlays. Los overlays son superposiciones en el mapa, objetos en el mapa que están vinculados a coordenadas de latitud y longitud, por lo que se mueven al arrastrar o realizar zoom en el mapa. Estas superposiciones reflejan los objetos que "añadimos" al mapa para designar puntos, líneas, áreas, o colecciones de objetos.

La API de Google Maps tiene varios tipos de superposiciones, por un lado las **individuales**, lugares en el mapa se muestran utilizando marcadores. Los marcadores en ocasiones pueden mostrar iconos con imágenes personalizadas, en cuyo caso se les suele llamar como "iconos". Los marcadores y los iconos son objetos de tipo Marker. Por otro lado existen las **líneas**, son las que se muestran en el mapa como líneas discontinuas (en representación de una secuencia ordenada de sitios). Las líneas son objetos de tipo Polilínea (Polyline). También existen los overlays para determinar **áreas**, se muestran de forma arbitraria en el mapa utilizando polígonos, que son similares a polilíneas. Al igual que las polilíneas, los polígonos son una secuencia ordenada de los lugares, a diferencia de las polilíneas, los polígonos definen una región encerrada. Por último, la **ventana de información** (infowindow) es también un tipo especial de overlay para mostrar el contenido (texto o imágenes por lo general) dentro de un globo emergente en la parte superior de un mapa en un lugar determinado. También se pueden poner en práctica sus propias plantillas personalizadas. Estas superposiciones personalizadas implementan la interfaz OverlayView. [10]

Layers. Los layers son capas, son objetos en el mapa que consisten en una o varias superposiciones, y se puede manipular como una sola unidad. Las capas en general, son colecciones de objetos que se agregan en la parte superior del mapa para que designe una asociación común. La API de Google Maps gestiona la presentación de los objetos dentro de las capas mediante la creación y autodestrucción de sus constituyentes superposiciones como los cambios de visualización del mapa. Las capas también pueden alterar la capa de presentación del mapa en sí, alterando ligeramente los cuadros prestados de manera coherente con la capa. [10]

Para añadir una capa dentro de un mapa, sólo tiene que llamar setMap(), pasándole el objeto de mapa en el que se mostrará la capa. Del mismo modo, para ocultar una capa, se invoca a setMap() , pasando nulo como parámetro.

La API de Google Maps tiene varios tipos de capas:

* **KmlLayer** para hacer objetos y elementos KML en Mapas GeorSS API V3. Utilizados comúnmente para mostrar información geográfica sobre el mapa.

* El objeto **BicyclingLayer** hace una capa con carriles para circular en bicicleta en una capa común, o default, que se encuentra en el mapa. Esta capa se devuelve de forma predeterminada en el DirectionsRenderer.

* El objeto **TrafficLayer** hace una capa que representa las condiciones del tráfico y superposiciones que representa el tráfico.

Map. Es la clase que hace referencia al mapa en sí, extiende del objeto MVCObject, y su invocación genera el mapa en el código HTML.

```
var map = new google.maps.Map(
    DOCUMENT.GETELEMENTBYID("MAP_CANVAS"),
    MYOPTIONS
);
```

Donde "map_canvas" es el tag <div> vacío donde se incluirá el mapa. El segundo parámetro es un array de opciones que a continuación detallamos algunas de las opciones más importantes. El centro del mapa es requerido y se especifica por medio del objeto LatLng, no es más que un par ordenado Latitud y longitud que se pasan como parámetros en la creación del mismo como strings.

center	LatLng	El centro inicial del mapa, es un parámetro requerido.
disableDoubleClickZoom	boolean	Habilita/Deshabilita el zoom por doble click.
draggable	boolean	Previene al mapa de permitir el arrastrar.
draggableCursor	string	Nombre o URL del cursor a mostrar cuando se arrastra.
keyboardShortcuts	boolean	En falso previene al mapa de ser controlado por teclado.
mapTypeId	MapTypeId	El tipo inicial de mapa. Es un parámetro requerido y puede contener los valores siguientes: HYBRID: Este tipo de mapa muestra una capa transparente de las calles principales sobre imágenes de satélite. ROADMAP: Muestra el mapa normal solo de calles. SATELLITE: Muestra el mapa con imágenes de satélite. TERRAIN: Mapa físico, muestra vegetación y elevación.
scrollwheel	boolean	En falso deshabilita el zoom por ruedita de mouse.
zoom	number	El zoom inicial del mapa. Es un parámetro requerido.

Markers. Los markers son una de las utilidades más antiguas de la API de Google Maps, es la posibilidad de marcar puntos en una determinada latitud y longitud, a estos puntos se los denomina markers y la manera de utilizarlos es la siguiente:

```
var marker = new google.maps.Marker({
    POSITION: LTLG,
    MAP: MAP,
```

```

        DRAGGABLE: TRUE,
        TITLE: 'MI PRIMER MARCADOR'
    });

```

Si se especifica un mapa, el marcador se añadirá al mapa automáticamente, de lo contrario el objeto queda creado en memoria pero no se visualiza hasta que se asigna uno. Tenga en cuenta que la posición se debe establecer para el marcador que se desea mostrar.

Polyline y Polygon, son objetos para dibujar formas sobre el mapa, de mucha utilidad, en nuestro caso en particular, para definir superficies de ubicaciones. Dado un conjunto de vértices que definen las esquinas de la ubicación se dibujan líneas entre cada vértice formando la superficie. Veamos un ejemplo:

```

var coords = [
    new google.maps.LatLng(25.774252, -80.190262),
    new google.maps.LatLng(18.466465, -66.118292),
    new google.maps.LatLng(32.321384, -64.75737),
    new google.maps.LatLng(25.774252, -80.190262)
];
var surface = new google.maps.Polygon({
    paths: coords,
    strokeColor: "#FF0000",
    strokeOpacity: 0.8,
    strokeWeight: 1,
    fillColor: "#224466",
    fillOpacity: 0.7
});

```

Pero si vamos al caso, definir puntos en un mapa es útil pero no sirve de mucho si a ese punto no le podemos poner algo de información adicional. Para esto existe la clase **InfoWindow**, que muestra en forma de globo de notificación, una porción de código HTML. La manera de utilizarlos es la siguiente:

```

var infowindow = new google.maps.InfoWindow({
    content: "<h1> Este es un globo de notificacion </h1>"
});

```

Una vez creado el globo con su contenido, necesitamos poder abrirlo y cerrarlo en determinados eventos. Para esto el objeto infowindow tiene dos mensajes open() y close(). Pero antes de pasar a detallar como funcionan debemos analizar como asociar eventos del usuario a los objetos del mapa. ¿Cómo podemos realizar una determinada acción, como abrir un globo de información, al hacer click en un marker?

```

/*Agrega el evento click al mapa para que muestre una
alerta con la latitud y la longitud donde se hizo click*/

```

```

google.maps.event.addListener(map, "click",
function(event) {
    alert(event.latLng.lat());
    alert(event.latLng.lng());
}

```

```

});

/*Agrega el evento dragend (se invoca al terminar de
arrastrar un marker), para que abra el globo de información de
la ubicación */

google.maps.event.addListener(marker, "dragend",
function() {
    infowindow.open(map, marker);
});

```

Como se muestra en el último ejemplo para abrir la ventana de información de una ubicación se reciben dos parámetros, el mapa y el marker donde se desea desplegar el globo. De esta forma ya podemos operar mediante acciones producidas por el usuario como clicks o drags.

Geocoder, DirectionsRenderer, DirectionsService, ElevationService, son los objetos mencionados anteriormente para operar con los servicios de Google: geodecodificación, direcciones y elevación. [\[10\]](#)

5.2 QR Codes como metodología de identificación de lugares

La identificación automática de objetos del mundo real es un área de estudio en constante desarrollo, con mejoras constantes acompañadas de los avances tecnológicos. En la plataforma desarrollada en esta tesina se utiliza el método de códigos de barra 2D Quick Response como metodología de sensado de ubicaciones. Es decir, que los agentes móviles puedan determinar dónde se encuentran, a partir de la toma de una fotografía del código de barras disponible en cada ubicación. En esta tesina se busca analizar y determinar el rendimiento de esta metodología de sensado en plataformas de hipermedia física.

Un código de barras QR (Quick Response Barcode) es un sistema para almacenar información en una matriz de puntos o un código de barras bidimensional creado por la compañía japonesa Denso-Wave en 1994; se caracterizan por los tres cuadrados que se encuentran en las esquinas y que permiten detectar la posición del código al lector. La sigla "QR" se derivó de la frase inglesa "Quick Response" pues el creador aspiraba a que el código permitiera que su contenido se leyera a alta velocidad. Los códigos QR son muy comunes en Japón y de hecho son el código bidimensional más popular en ese país.

Aunque inicialmente se usó para registrar repuestos en el área de la fabricación de vehículos, hoy, los códigos QR se usan para administración de inventarios en una gran variedad de industrias. Recientemente, la inclusión de software que lee códigos QR en teléfonos móviles japoneses, ha permitido nuevos usos orientados al consumidor, que se manifiestan en comodidades como el dejar de tener que introducir datos de forma manual en los teléfonos. Las direcciones y los URLs se están volviendo cada vez más comunes en revistas y anuncios japoneses. El agregado de códigos QR en tarjetas de presentación también se está haciendo común, simplificando en gran medida la tarea de introducir detalles individuales de un nuevo cliente en la agenda de un teléfono móvil.

Los consumidores que cuenten con dispositivos y programas de captura, en combinación con un PC con interfaz RS-232C pueden usar un escáner para leer los datos.

El estándar japonés para códigos QR (JIS X 0510) fue publicado en enero de 1999 y su correspondiente estándar internacional ISO (ISO/IEC18004) fue aprobado en junio de 2000. Un detalle muy importante sobre el código QR es que su código es abierto y que sus derechos de patente (propiedad de Denso Wave) no son ejercidos.

Si bien los códigos de barras convencionales son capaces de almacenar un máximo de 20 dígitos, el código QR es capaz de manejar cientos de veces más información. Un QR Code es capaz de manejar todo tipo de datos, tales como caracteres numéricos y alfabéticos, Kanji, Kana Hiragana, símbolos, binarios y códigos de control. Hasta 7.089 caracteres se pueden codificar en un solo símbolo.

Un código QR lleva la información tanto horizontales como verticales, es capaz de codificar la misma cantidad de datos en aproximadamente una décima parte del espacio de un código de barras tradicionales. (Para un tamaño de impresión más pequeños, existe Micro QR Code.) [7]

QR Code tiene la capacidad de corrección de errores. Los datos pueden ser restaurados, incluso si el símbolo es parcialmente sucio o dañado. Un máximo de 30% de las claves se puede restaurar. Ejemplo de dos claves que pueden ser restauradas:



Figura 5.2.1 Ejemplo de códigos deteriorados [7]

Capacidad de corrección de errores

Nivel L	7% de las claves se pueden restaurar
Nivel M	15% de las claves se pueden restaurar
Nivel Q	25% de las claves se pueden restaurar
Nivel H	30% de las claves se pueden restaurar

El ejemplo siguiente ilustra la forma en que el código QR maneja la distorsión. En estos casos se agregaron o eliminaron píxeles del código original para examinar el nivel de distorsión de los bordes. Las dos imágenes a las que se les alteraron los datos todavía son reconocibles y usan el nivel "L" de corrección de errores. [7]



Código fuente.

Datos borrados.

Datos agregados.

Figura 5.2.2 Proceso de recuperación de datos de un código QR [7]

El QR es capaz de ser leído en 360 grados (omni-direccional), es de lectura de alta velocidad. Un código QR lleva a cabo esta tarea a través de patrones de detección de posición situado en las tres esquinas del símbolo. Estos patrones de detección de posición dan garantía de lectura estable de alta velocidad, eludiendo los efectos negativos de las interferencias de fondo. Esto hace al código QR un gran candidato para la identificación de ubicaciones del mundo real dado su capacidad de lectura y supervivencia al deterioro que el exterior le pueda provocar.

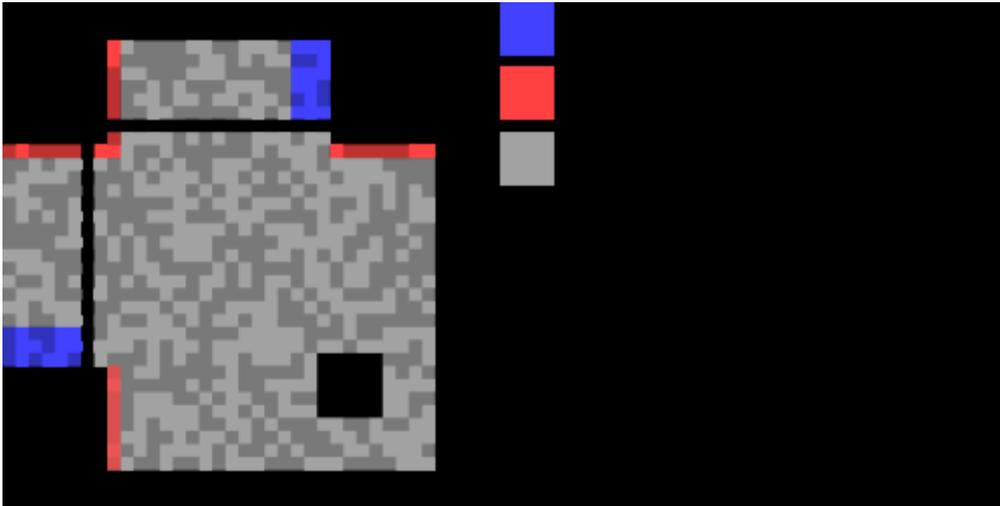


Figura 5.2.3 Estructura de un código QR [7]

Capacidad de datos del código QR

Solo numérico	Máx. 7.089 caracteres
Alfanumérico	Máx. 4.296 caracteres
Binario	Máx. 2.953 bytes
Kanji/Kana	Máx. 1.817 caracteres

Un código QR maneja diferentes versiones de simbología que varía desde 1 hasta 40. Cada versión tiene un modulo de configuración diferente. El modulo se refiere a los puntos blancos y negros que determinan el código QR. La “configuración de los módulos” se refiere al número de módulos contenidos en un símbolo, comenzando con la versión 1 de 21x21 módulos y terminando en la versión 40 de 177x177 módulos. Cada versión más grande tiene una diferencia de 4 módulos por lado.

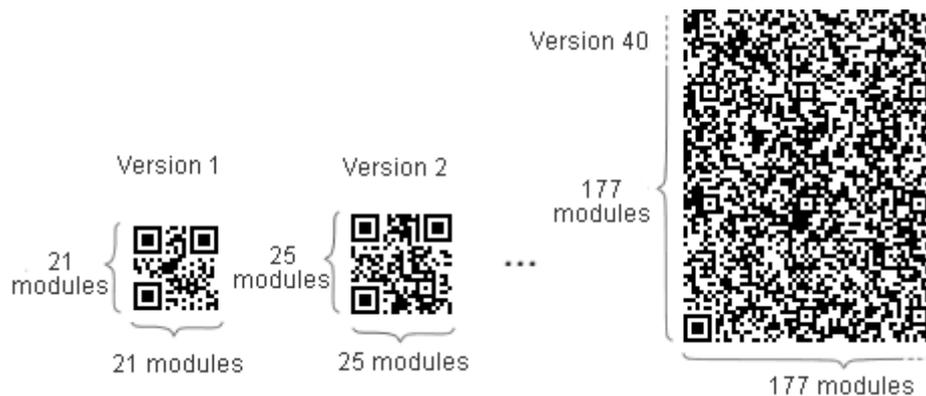


Figura 5.2.4 Comparación de versiones de código QR [7]

Luego de analizar la robustez que ofrece el código Quick Response, podemos decir que pueden resultar de mucha utilidad a la hora de crear una plataforma de hipermedia física. Nos ofrecen seguridad en la lectura, un aspecto moderadamente aceptable, economía en la implementación dado que solo necesitamos imprimir una hoja con el código, posibilidad de lectura por medio de cámaras de dispositivos móviles, y soporte para la mayoría de los lenguajes de programación actuales.

Por otro lado, vale destacar también la cantidad de aplicaciones de código abierto que existen actualmente para hacer lectura de estos códigos. Algunas de las más conocidas son: Google Zxing (Android y Barcodes para iPhone), I-Nigma, QuickMark (para Symbian y Windows Mobile), Nokia barcode reader, Kaywa Reader (funciona para la mayoría de los teléfonos, pero no es gratuita). Todas estas aplicaciones ofrecen la posibilidad de decodificar el contenido del código Quick Response y en caso de que se esté descifrando una URL, permiten acceder directamente mediante el explorador determinado en el sistema operativo móvil. Dado este análisis podemos decir que sería en vano desarrollar una aplicación especial que forme parte de una plataforma de hipermedia física que consista en la decodificación de los códigos QR. Es preferible dejar en manos de aplicaciones específicas que se dediquen a eso y mantengan el soporte activo para la comunidad.

5.3 Plataforma hipermedia sobre Web

El volumen de usuarios de Internet ha aumentado cada año que transcurre, cada día se añaden más personas en diferentes países y continentes. Según datos obtenidos en Agosto del 2001, se estima que de los 7 mil millones de personas que habitan el planeta tierra, unos 600 millones eran usuarios online. De ese total el 70% accedía por banda ancha. En la actualidad los valores han ido creciendo, y cada vez son más los usuarios que acceden por medio de dispositivos móviles. Esto fue impulsado por la mejoría de las señales que utilizan, como por ejemplo el caso de la señal 3G, que ofrece mejor velocidad y capacidad de transferencia de datos. De modo que podemos decir que la Web tiene un rol cada vez más importante en la vida de las personas. Por lo tanto, muchas empresas y organismos no dudan a la hora de comenzar un nuevo desarrollo, evaluar la posibilidad de implementarlo sobre Internet vía Web, o como muchas veces la llaman: "Aplicaciones en la nube".

Croover (el caso de prueba de esta tesina) está diseñado pensando en un ambiente global, público, donde cualquier persona podría crearse un usuario y administrar sus ubicaciones, se realizó de este modo para lograr una completa abstracción del uso de la plataforma en contextos particulares, como el académico el cual se busca implementar en esta tesina. A continuación vamos a ver el proceso de

diseño y desarrollo de Croover, como así también las tecnologías utilizadas, descripción de cada una y motivo de la elección de cada tecnología.

5.3.1 Tecnologías utilizadas en el caso de prueba

Luego de haber analizado las tecnologías disponibles para generar una plataforma de hipermedia física en los capítulos anteriores, podemos decir que resulta sencillo implementar una aplicación íntegra, es decir, tanto parte administrativa como móvil, sobre protocolo HTTP. Dado el constante y creciente avance de los dispositivos móviles para operar sobre la Web, resulta más eficiente generar una aplicación Web que muestre una página igual para todos los navegadores móviles, que una aplicación móvil que funcione correctamente en todos los sistemas operativos de cada teléfono celular del mercado. Si bien Java ofrece un factor de portabilidad muy importante ya se ha analizado que no siempre las aplicaciones funcionan de la misma manera bajo diferentes dispositivos.

A la hora de desarrollar nuevos sistemas, siempre se evalúan algunos factores respecto a que tecnología vamos a utilizar. Como ya sabemos, siempre surgen preguntas como: ¿Opensource o propietario?, la verdad es que depende de lo que vayamos a desarrollar, pero en nuestro caso, al querer implementar una plataforma que funcione sobre Web, existen muchas herramientas opensource que nos hacen el trabajo más fácil. Detrás de las tecnologías opensource existen comunidades de desarrolladores que utilizan estas herramientas y dan ayuda en foros y grupos de discusión, por lo cual ante algún problema que tengamos, podemos encontrar información disponible muy rápidamente en Internet.

Para demostrar lo analizado en los capítulos anteriores se realizó un caso de prueba para poder determinar la factibilidad de cada uno de las tecnologías nombradas. El caso de prueba es denominado Croover, y es una plataforma de hipermedia física desarrollada en Web que integra tanto la administración como la parte móvil sobre HTML. Las tecnologías elegidas fueron:

- Apache HTTP Server
- PHP (Symfony Framework)
- Doctrine (ORM)
- Linux
- MySQL

La fundación Apache software (www.apache.org) proporciona apoyo a la comunidad de proyectos Apache de código abierto. Los proyectos Apache se caracterizan por una colaboración, proceso consensuado de desarrollo basado en una licencia de software abierta y pragmática, y el deseo de crear software de alta calidad que es líder en su campo.

El Apache HTTP Server es un servidor HTTP de código abierto para los sistemas operativos modernos, incluyendo UNIX, MS-Windows, Macintosh y NetWare. El objetivo de este proyecto es proporcionar un servidor seguro, eficiente y extensible que proporcione servicios HTTP en sintonía con los actuales estándares de HTTP. Apache ha sido el servidor Web más popular en Internet desde abril de 1996. Por lo

tanto es sin dudas la mejor opción para comenzar un desarrollo Web sin dolores de cabeza. [1]

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas Web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK. PHP es un acrónimo recursivo que significa *PHP Hypertext Pre-processor* (inicialmente PHP Tools, o, *Personal HPTools*). Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo la implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar de facto para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre. [25]

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones.

Aunque todo en su diseño está orientado a facilitar la creación de página Web, es posible crear aplicaciones con una interfaz gráfica para el usuario utilizando la extensión PHP-Qt o PHP-GTK. También puede ser usado desde la línea de órdenes de la misma manera como Perl o Python pueden hacerlo; a esta versión de PHP se la llama PHP-CLI (*Command Line Interface*).

Cuando el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Mediante extensiones es también posible la generación de archivos PDF, Excel, Flash, así como imágenes en diferentes formatos. [25]

Symfony es un completo framework diseñado para optimizar el desarrollo de las aplicaciones Web mediante algunas de sus principales características. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación Web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación Web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación Web. [27]

Symfony está desarrollado completamente en PHP 5. Ha sido probado en numerosos proyectos reales y se utiliza en sitios Web de comercio electrónico de primer nivel. Symfony es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y Microsoft SQL Server. Se puede ejecutar tanto en plataformas unix (Unix, Linux, etc.) como en plataformas Windows.

Symfony fue diseñado para adaptarse a los siguientes requisitos:

- Fácil de instalar y configurar en la mayoría de plataformas (y con la garantía de

que funciona correctamente en los sistemas Windows y Unix estándares).

- Independiente del sistema gestor de bases de datos. Su capa de abstracción y el uso de Propel, permiten cambiar con facilidad de SGBD en cualquier fase del proyecto.
- Utiliza programación orientada a objetos, de ahí que sea imprescindible PHP 5.
- Sencillo de usar en la mayoría de casos, aunque es preferible para el desarrollo de grandes aplicaciones Web que para pequeños proyectos.
- Aunque utiliza MVC (Modelo Vista Controlador), tiene su propia forma de trabajo en este punto, con variantes del MVC clásico como la capa de abstracción de base de datos, el controlador frontal y las acciones.
- Basado en la premisa de “convenir en vez de configurar”, en la que el desarrollador sólo debe configurar aquello que no es convencional.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la Web.
- Preparado para aplicaciones empresariales y adaptables a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.
- Fácil de extender, lo que permite su integración con las bibliotecas de otros fabricantes.
- Una potente línea de comandos que facilitan generación de código, lo cual contribuye a ahorrar tiempo de trabajo. [\[27\]](#)

Las características más comunes para el desarrollo de proyectos Web están automatizadas en Symfony, tales como:

- Permite la internacionalización para la traducción del texto de la interfaz, los datos y el contenido de localización.
- La presentación usa *templates* y *layouts* que pueden ser construidos por diseñadores de HTML que no posean conocimientos del *framework*.
- Los formularios soportan la validación automática, lo cual asegura mejor calidad de los datos en las base de datos y una mejor experiencia para el usuario.
- El manejo de cache reduce el uso de banda ancha y la carga del servidor.

- La facilidad de soportar autenticación y credenciales facilita la creación de áreas restringidas y manejo de seguridad de los usuarios.
- El enrutamiento y las Urls inteligentes hacen amigable las direcciones de las páginas de la aplicación.
- Las listas son más amigables, ya que permite la paginación, clasificación y filtraje automáticos.
- Los plugins proveen un alto nivel de extensibilidad.
- La interacción con AJAX es mucho más sencilla. [\[27\]](#)

Doctrine es un Object Relational Mapper (ORM) para PHP con el soporte de una poderosa Database Abstraction Layer (DBAL). Una de sus características principales es la opción de escribir consultas a la base de datos en un dialecto propietario SQL denominado DQL, inspirado por el utilizado en Hibernate HQL. Esto les provee a los programadores de una alternativa robusta a SQL manteniendo la flexibilidad sin requerir duplicación de código. [\[8\]](#)

DQL como un lenguaje de consulta tiene instrucciones SELECT, UPDATE y DELETE que se asignan a sus tipos correspondiente de sentencia SQL. INSERT no se permiten en DQL, porque las entidades y sus relaciones tienen que ser introducidos en el contexto de persistencia a través EntityManager para garantizar la coherencia de su modelo de objetos.

Las sentencias DQL SELECT son una manera muy poderosa de recuperar partes de su modelo de dominio que no son accesibles a través de asociaciones. Además permiten recuperar las entidades y sus asociaciones en una sola instrucción SELECT de SQL que puede marcar una gran diferencia en el rendimiento en contraste de empleando varias consultas.

DQL UPDATE y DELETE ofrecen una manera de ejecutar cambios masivos en las entidades de su modelo de dominio. Esto es a menudo necesario cuando no se puede cargar todas las entidades afectadas de una actualización en masa en la memoria. [\[8\]](#)

Veamos un ejemplo de SELECT en DQL para observar la estructura de una consulta en doctrine:

```
<?php
//SELECT
$query = $em->createQuery('SELECT u FROM MyProject\Model\User u WHERE
u.age > 20');
$users = $query->getResult();

$query = $em->createQuery('SELECT u.username, u.name FROM CmsUser u');
$users = $query->getResults(); // CmsUser array with username, ids
echo $users[0]['username'];
```

```
//JOIN
$query = $em->createQuery("SELECT u FROM User u JOIN u.address a WHERE
a.city = 'Berlin'");
$users = $query->getResult();

¿>
```

Algunas de las funciones que se permiten en las estructuras SELECT, WHERE y HAVING de DQL son:

- ABS(arithmetic_expression)
- CONCAT(str1, str2)
- CURRENT_DATE() – Retorna la fecha actual.
- CURRENT_TIME() – Retorna la hora actual.
- CURRENT_TIMESTAMP() – Retorna el timestamp de la hora y le fecha actual.
- LENGTH(str) – Retorna la cantidad de caracteres del string.
- LOCATE(needle, haystack [, offset]) – Ubica la primera ocurrencia del substring en el string.
- LOWER(str) – Retorna el string en minusculas.
- MOD(a, b) - Retorna a MOD b.
- SIZE(collection) – Retorna la cantidad de elementos de la colección .
- SQRT(q) – Retorna la raiz cuadrada de q.
- SUBSTRING(str, start [, length]) – Retorna el substring del string.
- TRIM([LEADING | TRAILING | BOTH] ['trchar' FROM] str) – Separa el string por el caracter que se especifica por default es espacio en blanco.
- UPPER(str) – Retorna el string en mayusculas. [8]

La siguiente consulta es una función del caso de prueba de esta tesina, que retorna los mensajes pertenecientes a un usuario en particular:

```
public function getHappeningsForUser($user_id, $limit)
{
    return Doctrine_Core::getTable('Happening')-
>createQuery('h')
    ->where('h.location_id = ?', $this->getId())
    ->andWhere('h.created_for = ?', $user_id)
    ->limit($limit)
    ->execute();
}
```

Desarrollar sobre una plataforma Linux tiene beneficios en cuanto a seguridad y herramientas disponibles (también opensource) para operar con servidores desde línea de comandos. Programas como el editor de texto Vim, para utilizar desde consolas y no tener que programar desde IDE's propietarios.

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB (desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009) desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C. [22]

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y el copyright del código está en poder del autor individual, MySQL es propietario y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privadas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson y Michael Widenius.

5.3.2 Descripción del caso de prueba

En primer lugar se detallan cuestiones del modelo de datos. Se buscó ser lo más sencillo posible en el esquema de la base de datos para permitir la fácil comprensión y la escalabilidad del sistema en un futuro. Como podemos imaginar vamos a tener objetos ubicaciones (Locations), las cuales representarán a los distintos objetos del mundo real. Para cada una de ellas necesitamos guardar al menos 3 datos importantes; por un lado, la latitud y longitud que identifican el lugar donde se encuentra en el mundo y por otro lado, un nombre descriptivo. Pero además se agrega un dato que permite identificar a la persona que creó dicha ubicación, dado que en esta plataforma se suponen muchos usuarios administradores.

Como mencionamos en el capítulo anterior, las redes sociales tienen una forma de llegar a las personas la cual es fácil de implementar. Para comprobar que se puede integrar las metodologías de comunicación que utilizan las redes sociales y la hipermedia física se agregan tres conceptos de contenidos informativos que se encuentran en una ubicación: Mensaje, Evento y Encuestas.

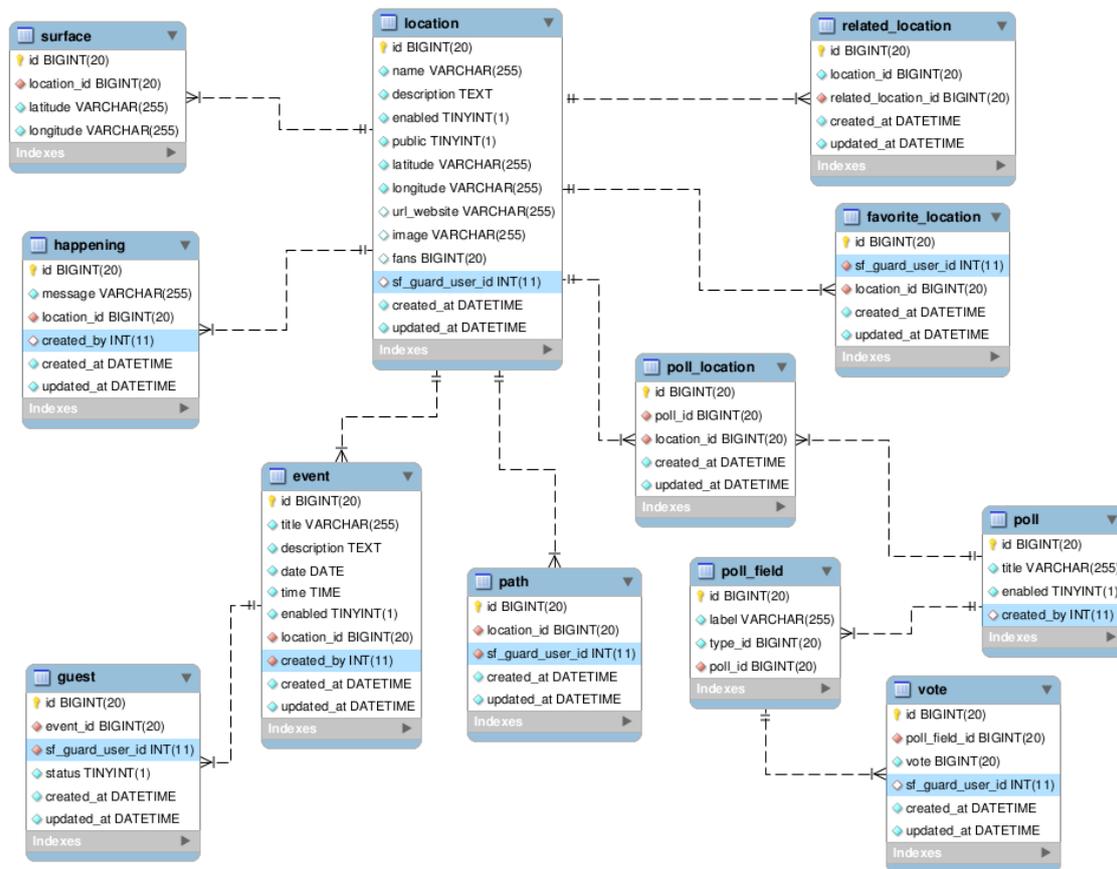


Figura 5.1 Modelo resumido de los elementos que componen a una ubicación

La figura anterior muestra como está compuesto el modelo para el funcionamiento básico de la plataforma de hipermedia, sin la inclusión del control de acceso a usuarios que vamos a definir más adelante.

Para el caso de los mensajes (Happening) basta con tres atributos para definir el modelo que los componen. Un mensaje intenta describir algo que sucedió o esta próximo a suceder. Inicialmente no existe límite en la cantidad de caracteres a utilizar por el usuario para componer un mensaje, pero la idea es que sea un mensaje corto. Un mensaje está asociado a una ubicación por lo tanto debe existir en la tabla de los mensajes la relación correspondiente mediante el ID de la ubicación.

En las encuestas (Poll), es necesario definir los campos por los cuales la encuesta está formada, y para cada campo el tipo de campo que es. El tipo de campo está dado por la cantidad de inputs existentes en HTML, para selección múltiple: Checkbox, para selección individual: Radiobutton y para ingresar texto un input normal. Por lo tanto tenemos la tabla PollField que expresa la relación entre un campo y la encuesta a la que pertenece. Luego que la encuesta esta correctamente definida, los usuarios que ingresan pueden ejercer su voto. El voto (vote) se registra en base al valor que seleccionó para los campos y el usuario que realizó la encuesta.

De los eventos (event) es necesario definir día y horario del evento a realizar. Y por otro lado mantener una lista de personas que van a concurrir a dicho evento, esto se almacena en la tabla invitados (guest).

Por otro lado, tenemos las ubicaciones relacionadas. Cada ubicación definida en cualquier parte del mundo puede estar relacionada a una o más ubicaciones en cualquier otro rincón del mundo. De modo que para almacenar esa relación existe la tabla `related_location`.

Como mencionamos anteriormente en una plataforma de hipermedia es importante reconocer de dónde viene el usuario para poder recomendarle a donde se puede dirigir. Para guardar la relación de la ubicación que un usuario visita y el usuario en sí, existe la tabla `path`, que marca el camino que el usuario va recorriendo a lo largo de un día.

Para cada ubicación el usuario podría querer definir la superficie que ocupa, para esto existe la tabla `surface`, cada punto que marca una esquina de la superficie es una tupla en esta tabla. Cada vértice está definido por una latitud y una longitud, y el conjunto de coordenadas se le envía a un objeto `polygon` de la API de Google Maps para que dibuje la superficie en el mapa.

Además, se permite a los usuarios tener ubicaciones como favoritas. Para cada ubicación que es agregada como favorita por el usuario se crea una tupla en la tabla `favorite_location`.

Como mencionamos anteriormente, en la imagen del modelo, está ausente el esquema de tablas del manejo de usuarios. Esta funcionalidad de administración de usuarios la provee un plugin de Symfony denominado `sfGuardDoctrinePlugin` [28]. Este plugin es testado y desarrollado por Sensio Labs, los creadores de Symfony, y garantizan el correcto funcionamiento del mismo. Por lo tanto no conviene ponerse a desarrollar algo nuevo para manejar los permisos y los usuarios.

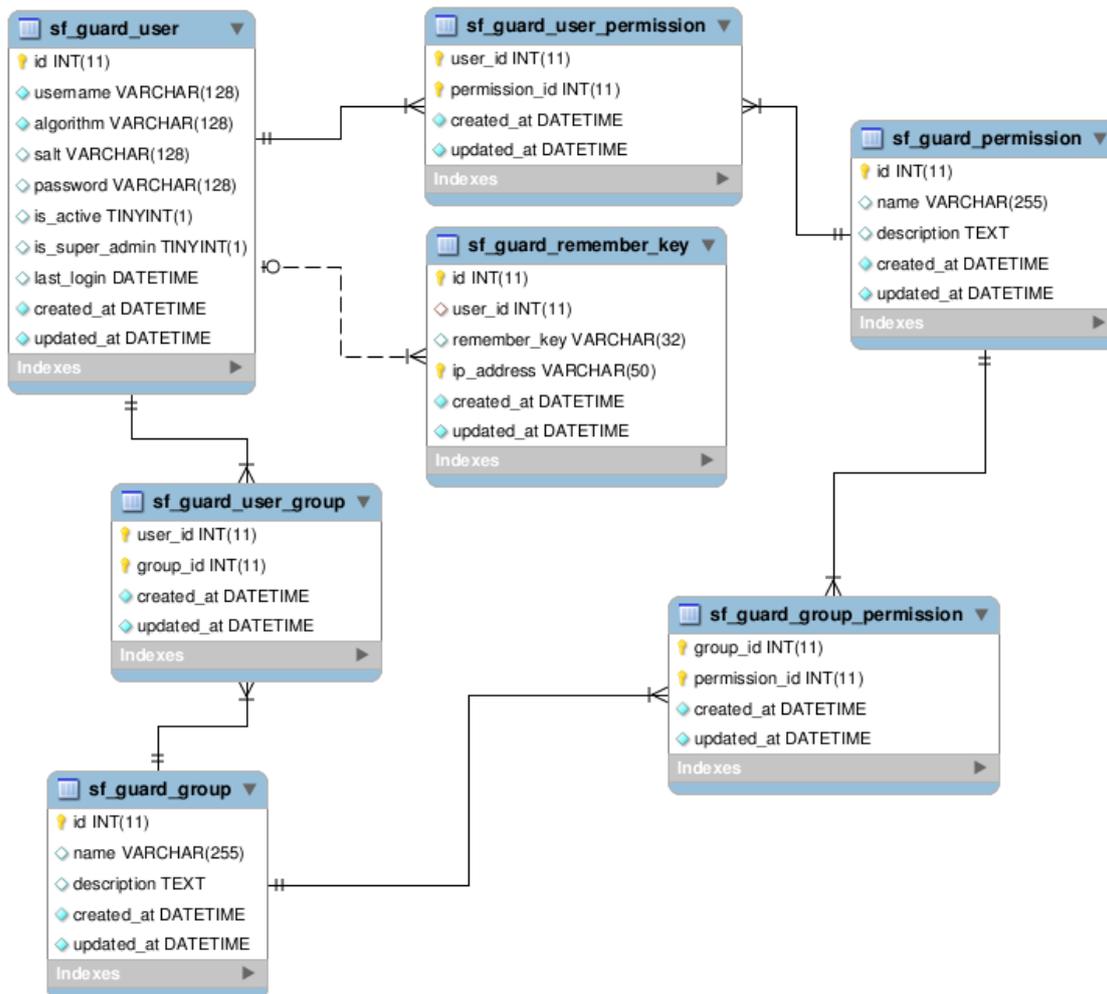


Figura 5.2 Imagen del modelo de datos del sfGuardDoctrinePlugin

Como vemos en la figura el plugin implementa toda la funcionalidad en siete tablas que se agregan al modelo de datos. Sf_guard_user es la tabla principal donde se almacenan todos los usuarios del sistema, con nombre de usuario y contraseña. Además, cada usuario puede estar relacionado a un grupo de usuarios (sf_guard_group) y de este modo se le asignan un conjunto de permisos al usuario. Por ejemplo el grupo “administrador” podría tener permisos para todas las acciones del sistema y el grupo “alumno” solo algunas acciones de visualización de contenidos.

Luego de analizar la estructura del plugin que ofrece la gestión de usuarios, se anexan estas tablas al primer modelo básico presentado páginas atrás en la figura 5.1. Ahora cada ubicación está relacionada a un objeto sf_guard user, como así también, los votos, los favoritos de un usuario, los mensajes, etc.

Como ya mencionamos existen dos tipos de usuarios los administradores y los móviles. ¿Cómo nos damos cuenta si es un agente móvil? Para resolver este problema, la mayoría de los sitios implementa un detector de exploradores móviles hardcodeando los distintos tipos de browsers del mercado, no está mal visto, porque es la forma más eficiente, pero si hay que estar atentos ante nuevos exploradores Web que no se detecten con nuestro “detector”. Veamos el siguiente ejemplo para detectar dispositivos móviles en PHP:

```
$mobile_browser = '0';
```

```

if(preg_match('/(up.browser|up.link|mmp|symbian|smartphone|midp|wap|p
hone)/i', strtolower($_SERVER['HTTP_USER_AGENT']))) {
    $mobile_browser++;
}

if((strpos(strtolower($_SERVER['HTTP_ACCEPT']),'application/vnd.wap.x
html+xml')>0) or ((isset($_SERVER['HTTP_X_WAP_PROFILE']) or isset($_
    $mobile_browser++;
}

$mobile_ua = strtolower(substr($_SERVER['HTTP_USER_AGENT'],0,4));

$mobile_agents = array(
    'w3c ', 'acs-', 'alav', 'alca', 'amoi', 'audi', 'avan', 'benq',
'bird', 'blac', 'blaz', 'brew', 'cell', 'cldc', 'cmd-', 'dang',
'doco', 'eric', 'hipt', 'inno', 'ipaq', 'java', 'jigs', 'kddi',
'keji', 'leno', 'lg-c', 'lg-d', 'lg-g', 'lge-', 'maui', 'maxo',
'midp', 'mits', 'mmef', 'mobi', 'mot-', 'moto', 'mwbp', 'nec-',
'newt', 'noki', 'oper', 'palm', 'pana', 'pant', 'phil', 'play',
'port', 'prox', 'qwap', 'sage', 'sams', 'sany', 'sch-', 'sec-',
'send', 'seri', 'sgh-', 'shar', 'sie-', 'siem', 'smal', 'smar',
'sony', 'sph-', 'symb', 't-mo', 'teli', 'tim-', 'tosh', 'tsm-', 'upg1',
'upsi', 'vk-v', 'voda', 'wap-', 'wapa', 'wapi', 'wapp', 'wapr',
'webc', 'winw', 'winw', 'xda', 'xda');

if(in_array($mobile_ua,$mobile_agents)) {
    $mobile_browser++;
}

if (strpos(strtolower($_SERVER['ALL_HTTP']),'OperaMini')>0) {
    $mobile_browser++;
}

if (strpos(strtolower($_SERVER['HTTP_USER_AGENT']),'windows')>0) {
    $mobile_browser=0;
}

if($mobile_browser>0)
    //redirect to mobile site
else
    //redirect to desktop site

```

Como hemos mencionado la plataforma Croover consta de 2 dos partes independientes. Una parte desktop, o administradora de ubicaciones. Y una parte móvil, o para agentes móviles que se desplazan por las ubicaciones. A continuación, detallamos las acciones que pueden ejecutar los usuarios desde la parte de administración.

Acceso y registro a la plataforma

El sistema ofrece la posibilidad de registrarse mediante un formulario de datos mínimos, el objetivo es no hacer del proceso de registro algo tedioso para el usuario.

Si la plataforma se está implementando en un contexto local, y éste lugar ya administra un conjunto de personas, se debe crear el web-service correspondiente para comparar los datos, o bien, migrar las personas a la base de datos de Croover

para que todas tengan acceso.

The screenshot shows the Croover website interface. At the top left is the Croover logo, a blue globe with white lines, followed by the text "CROOVER" in a bold, black, sans-serif font. Below the logo is the tagline "Red social de ubicaciones físicas". The main content area is divided into two panels. The left panel, titled "Entrar", contains a login form with fields for "Usuario" and "Contraseña", a "Recordarme" checkbox, and an "Entrar" button. The right panel, titled "Regístrate", contains a registration form with the sub-header "Es gratis y cualquiera puede unirse". It includes fields for "Nombre", "Apellido", "Correo electrónico", and "Contraseña", a "Sexo" dropdown menu, and a "Cumpleaños" section with three dropdown menus and a calendar icon. A "Regístrate" button is at the bottom of the registration form. At the bottom of the page, there is a footer with "[Español | Inglés]" on the left and "Facultad de Informática - UNLP" on the right.

Figura 5.3 Pantalla de acceso y registro a la plataforma

Como explicamos al principio del capítulo 5, Croover fue diseñada pensando en un ambiente público global. Por lo tanto, para el uso académico debe evaluarse la necesidad o no de un registro de usuarios, y si dicho registro se contrastará con otra base de datos.

Creación de una nueva ubicación

Una vez que el usuario accedió al sistema, se le permite definir ubicaciones en el mundo. Para cada una de ellas deberá especificar el nombre de la ubicación, una descripción, una URL en caso de que la ubicación tenga un sitio oficial, también se mostrará un checkbox para determinar si la ubicación es pública (en caso de serla cualquier persona puede escribir mensajes). Y por último el mapa para definir la superficie y puerta de entrada a la ubicación, en la figura 5.4 podemos ver la pantalla de creación de una ubicación en la plataforma Croover.



Nueva ubicación

Nombre

Descripción
Especifique el contenido que se mostrará cuando alguien abra la ubicación

URL Sitio Web
Especificar el sitio web oficial de la ubicación

Público Significa que cualquier persona puede publicar eventos y sucesos en esta ubicación

Mapa **Marcá la entrada y las esquinas de la superficie de tu ubicación**

POWERED BY Google

Chequear para marcar la superficie de la ubicación

[Volver al listado](#) [Guardar](#)

Figura 5.4 Pantalla de creación de una nueva ubicación

Para definir con más exactitud la superficie de una ubicación es conveniente cambiar el tipo de mapa, como nombramos en el apartado de Google Maps. En modo “Satélite” Google nos muestra el territorio por fotografías satelitales como vemos a continuación.

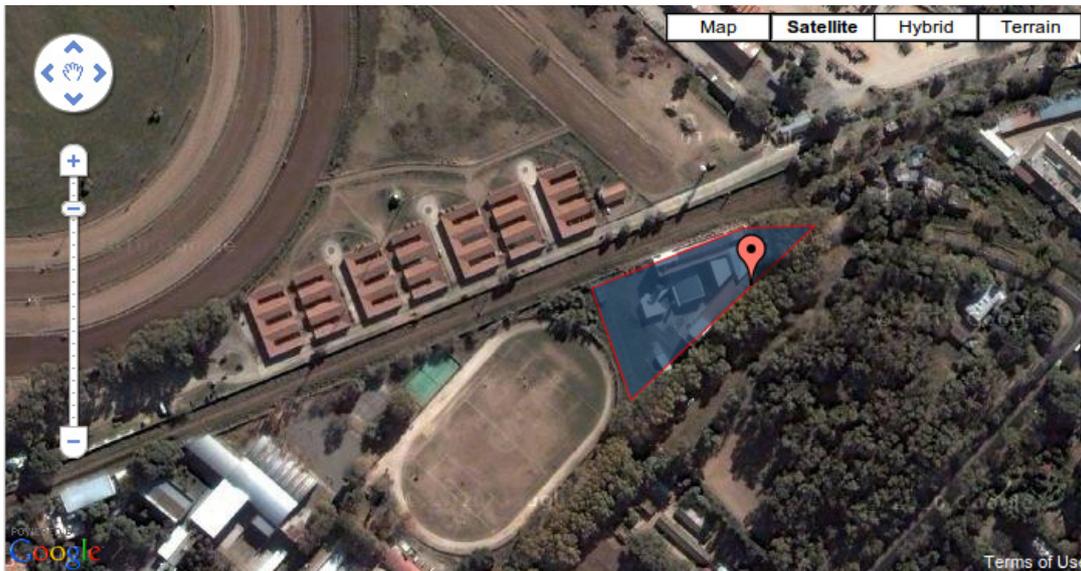


Figura 5.5 Ejemplo de la vista del mapa en modo Satelite

De este modo podemos determinar con exactitud las dimensiones de la superficie que ocupa una ubicación. Y luego se podrá ver del mismo modo en el mapa clásico:

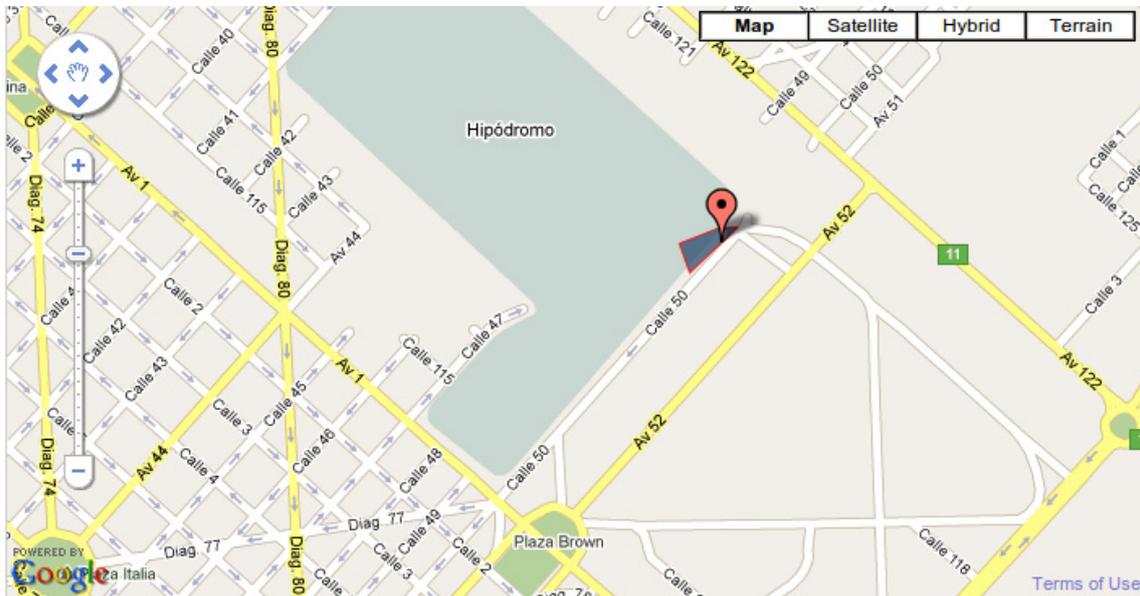


Figura 5.6 Ejemplo de vista del mapa en modo mapa clásico

Listado de ubicaciones

Una vez que el usuario definió sus ubicaciones podrá listarlas. Existen dos listados en el sistema: ubicaciones favoritas y ubicaciones propias. Las ubicaciones creadas por otros usuarios pueden ser marcadas como favoritas para un mejor acceso sin necesidad de ir al buscador de ubicaciones. Cada ubicación en los listados tiene 4 acciones básicas.

- **¿Qué pasa?** Consiste en una única pantalla de visualización de contenidos publicados por los administradores de esa ubicación. Se describe con mayor detalle más adelante en este capítulo.
- **¿Dónde está?** Es la pantalla que muestra en el mapa la ubicación junto con su superficie y ubicaciones relacionadas. Más adelante describimos esta pantalla.
- **Agregar/quitar de favoritos** Existen para las ubicaciones que no le pertenecen al usuario.
- **Ver detalles** Esta acción despliega un pequeño marco con detalles de la ubicación, estado actual (activa o inactiva), cantidad de seguidores (usuarios que la agregaron a sus favoritos), privacidad (si es pública o privada) y un URL en caso de que posea sitio Web oficial.

[Novedades](#) | [Mis ubicaciones](#) | [Mis favoritos](#) | [Mis caminos](#) | [Buscar ubicaciones](#) | [Crear ubicación](#)

Mis ubicaciones

 <p>Biblioteca Biblioteca de la facultad Ver detalles</p> <p>¿Qué pasa? ¿Dónde está?</p>	 <p>Alumnos Oficina de alumnos de la facultad Ver detalles</p> <p>¿Qué pasa? ¿Dónde está?</p>
 <p>Aula 3 Aula 3 Planta Baja Ver detalles</p> <p>¿Qué pasa? ¿Dónde está?</p>	 <p>Aula 4 Aula 4 Planta Baja Ver detalles</p> <p>¿Qué pasa? ¿Dónde está?</p>
 <p>Aula 5 Aula 5 Planta Baja Ver detalles</p> <p>¿Qué pasa? ¿Dónde está?</p>	 <p>Aula 6 Aula 6 Planta Baja Ver detalles</p> <p>¿Qué pasa? ¿Dónde está?</p>

10 resultados (página 1/2) ◀ 1 2 ▶

© 2010 Croover | About us Facultad de Informática - UNLP

Figura 5.7 Ejemplo del listado de ubicaciones que muestra la plataforma

Luego de que el usuario selecciona una ubicación del listado se dirige a la pantalla de publicación de contenidos de la misma, donde podrá visualizar los mensajes, eventos y encuestas publicados. Como podemos ver, también se muestra un header con un conjunto de acciones. Estas acciones varían según el usuario sea dueño de la ubicación o solo un visitante. Las figuras siguientes muestran dicho conjunto de acciones.

 <p>Biblioteca Biblioteca de la facultad 0 seguidores Ubicación pública</p>	<p> Editar Desactivar Obtener código QR Ubicaciones relacionadas </p> <p>¿Qué pasa? ¿Dónde está?</p>
---	---

Figura 5.8 Conjunto de acciones para el dueño de una ubicación

El conjunto de acciones para el dueño de la ubicación permiten la manipulación completa de la misma. Editar sus datos, superficie, ubicación, descripción, nombre y sitio Web. También permite la impresión del código QR, editar las ubicaciones relacionadas y la activación/desactivación de la ubicación.

 <p>Biblioteca Biblioteca de la facultad 0 seguidores Ubicación pública</p>	<p>¿Qué pasa? ¿Dónde está?</p>
---	------------------------------------

Figura 5.9 Conjunto de acciones para un usuario visitante de un ubicación.

Impresión de código QR para la ubicación

La impresión del código QR es una tarea indispensable para la parte administrativa de las ubicaciones, dado que con el código generado los agentes

móviles podrán acceder a los contenidos publicados en las ubicaciones. El contenido del código no es más que la URL que les permitirá acceder a los contenidos directamente. Cada usuario realizará la impresión de su código y lo publicará en la ubicación para que pueda ser fotografiado por el resto de la comunidad.

Para este caso de prueba se muestran dos inputs, uno permite seleccionar el modulo del código QR, como vimos en el apartado de este código de barra, el modulo permite especificar de algún modo el tamaño que va a poseer el mismo. Y en otro input se especifica el tamaño en píxeles que se le quiere dar al texto. Este texto se pone debajo del código QR, y es tomado del nombre de la ubicación, para poder identificar el código en caso de que se estén generando varios, o bien para que no se traspapele con otros códigos a la hora de establecerlos en cada ubicación.

The screenshot shows the CROOVER website interface. At the top, there is a navigation bar with the CROOVER logo and a user profile for Tomás Córdoba. Below the navigation bar, there is a section for the location 'Biblioteca', which includes a QR code, the name 'Biblioteca', and the description 'Biblioteca de la facultad'. To the right of this section, there are several action buttons: 'Editar', 'Borrar', 'Obtener código QR', 'Ubicaciones relacionadas', '¿Qué pasa?', '¿Dónde está?', and 'Agregar a favoritos'. Below this section, there is a dark bar with the text 'Imprimir código QR para la ubicación: "Biblioteca"'. Underneath this bar, there is a form with two input fields: 'Módulo' (set to 11) and 'Tamaño de la fuente' (set to 32). Below the form, there are two buttons: 'Imprimir' and 'Recargar'. Below the form and buttons, there is a large QR code. Below the QR code, the word 'Biblioteca' is printed in a bold, black font.

Figura 5.10 Impresión del código QR de la ubicación

La integración de PHP y los códigos de barra Quick Response, se da por medio de clases que permiten la generación del mismo. En este caso específico, se utiliza un plugin de Symfony denominado QRcodePlugin [28]. Este plugin contiene un helper, que se utiliza de la siguiente manera en cualquier template del proyecto:

```
<?php use_helper('QRcode')?>

<?php echo QRcode('I love Symfony');?>
```

Esa dos líneas de código, la primera invoca al helper para poder hacer uso de sus funciones en el template, y la segunda línea llama a la función del helper `QRcode()`; que genera una imagen de un código QR con el contenido 'I love Symfony'.

A continuación se detalla la cabecera de la función `QRcode()` del plugin:

```
# d= data          URL encoded data.
# e= ECC level     L or M or Q or H   (default M)
# s= module size  (dafault PNG:4 JPEG:8)
# v= version      1-40 or Auto select if you do not set.
# t= image type   J:jpeg image , other: PNG image
#
# structured append m of n (experimental)
# n= structure append n (2-16)
# m= structure append m (1-16)
# p= parity
# o= original data (URL encoded data) for calculating parity
#
function QRcode($d, $e = 'M', $s = null, $v = null, $t = 'P',
$m = null, $n = null, $o = null, $p = null)
```

Como se puede observar, el primer parámetro es el único requerido y es el contenido del código de barras a generar. Los demás parámetros como se explican en el encabezado son para los parámetros de la generación del código, tamaño del modulo, versión, tipo de imagen a generar (jpg, png), datos de paridad, etc.

¿Dónde se encuentra una ubicación?

Esta funcionalidad sirve para la visualización de la ubicación una vez que fue creada, tanto para el usuario dueño como para los demás usuarios. Además se muestra una lista de las ubicaciones relacionadas y una lista de las ubicaciones cercanas, que se calculan en base a un radio de cuatro cuadras aproximadamente.

Novedades | Mis ubicaciones | Mis favoritos | Mis caminos | Buscar ubicaciones | Crear ubicación



Biblioteca
Biblioteca de la facultad
0 seguidores
Ubicación pública

Editar
¿Qué pasa?

Desactivar
¿Dónde está?

Obtener código QR

Ubicaciones relacionadas

Ubicación y lugares relacionados

La ubicación se muestra en el mapa. A la izquierda podrás encontrar algunas ubicaciones relacionadas o cercanas a esta ubicación. Si haces click sobre ellas, un marcador aparecerá en el mapa para visualizarla. Y si haces click sobre el marcador en el mapa, el sistema calculará la distancia y el camino que tendrás que realizar para llegar desde la ubicación actual.

Ubicaciones relacionadas

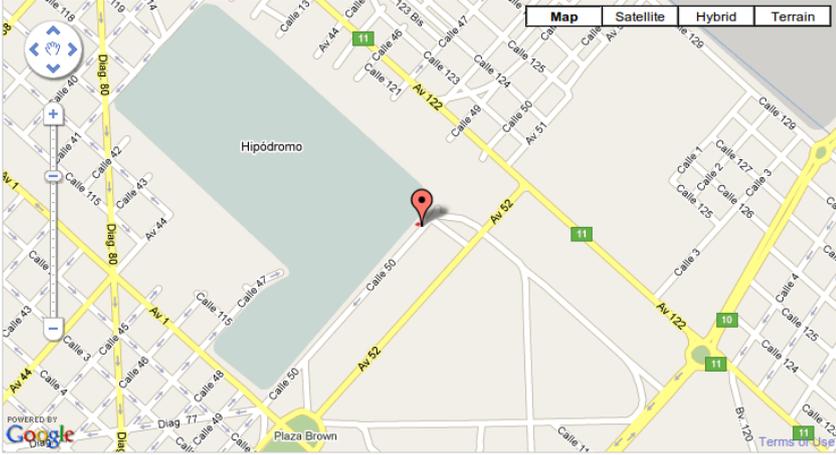
Alumnos

Aula 3

Ver todas las ubicaciones

Ubicaciones cercanas

Sin resultados



© 2010 Croover | About us
Facultad de Informática - UNLP

Figura 5.11 Dónde se encuentra una ubicación

Como podemos ver inicialmente en el mapa se muestra la ubicación que estamos consultando. Sobre la izquierda se puede ver una barra con las ubicaciones relacionadas y las ubicaciones cercanas. Si hacemos click sobre alguna de ellas el mapa se va a actualizar con el marcador de la ubicación que hicimos click, el color del marcador depende del tipo de relación. Las relaciones por cercanía se muestran con marcadores amarillos y las ubicaciones relacionadas con marcadores azules, como se ve en la imagen anterior.

En esta pantalla, además, podemos ver el camino que tendríamos que recorrer si quisiéramos ir desde la ubicación que estamos consultando a cualquiera de las ubicaciones relacionadas o cercanas. Esto se realiza por medio de Driving directions explicadas en el capítulo 4. Para que se muestre dicho camino se debe hacer click sobre el marcador de la ubicación a la cual nos queremos dirigir. Esta funcionalidad quizás no es de mucho uso en la aplicación desktop pero es el corazón de la aplicación móvil, ya que los usuarios se están moviendo físicamente y necesitan saber cómo llegar.

¿Qué está pasando en una ubicación?

Nos muestra en una sola pantalla los últimos mensajes publicados, las encuestas y eventos activos. Es la pantalla de comunicación central, donde se resumen todos los “movimientos” que hubo o que habrá en dicha ubicación.

The screenshot shows the Croover interface for a location named 'Facultad de Informatica - UNLP'. At the top, there is a navigation bar with links: 'Mis ubicaciones', 'Mis favoritos', 'Mis caminos', 'Buscar ubicaciones', and 'Crear ubicación'. The location details include a QR code, '0 seguidores', and 'Ubicación privada'. Action buttons include 'Editar', 'Borrar', 'Obtener código QR', 'Ubicaciones relacionadas', '¿Qué pasa?', '¿Dónde está?', and 'Agregar a favoritos'. Below the location details, there are three sections: 'Encuestas' (No hay encuestas activas), 'Mensajes' (No hay mensajes nuevos), and 'Eventos' (No hay eventos activos). Each section has a 'Ver todos' button and a '+ Crear nuevo...' button.

Figura 5.12 ¿Qué está pasando en la ubicación? Ejemplo sin eventos, ni mensajes, ni encuestas publicadas

This screenshot shows the same location as Figure 5.12, but with active content. The 'Encuestas' section displays a poll titled '¿Que opina del nuevo edificio de la facultad?' with a bar chart showing 100% 'Muy Bueno' (1 vote) and 0% for 'Bueno', 'Regular', and 'Malo'. The 'Mensajes' section shows a message from Tomás Córdoba about a new office hours schedule for students, posted 5 minutes ago. The 'Eventos' section features an event titled 'Conferencia Software Libre' scheduled for 2010-09-17 at 14:00:00 hs, with a list of topics and a '¿Vas a venir?' poll.

Figura 5.13 ¿Qué está pasando en la ubicación? Ejemplo con eventos mensajes y encuestas publicadas

Cargar un nuevo mensaje en una ubicación

Croover maneja un apartado para el tratamiento de mensajes, tanto públicos (llegan a todos los usuario y todos los pueden ver) como privados (solo pueden ser vistos por la persona a la cual va dirigido). Como vemos en la figura siguiente tenemos

dos acciones en la parte de mensajes. Un enlace con la leyenda “Contar que sucede” y otro con la leyenda “Nuevo mensaje a usuario”.

The screenshot shows a messaging interface with a dark header labeled "Mensajes". Below the header is a message card with the text: "De: Tomás Córdoba", "Nuevo catálogo de libros en biblioteca!", and "Hace aproximadamente 4 minutos". Underneath the message card is a light blue form area. It starts with a "Para:" label followed by a large empty text input field. At the bottom right of this form area is an "Enviar" button. Below the form area, there are three navigation links: "Ver todos" (with a document icon), "Contar que sucede" (with a plus icon), and "Crear mensaje para usuario" (with a plus icon).

Figura 5.14 Envío de mensajes a un usuario específico

En el campo de destinatario “para” el usuario comienza a tipear el nombre, email o usuario Twitter de la persona a la cual le desea mandar un mensaje y se despliega una lista de opciones, es decir, funciona con el autocomplete de jQuery.

The screenshot shows a messaging interface with a dark header labeled "Mensajes". Below the header is a message card with the text: "De: Tomás Córdoba", "Nuevo catálogo de libros en biblioteca!", and "Hace aproximadamente 42 minutos". Underneath the message card is a light blue form area. It features a large empty text input field. Below the input field, there is a "Publicar en" label followed by two social media icons: Twitter and Facebook, each with a small square selection box next to it. At the bottom right of this form area is an "Enviar" button. Below the form area, there are three navigation links: "Ver todos" (with a document icon), "Contar que sucede" (with a plus icon), and "Crear mensaje para usuario" (with a plus icon).

Figura 5.15 Ejemplo de publicación de un mensaje público

En este caso el mensaje es visto por todas las personas que visitan la ubicación y trae además la posibilidad de publicar el mismo mensaje en las redes sociales más conocidas actualmente: Twitter y Facebook.

Encuestas y eventos

Croover permite la definición de encuestas y eventos, estos se muestran en la pantalla de contenidos como vimos en la figura 5.12. Podemos apreciar mejor la vista en la figura siguiente, que nos muestra la sección de eventos y encuestas vacíos.



Figura 5.16 Pantalla de eventos y encuestas vacíos

Los eventos se definen en un día y horario específico en una ubicación. Se solicita también un nombre y una breve descripción en la creación de un nuevo evento, luego los usuarios que visitan la ubicación pueden marcar si asistirán o no dicho evento como lo muestra la figura 5.17. El administrador, por otra parte, puede descargar un archivo XLS con el listado de las personas que asistirán para tener un mejor control de las personas que concurrirán.



Figura 5.17 Ejemplo de un evento publicado, con la acción para administradores de descargar el listado de personas que asistirán

Las encuestas están compuestas de un formato muy sencillo. Se permite la creación de campos radio, para una selección individual, o checkbox, para una selección múltiple. Como vemos en la figura 5.18, luego de establecer el nombre de la encuesta, Croover nos permite editar los campos de la misma. El usuario debe definir una etiqueta y un tipo (que establece si es un campo radiobutton o checkbox) para cada uno de ellos. La encuesta, una vez creada, aparecerá en la pantalla de visualización de contenidos donde los usuarios podrán realizar su voto como se muestra en la figura 5.19. Una vez que el usuario vota, puede ver los resultados

temporales de la encuesta como muestra la figura 5.20.

Logo: CROOVER

Salir
Tomás Córdoba [Español | Inglés]

Novedades | Mis ubicaciones | Mis favoritos | Mis caminos | Buscar ubicaciones | Crear ubicación

Editar campos de la encuesta

✓ Encuesta creada, defina los campos que la componen

Agregar un campo nuevo

Etiqueta

Tipo

Agregar campo

Vista previa de la encuesta

¿Piensa que esta plataforma tendrá éxito?

Terminar encuesta

© 2010 Croover | Quiénes somos Facultad de Informática - UNLP

Figura 5.18 Pantalla de edición de campos de una encuesta sobre la derecha se visualiza la vista previa de cómo está quedando la encuesta.

Encuestas

¿Piensa que esta plataforma tendrá éxito?

- Si, estoy seguro
- Debemos esperar la respuesta de la gente
- Es poco probable
- No, no lo creo

Votar

Ver todos + Crear nueva encuesta

Figura 5.19 Ejemplo de una encuesta que aún no fue respondida por el usuario

Encuestas

¿Piensa que esta plataforma tendrá éxito?

Si, estoy seguro: 43% 3 votos

Deberíamos esperar la respuesta de la gente: 29% 2 votos

Es poco probable: 29% 2 votos

No, no lo creo: 0% 0 votos

Ver todos + Crear nueva encuesta

Figura 5.20 Ejemplo de una encuesta en la que el usuario ya realizó su voto y visualiza los resultados.

Camino recorrido

Cuando un usuario vuelve a su casa, puede querer saber por dónde anduvo para observar mejor los contenidos o por simple curiosidad. Esta funcionalidad permite al usuario dar un recorrido virtual por las ubicaciones que ha visitado. Permite seleccionar una fecha y visualizar todos los lugares que visitó desde la aplicación móvil únicamente, uniendo cada punto mediante el uso de Driving directions de la API de Google Maps y en el orden correcto en el que fueron visitados.

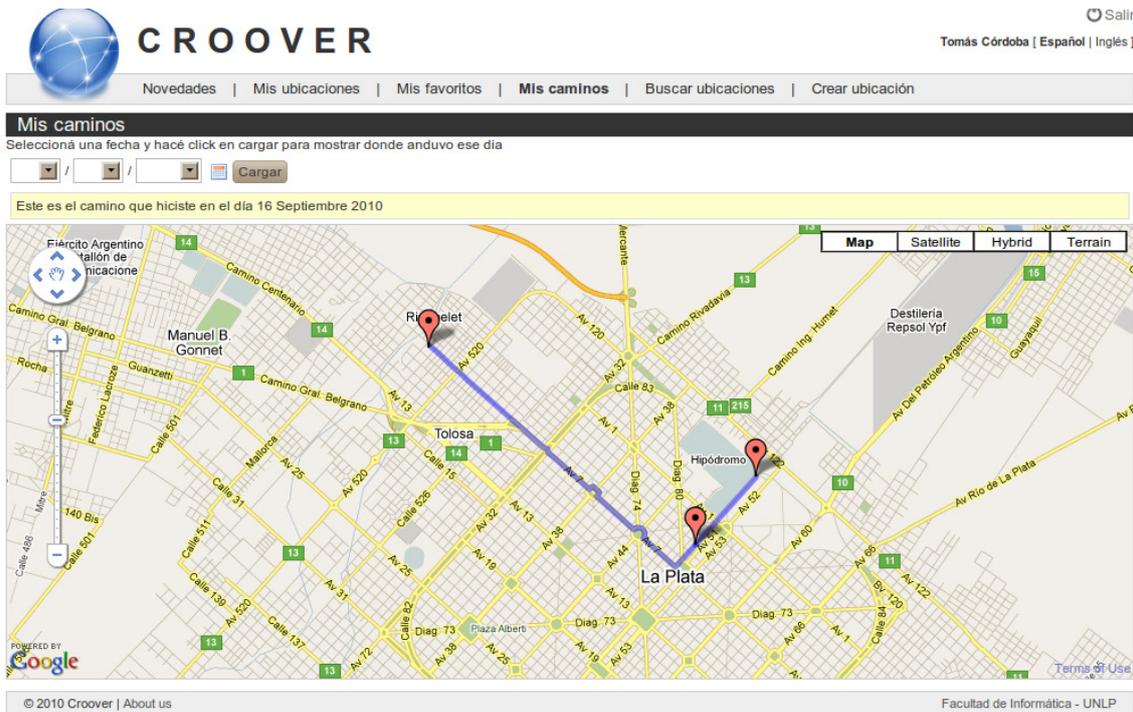


Figura 5.21 Ejemplo de la pantalla caminos recorridos en un día específico

Los usuarios además disponen de un resumen de los movimientos realizados en sus ubicaciones y en las ubicaciones favoritas de ellos. Esta sección se denomina “Novedades”. Y es lo primero que ve el usuario al ingresar al sistema. La imagen siguiente nos muestra la apariencia que posee.

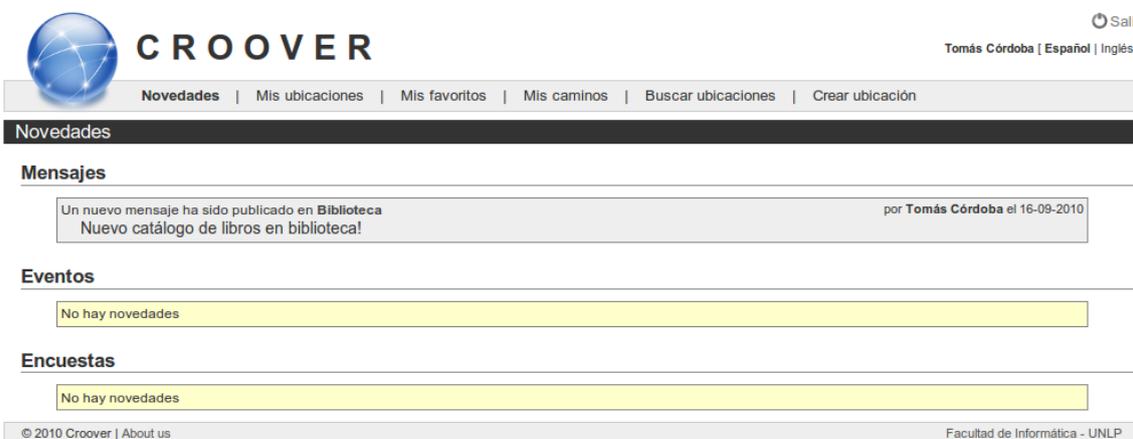


Figura 5.22 Resumen de actividades en las ubicaciones: “Novedades”

Luego de haber analizado la parte desktop para los usuarios que gestionan las ubicaciones, podemos enfocarnos en la aplicación móvil. Como se hizo mención anteriormente la parte móvil también se realizará sobre Web y a diferencia de la desktop, tendrá un conjunto de acciones más limitados y un estilo CSS acorde al tamaño de la pantalla. En este caso se desarrollo y testeó sobre el browser Safari para iPhone e iPod.

La pantalla de inicio de sesión es similar a la desktop a diferencia de que no se pueden registrar nuevos usuarios desde la parte móvil. Se requiere un primer registro vía Web desktop para comenzar a navegar las ubicaciones.



Figura 5.23 Pantalla de inicio de sesión en dispositivo móvil

Una vez dentro del sistema podemos ver las noticias de la misma manera que lo hacíamos en la aplicación desktop a diferencia de algunas cuestiones de estilos y tamaño de pantalla. El menú principal de navegación solo tiene dos ítems “Novedades” y “Ubicaciones”, el objetivo es minimizar las acciones y optimizar el acceso a los contenidos mediante una interfaz sencilla y fácil de usar.

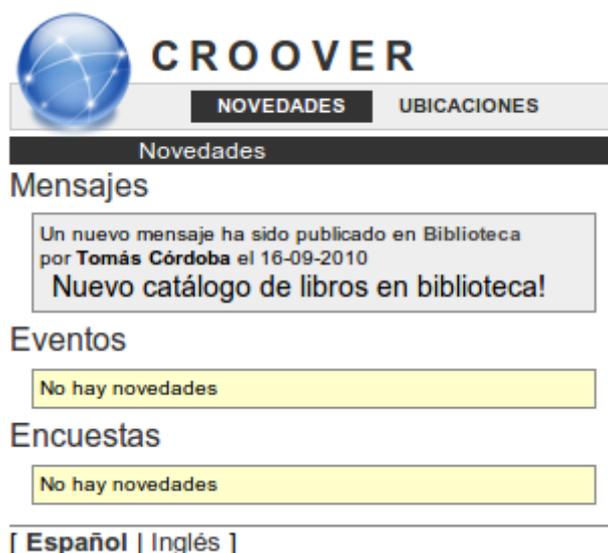


Figura 5.24 Pantalla de novedades de la aplicación móvil

Para facilitar el uso, uno de los cambios más importantes se produjo en el listado de ubicaciones. En este se incluyó un input para realizar búsqueda de lugares más rápidamente y se quitó la imagen del código QR dejando solo el nombre y la descripción de la ubicación. El objetivo es facilitar el movimiento del usuario entre los distintos spots en caso de que no quiera tomar la fotografía correspondiente a dicha ubicación o tenga problemas con su cámara o bien no se encuentre el código QR.



Figura 5.25 Listado de ubicaciones en la aplicación móvil

Cuando el usuario hace click en las acciones del listado, puede dirigirse al mapa que indica donde está la ubicación o a la acción "¿Qué pasa?" que muestra todos los contenidos de la ubicación. A diferencia de la aplicación gestora de ubicaciones, la aplicación móvil muestra los contenidos en una sola columna para mejorar la interfaz de usuario. También se agrega una columna que muestra los mensajes que van dirigidos especialmente al usuario.



Figura 5.26 ¿Qué está pasando? En la aplicación móvil

En la acción “¿Dónde está?” se muestra el mapa con la ubicación geográfica de la misma, y además se muestran todas las ubicaciones relacionadas y cercanas. En la siguiente figura vemos la ubicación y dos ubicaciones relacionadas que se muestran con marcadores azules.



Figura 5.27 Mapa con ubicaciones relacionadas

Si el usuario quiere dirigirse a alguna de estas ubicaciones relacionadas debe hacer click en el marcador para que se le muestre la ventana de información como se

muestra en la figura 5.28. En la ventana encontrará la acción “Ir hasta aquí”, la cual mediante Google Driving Directions le mostrara el camino que debe recorrer como se muestra en la figura 5.29.



Figura 5.28 Mapa con ventana de información desplegada de una ubicación relacionada

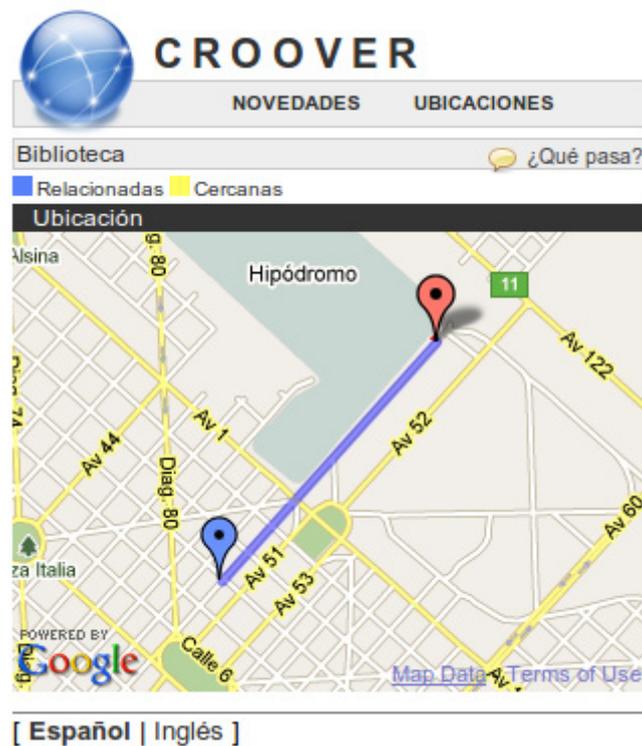


Figura 5.29 Mapa luego de haber calculado la ruta con Google Driving Directions

Con esta última acción completamos la navegación física de los elementos gracias al cálculo de rutas ofrecido por Google. Pero aún quedan algunos cabos sueltos respecto al ciclo completo de interacción del usuario y el sistema. ¿Por qué no desarrollar una aplicación completa que tome la fotografía del código QR? La verdad es que existen tantos softwares que realizan esta acción que resulta poco productivo implementar uno nuevo y estar atentos de ofrecer soporte por si se producen errores o necesita mantenimiento. Es por esto que se decide tercerizar esta parte. Por otro lado Croover toma al agente móvil como un usuario de consulta, es decir, no tiene la capacidad de publicar contenido en una ubicación.

Idioma de la aplicación

El idioma de la aplicación puede ser cambiado desde los links correspondientes: en la aplicación móvil desde abajo a la izquierda y en la aplicación desktop desde arriba a la derecha al lado del nombre de la persona. Croover fue desarrollada en su totalidad en idioma ingles y traducida frase por frase en dos archivos XML que se encuentran en el directorio "apps/backend/i18n/es/". La estructura de los mismos es la siguiente y es la necesaria para que el sistema de traducción de Symfony funcione correctamente:

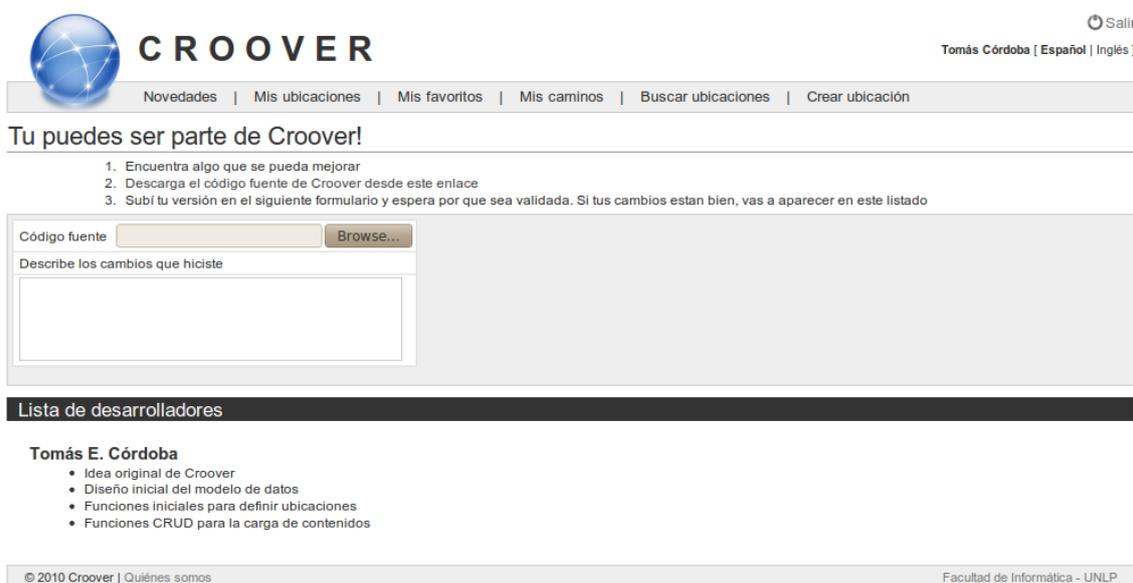
```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE xliiff (View Source for full doctype...)>
<xliiff version="1.0">
<file source-language="EN" target-language="es" datatype="plaintext"
original="messages" date="2010-05-09T19:26:18Z" product-
name="messages" xml:space="default">
<header />
<body>
  <trans-unit id="1">
    <source>Homepage</source>
    <target>Inicio</target>
  </trans-unit>
  <trans-unit id="2">
    <source>Something</source>
    <target>Algo</target>
  </trans-unit>
</body>
```

El XML de Croover está compuesto de 1500 líneas de código aproximadamente. Si se quisieran anexar algunos idiomas mas, Symfony provee una tarea que genera el XML correspondiente para todas las frases traducidas encontradas en el sistema. De este modo solo se necesita buscar las traducciones correspondientes a cada frase e ir actualizando el XML hasta que quede completamente traducido. La función que se encarga de traducir una frase en el código PHP es "__('Frase a traducir')", doble guión bajo y la palabra o frase a traducir.

El modelo presentado en el caso de prueba es un modelo muy simplificado de lo que puede llegar a ser una plataforma concluida en su totalidad. Solo se presenta un funcionamiento básico, para la definición de ubicaciones, recorridos y administración de contenidos. La idea es dejar un sistema escalable para su futura ampliación.

En primera instancia se realizará una prueba a nivel global. Se publicará la plataforma en la URL www.croover.com.ar para que todas las personas puedan hacer uso de ella. De esta forma se evaluará la aceptación de la sociedad de una plataforma de hipermedia física global.

Además se pondrá a disposición de los desarrolladores el código fuente completo de la plataforma, para que puedan realizar mejoras e implementar nuevas ideas. Luego de que un desarrollador realice un cambio o mejora, se evaluará los archivos que este modificó y si es válida la modificación se actualizará el servidor con la nueva versión. Desde la plataforma se podrá encontrar un link "About us", a un listado de las personas que llevaron a cabo la plataforma como se ve en la figura 5.30. Cada cambio validado que realice una persona será agregado a esta lista. Haciendo de Croover una plataforma desarrollada por toda la sociedad.



The screenshot shows the Croover website interface. At the top left is the Croover logo, a blue globe with a network pattern. To its right is the word "CROOVER" in a bold, sans-serif font. Further right, there is a "Salir" button and the name "Tomás Córdoba" with language options "Español" and "Inglés". Below the logo and name is a navigation bar with links: "Novedades", "Mis ubicaciones", "Mis favoritos", "Mis caminos", "Buscar ubicaciones", and "Crear ubicación".

Below the navigation bar is a section titled "Tu puedes ser parte de Croover!". It contains a numbered list of three steps: 1. Encuentra algo que se pueda mejorar; 2. Descarga el código fuente de Croover desde este enlace; 3. Subí tu versión en el siguiente formulario y espera por que sea validada. Si tus cambios estan bien, vas a aparecer en este listado.

Below the list is a form for submitting source code. It has a text input field labeled "Código fuente" with a "Browse..." button next to it. Below this is a larger text area labeled "Describe los cambios que hiciste".

Below the form is a section titled "Lista de desarrolladores". It lists "Tomás E. Córdoba" with a bulleted list of his contributions: "Idea original de Croover", "Diseño inicial del modelo de datos", "Funciones iniciales para definir ubicaciones", and "Funciones CRUD para la carga de contenidos".

At the bottom of the page, there is a footer with "© 2010 Croover | Quiénes somos" on the left and "Facultad de Informática - UNLP" on the right.

Figura 5.30 Listado de desarrolladores que realizaron mejoras y se consideran parte de Croover

Como se puede ver en la figura 5.30 aparecen una serie de pasos a realizar para ser parte de la plataforma. Primero la persona debe encontrar alguna funcionalidad que se pueda mejorar, o bien, que se pueda agregar. Luego el usuario se descarga el código fuente y realiza las modificaciones necesarias. Una vez que termina, por medio del formulario que aparece más abajo, sube su versión mejorada. En el campo de descripción debe especificar cuáles fueron sus cambios introducidos y podría especificar además los archivos que modificó. Luego, la nueva versión es evaluada por el equipo de desarrollo de Croover para que no existan problemas de seguridad o código malicioso. Una vez aceptadas las modificaciones se actualiza el servidor y en el listado de desarrolladores aparecerá el nombre de la persona y los cambios que realizó.

5.4 Desarrollo en un ámbito académico

Una plataforma de hipermedia física puede implementarse en muchos contextos diferentes, en un ambiente local, como puede ser una facultad, un shopping,

una empresa, etc. O en un ambiente global, en una dirección pública donde todas las personas accedan y creen sus ubicaciones, como si fuese una red social.

En el caso particular de la implementación en una universidad debemos definir algunas cuestiones antes de comenzar. Como primer paso la identificación de los usuarios ¿todos los alumnos pueden acceder?, ¿profesores?, ¿personal administrativo? Se supone que se configura este tipo de plataforma para mejorar la comunicación entre alumnos y personal de la universidad, pero es importante detallar que rol cumple cada uno.

Los alumnos deben poder acceder a los contenidos pero no publicar nuevos contenidos en ellas, por lo tanto las ubicaciones son de tipo “privadas” como mencionamos anteriormente en la creación de una ubicación. Los profesores en cambio, deben tener la posibilidad de publicar contenidos en cada ubicación.

¿Qué pasa cuando se quiere identificar un aula como materia?. Pensemos lo siguiente: si tenemos un aula que dicta muchas materias ¿tenemos muchas ubicaciones en un mismo lugar? Lo mejor para estos casos depende directamente de la institución en la cual se está implementando. Puede resolverse creando varias ubicaciones en el mismo lugar, cada una de ellas representando la materia que se dicta en ese lugar, de modo que quedaría en el listado de ubicaciones algo así como:

- Aula 5 – Programación de computadoras
- Aula 5 – Arquitectura de computadoras
- Aula 5 – Organización de computadoras

Esto funcionaría y sería de utilidad si se publica mucho contenido en para cada una de las ubicaciones, entonces tener todo como una única ubicación generaría mucha confusión de publicaciones. Pero si los contenidos a publicar son mínimos y se quiere mantener un orden en las ubicaciones es mejor tener solo una ubicación física “Aula 5”.

Debemos hablar también de cómo los datos de los alumnos son obtenidos ¿a que nos referimos con esto? Es normal que todas las instituciones tengan la información de sus alumnos en algún sistema de gestión, por lo tanto se debe analizar si se desea obtener todos los datos por medio de un Web Service o se obligará a los alumnos a registrarse en la plataforma de hipermedia física para poder utilizarla. Si se decide que los alumnos deben registrarse para utilizar la plataforma entonces se debe establecer si queremos que los datos ingresados en ese registro sean correspondidos con los datos cargados en el sistema de gestión de alumnos de la institución. Para evitar tener usuarios falsos dentro de un contexto local como lo es una universidad.

Otro aspecto a tener en cuenta en un ámbito académico es la posibilidad de informar a los alumnos sobre su situación actual con una ubicación. Por ejemplo, en biblioteca si un alumno adeuda un libro, o bien tiene una reserva que ya puede retirar, o el caso de la oficina de alumnos si tuviese tramites sin terminar, o ya terminados y listos para retirar. Por lo tanto, es necesario poder enviar mensajes directos a un usuario para que este cuando visite la ubicación desde su dispositivo móvil sea notificado automáticamente, sin necesidad de que pregunte personalmente. Ahorrando así tiempo de espera por atención, en caso de que haya colas de personas esperando por ser atendidas.

Los profesores deben ser administradores de ubicaciones, para publicar contenidos y que estén disponibles para los alumnos. Con respecto a los contenidos que estos publican debemos saber que el caso de prueba presentado funciona de la

misma manera que lo hace Twitter, si se desea publicar un documento, sitio Web o video se realiza mediante un link externo. Para no duplicar información sin sentido y mantener el servidor donde corre la plataforma con un bajo consumo de recursos.

5.5 Otros contextos de aplicabilidad

Existen muchos usos diferentes para este tipo de plataformas, si bien todas están orientadas a identificación de ubicaciones en el mundo real, no todos los usuarios acceden a este tipo de sistemas en busca de lo mismo. Los usuarios que desean publicar su empresa no pretende solo estar en contacto con el mundo y ser ubicado, también puede querer promocionar su negocio por medio de este tipo de herramientas. Para lo cual se puede anexar agregados que permitan este tipo de funcionalidades, como publicación de promociones temporales, del día por ejemplo.

Hoy en día las empresas se enfocan en captar los intereses individuales y colectivos de los clientes sin ser invasivos. Se ha comprobado que las ganancias producidas por el marketing invasivo son mínimas y dejan a la empresa mal vista. Todos hemos sufrido campañas invasivas en nuestros teléfonos celulares vía SMS o en nuestras cuentas de emails, y eso hace que nos acordemos de las empresas que nos hacen eso, pero nos acordamos de una mala manera.

El concepto de “proximity marketing” es lo que hoy las organizaciones buscan desarrollar para llegar mejor a sus clientes sin espantarlos y sin dejar una mala imagen de la empresa. Marketing de proximidad es la distribución inalámbrica localizada de contenido publicitario asociado a un determinado lugar. Las transmisiones pueden ser recibidas por las personas en ese lugar que deseen recibirlos y tienen el equipo necesario para hacerlo.

La distribución puede ser a través de una emisión tradicional localizada, o más comúnmente, específicamente dirigido a los dispositivos que se sabe en un área particular.

La ubicación de un dispositivo puede ser determinada por:

- Un teléfono celular estando en una celda en particular
- Un dispositivo Bluetooth o Wi-Fi están dentro del alcance de un transmisor.
- Un dispositivo conectado con Internet con el GPS que le permite solicitar contenido localizado desde los servidores de Internet.

Las comunicaciones podrán ser más orientadas a grupos específicos dentro de un lugar determinado, por ejemplo, en el contenido de los puntos calientes turísticos sólo podrá ser distribuida entre los dispositivos registrados fuera del área local.

Las comunicaciones podrán ser tanto tiempo y lugar específico, por ejemplo, contenido en un lugar de la conferencia puede depender del caso en curso.

Utilización del marketing de proximidad incluyen la distribución de los medios de comunicación en los conciertos, la información (enlaces en los servicios locales), juegos y aplicaciones sociales, y la publicidad.

La utilización de una plataforma de hipermedia física con códigos QR para acceso hace que el marketing de proximidad sea aun más eficiente ya que el usuario accede tomando la fotografía al código de barra por su propia voluntad, sin que la plataforma realice el sondeo de dispositivos. Utilizando así la premisa de que si el usuario accede al contenido es porque siente interés.

Pero, ¿Cómo utilizar la plataforma a favor de una organización? Uno de los recursos más caros que enfrenta una empresa es el acceso a información. La información es difícil de conseguir, y difícil de mantener al día. Una plataforma de hipermedia física, puede ofrecer beneficios significativos a la hora de hacer encuestas, una encuesta puede ser combinada con una promoción a favor de los clientes para que estos se sientan atraídos. Es decir, responder una breve encuesta a cambio de acceder a un beneficio. La empresa recibe la opinión de los clientes la cual es de mucho valor en la hora de toma de decisiones en el futuro de la organización.

5.6 Actualidad Web, browsers y accesibilidad

Día a día la W3C actualiza sus documentos de especificaciones para lograr una Web más funcional, accesible y completa. Los documentos que más a la espera están por parte de los desarrolladores son:

- **HTML5 - A vocabulary and associated APIs for HTML and XHTML**
- **Introduction to CSS3**
- **Mobile Web Best Practices**

El primer documento es la referencia completa para desarrollar aplicaciones Web en HTML5. HTML5 promete muchas mejoras y cambios significativos, como la introducción de audio y video sin necesidad de Adobe Flash como en la versión actual.

Por otra parte, el documento que especifica cómo es la estructura de las hojas de estilo CSS. En un principio los estilos CSS fueron muy acotados, hoy en día ya se pueden utilizar fuentes para texto propias y muchas cosas más. Si bien existe una lucha por parte de los desarrolladores con respecto a los motores de traducción CSS de los browsers, un documento que defina la estructura completa puede solucionar muchos de estos problemas, ya que una página Web en IE no se visualiza de la misma forma que en Mozilla o Chrome.

Mobile Web Best Practices, es una propuesta de la W3C para generar un documento que especifique todas las normas y puntos a tener en cuenta a la hora de generar una plataforma Web para dispositivos móviles. A continuación se detallan los puntos más importantes de cada uno de estos documentos.

HTML5 y CSS3 son dos de las novedades que se discuten hoy en día en el sector de desarrollo Web. Aunque actualmente no hay lanzamiento oficial desde la W3C, si existen documentos parciales de cómo va tomando forma la especificación de la nueva versión de HTML5.

HTML ha estado en continua evolución desde su introducción a la Internet a principios de 1990. Algunas características fueron introducidas en las especificaciones,

mientras que otros se introdujeron en versiones de software. En algunos aspectos, las implementaciones y prácticas de autores han convergido entre sí.

HTML4 se convirtió en una Recomendación del W3C en 1997. A pesar de que sigue sirviendo como una guía aproximada de muchas de las características básicas de HTML, no proporciona suficiente información para construir implementaciones que operen entre sí y, sobre todo, con una masa crítica de contenido desplegado. Lo mismo ocurre con XHTML1, que define una serialización XML para HTML4 y DOM Level 2 HTML, que define las API de JavaScript para HTML y XHTML. HTML5 sustituirá a estos documentos.

El proyecto de HTML5 refleja un esfuerzo, iniciado en 2004, para estudiar las contemporáneas implementaciones HTML y el contenido desplegado. El proyecto consiste en:

1. Definir un lenguaje sencillo llamado HTML5 que puede escribirse en la sintaxis HTML y en la sintaxis XML.
2. Definir los modelos detallados de procesamiento para fomentar las implementaciones interoperables.
3. Mejorar el marcado de documentos.
4. Presentar marcado y APIs para lenguajes emergentes, tales como la aplicación Web.

Los siguientes elementos se han introducido para mejorar la estructura:

- **section** representa un documento genérico o de la sección de solicitud. Se puede utilizar junto con los h1, h2, h3, h4, h5, h6 para indicar la estructura del documento.
- **article** representa una pieza independiente del contenido de un documento, como una entrada de blog o artículo de periódico.
- **aside** representa una pieza de contenido que es sólo ligeramente en relación con el resto de la página.
- **hgroup** representa el encabezado de una sección.
- **header** representa un grupo de ayudas de introducción o la navegación.
- **footer** representa un pie de página de una sección y puede contener información sobre el autor, la información de copyright, etc.
- **nav** representa una sección del documento se destina para la navegación.
- **figure** representa un fragmento de contenido de flujo en sí misma, generalmente se hace referencia como una sola unidad de la corriente principal del documento.

```
<figure>
  <video src="ogg"> </video>
  <figcaption> Ejemplo </figcaption>
</figure>
```

figcaption se puede utilizar como título (es opcional).

Luego, hay otros varios elementos nuevos:

- **video** y **audio** para el contenido multimedia. Ambos proporcionan una API de manera que los autores aplicación puede script su propia interfaz de usuario, pero también hay una forma de activar una interfaz de usuario proporcionada por el agente de usuario. elementos de origen se utilizan junto con estos elementos si hay varias secuencias disponibles de los diferentes tipos.
- **embed** se utiliza para integrar el contenido del plugin.
- **mark** representa una racha de texto en un documento marcado o marcado para fines de referencia, debido a su importancia en otro contexto.
- **progress** representa una realización de una tarea, como la descarga o al realizar una serie de costosas operaciones.
- **meter** representa una medición, como el uso del disco.
- **time** representa una fecha y / u hora.
- **rubí, rt** y **rp** permiten anotaciones para marcar rubí.
- **wbr** representa una oportunidad de salto de línea.
- **canvas** es usada para la renderización de gráficos de mapas de bits dinámicos sobre la marcha, como gráficos o juegos.
- **command** representa un comando que el usuario puede invocar.
- **details** representa la información adicional o los controles que el usuario puede obtener en la demanda. El elemento de resumen se ofrece el resumen, la leyenda, o el título.
- **datalist** junto con el atributo una nueva lista para la entrada puede ser utilizado para hacer cuadros combinados:


```
<input list="browsers">
  <datalist id="browsers">
    <option value="Safari">
    <option value="Internet Explorer">
    <option value="Opera">
    <option value="Firefox">
  </datalist>
```
- **keygen** representa el control para la generación de par de claves.
- **output** representa algún tipo de salida, como de un cálculo realizado a través de secuencias de comandos. [\[14\]](#)

El atributo del elemento **input** denominado **type**, que determina el tipo de campo que se está definiendo ahora tiene los siguientes valores nuevos:

- tel
- search

- url
- email
- datetime
- date
- month
- week
- time
- datetime-local
- number
- range
- color [14]

La idea de estos nuevos tipos es que el agente de usuario puede proporcionar la interfaz de usuario, como un selector de fechas de calendario o la integración con la libreta de direcciones del usuario, y presentar un formato definido en el servidor. Se le da al usuario una mejor experiencia como su entrada se comprueba antes de enviarlo al servidor de lo que significa que es menos tiempo de espera para la retroalimentación.

HTML5 ha introducido varios nuevos atributos a varios elementos que ya formaban parte de HTML4:

- El **a** y **area** ahora tienen un atributo **media** para la consistencia con el elemento **link**.
- El elemento **area**, por coherencia con el elemento **a** y los elementos **link**, ahora también tiene la **hreflang** y atributos **rel**.
- El elemento **base** puede tener ahora un atributo **target** y, para la coherencia con el elemento **a**. (Esto ya está ampliamente respaldado.) También, el atributo de **target** para el elemento **a** y los elementos **area** ya no es obsoleto, ya que es útil en aplicaciones Web, por ejemplo, en relación con el **iframe**.
- El atributo **value** para el elemento **li** ya no es obsoleto ya que no es de presentación. Lo mismo ocurre con el atributo **start** del elemento **ol**.
- El elemento **meta** tiene un atributo **charset** ahora como ésta era ampliamente apoyado y proporciona una buena manera de especificar la codificación de caracteres para el documento.
- Un nuevo atributo **autofocus** puede ser especificado (excepto cuando el atributo **type** está oculto) en elementos **select**, **textarea** y **button**
- Un nuevo atributo **placeholder** puede ser especificado en los elementos **input** y **textarea**. Representan una ayuda como guía en el ingreso de datos del usuario.

```
<input placeholder="a@b.com" type=email>
```

El nuevo atributo **form** para los elementos **input**, **output**, **select**, **textarea**, **button** y **fieldset** permiten asociar controles a un formulario. Por ejemplo, estos elementos pueden ahora ser posicionados en cualquier lugar de la página no solo como hijos de tag **form**. [14]

```
<label>Email:
<input type=email form=x name=email>
```

```
</label>  
<form id=x></form>
```

Todos estos cambios y nuevas especificaciones pueden ser leídos directamente desde el sitio Web de la W3C. Según informan en la página de la organización WHATWG, HTML 5 se prevé esté listo como especificación de implementación recomendada en el 2012. ¿Quiere esto decir que vamos a tener que esperar hasta 2012 para aprovechar las ventajas de HTML 5? realmente no es justamente así, puesto que algunos navegadores ya implementan muchas de las características del moderno lenguaje.

Resulta que HTML 5 está formado por muchos módulos distintos, cuyo grado de especificación está en niveles dispares. Por tanto, muchas de las características de HTML 5 están ya listas para ser implementadas, en un punto de desarrollo que se encuentra cercano al que finalmente será presentado. Otras muchas características están todavía simplemente en el tintero, a modo de ideas o borradores iniciales.

De hecho, las versiones más nuevas de casi todos los navegadores, incluido el polémico Internet Explorer 8, implementan algunas de las características de HTML 5. Claro que, para que una web se vea bien en todos los sistemas, hay que utilizar sólo aquellas partes que funcionan en todos los navegadores, por lo que a día de hoy, pocas son las utilidades realmente disponibles del lenguaje, si queremos hacer un sitio web compatible. No obstante, en el peor de los casos, podemos empezar a usar a nivel experimental estas características, aunque sólo sea para frotarnos las manos en espera de incorporarlas realmente en nuestras prácticas de desarrollo habituales.

CSS3. A medida que la popularidad de CSS aumenta, lo hace también el interés por agregarle nuevas definiciones a la especificación. En lugar de intentar meter decenas de cambios en una especificación monolítica, será mucho más fácil y eficiente para que pueda actualizar las piezas individuales de la especificación. Los módulos permiten CSS que se actualizará en forma más oportuna y precisa, permitiendo así una evolución más flexible y oportuna del pliego de condiciones en su conjunto.

Para dispositivos de recursos limitados, puede ser poco práctica para dar soporte a todos lo especificado en CSS. Por ejemplo, un navegador aural puede ser que se trate sólo con estilos de fonética, mientras que un navegador visual puede cuidar nada para los estilos de fonética. En tales casos, un agente de usuario puede implementar un subconjunto de CSS. Subconjuntos de CSS se limitan a la combinación de módulos CSS seleccionado, y una vez que un módulo ha sido elegido, de todas sus funciones debe ser apoyada. [\[19\]](#)

Safari, es un navegador Web de código cerrado desarrollado por Apple Inc. Está disponible para Mac OS X, iOS (el sistema usado por el iPhone, el iPod Touch y el iPad) y Microsoft Windows.

Incluye navegación por pestañas, corrector ortográfico, búsqueda progresiva, vista del historial en CoverFlow, administrador de descargas y un sistema de búsqueda integrado.

La primera versión beta de Safari fue presentada en la exposición Macworld el 7 de enero de 2003 y fue liberada en forma de beta pública. Su versión 1.0 se lanzó en junio de 2003. La versión 1.1 se publicó en octubre del mismo año y se convirtió en la primera versión de Safari en ser el navegador predeterminado para Mac OS X.

La versión 2.0 hizo su aparición el 29 de abril de 2005 formando parte de Mac OS X v10.4. El día 31 de octubre de 2005, en una actualización de Tiger, liberada la versión 2.0.2, convirtiendo a Safari en el primer navegador que pasaba el test Acid2.

El 9 de enero de 2007, Steve Jobs anunció que el smartphone de Apple Inc. (iPhone) usaría Safari para la exploración de sitios Web. Safari 3 fue anunciado el 11 de junio de 2007 y una beta pública fue puesta a disposición de los usuarios, siendo esta la primera versión en funcionar tanto en Mac OS X como en Microsoft Windows. Con el lanzamiento de Mac OS X v10.5 el 26 de octubre de 2007, fue incluida primera versión estable de Safari 3.

El 24 de febrero de 2009 se lanzó una beta de la versión 4 del navegador con una interfaz rediseñada, y utilizando su nuevo motor Javascript Nitro, que en pruebas sintéticas supera al rendimiento de Google Chrome y Firefox 3.5 beta. Finalmente, el 8 de junio del mismo año, y anunciado en el WWDC Keynote, se introdujo la versión final del Safari 4. En la versión para Windows eliminaron la pequeña interfaz Aqua del navegador que le daba un aspecto de usabilidad similar al Mac OS X quedando un aspecto más parecido al nativo de Windows, esto no le gustó mucho a los usuarios cambiando muchos de ellos a la competencia.

Actualmente (Enero de 2010) Microsoft negocia con Apple para que Bing sea el buscador por defecto en su navegador Safari, tanto en su versión para Mac OS X como para el iPhone y el iPod Touch.

Safari incluye navegación por pestañas, corrector ortográfico, marcadores, bloqueador de ventanas emergentes, atajos del teclado, soporte para motores de búsqueda, un gestor de descargas, CoverFlow para vista del historial y los Top Sites.

Proporciona un entorno para los desarrolladores Web, como el "User Agent" en el cual se selecciona el motor de renderizado para abrir las páginas Web. Inspector Web, Consola de errores además de herramientas para JavaScript.

Está escrito sobre el framework WebKit, que incluye a WebCore, el motor de renderizado, y JavaScriptCore, el intérprete de JavaScript. Por su parte, WebKit (el motor de renderizado del navegador) está basado en el motor KHTML, creado por el proyecto KDE para su navegador Konqueror. Como resultado de esto, el motor interno de Safari es software libre y es liberado bajo los términos de la licencia LGPL. Las mejoras al código de KHTML por parte de Apple son incorporadas al código de KDE rápidamente.

Safari es un gran explorador para dispositivos móviles y desktop. Ofrece soporte tanto para HTML5 como para CSS3 por lo que lo hace estar en los primeros navegadores del mercado.

Opera Mobile, puede formatear automáticamente cualquier página Web, ajustando el tamaño de las imágenes, texto, tablas y demás a la resolución del dispositivo móvil sobre el cual se esté utilizando, esta tecnología es llamada Small Screen Rendering de manera abreviada por sus nombre en inglés (SSR), que al español traduciría algo así como *Presentación sobre Pantalla Pequeña*. Dicha

tecnología facilita la navegación en dispositivos con limitaciones para mostrar pantallas de resoluciones mayores. La tecnología en si es propietaria pero el navegador es capaz de leer y traducir páginas SSR sin ningún problema. Además el navegador incluye las siguientes características:

- Icono de búsqueda en la barra de navegación
- Guardar imágenes
- Copiar y Pegar texto
- Enviar un link como correo electrónico y SMS
- Importar los favoritos de IE
- Soporte para Macromedia Flash Player 7 para Pocket PC
- Soporte para FlashLite 2.1 de Macromedia
- Múltiples ventanas
- Navegación por Tabs (sólo para Pocket PC)
- Zoom
- Descargar archivos
- Bookmarks
- Icono para identificar los sitios seguros
- Preferencias de usuario
- Menú contextual
- Diferentes modos de vista: Modo escritorio (PC), modo a una sola columna y ajustado

Ahora Opera ha anunciado la versión 9 de su Navegador para dispositivos tipo Pocket PC, con interesantes adiciones tales como Zoom inteligente AJAX Web 2.0 y la posibilidad de adicionar widgets como en el Opera para PC.

Opera hace realidad la navegación sobre dispositivos móviles y desde celulares. En la actualidad Opera abarca más de 50 millones de instalaciones sobre dispositivos móviles y se considera como el mejor navegador de su tipo.

Accesibilidad

La accesibilidad Web significa que personas con algún tipo de discapacidad van a poder hacer uso de la Web. En concreto, al hablar de accesibilidad Web se está haciendo referencia a un diseño Web que va a permitir que estas personas puedan percibir, entender, navegar e interactuar con la Web, aportando a su vez contenidos. La accesibilidad Web también beneficia a otras personas, incluyendo personas de edad avanzada que han visto mermadas sus habilidad a consecuencia de la edad.

La accesibilidad Web engloba muchos tipos de discapacidades, incluyendo problemas visuales, auditivos, físicos, cognitivos, neurológicos y del habla. El documento "Cómo utilizan la Web personas con discapacidad" muestra la forma en la que diferentes discapacidades pueden dificultar la utilización de la Web_e incluye algunos escenarios de personas con discapacidad utilizando la Web.

Es esencial que los diferentes componentes de desarrollo Web e interacción trabajen conjuntamente para que la Web sea accesible para aquellas personas con discapacidad. Estos componentes incluyen:

- **contenido** - información presente en una página Web o en una aplicación Web, incluyendo:
 - información como por ejemplo texto, imágenes y sonidos
 - código o etiquetado que define estructura, presentación, etc.
- **navegadores Web, reproductores multimedia** y otros "agentes de usuario"
- **tecnología asistida**, en algunos casos - lectores de pantalla, teclados alternativos, intercambiadores, software de escaneo, etc.
- Conocimiento de los **usuarios**, experiencias y, en ocasiones, estrategias de adaptación para la utilización de la Web
- **desarrolladores** - diseñadores, codificadores, autores, etc., incluyendo desarrolladores que tienen alguna discapacidad y usuarios que proporcionan contenido
- **herramientas de autor** - software para crea sitios Web
- **herramientas de evaluación** - herramientas para evaluar la accesibilidad Web, validadores de HTML, validadores de CSS, etc.

Entre los componentes existen interdependencias de gran importancia, lo que significa que los componentes deben funcionar de forma conjunta para que la Web sea accesible. Por ejemplo, el texto alternativo en las imágenes:

- Las **especificaciones técnicas** contemplan la utilización del texto alternativo (por ejemplo, HTML define el atributo de texto alternativo (alt) del elemento imagen (img))
- Las **pautas de WAI** - WCAG, ATAG y UAAG, definen cómo utilizar texto alternativo en relación a la accesibilidad en los diferentes componentes
- Los **desarrolladores** proporcionan palabras con un significado adecuado para el texto alternativo
- Las **herramientas de autor** permiten, facilitan y promueven la aportación de texto alternativo en una página Web
- Las **herramientas de evaluación** son utilizadas para ayudar a comprobar la existencia de texto alternativo
- Los **agentes de usuario** proporcionan interfaz humana y automática al texto alternativo
- Las **tecnologías asistidas** proporcionan interfaz humana al texto alternativo en varias modalidades
- Los **usuarios** saben cómo obtener el texto alternativo desde su agente de usuario y/o tecnología asistida en función de sus necesidades

Existen millones de personas con discapacidad que no pueden utilizar la Web. Actualmente, la mayoría de los sitios Web y los software Web presentan barreras de accesibilidad, lo que dificulta o imposibilita la utilización de la Web para muchas personas con discapacidad. Cuanto más software y sitios Web accesibles estén disponibles, más personas con discapacidad podrán utilizar la Web y contribuir de forma más eficiente.

Pero la accesibilidad Web beneficia también a organizaciones y a personas sin discapacidad. Por ejemplo, un principio básico de la accesibilidad Web es la flexibilidad con el objetivo de satisfacer diferentes necesidades, situaciones y preferencias. Esta flexibilidad va a beneficiar a todas aquellas personas que utilizan la Web, incluyendo personas que no tienen ninguna discapacidad pero que, debido a determinadas situaciones, tienen dificultades para acceder a la Web (por ejemplo, una

conexión lenta), también estaríamos hablando de aquellas personas que sufren una incapacidad transitoria (por ejemplo, un brazo roto), y de personas de edad avanzada. El documento "Desarrollo de un Proyecto de Accesibilidad Web para su organización" describe los diferentes beneficios de la accesibilidad Web, incluyendo los beneficios obtenidos por las empresas.

La Web es un recurso muy importante para diferentes aspectos de la vida: educación, empleo, gobierno, comercio, sanidad, entretenimiento y muchos otros. Es muy importante que la Web sea accesible para así proporcionar un acceso equitativo e igualdad de oportunidades a las personas con discapacidad. Una página Web accesible puede ayudar a personas con discapacidad a que participen más activamente en la sociedad.

Otra consideración importante para las empresas es que la accesibilidad Web es un requisito establecido en algunos casos por leyes y políticas. El documento "Recursos de Políticas de Accesibilidad Web de WAI" proporciona enlaces a recursos en relación a factores legales y políticos dentro de las empresas, incluyendo una lista con las leyes y políticas más importantes a nivel mundial.

La accesibilidad Web se ha entendido siempre como responsabilidad de los desarrolladores Web. Pero, el software Web tiene también un papel importante en la accesibilidad Web. Es importante que el software ayude a los desarrolladores a generar y evaluar sitios Web accesibles para que las personas con discapacidad puedan utilizarlos.

Una de las funciones de la Iniciativa de Accesibilidad Web ([WAI](#)) es desarrollar pautas y técnicas que proporcionen soluciones accesibles para el software Web y para los desarrolladores Web. Las pautas de WAI son consideradas como estándares internacionales de accesibilidad Web.

Hacer un sitio Web accesible puede ser algo sencillo o complejo, depende de muchos factores como por ejemplo, el tipo de contenido, el tamaño y la complejidad del sitio, así como de las herramientas de desarrollo y el entorno.

Muchas de las características accesibles de un sitio se implementan de forma sencilla si se planean desde el principio del desarrollo del sitio Web o al comienzo de su rediseño. La modificación de sitios Web inaccesible puede requerir un gran esfuerzo, sobre todo aquellos que no se "etiquetaron" correctamente con etiquetas estándares de XHTML, y sitios con cierto tipo de contenido, como multimedia.

Cuando se desarrolla o rediseña un sitio Web, la evaluación de la accesibilidad de forma temprana y a lo largo del desarrollo permite encontrar al principio problemas de accesibilidad, cuando es más fácil resolverlos. Técnicas sencillas, como es cambiar la configuración en un buscador, pueden determinar si una página Web cumple algunas de las pautas de accesibilidad. Una evaluación exhaustiva, para determinar el cumplimiento de las pautas, es mucho más compleja.

Hay herramientas de evaluación que ayudan a realizar evaluaciones de accesibilidad. No obstante, ninguna herramienta en sí misma puede determinar si un sitio cumple o no las pautas de accesibilidad. Para determinar si un sitio Web es accesible, es necesaria la evaluación humana. [\[13\]](#)

Iniciativa de Web móvil

Cuando se habla de Web móvil se está haciendo referencia a una Web en la que el usuario puede acceder a la información desde cualquier lugar, independientemente del tipo de dispositivo que utilice para ello.

Actualmente, existe una demanda cada vez mayor por parte de los usuarios en lo referente a una disponibilidad incondicional de la Web, pero la realidad en el mercado es otra ya que aunque la oferta de dispositivos móviles está creciendo de forma asombrosa en los últimos años, ofreciéndonos infinidad de dispositivos desde los que llevar a cabo operaciones que normalmente realizábamos desde el equipo de sobremesa, existen limitaciones a la hora de acceder a los servicios desde esos dispositivos móviles. En la mayoría de las ocasiones, el resultado es una experiencia de usuario poco satisfactoria al encontrarnos con numerosos problemas para acceder a la Web desde los dispositivos móviles. [29]

El W3C, con el objetivo de convertir el acceso a la Web desde un dispositivo móvil en algo tan sencillo y cómodo como lo es desde los equipos de sobremesa, ha puesto en marcha la Iniciativa de Web Móvil que busca resolver los problemas de interoperabilidad y usabilidad que actualmente dificultan el acceso a la Web desde dispositivos móviles y hacer posible uno de los objetivos principales del W3C que consiste en alcanzar una Web única. Se trataría de transformar la Web en una plataforma omnipresente de servicios completamente transparentes para el usuario. [29]

¿Para qué sirve?

Con la aparición de un número cada vez mayor de dispositivos diferentes tanto en tamaños como formas, y con características muy diversas tales como tecnologías de localización, cámaras, reconocimiento de voz, pantallas táctiles, etc., la Web puede alcanzar a un mayor número de usuarios en todo momento y en cualquier situación. La Web móvil llega a lugares donde el cable no puede llegar, lugares que anteriormente eran impensables, como está ocurriendo en los países en vías de desarrollo. Esto trae consigo nuevas oportunidades para realizar negocios, para trabajar, para gestionar nuestro tiempo de ocio, y para muchas otras cosas, lo que nos permite incrementar nuestra capacidad de movimiento.

Pero para poder hacer uso de esa Web móvil sin encontrarnos con problemas es necesaria una total adecuación de los contenidos a los dispositivos utilizados independientemente del lugar en el que estemos y del dispositivo que utilicemos. El principal objetivo de las iniciativas puestas en marcha en torno a la Web móvil es la búsqueda de una Web no fragmentada como consecuencia del surgimiento de una multitud de nuevos dispositivos móviles, navegadores, operadores, proveedores de contenido, etc. Una Web donde sea posible obtener el contenido correcto, en el momento oportuno y en el lugar adecuado. [29]

¿Cómo funciona?

La Web móvil se presenta como un auténtico reto tanto para usuarios como para desarrolladores ya que, por un lado, el usuario encuentra problemas al intentar

acceder a los sitios Web desde los dispositivos móviles, y por otro, los proveedores de contenido encuentran dificultades para crear sitios Web que funcionen adecuadamente en todos los tipos de dispositivos y configuraciones.

Es importante tener en cuenta que existen grandes diferencias entre usuarios móviles y usuarios fijos, como son los diferentes tipos de contenido que manejan, las capacidades de los dispositivos que utilizan (pantallas pequeñas) y el contexto en el cual el usuario recibe el contenido (por ejemplo, en el autobús). El acceso a la información desde cualquier lugar, en cualquier momento e independientemente del dispositivo utilizado puede alcanzarse a través de aplicaciones que se adapten dinámicamente a las necesidades del usuario, a las capacidades del dispositivo y a las condiciones del entorno. Dotar a las aplicaciones de **movilidad** permitirá a los usuarios utilizar diferentes dispositivos para acceder a la misma información. Como usuarios, podremos elegir la forma de interactuar con estas aplicaciones en función de nuestras necesidades y de las características del dispositivo utilizado. Para ello es necesaria una infraestructura global basada en estándares que permita la interoperabilidad.

En la actualidad, el trabajo del W3C en temas de Web móvil se centra principalmente en dos áreas: *generación de buenas prácticas* y *descripción de dispositivos móviles*. El “Grupo de Trabajo de Buenas Prácticas en Web Móvil” ha sido creado para desarrollar pautas, puntos de verificación y buenas prácticas con el objetivo de ayudar a los proveedores de contenido a desarrollar contenido Web que funcione correctamente en dispositivos móviles. Por otro lado, el “Grupo de Trabajo de Descripción de Dispositivo” se ha creado para guiar el desarrollo de mecanismos de descripción de dispositivos, es decir, bases de datos de descripciones que los desarrolladores de contenido podrán utilizar para adaptar los contenidos a los diferentes dispositivos.

Algunos de los estándares del W3C utilizados en la consecución de la Web móvil son las tecnologías multimedia y de gráficos como SMIL para multimedia, y SVG para la representación de gráficos vectoriales y animaciones en dispositivos de pantalla pequeña, que cuenta con dos perfiles, SVG Tiny para la representación de gráficos vectoriales en teléfonos móviles y SVG Basic para PDAs. La adaptación del contenido a dispositivos concretos y a las preferencias de los usuarios dispone de algunas tecnologías estándares como CC/PP y DPF que permiten obtener datos sobre las características de los dispositivos, las preferencias del usuario y las condiciones del entorno, o DSelect para la selección y filtrado de contenido. A través de este mecanismo se seleccionan ciertas partes de la información, mientras que otras son suprimidas.

En el ámbito de la interacción con dispositivos móviles, el W3C proporciona tecnologías que facilitan la interacción a través del oído, la vista y el tacto. Entre algunas de estas tecnologías encontramos la “Infraestructura de Interfaz del Habla” donde el W3C trabaja en lenguajes de etiquetado para hacer posible la utilización de aplicaciones de voz en diferentes plataformas de software y hardware. En concreto se trabaja diseñando lenguajes de etiquetado para la gramática de reconocimiento de voz, síntesis del habla, semántica de lenguaje natural y otros componentes de diálogo reutilizables. Algunos de los lenguajes de etiquetado son la “Especificación de Gramática de Reconocimiento del Habla” y el “Lenguaje de Etiquetado de Síntesis del Habla”. También hay que destacar EMMA que actúa como mecanismo de comunicación dentro de un sistema multimodal, VoiceXML 2.0, como lenguaje para crear diálogos entre el usuario y la aplicación e InkML para el intercambio de escritura electrónica entre distintas aplicaciones. [29]

Las siguientes imágenes muestran la página Web de la Oficina Española del W3C en diferentes dispositivos, a través de los cuales es posible ver de forma correcta el contenido.



Funcionamiento correcto en distintos dispositivos. [29]

Capítulo 6. Conclusiones

Antes de evaluar las conclusiones recordemos los objetivos de esta tesina de grado:

- En primer lugar, el análisis de aplicabilidad de una plataforma de hipermedia física en un ámbito académico y en otros contextos.
- En segundo lugar, el análisis de los códigos QR como metodología de sensado en una plataforma de este tipo.
- Por último, se buscó desarrollar una plataforma de hipermedia física como caso de prueba, analizando las tecnologías actuales para desarrollar este tipo de sistemas.

6.1 Aplicabilidad de la plataforma de hipermedia física

Luego de analizar los diferentes contextos de aplicabilidad de una plataforma de hipermedia física, determinamos que existen muchos usos que se le pueden dar a un sistema de este tipo. Ya que el objetivo es brindarles a las personas información de donde se encuentran parados. Muchas organizaciones pueden sacar provecho para promocionar lugares con fines de lucro.

La implementación en un ámbito académico es una situación muy particular. Los jóvenes actuales tienen facilidad para las nuevas tecnologías lo cual reduce los riesgos de fracaso en la aplicabilidad. Pero la forma de comunicar y llegar a ellos con los contenidos debe ser similar a la que utilizan las redes sociales actuales. De este modo, los adolescentes tendrán una mayor predisposición para hacer uso de una tecnología de este tipo.

Luego de analizar los beneficios que puede ofrecer una plataforma de este tipo, determinamos que algunos de ellos son:

- Beneficios importantes para mejorar la comunicación entre todos los individuos de una institución.
- Mejorar la ubicación de las personas nuevas en el lugar.
- Contenidos más fácil de manejar y acceder.
- Es una plataforma económica de implementar. Solo se necesita un servidor http mediano.
- Noticias en tiempo real sobre acontecimientos específicos. Si un profesor suspende una clase, o si va a llegar más tarde, o bien un cambio de aula repentino para una cursada.
- Mejorar la calidad de la institución mediante estadísticas obtenidas de las opiniones de los alumnos respondiendo encuestas.

Estos y algunos más son los beneficios que se pueden obtener mediante un sistema de hipermedia física que integre comunicación tipo red social y navegación física de elementos.

Como se mencionó anteriormente, muchas organizaciones pueden sacar provecho de aplicaciones de este tipo. Se puede utilizar como mecanismo para el sensado de información, para captar una opinión general de los clientes, realizar promociones temporales online a cambio de responder alguna encuesta, o la simple distribución de información por medio de mensajes o eventos.

6.2 Códigos QR como metodología de sensado

Analizando los códigos QR se pudo determinar que son una buena elección a la hora de abaratar costos de implementación en un sistema. Pero esta no es el único beneficio. Se pudo observar los pro y contra que estos presentan:

Beneficios

- Fácil implementación, mediante un soporte impreso.
- Licencia sin ejercicio de derechos. Libre y gratuito.
- Soporte en la mayoría de los lenguajes de programación actuales.
- No se necesita impresora de alta calidad para su emisión.
- Soportan un desgaste importante. Puede ser leído desde cualquier posición e incluso si está dañado o faltan partes.
- Cantidad de aplicaciones móviles para tomar fotografía y decodificarlos

Contras

- No todos los dispositivos móviles tienen cámara digital (Ej. iPods)
- Los códigos disponibles en las ubicaciones suelen ser quitados, perdidos y desgastados con el paso del tiempo.

De este modo los códigos de barra quick response ofrecen una buena alternativa cuando se desea implementar una metodología de identificación automática robusta, que sea fácil de implementar, económica y eficiente.

6.3 Desarrollo del caso de prueba Croover

Luego de realizar una investigación sobre las tecnologías disponibles para la implementación de una plataforma de este tipo, se determino que puede desarrollarse en su totalidad con herramientas de software libre (opensource). Por lo tanto, realizar una plataforma de hipermedia física no insume costos significativos, solo se necesita la mano de obra para poner en funcionamiento cada tecnología.

Se desarrolló una plataforma de hipermedia física sobre Web que permite la

gestión de ubicaciones del mundo real. Se determinó que el uso de PHP como lenguaje de programación central ofrece un desempeño eficiente para el funcionamiento estable de la plataforma.

Se desarrolló de forma conjunta una aplicación Web para agentes móviles testeada en el explorador móvil Safari, utilizado en productos Apple como el iPhone y el iPod. Se intentó realizar una interfaz grafica amigable y sencilla ya que este tipo de dispositivos, como muchos otros, tienen pantalla táctil, lo cual necesita que los contenidos sean bien distribuidos en los 300px que ofrece la pantalla.

Se analizó el uso de Google Maps para posicionamiento y ubicación de los elementos del mundo real. Determinando que el uso en dispositivos móviles tiene un desempeño menor que el presentado en la aplicación de escritorio. El retardo producido por la descarga de la API, es de aproximadamente cinco veces mayor. En un dispositivo móvil, dependiendo de la conexión, suele tardar entre cinco y siete segundos. De todos modos, las conexiones cada vez son más veloces y Google promete solucionar problemas de esta índole y muchos otros más.

Se integró a la plataforma Croover, una comunicación del tipo utilizado por redes sociales. Como así también, la interconexión entre redes sociales como Facebook y Twitter para producir publicaciones simultaneas. Hoy en día muchas organizaciones tienen cuenta en estas redes sociales, de modo que resulta indispensable tener este tipo de enlace. Las API de las dos redes sociales, es muy sencilla de entender, tanto en PHP que fue el lenguaje elegido como en otros lenguajes Web.

Capítulo 7. Bibliografía

- [1] Apache Documentation and reference - www.apache.org - Copyright 2009 The Apache Software Foundation.
- [2] R.T. Azuma, Y. Baillet, R. Behringer, St. Feiner, S. Julier, B. MacIntyre: "Recent Advances in Augmented Reality". IEEE, Nov. 2001, pp 255-385.
- [3] J. C. Almenara: "Navegando, construyendo: la utilización de los Hipertextos en la enseñanza". Medios de comunicación, recursos y materiales para la mejora educativa II, Sevilla, CMIDE del Excmo. Ayuntamiento de Sevilla y SAV de la Universidad de Sevilla, 227-243.
<http://tecnologiaedu.us.es/revistaslibros/hiper.html>.
- [4] C. Challiol, De Cófolo, 2006: "Extensión del MVC para resolver problemas de Hipermedia Física", Tesina de grado, Facultad de Informática – UNLP
- [5] C. Challiol, A. Fortier, G. Rossi, S. Gordillo: "Architectural and implementation issues for a context-aware hypermedia platform" - Journal of Mobile Multimedia, Vol. 1, No.1 (2005) 000-000
- [6] C. Challiol, A. Fortier, G. Rossi, S. Gordillo: Physical Hypermedia: a Context-Aware approach
- [7] Denso Wave Corp. – Documentation of the QRcodes
<http://www.denso-wave.com/qrcode/index-e.html>
- [8] Doctrine - Documentation and reference
<http://www.doctrine-project.org/projects/orm/2.0/docs/en>
- [9] D. Estrin, D. Culler, K. Pister, G. Sukhatme: "Connecting the Physical World with Pervasive Networks". IEEE Pervasive Computing, vol. 01, no. 1, pp. 59-69, January-March 2002.
- [10] Google Maps Javascript API and reference - junio de 2010
<http://code.google.com/intl/es-ES/apis/maps/documentation/javascript/examples/index.html>
- [11] E. Hammer-Lahav - The OAuth 1.0 Protocol, 2010 for the Internet Engineering Task Force (IETF).
- [12] F. A. Hansen, Niels Olof Bouvin January 2009: Mobile Learning in Context – Context-aware Hypermedia in the Wild
- [13] S. L. Henry - W3C – Accesibilidad Web
<http://www.w3.org/WAI/intro/accessibility>
- [14] Hickson, Google, Inc.W3C – HTML5 Draft Specification
<http://www.w3.org/TR/html5/>
- [15] R.E. Horn: "Mapping Hypertext". Lexington Institute.USA - Paperback- February 1990

- [16] D. Klamm - How Universities Can Win Big with Location-Based Apps
<http://mashable.com/2010/09/22/universities-geo-location/>
- [17] K. J. Kortbek, F. A. Hansen, , and K. Grønbæk: Mobile Urban Drama – Setting the Stage with Location Based Technologies - Center for Interactive Spaces, Department of Computer Science, University of Aarhus, Aabogade 34, DK-8200, Aarhus N, Denmark
- [18] W.E. Mackay: “Augmented Reality: Linking real and virtual worlds. A new paradigm for interacting with computers”. Proceedings of the ACM Conference on Advanced Visual Interfaces (AVI 98), ACM Press, pp13-21.
- [19] E. A. Meyer, B. Bos - W3C – CSS3 Specification
<http://www.w3.org/TR/css3-roadmap/>
- [20] P. Milgram, F. Kishino: “A Taxonomy of Mixed Reality Visual Displays”. IEICE Trans. Information Systems, vol. E77-D, no. 12, 1994, pp. 1321-1329.
- [21] Y. H. Montero, 2002: “Diseño Hipermedia centrado en el usuario”
- [22] MySQL Documentation and Reference
www.mysql.com
- [23] T. H. Nelson: “Literary Machines”. Own publisher. 1990.
- [24] C. O. Nueda: "Educación y futuro. Textos para una encrucijada". Colección Documentos de la Red. Consejería de Educación de la Comunidad de Madrid y Entimema. Madrid, 2001. ISBN 84-8198--367-5.
- [25] PHP Documentation and API reference
www.php.net
- [26] L. Romero, N. Correia: “HyperReal: A Hypermedia model for Mixed Reality”. Proceedings of the 14th ACM International Conference of Hypertext and Hypermedia (Hypertext 2003), pp 2-9, ACM Press.
- [27] Symfony Project
www.symfony-project.org
- [28] Symfony Project Plugins
<http://www.symfony-project.org/plugins/>
- [29] W3C Web móvil – Iniciativa web móvil – 2005
<http://www.w3c.es/Divulgacion/GuiasBreves/webmovil>