



Universidad Nacional de La Plata  
Licenciatura en Informática



BIBLIOTECA  
FAC. DE INFORMÁTICA  
U.N.L.P.

## Trabajo de Grado

# “Hipermmedia Física aplicada en el contexto de un Centro Comercial”

Autores:

A.C. Federico E. Carpi  
Gabriel García Peña

Director de Tesis

Dra. Silvia Gordillo

Codirector de Tesis

Dr. Gustavo Rossi

<b>TES</b> <b>05/25</b> <b>DIF-03108</b> <b>SALA</b>	 <b>UNIVERSIDAD NACIONAL DE LA PLATA</b> <b>FACULTAD DE INFORMÁTICA</b> Biblioteca 50 y 120 La Plata catalogo.info.unlp.edu.ar biblioteca@info.unlp.edu.ar
	 DIF-03108



BIBLIOTECA  
FAC. DE INFORMÁTICA  
U.N.L.P.

IDONACION? FACULTAD..... TES  
\$..... 05/25  
Fecha..... 12-3-08  
Inv. E..... Inv. B..... 003108

## Agradecimientos:

*Queremos agradecer al LIFIA, laboratorio en el que desde hace varios años estamos participando en grupos de investigación y desarrollo, el cuál nos ha brindado la posibilidad de aprender conceptos fundamentales para nuestro desarrollo personal y profesional.*

*En particular, a nuestros directores de Trabajo de Grado, Silvia Gordillo y Gustavo Rossi, por orientarnos en el desarrollo del mismo.*

*También queremos realizar una mención especial a la Facultad de Informática, la cuál nos ha entregado sus enseñanzas e instalaciones para poder lograr nuestro objetivo.*

### *Federico Carpi:*

*Personalmente quiero agradecer en primer lugar a mi familia: a mis padres, Ana María y Eduardo, y a mis hermanas Vero y Mili, por su apoyo constante en todas las situaciones, buenas y malas. Seguramente sin ellos no hubiera llegado a donde estoy hoy. A Mao, por las largas charlas que solíamos tener y que añoro enormemente, espero que en algún momento se vuelvan a repetir. A mis sobrinos Julián y Alejo (y al próximo que esta por venir) por todo su amor.*

*No puedo olvidarme de agradecer también a mis amigos cosechados en la Facultad, por todas sus opiniones y sugerencias para poder realizar este trabajo y por aguantar mis eternas charlas técnicas.*

*Por último quiero agradecer a la persona que está a mi lado desde hace 6 años, por todo el apoyo y amor que me brindó siempre y sobre todo por el aguante que siempre supo tener. Para vos Yami, todo mi amor.*

### *Gabriel García Peña:*

*Quiero agradecer a mi papá Omar, por haberme brindado la posibilidad de elegir y de estudiar esta carrera. A mi hermana Cecilia por haberme mostrado los caminos del estudio universitario y haberme ayudado en muchos momentos. A mis abuelas, Julia y Generosa que iniciaron mi educación y a mi mamá Estér que me ayuda todos los días acompañándome desde el cielo.*

*En especial, para vos Vale, no encuentro las palabras que expresen mi agradecimiento, fuiste mi sostén en momentos difíciles, me acompañas siempre y me enseñas a ver de otra manera las cosas, sin vos hubiese sido imposible llegar a este momento. Gracias.*

*Por último, agradezco a todos mis amigos que me ayudaron y me alentaron a lo largo de toda mi carrera.*

# Índice

<b>Índice</b> .....	<b>3</b>
<b>Resumen</b> .....	<b>6</b>
<b>Capítulo 1: Introducción</b> .....	<b>7</b>
1.1 Motivación .....	7
1.2 Desarrollos propuestos .....	7
1.3 Resultados esperados.....	7
<b>Capítulo 2: Sistemas móviles y sensibles al contexto</b> .....	<b>9</b>
2.1 Introducción .....	9
2.2 Computación móvil (Mobile computing).....	10
2.2.1 Tecnología WAP .....	11
2.3 Computación Ubicua (Ubiquitous computing) .....	11
2.4 Conceptos de Contexto y sensibilidad al contexto.....	12
2.4.1 Contexto .....	12
2.4.2 Sensibilidad al contexto (Context awareness).....	12
2.5 Servicio basados en la ubicación (Location-based services, LBS) .....	13
2.5.1 Tecnologías .....	13
2.5.2 Servicios.....	13
<b>Capítulo 3: E-Commerce y M-Commerce</b> .....	<b>15</b>
3.1 Comercio electrónico .....	15
3.1.1 Desarrollo histórico .....	15
3.1.2 Claves del éxito del comercio electrónico.....	16
3.1.3 Problemas asociados al comercio electrónico .....	16
3.1.4 Productos adecuados para la venta electrónica .....	17
3.1.5 Aceptación del comercio electrónico .....	17
3.2 Comercio móvil.....	18
3.2.1 Tendencias del comercio móvil.....	18
3.2.2 Estado actual del comercio móvil .....	19
<b>Capítulo 4: Hipermedia e Hipermedia Física</b> .....	<b>20</b>
4.1 Hipermedia.....	20
4.1.1 Introducción .....	20
4.1.2 Características principales de un Sistema Hipertexto .....	21
4.1.3 Modelos de hipertexto .....	22
4.2 Hipermedia Física .....	23
4.2.1 Introducción .....	23
4.2.2 Características de Hipermedia Física .....	24
4.2.3 Criterios de diseño y desarrollo de un Sistema Físico Hipermedial .....	26
4.2.4 Ejemplos de escenarios de aplicación de Hipermedia Física .....	26
a- Museo .....	26
b- Sistema de orientación para no videntes.....	27
c- Centro Comercial .....	27
4.2.5 Conclusiones de Hipermedia Física .....	28

<b>Capítulo 5: Descripción del sistema basado en Hipermedia Física</b> .....	<b>30</b>
5.1 Introducción .....	30
5.2 Descripción y características generales del Sistema .....	30
5.3 Especificación funcional del Sistema.....	31
5.3.1 Servidor .....	33
5.3.2 Consola de Administración del Servidor .....	34
5.3.4 Aplicación Cliente (Usuario móviles).....	35
5.4 Alternativas de conectividad .....	38
5.4.1 Conectividad cliente/servidor vía Bluetooth .....	38
5.4.2 Conectividad cliente/servidor vía GPRS.....	40
5.4.3 Conectividad Cliente/Locales .....	41
<b>Capítulo 6: Diseño</b> .....	<b>43</b>
6.1 Introducción .....	43
6.2 Descripción del modelo.....	43
6.2.1 Descripción de los módulos del servidor .....	43
6.2.2 Descripción de los modelos de Usuario Móviles y Dispositivos.....	45
6.2.3 Descripción del modelo de compras .....	46
6.2.4 Descripción del modelo de locales y productos .....	47
6.2.5 Diseño de Hipermedia Física .....	48
6.2.5.1 Modelo de Nodo Físico .....	48
6.2.5.2 Modelo de Link Físico .....	49
6.2.5.3 Modelo de Link Hipermedial .....	49
6.2.6 Relaciones entre el modelo de Hipermedia Física y el modelo del dominio .....	49
<b>Capítulo 7: Implementación</b> .....	<b>50</b>
7.1 Introducción .....	50
7.2 Objetivo y alcance del prototipo .....	50
7.2.1 Funcionalidad provista en el servidor .....	50
7.2.2 Funcionalidad provista en el cliente.....	51
7.3 Implementación del servidor.....	52
7.3.1 Herramientas utilizadas para realizar la implementación .....	52
7.3.2 Implementación de la Consola del Servidor .....	55
7.3.3 Implementación del algoritmo de búsquedas entre locales.....	60
7.3.3.1 Algoritmos de búsquedas sobre grafos.....	60
7.3.3.2 Haciendo búsquedas sobre el prototipo.....	62
7.3.4 Usuario del sistema .....	64
7.3.5 Locales del Centro Comercial .....	65
7.3.6 Persistencia de los datos.....	66
7.3.6.1 Diseño e implementación del módulo de persistencia .....	67
7.4 Implementación del Cliente .....	70
7.4.1 Herramientas utilizadas para realizar la implementación .....	70
7.4.2 Implementación de la aplicación cliente .....	70
7.4.2.1 Funcionalidades de la aplicación cliente.....	71
7.4.3 Simulación del movimiento del usuario.....	75
7.4.4 Visualización de productos en venta.....	77
<b>Capítulo 8: Conclusiones y Trabajo Futuro</b> .....	<b>79</b>
8.1 Conclusiones .....	79

8.2 Trabajo Futuro.....	80
<b>Apéndice A: Map Obejct-Java .....</b>	<b>81</b>
A.1 Objetos de visualización.....	81
A.2 Objetos de contenido .....	85
A.3 Objetos de acceso a datos.....	87
<b>Apéndice B: Bluetooth .....</b>	<b>89</b>
B.1 ¿Qué es bluetooth?.....	89
B.2 Características.....	89
B.3 Bluetooth versus infrarojo .....	90
B.4 Aplicaciones con Java .....	90
<b>Apéndice C: Java 2 Micro Edition (J2ME).....</b>	<b>92</b>
C.1 ¿Qué es la plataforma J2ME?.....	92
C.2 Configuración .....	93
C.3 Perfiles .....	94
C.4 MIDlets.....	94
<b>Apéndice D: General Packet Radio Service (GPRS) .....</b>	<b>96</b>
C.1 Introducción.....	96
C.2 Protocolo.....	97
<b>Referencias .....</b>	<b>98</b>

## Resumen

El objetivo de esta Tesis será implementar un entorno en el que se ejecutara un sistema de información cuyas principales características son la adaptación de la información en función de la "posición geográfica" en la que se encuentra el usuario (Sistema móvil) y al entorno en el que se encuentra (Sistema sensible al contexto) permitiendo la conexión física y/o lógica entre distintos puntos de interés del dominio y brindando relaciones entre los mismos (Physical Hypermedia) [1]. Bajo este sistema, el entorno se convierte en un "espacio interactivo" en el que el usuario por medio de dispositivos móviles (celular, Table PC, PDA, etc.) puede obtener información sobre los elementos que le rodean en cada momento. Dicha información se puede clasificar en:

- Características físicas sobre como se puede acceder o llegar a los distintos nodos físicos.
- Características informativas en las distintas relaciones existentes entre los mismos (información del objeto, links multimedia en la web, por ejemplo, en una exposición de cuadros, seguir un hilo temático de las obras expuestas y obtener información sobre otros cuadros que traten sobre el mismo tema que el que estamos observando en la exposición).

El sistema estará basado en una estructura cliente/servidor donde el servidor se encarga de transmitir la información a los clientes en base a los parámetros que recibe de éstos. Cada uno de los clientes enviará al servidor información sobre las características de su dispositivo móvil, su posición y el conjunto de consultas que realice sobre la información. Luego recibirá del servidor la información requerida representada con el formato apropiado.

El diseño y la implementación de la arquitectura se basará en el patrón MVC (Model View Controller) [3], siendo el modelo la aplicación del dominio sobre la que se trabajará, el controlador será toda la lógica relacionada con el manejo de los peticiones de los clientes y el envío de información a los mismos, y las vistas se adaptarán a los distintos dispositivos móviles existentes (celular, Table PC, PDA, etc.)

El dominio del sistema se centrara en un Centro Comercial, con sus respectivos clientes, proveedores, locales, etc. Parte de la funcionalidad que proveerá se basara en:

- Cuando un usuario visite el Shopping podrá indicarle al sistema cuales son sus áreas de interés dentro de un conjunto de áreas predefinidas, por ejemplo: "Deportes", "Comidas", "Salud", etc.
- A medida que el cliente camine por el Shopping recibirá información desde el servidor en función de su posición y las áreas de interés que haya preseleccionado. En otras palabras cuando el cliente se encuentre en la cercanía de un local que esta incluido en alguna de sus áreas de interés, el local enviará su ID al dispositivo móvil. La aplicación cliente recibirá dicho ID y realizará un requerimiento al servidor peticionando información sobre el local. El servidor enviará información referente a la mercadería del mismo (ofertas, exclusividades, promociones, servicios) junto con información adicional como: otros locales relacionados que vendan artículos similares en el Centro Comercial, otras direcciones de locales que administre la misma firma, como comunicarnos con la firma o donde podemos hacer reclamos si no estamos conformes con la atención, etc.

# Capítulo 1

## Introducción

### 1.1 Motivación

En los últimos años ha habido una gran evolución en los sistemas de cómputos, uno de estos avances es que los componentes electrónicos son cada vez más pequeños, esto permite que muchos dispositivos que antes no poseían unidades de procesamiento hoy en día si la tengan (Pervasive Computing) [4], en particular, los dispositivos móviles: celular, Table PC, PDA, etc. poseen una gran capacidad de procesamiento (que muchas veces no es aprovechada en su totalidad).

Por otro lado en un pasado cercano la Hipermedia [5] nos permitió analizar distintos dominios de información en un entorno digital. En la actualidad ha surgido un nuevo concepto que es el de Hipermedia Física, el cual aparte de integrar las ideas de Hipermedia, agrega el manejo de información física (distancia a un objeto, etc.) en un sistema de información.

Por tales motivos hemos decidido realizar un trabajo que integre o relacione el uso de dispositivos móviles junto con los conceptos de Hipermedia Física, y sumando a esto tomamos con motivación extra la investigación, el desarrollo y la integración de nuestra aplicación con el concepto de m-commerce [6].

### 1.2 Desarrollos propuestos

A continuación, luego de presentar en el resumen los objetivos de este trabajo, junto con su motivación, presentamos los desarrollos propuestos como así también las diferentes áreas de estudio que deberemos abarcar.

- Estudiar el modelo de datos a utilizar tanto para la información multimedia como para la geográfica.
- Estudiar los parámetros a almacenar para cada uno de los usuarios y dispositivos del sistema de manera que se pueda realizar una adaptación dinámica del sistema a sus características.
- Diseño e implementación del servidor de información.
- Diseño e implementación de los clientes para los dispositivos móviles.
- Analizar las diferentes tecnologías para implementar las funciones necesarias para la transmisión inalámbrica de información entre los locales y los dispositivos móviles y entre estos con el servidor.
- Estudiar los diferentes mecanismos para hacer transparente el proceso de instalación y ejecución de la aplicación cliente en los dispositivos móviles.

### 1.3 Resultados esperados

- Contribuir al desarrollo del Concepto de Hipermedia Física, aplicando este concepto a un sistema real de información e investigando, sumando y relacionando este concepto, con el desarrollo de aplicaciones m-commerce [6] en un contexto auténtico.

- Resolver las dificultades que presenta la implementación y desarrollo de un sistema de m-commerce [6]. Inconvenientes que se relacionan con las características del ambiente inalámbrico, limitaciones en los equipos móviles y seguridad en las transacciones.

# Capítulo 2

## Sistemas Móviles y Sensibles al Contexto

### 2.1 Introducción

Un sistema de computación móvil puede definirse como una serie de artefactos y equipos portátiles que hacen uso de la computación para lograr su funcionamiento, por ejemplo: las computadoras portátiles, los teléfonos celulares, los cuadernos de notas computarizados, las calculadoras de bolsillo, etc.

Actualmente no caben dudas que la computación móvil e inalámbrica está jugando un papel destacado en la industria de las comunicaciones. En los últimos años los dispositivos móviles han sido dotados de capacidades de procesamiento que años atrás hubiesen sido impensadas. Esta es una de las razones para que exista una gran diversidad de aplicaciones. Hay varios campos en los que no solo es de utilidad, sino de competencia esencial para las empresas. Aquí algunos ejemplos:

- **Manejo de pacientes:**  
La computación móvil permite al médico o institución mantener contacto con un paciente cuyo estado requiere continua vigilancia. Y esto no se limita al envío o recepción de mensajes (hablados o escritos), sino que incluye también el monitoreo constante de signos vitales críticos que pueden anticipar una emergencia.
- **Ventas directas:**  
Posiblemente ésta sea la aplicación más evidente: poder consultar inventarios, precios y realizar pedidos en forma inmediata resulta de particular interés para cualquier empresa que se dedique a la comercialización de productos.
- **Servicio a clientes:**  
La asesoría, servicio técnico y consultoría es un área en donde la computación móvil es vital. La consulta a bancos de información, bases de datos inteligentes, acopio de información actualizada y consulta de especialistas, es sólo una pequeña muestra de todo lo que puede impactar esta tecnología, sin mencionar la posibilidad de contacto permanente con el cliente.
- **Personal móvil en oficinas:**  
No es extraño encontrar a personal que, pese a encontrarse siempre en el mismo edificio, se mudan de lugar con frecuencia para, por ejemplo, dar soporte técnico al personal o revisar proyectos. La computación móvil no sólo les permite ser localizados con facilidad, sino que también le auxilia en la consulta de datos que por lo regular estarían en su oficina.
- **Profesionales viajeros:**  
Estos profesionales son usuarios potenciales que pueden aprovechar esta herramienta: contadores con los registros de una empresa bajo el brazo, gerentes regionales que integran metas empresariales, dirigentes corporativos que requieren información actualizada y, en fin, una gran cantidad de personas que necesitan viajar para realizar su trabajo.

- **Manejo de sucursales:**

En un mundo en el que las empresas han dejado de tener una sola oficina para expandirse, las sucursales aparecen casi sin desearlo, con una gran cantidad de datos que deben consolidarse.

- **Grupos de trabajo:**

La globalización y expansión de empresas hace que sea cada vez más común atacar proyectos con el personal adecuado, el cual no siempre trabaja bajo un mismo techo y, en ocasiones, ni siquiera en la misma ciudad o país.

Tal como hemos presentado es enorme el número de áreas en las que puede ser aplicada la computación móvil. Este trabajo abarcará el área de “Ventas Directas”, entre otros servicios complementarios (ofertas, pedidos de información, etc.)

## 2.2 Computación móvil (Mobile computing)

El concepto de Mobile Computing es de suma importancia para este trabajo, es por esto que incluimos algunas definiciones que pueden esclarecer su significado.

Según Simerman [14], el término mobile computing o computación móvil se utiliza para describir el uso de dispositivos de cómputo, los cuales usualmente interactúan con un sistema de información central, a pesar de que el lugar de trabajo no es fijo. La tecnología de mobile computing permite al usuario: crear, acceder, procesar, almacenar y comunicar información sin restringirse a una única ubicación. La computación móvil se puede realizar gracias a la combinación de: hardware, sistemas y aplicaciones de software; y alguna forma de medio de comunicación. En los últimos tiempos han surgido soluciones móviles poderosas, esto se debe a causa de la disponibilidad de: dispositivos de cómputos extremadamente pequeños y poderosos, la existencia de software especializado y las crecientes mejoras que están teniendo las telecomunicaciones.

Por otro lado, según Sandeep [15], Mobile computing es un término usado para describir tecnologías que permiten a las personas acceder a servicios en una red, desde cualquier lugar, en cualquier momento, y en cualquier ubicación. Los términos “Ubiquitous computing” y “Nomadic computing” son sinónimos de mobile computing.

El acceso a la información vía dispositivos móviles es restringido por el bajo ancho de banda disponible, un mantenimiento pobre de las conexiones, escasa seguridad, y problemas de direccionamiento. Como contraparte con los sistemas “cableados”, el diseño de software para los dispositivos móviles debe considerar las limitaciones de los recursos, la capacidad de las baterías y el tamaño de la pantalla del dispositivo. Consecuentemente, nuevo hardware y técnicas de software deben ser desarrollados. Por ejemplo, las aplicaciones para dispositivos móviles necesitan ser altamente optimizadas en cuanto a su tamaño, debido a las limitaciones de memoria de dichos dispositivos. Para dispositivos con conexión a Internet el protocolo TCP/IP no puede ser utilizado, debido a que requiere demasiado espacio en memoria y no optimiza el consumo de energía. Dada la gran cantidad de tecnologías celulares que han emergido en tal mercado, es extremadamente difícil proveer soporte para la comunicación entre los dispositivos. Para resolver este problema, una nueva tecnología de hardware a sido propuesta “Bluetooth” [25]. Cualquier dispositivo con un chip Bluetooth podrá comunicarse con otro dispositivo que posea un chip similar, independientemente de las tecnologías de comunicación que estén usando.

En el pasado reciente, las compañías de teléfonos celulares han presentado un importante patrón de crecimiento. El número de usuarios ha estado aumentando constantemente, pero el tiempo de aire por usuario tiende a ser constante. Es por esto que muchas compañías, para incrementar el promedio de tiempo de conexión por usuario, han comenzado a proporcionar servicios de datos sobre sus redes. Es decir, para que el usuario use su celular tanto para comunicación por voz como para comunicación por datos. Servicios de datos típicos incluyen: Chat, e-mail, navegación en Internet. Un ejemplo de este tipo de servicios es el "Servicio de Mensajes Cortos", SMS por sus siglas en inglés (Short Message Service), el cual es un servicio de datos sobre una red GSM que permite al usuario enviar mensajes de 160 caracteres como máximo (similar a un pager). Obviamente con este servicio no se puede navegar la red, chequear e-mail, o chatear. Para esto, las redes GSM, proveen otro servicio llamado GPRS (General Packet Radio Service) que permite el envío de información a través de la red celular.

### **2.2.1 Tecnología WAP**

En los últimos tiempos ha habido un esfuerzo constante, por parte de las empresas y grupos de investigación, para el desarrollo de estándares que provean servicio de datos sobre dispositivos "hand-held". WAP (Wireless Application Protocol) es un protocolo que abarca los protocolos adaptados para pequeños dispositivos. WAP ha sido desarrollado por el "WAP Forum" [16] y corre sobre protocolos subyacentes como IP o SMS. En el modelo WAP, un proveedor de servicios opera sobre un gateway (puerta de enlace) WAP para convertir el contenido de Internet en un subconjunto de tags HTML que son mostrados por el mini-navegador de un dispositivo móvil.

HTML, el lenguaje estándar de Internet, no está optimizado para estos dispositivos. Los dispositivos móviles están caracterizados por pantallas de pequeños tamaños, capacidades limitadas de entrada y un ancho de banda también limitado. Un documento modelo de HTML consiste en encabezados, títulos, párrafos, etc. los cuales no sirven para una pantalla que posee, por ejemplo, 10 filas de alto, y 15 caracteres de ancho. Teniendo en mente las restricciones de memoria de los dispositivos móviles, el navegador no debe ser pesado, para esto existen nuevos estándares tales como: HDML, WML y Compact HTML [17].

## **2.3 Computación Ubicua (Ubiquitous Computing)**

El término "computación ubicua" fue acuñado desde el Centro de Investigación de Xerox en Palo Alto en 1991 por Mark Weiser [12]. La propuesta de Weiser se fundamenta en que la interacción actual usuario-computadora no es la adecuada. La computadora es un dispositivo demasiado complejo, su manipulación requiere mucha dedicación exclusiva, distrayendo la atención del usuario de la tarea que tiene que realizar. Por tanto, parece lógico defender la "desaparición" de las computadoras. Entiéndase bien este término, los dispositivos de computación quedan ocultos, invisibles a los usuarios a los cuales dan servicios [13].

Según M. Weiser [12], precursor de la computación ubicua, establece que esta tecnología tiene como meta incrementar el uso de la computación haciendo que muchas computadoras estén disponibles a través de entornos físicos, pero haciendo que estas estén invisibles a los usuarios. Actualmente un gran número de investigadores alrededor del mundo están trabajando en el framework de computación ubicua. Estas

investigaciones tienen un gran impacto en todas las áreas de la ciencia de la computación, incluyendo componentes de hardware (por ejemplo chips), protocolos de red, aplicaciones, privacidad y métodos computacionales.

## 2.4 Conceptos de Contexto y Sensibilidad al Contexto

En esta sección presentaremos algunas definiciones de contexto y sensibilidad al contexto junto con las diferentes clases de adaptaciones. Es importante analizar estos conceptos para luego poderlos aplicarlos al prototipo que vamos a realizar.

### 2.4.1 Contexto

Un gran número de investigadores en los últimos años ha realizado definiciones de contexto y sensibilidad al contexto. A continuación presentaremos algunas de las definiciones más relevantes, principalmente la de los pioneros en este tema.

Según Schilit y Theimer [49], pioneros en las áreas de investigación de context-aware computing con el trabajo “Disseminating Active Map Information to Mobile Host” se refirieron al contexto como ubicación, identidad de la cercanía de personas y objetos, y los cambios que ocurren para esos objetos.

Schilit y Theimer proponen dos categorías para el contexto:

- Los factores humanos:
  - Usuario: Esta caracterizado por los hábitos del usuario, el estado mental o características fisiológicas.
  - Entorno social: esta caracterizado por la proximidad de otros, por sus relaciones sociales y tareas colaborativas.
  - Tareas: define las actividades dirigidas al objetivo o los objetivos generales del usuario.
- El entorno físico:
  - Ubicación: la definen como absoluta, por ejemplo una coordenada GPS, o relativa, por ejemplo una habitación particular.
  - Infraestructura: describe el ambiente informático y el entorno de interacción.
  - Condiciones: Describe las condiciones físicas del entorno tales como, ruido, claridad, presión, etc.

Otra definición establecida por Brown et al [50] define al contexto como ubicación, identidad de las personas alrededor del usuario, hora, estación, temperatura, etc.

Por ultimo, para dar una visión más abstracta del tema, Dey [51] establece una definición en relación a su trabajo sobre herramientas de contexto. Según éste, el contexto es cualquier información que puede ser usada para caracterizar la situación de una entidad. Una entidad es una persona, lugar u objeto que es considerado relevante para la interacción entre un usuario y una aplicación, incluyendo al usuario y la aplicación.

### 2.4.2 Sensibilidad al Contexto (Context-awareness)

Según Schilit y Theimer [49], las aplicaciones sensibles al contexto son aquellas que se adaptan de acuerdo a ciertos parámetros, tales como: ubicación, cercanía a

objetos o personas, tipos de dispositivos accesibles, como así también los cambios que les pueden ocurrir a estos parámetros en el tiempo. Para cada uno de estos parámetros existen tres aspectos importantes a tener en cuenta: donde esta, con quien esta, y que recursos están cercanos y disponibles.

Otra definición de sensibilidad al contexto es la propuesta por Dey [51], la cual dice que: un sistema es sensible al contexto si este usa el contexto para proporcionar información relevante y/o servicios al usuario, donde las relevancias dependen de las acciones del usuario.

Por último, desde nuestro punto de vista, un dispositivo será sensible al contexto si posee información acerca de las “circunstancias” en las que se encuentra y dependiendo de estas pueda reaccionar de diferentes maneras. Por ejemplo: un celular sensible al contexto podría determinar que se encuentra en una sala de reunión y que el usuario también se encuentra allí. Con lo cual este dispositivo podría concluir que el usuario esta actualmente en una reunión y por este motivo rechazar aquellas llamadas que no sean importantes.

## 2.5 Servicios Basados en la Ubicación (Location-Based Service, LBS)

En estos tiempos en los cuales existe una significativa competencia en las telecomunicaciones, las empresas de redes móviles buscan continuamente nuevas e innovadoras maneras de crear o proveer servicios a sus clientes. Una de estas innovaciones se basa en el surgimiento de servicios altamente personalizados. Un método para brindar estos servicios es basarse en la posición del usuario. A continuación explicaremos los “Servicios Basados en la Ubicación” (Location-Based Service, LBS).

### 2.5.1 Tecnologías

Una de las tecnologías más obvias para el uso de LBS es el uso del ampliamente conocido Sistema de Posicionamiento Global (Global Position System, GPS) [52]. Sin embargo existen otras tecnologías de posicionamiento además de GPS, tales como, redes celulares las cuales determinan la posición de un dispositivo móvil realizando una triangulación en función de la señal enviada desde el dispositivo hacia las celdas de la red.

Por otro lado, los datos geográficos son un aspecto importante en cualquier tipo de sistema basado en la posición. Los Sistemas de Información Geográficos (Geographic Information Systems, GIS) proveen herramientas para administrar datos basados en mapas tales como las construcciones realizadas por el hombre (calles, edificios, etc.), características del terreno (montañas, ríos, etc.), etc. Los GIS también se utilizan para administrar puntos de interés tales como: estaciones de servicios, restaurantes, clubes, etc.

### 2.5.2 Servicios

Existen diferentes tipos de servicios que se pueden proveer en función de la posición del dispositivo móvil

- **Información basada en la posición:** por ejemplo, supongamos a una persona buscando restaurantes usando su dispositivo móvil basado en la tecnología WAP [16]. Una aplicación LBS podría interactuar con otros componentes para determinar la posición del usuario y proveer un listado de restaurantes que se encuentren a una proximidad  $x$  del usuario móvil.
- **Servicios de emergencia:** estos son otros tipos de servicios que se pueden proveer haciendo uso de la posición del usuario. Supongamos que un usuario móvil realiza un llamado a un número de emergencia  $X$ . Una información importante, y que el usuario podría no saber, es la posición en la que se encuentra.
- **Seguimiento:** por último este es otro tipo de servicio que se puede proveer utilizando como punto de partida la posición del usuario. Básicamente el termino “seguimiento” se refiere a monitorear las posiciones por las que el usuario se ha movilizado. Cabe acotar que con este tipo de servicio podemos hablar de seguimiento no solo de usuarios, sino también de vehículos o cualquier otro dispositivo que pueda movilizarse de un punto a otro. Por ejemplo, existen aplicaciones de seguimiento vehicular [53] que le permiten determinar a una empresa en donde se encuentran cada uno de los móviles de su flota de vehículos.

## Capítulo 3

### E-Commerce y M-Commerce

#### 3.1 Comercio electrónico (e-commerce)

El comercio electrónico o e-commerce consiste en la compra, venta y comercialización de productos o servicios en una red de computadoras.

Una definición alternativa para el comercio electrónico puede ser descripta como la conducta de comunicaciones comerciales de negocio y gerenciamiento a través de métodos electrónicos, tales como Intercambio Electrónico de Datos (EDI, por sus siglas en ingles) y sistemas automatizados de recolección de datos.

El comercio electrónico puede también implicar la transferencia de electrónica de la información entre negocios.

Según un estudio realizado por la compañía “Forrester” [11] (citado en Kessler, 2003) el valor de las ventas generado por el comercio electrónico en los Estados Unidos fue U\$S 12.2 mil millones en el año 2003.

##### 3.1.1 Desarrollo Histórico

El significado de comercio electrónico ha ido cambiado a lo largo del tiempo. Originalmente, “comercio electrónico” significaba la facilidad de realizar transacciones comerciales electrónicamente, generalmente usando una tecnología como “Electronic Data Interchange” [18] (EDI, introducida al final de los años 70) para enviar documentos comerciales como ordenes de compras o facturas electrónicamente.

Con el surgimiento de nuevas actividades en el comercio electrónico se debió refinar el término a “Comercio Web” – la compra de bienes y servicios sobre la World Wide Web vía servidores seguros (HTTPS, un protocolo especial el cual cifra los datos confidenciales para la protección del cliente).

### **3.1.2 Claves del éxito del comercio electrónico**

Son muchos los factores [19] que existen para que el comercio electrónico tenga éxito. Para el objetivo de este trabajo no es necesario indicarlos a todos, solo tenemos la intención de mostrar algunas de los ítems más importantes:

- 1- Valorar a los clientes. Los vendedores pueden alcanzar este objetivo ofreciendo un producto en línea que atraiga a clientes potenciales con un precio competitivo, tal como ocurre en el comercio no electrónico.
- 2- Otorgar servicio y performance. La posibilidad de realizar transacciones comerciales fácilmente, junto con la experiencia del vendedor puede ser una manera de alcanzar estas metas.
- 3- Proveer un website atractivo. El buen uso de colores, gráficos, animaciones, fotografías, fuentes, y un correcto porcentaje de espacios en blanco pueden ayudar para crear un website atractivo.
- 4- Incentivar a los clientes para que luego de una compra vuelvan a realizar otra en un periodo de tiempo. Para este fin se puede hacer uso de cupones de descuento, ofertas especiales, etc.

### **3.1.3 Problemas asociados al comercio electrónico**

Más allá de que una empresa ofrezca los servicios de comercio electrónico siguiendo las pautas que se presentaron anteriormente para idear una estrategia de ventas, los problemas pueden presentarse. Algunos de los orígenes de tales problemas son:

- 1- Falta de conocimiento de los clientes, por qué y cómo compran. Un producto con grandes posibilidades de venta puede fallar si la empresa no entiende los hábitos, expectativas y motivaciones del cliente.
- 2- No considerar la situación competitiva. Uno puede tener la habilidad para construir un modelo de negocios viable para la venta de libros vía medios electrónicos, aunque quizás vaya a ser imposible competir con empresas como “Amazon.com” [20].
- 3- Inhabilidad para predecir la reacción del entorno. ¿Qué harán los competidores? Podrían introducir marcas de fábricas competitivas o desarrollar nuevos websites competitivos. ¿Se iniciara una guerra de precios?, ¿Qué hará el gobierno? La investigación de los competidores, las industrias y los mercados puede atenuar algunas consecuencias.
- 4- Sobrestimación de las capacidades de los recursos de la empresa. ¿Pueden el personal, el hardware, el software y los procesos manejar la estrategia propuesta? Estos problemas pueden resolverse mediante un buen planeamiento de los recursos y entrenamiento del personal.
- 5- Subestimación de los tiempos requeridos. Ajustar el funcionamiento de una empresa de comercio electrónico puede tomar considerable tiempo y dinero, y el error en entender secuencialidad y el orden de las tareas puede implicar también importantes costos para la empresa.

Existen otros problemas asociados al comercio electrónico que no son relevantes para el objetivo de este trabajo, es por esto que no han sido incluidos, para más información [19].

### **3.1.4 Productos convenientes para la venta electrónica**

Ciertos productos resultan ser más convenientes para la venta en línea, otros resultan ser todo lo contrario ya que es casi imposible realizar u ofrecer la venta por medio electrónico. Muchas compañías puramente virtuales ofrecen productos digitales, incluyendo almacenamiento, recuperación y modificación de datos, música, películas, educación, comunicación, software, fotografía, y transacciones financieras. Ejemplos de este tipo de compañías pueden ser: Amazon [20], Google [21], eBay [22], y Paypal [23].

Las empresas virtuales también pueden ofrecer con éxito productos y servicios que no sean digitales. Tales productos suelen tener una elevada relación costo-peso. Es por esto que muchas veces se hace difícil el envío a lugares remotos. A diferencia de, por ejemplo, CDs, DVDs o libros, los cuales pueden ser fácilmente empaquetados y enviados.

Los productos tales como piezas de repuestos también son buenos candidatos para la venta electrónica o en línea. Un factor para el éxito de este tipo de ventas puede consistir en ofrecer la información al cliente lo mas exacta posible, por ejemplo, indicando que número de pieza utiliza la versión de su producto.

Otro campo de venta importante es el de la pornografía dado que los productos y servicios cumplen con los requisitos para las ventas virtuales.

Los productos inadecuados para el comercio electrónico abarcan productos con una baja relación costo-peso así como también productos que tienen que ser probados (perfumes, ropa, etc.) antes de efectuar la compra.

### **3.1.5 Aceptación del e-commerce**

Los consumidores han tenido una lenta aceptación comparada con la expectativa que tenían los autores. Incluso en las categorías de productos convenientes para la venta electrónica, las compras avanzaron lentamente. Esto se debe a varias razones, entre las que se encuentran:

- Preocupación de los usuarios acerca de la seguridad de la tecnología. Mucha gente no desea usar su tarjeta de crédito sobre la Internet por miedo al robo de sus datos.
- Problemas para acceder a la web, particularmente en países en desarrollo. El bajo grado de penetración de acceso a Internet en algunos sectores reduce ampliamente el potencial del comercio electrónico.

## 3.2 Comercio móvil (m-commerce)

De la misma manera que el e-commerce, el m-commerce, consiste en la compra, venta y comercialización de productos o servicios en una red de computadoras. La diferencia con el comercio electrónico se encuentra en que el m-commerce es realizado a través de dispositivos móviles [6].

Actualmente el m-commerce es utilizado principalmente para la venta de ringtones y juegos. Por otro lado los servicios que proveen las nuevas redes 3G/UMTS [32, 33] posibilitan el pago de servicios basados en la posición, tales como mapas, audio y video. Otros tipos de servicios es el envío de información vía mensajes SMS [34], ejemplos de este tipo de servicio pueden ser envíos de:

- Resultados de un partido de fútbol.
- Estado y pronóstico del clima.
- Estado del tránsito en autopistas.
- Carteleras de cines.
- Etc.

En la actualidad los principales métodos de pago utilizados por el comercio móvil son:

- Llamadas a números especiales que debitan de la cuenta telefónica móvil un determinado monto (sistema de telefonía prepaga).
- Registro del monto de la compra en la factura telefónica móvil (sistema de telefonía con facturación).

El término “comercio móvil” comenzó a utilizarse a finales de los años 90 durante el auge de las .com. El surgimiento de aplicaciones m-commerce altamente provechosas fue posible gracias a la telefonía móvil de banda ancha provista por los servicios de los celulares de tercera generación.

### 3.2.1 Tendencias del Comercio móvil

Las PDAs y los teléfonos móviles han llegado a ser tan populares que muchos negocios están comenzando a utilizar el comercio móvil como el método para más eficiente para satisfacer las demandas de sus clientes [6]. Aunque la mayor cantidad de avances en comercio móvil se ven en algunos países de Asia y Europa, Canadá y Estados Unidos están comenzando a hacer uso de esta tecnología. Los bancos y otros servicios financieros están explorando el uso del comercio móvil para agrandar su negocio permitiendo que sus usuarios tengan acceso a la información en donde sea que estén, por ejemplo balances de cuentas, cotizaciones de la bolsa, etc. Las noticias también están siendo un campo de acción en el que se mueve el comercio móvil. Los usuarios pueden suscribirse a los títulos de diarios y recibir esta información en sus dispositivos móviles.

Queda claro, después de párrafo anterior, que el uso del comercio móvil esta en etapa de exploración. Actualmente se están estableciendo regulaciones sobre el uso de esta tecnología. Uno de las aristas más importantes, en la cual las compañías de telefonía móvil están haciendo importantes inversiones, es la seguridad en las transacciones para proteger los datos de los clientes contra cualquier tipo de intrusiones en línea.

### 3.2.2 Estado actual del Comercio móvil

Si bien el comercio móvil es una tecnología que está surgiendo ya se puede encontrar en el mercado diferentes utilidades de la misma. Tal es el caso de la empresa mobipay [35], una empresa española que pone a disposición de clientes y comercios la posibilidad de realizar transacciones de pago seguro a través del teléfono móvil.

Mobipay ofrece a los usuarios la posibilidad de realizar sus compras habituales con su teléfono móvil. El móvil se convierte así en una cartera donde se pueden incorporar diferentes medios de pagos: tarjetas financieras o el propio saldo telefónico, contra los que realizar los pagos habituales.

De esta manera, realizar un pago por móvil, es autorizar un cargo contra la cuenta asociada al medio de pago seleccionado, ya sea contra la cuenta de nuestra tarjeta financiera o bien contra nuestro saldo telefónico.

De esta forma, con mobipay se puede utilizar el móvil para pagar en un taxi, en una máquina expendedora, las compras en Internet, o incluso pagar el estacionamiento en zona de aparcamiento regulado.

La comunicación se realiza mediante mensajes interactivos entre el teléfono móvil del comprador y mobipay, a modo de pregunta-respuesta, en la que el cliente recibe toda la información sobre la operación de pago que se está llevando a cabo, y la solicitud para que, también a través de su móvil, la autorice.

Por otro lado, los nuevos teléfonos móviles desarrollados por las empresas más renombradas (Nokia [36], Sony-Ericsson [37], Motorola [38], etc.) están siendo ideados para que provean soporte o complementos para m-commerce. Por ejemplo, los nuevos teléfonos de Nokia poseen una aplicación denominada “Monedero Electrónico” [39]. Esto es una aplicación que ofrece un espacio seguro de almacenamiento en el teléfono para la información utilizada en los pagos móviles, como por ejemplo, números de tarjetas de crédito. Incluso es posible autenticar las transacciones mediante una firma digital. Para más información sobre los dispositivos Nokia con soporte de dicha aplicación dirigirse a la referencia [40].

# Capítulo 4

## Hipermedia e Hipermedia Física

### 4.1 Hipermedia

En nuestra cultura, antiguamente, tras la invención de la impresión, la organización de la información en un mensaje y la correspondiente utilización del mismo se ha basado en un modelo secuencial: un libro es una secuencia de textos organizados por capítulos. Las páginas se leen de la primera a la última, etc.

El hipertexto es una tecnología que surgió como una alternativa a esta estructura de linealidad que organiza una base de información en bloques distintos de contenidos, conectados a través de una serie de enlaces cuya activación o selección provoca la recuperación de información [41].

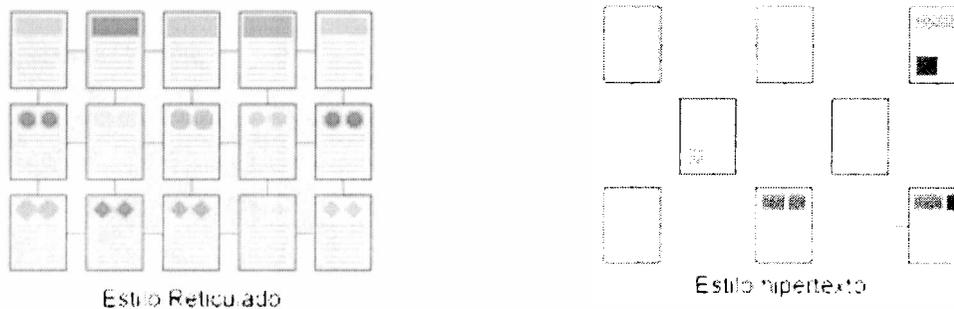
#### 4.1.1 Introducción

El hipertexto ha sido definido como un enfoque para manejar y organizar información, en el cual los datos se almacenan en una red de nodos conectados por enlaces. Los nodos contienen textos y si contienen además gráficos, imágenes, audio, animaciones y video, así como código ejecutable u otra forma de datos se les da el nombre de hipermedia, es decir, una generalización de hipertexto.

Considerando cómo se representa el conocimiento humano, el hombre opera por asociación, saltando de un ítem al próximo, en forma casi instantánea. El paradigma hipermedia intenta modelar este proceso con enlaces entre segmentos de información contenidos en nodos.

A diferencia de los libros impresos, en los cuales la lectura se realiza en forma secuencial desde el principio hasta el final, en un ambiente hipermedial la "lectura" puede realizarse en forma no lineal, y los usuarios no están obligados a seguir una secuencia establecida, sino que pueden moverse a través de la información y hojear intuitivamente los contenidos por asociación, siguiendo sus intereses en búsqueda de un término o concepto. En consecuencia, se pueden presentar distintos estilos para recorrer un mismo libro o conjunto de documentos. En la **Figura 4.1**, a continuación, se representan el estilo secuencial, el estilo jerárquico, el estilo reticulado y el de hipermedia:





**Figura 4.1: Estilos de movimiento en un libro o conjunto de documentos**

En términos más sencillos, y a la vez más amplio, hipermedia es un sistema de bases de datos que provee al usuario una forma libre y única de acceder y explorar la información realizando saltos entre un documento y otro.

#### **4.1.2 Características principales de un Sistema Hipertexto**

Esta tecnología de información ha sido defendida y elogiada debido a las grandes ventajas que proporciona; sin embargo no todos los hipertextos que se han implementado y están disponibles en distintas plataformas e instalaciones cumplen cabalmente con todas las expectativas de los usuarios. Un sistema hipertexto, en términos ideales, debe cumplir con las siguientes características:

- Debe proveer un medio adecuado para organizar y presentar información poco o nada estructurada, no ajustada a esquemas tradicionales y rígidos como es el caso de las bases de datos. Pueden utilizarse esquemas jerárquicos para la utilización de sistemas de documentación de texto tradicionales, muy organizados o simplemente creando estructuras de redes con poco o ningún atributo de precedencia.
- Tener asociada una interfaz de usuario muy intuitiva, pues se pretende imitar el funcionamiento de la mente humana, haciendo uso de modelos cognitivos, por lo que el usuario no debería realizar grandes esfuerzos para obtener la información requerida.
- La información se encuentra distribuida y puede ser accedida en forma concurrente por varios usuarios, por lo tanto es un ambiente compartido.
- Tiene asociados varios mecanismos de recuperación y búsqueda de información a través de las navegaciones, ya sean dirigidas o no dirigidas.

Estas características hacen que la hipermedia sea utilizada en una variedad muy amplia de aplicaciones, en las cuales se tienen al menos los siguientes requerimientos: gran cantidad de información organizada en distintos fragmentos y contextos, los cuales pueden estar relacionados entre sí, que el usuario necesita en forma discreta, y que pueda estar implantado en un ambiente electrónico o computacional. Dados estos requerimientos, el dominio de las aplicaciones hipermediales incluye: ayudas y documentación, diccionarios y enciclopedias electrónicas, herramientas CASE para desarrollo de software, organizadores de ideas, sistemas de información turísticos y

geográficos, venta electrónica, soporte para enseñanza y aprendizaje, trabajo colaborativo y comunicaciones. Estas aplicaciones pueden ser implementadas tanto en ambientes cerrados o en ambientes abiertos. Un ambiente cerrado es aquel donde todo el repositorio de información se encuentra concentrado en una única unidad de almacenamiento o servidor, y los enlaces entre hiperdocumentos sólo pueden realizarse entre fragmentos de información que residen en el mismo servidor. En cambio, cuando el ambiente es abierto, los contenidos y fragmentos de información pueden encontrarse distribuidos en diversos repositorios de almacenamiento o varios servidores, es decir la información se encuentra físicamente distribuida en servidores distintos y se permiten hacer referencias entre hiperdocumentos que residen en servidores distintos: este es el caso del World Wide Web [42].

### 4.1.3 Modelos de hipertexto

Según [43], un modelo es la expresión de una realidad o sistema complejo mediante algún lenguaje formal o simbolismo gráfico que facilita su comprensión y el estudio de su comportamiento. Por su propia definición, un modelo debe cumplir con tres requisitos básicos:

- General, es decir, debe ser válido para cualquier aplicación del campo que formaliza.
- Abstracto, ya que con esto se puede separar las características particulares del objeto de estudio para extraer su esencia.
- Consistente, para lograr que cada elemento tenga una única definición, acorde con la función que se espera que represente y coherente con el resto de componentes del modelo.

Aplicando estas definiciones al campo de la tecnología hipermedial, un modelo es un marco general y teórico que formaliza todas las características y funciones esenciales o deseables, que se puedan incluir en cualquier aplicación de este tipo. Evidentemente el modelo será más completo, en la medida que cuente y exprese más características.

Un modelo en el contexto de sistemas hipermediales, debe ser capaz de representar tanto la estructura estática como el funcionamiento dinámico de sus componentes. Se debe agregar al modelo, sin embargo, los aspectos relacionados con el intercambio de información entre el sistema y los usuarios, siendo éstos el autor y el lector del hipermedia, con la finalidad de describir aquellos elementos de interface que facilitan la observación del documento en un computador.

En el nivel más elemental, todos los sistemas hipertexto están basados en un modelo básico, el cual sigue siendo utilizado y asumido por una variedad de investigadores. El modelo básico está dividido en: submodelo de datos y submodelo de procesos.

En el submodelo de datos, los nodos se interrelacionan mediante enlaces dirigidos, para formar la estructura de un dígrafo [44]. Todas las operaciones de inserción, eliminación y actualización de nodos y enlaces están provistas de las operaciones de grafos. Los nodos, son las unidades primitivas para organizar la información en el espacio hipermedial: funcionan como colecciones de datos no estructurados los cuales son agrupados de forma tal de crear una entidad lógica. Esta entidad existe para proveer abstracciones coherentes de información dentro del espacio

de información representado en la estructura hipermedial. Los enlaces son las otras unidades fundamentales del submodelo de datos: implementan la dirección de las conexiones entre nodos, están asociados como anclas a un nodo fuente y proveen al usuario la habilidad de activarlos para moverse hacia un nodo destino. Algunos autores distinguen dos tipos de enlace, los referenciales que funcionan como referencias cruzadas entre documentos, y los enlaces organizacionales, que se utilizan para establecer estructuras explícitas por la conexión de un nodo padre a sus nodos hijos, formando árboles u otro tipo de jerarquías.

El submodelo de procesos se refiere, en el nivel básico, a los mecanismos de acceso a la información representada en el hiperdocumento, se requiere de la descripción de la dinámica, es decir las navegaciones que constituye el primer medio que tiene el usuario para acceder a los contenidos y los demás procesos que intervienen en la interacción con el usuario. Por lo tanto, el submodelo de procesos debe contener, además del proceso propio de exploración vía enlaces, otros procesos que pueden influir sobre la forma en que los usuarios inspeccionan la estructura de los hiperdocumentos y que permiten al usuario obtener información de acuerdo a distintas necesidades estudiadas para caracterizar este paradigma.

## 4.2 Hipermedia Física

Si quisiéramos extender el paradigma de Hipermedia al mundo físico real, tendríamos que hablar de la existencia de un modo de representación de los objetos físicos y un modo de relación entre los mismos, de la misma manera que existen una conexión o link en la relación de objetos digitales en un sistema de Hipermedia tradicional. Es decir, necesitaríamos cierto conocimiento de los objetos físicos que nos permita obtener los links que nos lleven a otros objetos o a información relacionada con el mismo, para lograr así un espacio hipermedial.

### 4.2.1 Introducción

Teniendo en cuenta las características definidas en el párrafo anterior, en este punto surge el concepto de Hipermedia Física (*Physical Hypermedia, PH*), cuya definición podríamos sintetizarla como un término que se utiliza para describir cualquier sistema de hipermedia relacionado con conceptos de ubicación en un espacio físico del mundo real. En otras palabras, Hipermedia Física modela el espacio físico utilizando el paradigma de hipertexto, permitiendo navegar al mismo tiempo, tanto objetos verdaderos como virtuales integrados en un mismo sistema.

La idea del PH primero fue introducida por [47], quien explica que en muchos dominios existen actividades caracterizadas por el hecho de tener que relacionar información de medios físicos y digitales, por lo que es necesario poder relacionar estos medios con sus respectivos objetos y relaciones en un sistema de información total, teniendo en cuenta los distintos problemas que presentan cada uno de estos medios. En particular para tratar el problema de llevar los objetos del mundo físico a un sistema hipermedial surge el concepto de Hipermedia Física.

Según [46], Hipermedia Física es un término que formaliza y define la computación ubicua, y si bien es correcto pensar que el problema que trata PH está muy relacionado con Ubiquitous Computing [12], no podemos afirmar que ambas sean lo mismo o traten exactamente los mismos problemas, dado que en los trabajos existentes referentes a Ubiquitous Computing, como [48] y [54], la atención principal ha sido el

desarrollo de las infraestructuras para adaptar los dispositivos que nos brindan información relacionada con los objetos físicos reales, mientras que, el concepto de Hipermedia Física centra su definición en los métodos para asociar estos objetos físicos más allá de los dispositivos en cuestión. Por lo tanto podemos separar ambos conceptos y centrar la Ubiquitous Computing en la interacción dispositivo-objeto, y la Hipermedia Física en la relación de objetos en un espacio físico. Por supuesto que ambas disciplinas se cruzarán e interactuarán en distintos puntos de análisis, definiciones y ejemplos, pero el presente trabajo gira en torno a Hipermedia Física por lo que si bien se harán referencias a los trabajos realizados de Ubiquitous Computing, todos los análisis, definiciones y formulaciones están orientadas a las características y aplicaciones de la Hipermedia Física.

#### 4.2.2 Características de Hipermedia Física

Como venimos explicando hasta el momento, la hipermedia física, relaciona objetos físicos con el espacio hipermedial, de manera de aumentar la dimensión de los mismos con la información digital que puede ser alcanzada por un usuario móvil mientras dicho usuario se encuentra en un Centro Comercial enfrente a una oferta de un televisor en una casa de electrodomésticos.

En este punto, conviene que analicemos los objetos físicos más detenidamente e introduzcamos algunos conceptos que nos permitan una mayor comprensión de la definición y utilización de la Hipermedia Física de manera de familiarizarnos con los conceptos introducidos y adquirir una definición más técnica:

- Objeto Físico: Es cualquier objeto “real” que puede ser explorado físicamente, el cuál tendrá una presencia física en el sistema, y un mecanismo para detectar si un usuario se encuentra cerca de él.
- Nodo Físico: Representa un Objeto Físico y sus interrelaciones con otros objetos dentro de un sistema hipermedial. Un Nodo Físico tiene asociado:
  - Un sistema de localización que será independiente de cada dominio donde se implementa una arquitectura de Hipermedia Física.
  - Un conjunto de nodos adyacentes o vecinos que estarán cercanos a él en función a un criterio predeterminado que puede estar relacionado con el sistema de localización.
- Link Física: Representa un link en el mundo real. Es un conjunto de pasos que deben ser seguidos por un usuario para ir desde un Nodo Físico origen (donde se encuentra el usuario) a un Nodo Físico destino. Por ejemplo, un usuario en un Centro Comercial esta observando un par de zapatos y desea tener una referencia a todos los locales que se encuentra en dicho Centro que realicen venta de zapatos, y como llegar hasta ellos.

Un Link Física, presenta tres tipos de información:

- Información de destino: La que describe porque deberíamos trasladarnos al nodo destino indicado por el link.
- Información Trasversal: Indica como el usuario puede trasladarse

físicamente desde el Nodo Origen al Nodo destino.

- Información de relación con el origen: Nos provee la información que describe al Nodo Destino en el contexto de trasladarse desde el Nodo Origen.
- Link Digital: Es un link tradicional de la Web, basado en el concepto de link Hipermedial. El link Digital provee información adicional de un objeto física al que esta relacionado. En el ejemplo de los zapatos, un link digital podría permitirle al usuario informarse sobre donde y como se elaboran los zapatos que esta a punto de adquirir.
- Documento Digital: Es cualquier componente de la web; un site, una página, un documento hipermedial, etc.

Teniendo en cuenta estos conceptos, estamos en condiciones de describir los distintos tipos de relaciones que surgirán entre los distintos componentes de una arquitectura de Hipermedia Física:

- Digital a digital: Es una relación equivalente a link hipermedial de la Web. La fuente es el presente documento digital que esta siendo visto por el usuario y el destino es un documento electrónico relacionado con el origen por algún criterio o contexto.
- Física a digital: Es una relación entre un objeto físico y un documento digital. Dicha relación se establece utilizando un link digital, que será mostrado al usuario en función de la cercanía de este con el objeto físico en cuestión. El objetivo de esta relación es que el usuario pueda obtener información en la web sobre el objeto real ante el cual se encuentra.
- Digital a física: Aquí, se relacionan la información digital que el usuario esta viendo en una página, con objetos físicos que se encuentran cercanos a la posición actual del mismo. La relación se establece mediante un link físico que le indica al usuario como tiene que desplazarse para llegar al objeto en cuestión. El objetivo de esta relación es orientar al usuario en la búsqueda de objetos reales que son de su interés y se encuentran en su “cercanía”. El término “cercanía” esta determinado por el nodo físico del sistema donde se encuentra, y sus nodos adyacentes, pero es importante resaltar que la “cercanía” debe tener un nexo importante con la cercanía de los objetos reales de interés, de manera que la posibilidad de “recorrer” un link físico este al alcance del usuario en cuestión.
- Física a física: Esta relación representa el movimiento real de un usuario entre diversos objetos físicos, donde el sistema deberá interpretar de manera inteligente la cercanía del usuario con cada uno de los objetos físicos que se encuentran en el sistema, de manera de brindarle al usuario la información correspondiente a cada uno de ellos en función de donde se encuentre.

Para comprender mejor estas relaciones es importante tener en claro que cualquier sistema de **Hipermedia Física** estará compuesto esencialmente por nodos físicos interrelacionados entre sí, los cuales pueden considerarse, analizándolos más en profundidad, como nodos que se encuentran en una red del hipermedia ligados entre sí de manera física o digital. En el caso de que la conexión sea digital se utiliza el conocido paradigma de navegación de la web (www) para desplazarse en el sistema, mientras que, cuando la relación es física, la conexión tiene que ser "caminada" por el usuario.

#### **4.2.3 Criterios de diseño y desarrollo de un Sistema Físico Hipermedial**

En el inciso anterior cuando nos referimos a conexiones en un sistema físico hipermedial mencionamos el paradigma de navegación de la web (links digitales) y también el hecho de tener que "caminar" de un punto a otro en un espacio físico determinado (links físicos) que puede variar en función al dominio de la aplicación, por lo que es necesario que exista una relación entre los objetos del dominio más allá de los links en cuestión. Ahora bien, a partir de esta relación surge la pregunta que da origen a esta sección del trabajo: ¿Cuáles es el criterio único en el que nos podemos basar para definir la relación entre los distintos objetos del dominio? La respuesta a esta pregunta es que no existe un único criterio para representar y definir la relación entre los objetos del dominio sino que, tal como en el mundo real existen múltiples criterios para "recorrer" los objetos relacionados en un Sistema Físico Hipermedial, donde entendemos que "recorrer" significa desde hacer un click hasta trasladarse físicamente de un lugar a otro. Suponiendo un dominio de un Sistema de Hipermedia Física aplicado en un Hipermercado, ejemplos de estos criterios serían, el gusto personal de un cliente, indicándole los productos que son de su interés y como acceder a ellos. O también el interés de venta de la firma, o los distintos rubros de los productos y las necesidades complementarias entre uno y otro, o bien creando links relacionados por la calidad del producto, etc. La realidad indica que hay múltiples criterios de relación entre objetos, y la Hipermedia Física nos brinda la herramienta que nos permite crear relaciones abarcando todos los criterios posibles y uniendo los objetos reales y virtuales de un Sistema Físico Hipermedial.

Por lo tanto es importante que quede claro que para diseñar y desarrollar un Sistema Físico seguimos múltiples criterios donde los links físicos pueden indicarnos como llegar caminando, en auto, o hasta volando desde un punto de partida que representa un lugar u objeto físico hasta un punto de llegada, siguiendo detalladamente un conjunto de referencias físicas de desplazamiento, las cuáles están directamente relacionadas con el dominio sobre el cuál se implementa el sistema.

#### **4.2.4 Ejemplos de escenarios de aplicación de Hipermedia Física**

a – Museo

Tomando como referencia el ejemplo presentado por [63], supongamos un Museo de Historia en el que se ha montado un sistema de hipermedia Física. Un turista va a visitar el museo, se registra en el acceso ingresando en el sistema los temas que son de su interés, lugar de procedencia, si es la primera vez que viene, datos personales, etc., conecta su PDA o celular al sistema y se apresta a recorrer el museo.

A partir de este momento, el sistema del Museo podrá registrar y ubicar al viajero a medida que este se desplace por las instalaciones, pudiendo enviarle información sobre las galerías o secciones que debe visitar (e indicándoles como llegar a las mismas) en función de los temas de interés registrados en el acceso. Además, se podrá producir una interacción donde el turista puede solicitar información sobre áreas de servicios, salidas o algún tema transversal, y el sistema del museo además de esos datos también podrá brindar información sobre los lugares u objetos que el viajero va recorriendo, con datos adicionales obtenidos de la web o de una biblioteca virtual sobre las esculturas u objetos del museo que el usuario va visitando.

Tomando un Ejemplo de este tipo de interacción de información, digamos que el viajero ha entrado al museo siguiendo una galería donde se presenta pinturas egipcias, y el sistema tiene registrado que como un tema de interés para este usuario la cultura egipcia, entonces enviará a la PDA de usuario todas las áreas del museo donde se podrán encontrar objetos relacionados con esta cultura y que pasos debe seguir para acceder al mismo, así como también información virtual sobre la historia egipcia obtenida de la web o de bibliotecas virtuales, donde existen otros museos que también brinden exposiciones sobre los egipcios, en que lugar quedan, etc.

De esta manera, en este ejemplo, queda bien expuesta las dos formas en que podemos recorrer la información que un Sistema de Hipermedia Física provee, por un lado el usuario navega virtualmente por la información provista, y por otro navega físicamente, caminando los links de un lugar a otro siguiendo los datos que el link físico le brinda.

#### b – Sistemas de Orientación para no Videntes.

Consideremos una persona no vidente que se dirige a un supermercado a realizar una compra de diversos productos que son de su necesidad. La persona llega y se registra en el Sistema de Hipermedia Física del supermercado conectando su PDA o celular al Sistema e ingresando en el mismo los productos que desea adquirir. Luego, comienza a recorrer el supermercado y cuando llega a la primer góndola, la persona recibe en su PDA información sobre el rubro de los productos que hay en la misma, y si hay en ella alguno de los productos que el cliente desea comprar. Sumado a esta información el cliente también puede consultar al sistema las ofertas que se presentan en dicha góndola o si recuerda algún producto que debe comprar que no ingreso cuando se registro puede consultarlo en cualquier momento y pedirle al sistema que le indique como llegar. El sistema ante esto puede indicarle al usuario, en función de la posición donde se encuentra el mismo, en que góndola esta el producto deseado y cuales son los pasos exactos que debe dar en uno u otro sentido para llegar al mismo.

#### c - Centro Comercial.

Un escenario simple, y de uso cotidiano, de aplicación puede ser un Centro Comercial, compuesto por un conjunto de locales y stand, al cuál concurren visitantes equipados con dispositivos móviles, para realizar compras y observar artículos que pueden ser de su interés.

Cuando alguno de los visitantes se detiene frente a una vidriera o stand de alguno de los locales, su dispositivo móvil recibe la información de los productos y ofertas que ofrece dicho local, y en función al producto que puede ser de su interés recibe información para conocer más detalles del mismo en la web (links digitales) o información invitándolo a visitar otro locales del mismo rubro (y como llegar hasta los

mismos), con los mismo productos, etc. (links físicos). De esta manera el usuario podrá ir recorriendo distintos locales o stand (Nodos físicos) en función de su interés y/o de lo que desea comprar, caminando el link físico al guiarse con la información provista por el mismo.

Este último ejemplo será el escenario de desarrollo donde se centrará nuestro trabajo, en el capítulo siguiente explicaremos detalladamente de que manera diseñaremos un Sistema de Hipermedia Física en un Centro Comercial, la funcionalidad que se proveerá y los elementos hipermediales, físicos y digitales que el mismo proveerá.

#### **4.2.5 Conclusiones de la aplicación de Hipermedia Física**

Hay varios puntos que son interesantes comentar, que pueden ser vistos como consecuencias o conclusiones al tratar de diseñar y/o implementar un sistema basado en Hipermedia Física:

- **Generación automática de información en relación al contexto:**  
La hipermedia aplicada en un mundo físico necesita Context-Awareness [51] debido a la necesidad que tiene el sistema de ir presentando la información en función de la ubicación del usuario y su relación con los objetos reales que integran el sistema de Hipermedia Física. Esto se puede lograr de forma automatizada poniendo sensores en todos los objetos físicos del sistema o mediante la interacción del usuario.
- **Ofrecimiento de Computación Social:**  
Un sistema o arquitectura de Hipermedia Física puede permitir que distintos grupos de usuarios dejen comentarios o sugerencias para otro grupo de usuarios que visitarán los mismos lugares y/u objetos pertenecientes al sistema, de manera de poder reutilizar y distribuir el conocimiento y la experiencia adquirida por cada uno de los grupos, a través del tiempo.
- **Soporte de un grupo heterogéneo de dispositivos móviles:**  
Al desarrollar aplicaciones que integren sistema de Hipermedia Física hay que tener en cuenta las tecnologías y técnicas existentes de manera de que dichas aplicaciones pueden ser utilizadas por mayor cantidad de distintos dispositivos móviles, lo que permitirá un mayor éxito y flexibilidad del sistema desarrollado teniendo en cuenta la gran cantidad de dispositivos que se presentan hoy en día en el mercado y que son adquiridos y utilizados por diversos usuarios.  
En este punto, es fundamental comprender, que el paradigma PH se basa en un sistema compuesto por aplicaciones que funcionan en dispositivos inalámbricos, lo que posibilitará el desplazamiento físico del usuario entre los distintos nodos que componen dicho sistema.
- **Transparencia en la arquitectura de Hipermedia Físico:**  
Al diseñar un sistema de Hipermedia Física, debemos poder identificar y definir, con claridad y sencillez, y de manera independiente a la implementación, cuales son los objetos de interés presentes en el dominio, junto

con las relaciones existentes entre los mismos, de manera que sea transparente para el usuario cuáles de estas relaciones debe “caminar” e implican un movimiento físico, y cuáles otras debe “browsear” para poder obtener la información que es de su interés.

A esto, hay que sumarle la importancia de que el diseño pueda hacer frente a la vertiginosa evolución y heterogeneidad que presenta la tecnología.

## Capítulo 5

### Descripción del Sistema del Centro Comercial

#### 5.1 Introducción

En las siguientes secciones explicaremos detalladamente el Sistema del Centro Comercial. Se presenta también el objetivo y alcance del mismo, junto con la funcionalidad que proveerá. Una vez terminada la descripción, en los capítulos siguientes realizaremos el diseño e implementación de un prototipo para mostrar parte de la funcionalidad descripta.

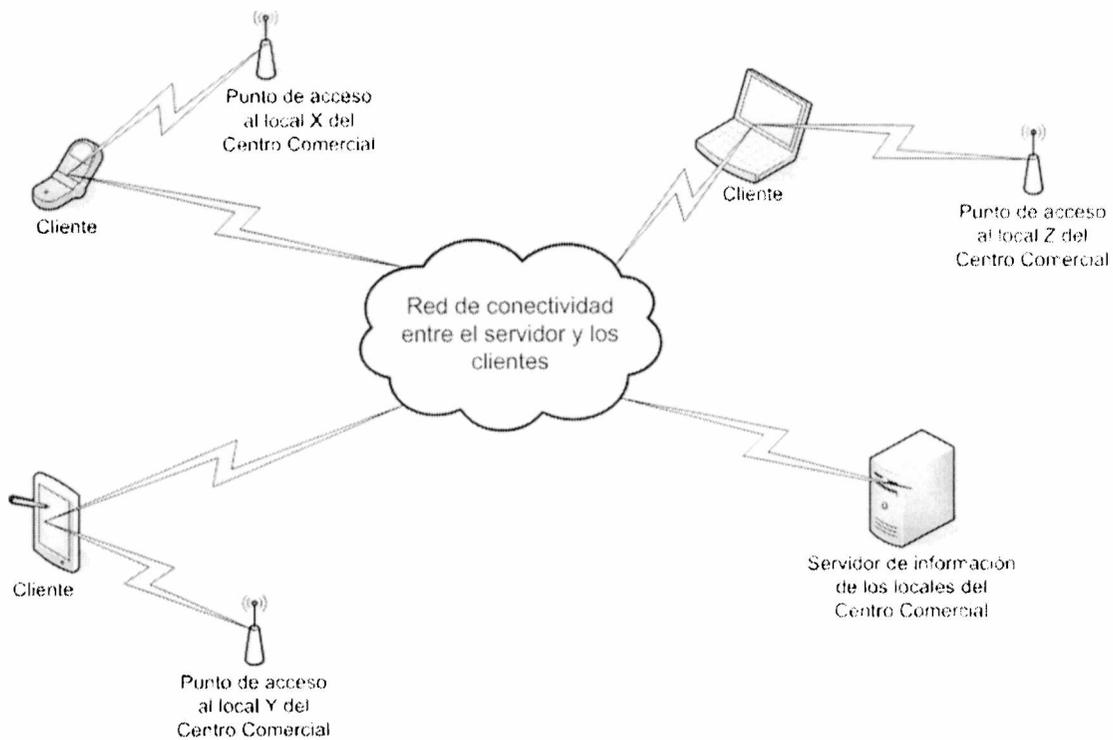
#### 5.2 Descripción y características generales del Sistema

El principal objetivo del sistema es proveer la adecuada adaptación de la información en función de la "posición relativa" (dentro del Centro Comercial) en la que se encuentra el usuario móvil (Sistema móvil) y al entorno en el que se encuentra (Sistema sensible al contexto) permitiendo la conexión física y/o lógica entre distintos puntos de interés del dominio, en nuestro caso los locales de un Centro Comercial, y brindando relaciones entre los mismos (Hipermedia Física [1]).

Teniendo en cuenta el escenario planteado nos encontramos con las siguientes componentes del sistema.

- **Locales del Centro Comercial:** son el conjunto de entidades que proveerán información al sistema acerca de los productos que desean comerciar.
- **Los clientes:** integran el conjunto de usuarios móviles que interactuarán con el sistema. Dichos clientes podrán acceder a los servicios brindados por el sistema, como por ejemplo realizar compras a través de sus dispositivos móviles (m-commerce [6]).
- **Un servidor** en el cual reside la información de los locales que componen el Centro Comercial junto con la información de los usuarios registrados.

A continuación, la **Figura 5.1** muestra las relaciones entre las componentes presentadas anteriormente y cómo los usuarios móviles del sistema (clientes) interactúan con los locales y el servidor de información.



**Figura 5.1: Arquitectura del sistema**

En la figura anterior, los puntos de acceso simbolizan a los sensores que identificarán a distintos locales del Centro Comercial.

### 5.3 Especificación funcional del sistema

A continuación presentamos la especificación funcionalidad del sistema, en donde detallamos la interacción entre las distintas entidades.

La comunicación de los clientes con el sistema estará compuesta por las siguientes etapas:

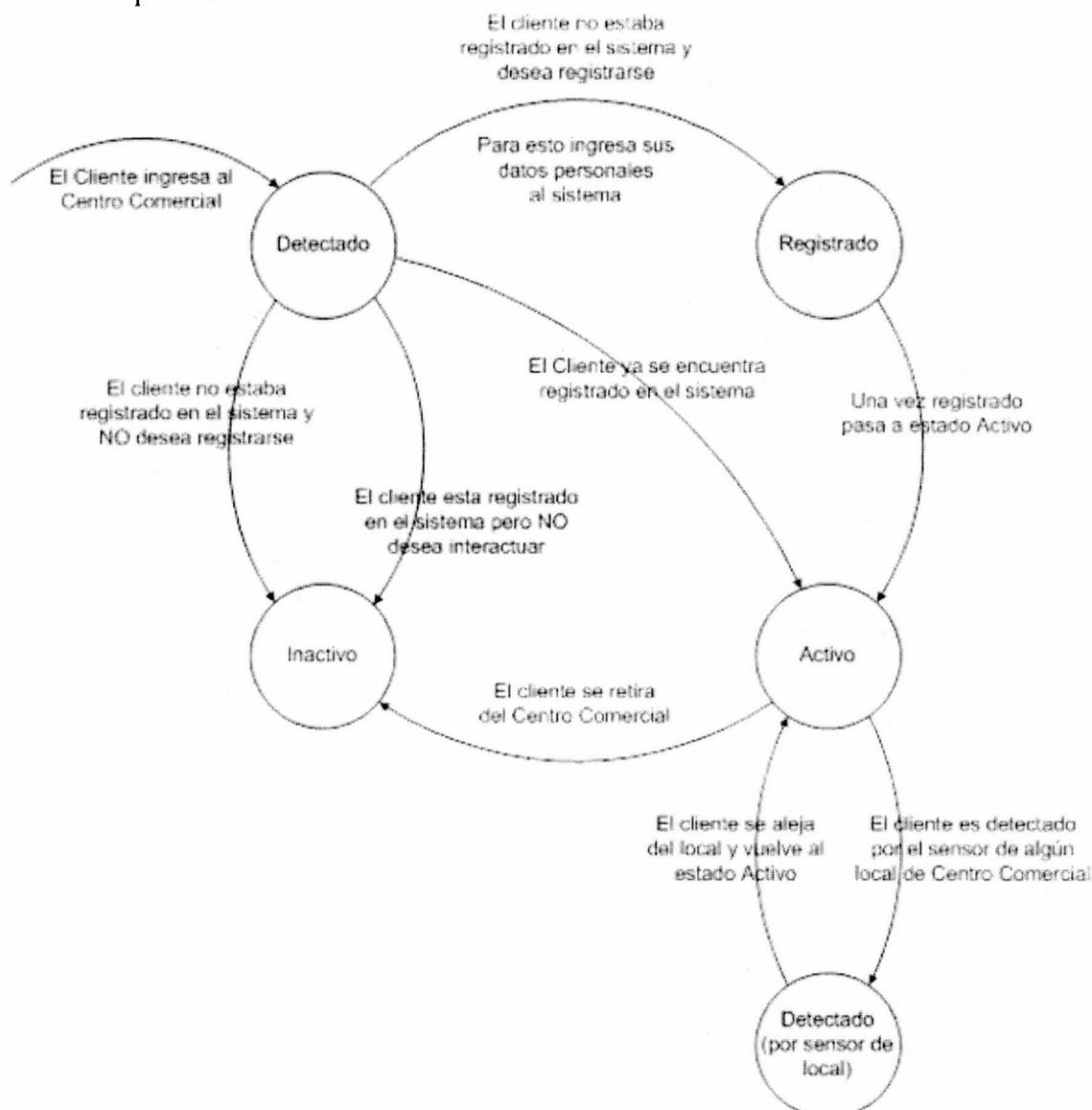
1. Una persona ingresa al Centro Comercial con su un dispositivo móvil. En ese instante el dispositivo es detectado por el servidor.
2. Una vez que ha ocurrido esto, el cliente es invitado a registrarse en el sistema del Centro Comercial. Dependiendo del interés de la persona por ingresar al sistema se llevaran a cabo algunas de las siguientes acciones:
  - i. Si el cliente no acepta la invitación, termina la interacción entre el sistema y el mismo.
  - ii. Si el cliente acepta la invitación, se instalará de manera transparente la aplicación en el dispositivo móvil perteneciente al mismo.
3. La aplicación cliente solicitará al usuario que registre sus datos en el servidor en caso de que no se encuentre registrado (si ya estuviese registrado los

pasos 2 y 3 no serán tenidos en cuenta). En este momento el usuario pasará a estar registrado en el sistema, y en estado Activo.

4. Una vez registrado el usuario y con la aplicación cliente ejecutándose en su dispositivo móvil, este comenzará a interactuar con el servidor a medida que avance por el Centro Comercial.
5. En el momento en que el usuario es detectado por el sensor de alguno de los locales del Centro Comercial, este le enviará la identificación del local (única) al dispositivo móvil, mediante un sensor equipado con una tecnología de comunicación inalámbrica.
6. Recibida la identificación del local, la aplicación cliente interactuará con el servidor solicitando el envío de la información del local con la identificación que acaba de recibir.
7. Cuando el cliente recibe la información, obtiene el listado de productos que el local ha provisto al servidor, pudiendo seleccionar aquellos que les sean de interés para su posterior compra. Además, el cliente también recibirá información sobre:
  - Productos de otros locales relacionados con los productos del local actual que son de su interés, y la forma de trasladarse hasta los locales que contienen los mismos.
  - La ubicación y la forma de llegar hasta distintos locales, que pertenezcan al Centro Comercial y estén relacionados con los productos que el usuario incluyo como de su interés, y que todavía no ha visitado.
  - Origen de Fabricación, materiales, red de comercialización etc., y diversos temas relacionados con el, o los productos, que el usuario se encuentra observando en ese mismo momento.
8. De esta manera, como describen los puntos anteriores, el cliente recorrerá el Centro Comercial, interactuando con el sistema, el cual le ira sugiriendo los distintos locales en los que podrá visualizar y/o seleccionar los productos que son de su agrado.
9. Cuando el cliente finalice su visita al Centro Comercial y decida retirarse del mismo podrá efectuar un “checkout” para realizar la compra real de sus productos. Dicha compra se puede realizar de las siguientes maneras:
  - a- Realizando la compra localmente en el Centro Comercial, esto es, acercarse a un espacio destinado para tal fin en donde se le presentaran los productos que selecciono y si le satisface su calidad efectuara el pago de los mismos (contado, tarjeta de débito, cheque, tickets, etc.).

- b- Realizando la compra con modalidad “envío a domicilio”, donde los productos que haya seleccionado el cliente les serán enviados al domicilio que ingreso el usuario al registrarse al sistema o a otro domicilio que el usuario desee, el cuál deberá ser previamente ingresado al sistema. El pago en esta modalidad de entrega se podrá realizar en forma de contrareembolso (pagar cuando se recibe el producto en el domicilio) o con tarjeta de crédito en ese mismo momento mediante la utilización de su dispositivo móvil (m-commerce [6]).

En la **Figura 5.2** se representa gráficamente las 8 etapas de la especificación funcional presentados anteriormente:



**Figura 5.2:** Diagrama de estados de un cliente

### 5.3.1 Servidor

En esta sección se describen las características del servidor junto con las funcionalidades que este proveerá a los clientes. También se presenta la “Consola de

Administración del Servidor” la cual es una herramienta que permite acceder a funcionalidades o servicios del servidor que no pueden ser accedidos desde las aplicaciones clientes.

### 5.3.2 Consola de Administración del Servidor

Esta herramienta provee las funcionalidades y características necesarias para realizar el monitoreo de todos los usuarios móviles registrados en el sistema, junto con las operaciones requeridas para la administración de los mismos, las cuales son detalladas a continuación:

1. **Interfaz para visualización de usuarios conectados:** El servidor recibe peticiones de los clientes conectados en el sistema. La información que se le envía en cada petición será el ID del local al que se encuentra próximo cada uno de los clientes. Dado que en función de dicho ID es posible determinar la posición relativa del usuario móvil dentro del Centro Comercial, los datos de este serán presentados en la interfase de la consola sobre un mapa del Centro Comercial.
2. **Alta de usuario móvil:** Permite realizar el alta de un nuevo usuario en el sistema. Algunos de los datos requeridos son: nombre, apellido, tipo de dispositivo móvil (celular, palm, etc.)
3. **Baja de usuario móvil:** Permite realizar la baja de un usuario del sistema. El servidor realizará las siguientes validaciones antes de permitir realizar la eliminación del usuario:
  - a. Que el usuario no se encuentre conectado al sistema en el momento de realizar la baja.
  - b. Y que el usuario no posea envíos de compra pendientes.En cualquiera de los dos casos anteriores el sistema informará de tal situación y denegará la petición de ejecución de la operación de eliminación.
4. **Modificación de usuario móvil:** Permite realizar modificaciones a los datos asociados a un usuario previamente registrado.
5. **Búsqueda de usuarios por algún criterio:** Esta función permite realizar la búsqueda de usuarios por diferentes criterios. Algunos criterios de búsqueda son: nombre o apellido del usuario, usuarios con mayor cantidad de compras, usuarios con menor cantidad de compras, usuarios que compraron un determinado producto o que visitaron un local específico, etc.
6. **Alta de Local:** Permite realizar el alta de un nuevo local en el sistema. Esta operación permite asociarle a un local los diferentes productos, ofertas, descuentos, promociones, etc. que este provea, junto con los datos típicos tales como: nombre, propietario, etc.
7. **Baja de Local:** Esta operación permite realizar la baja de un local del sistema. En este caso, es importante tener en cuenta que mientras el censor (dispositivo que detecta a los usuarios móviles en la proximidad de un local)

no sea “deshabilitado”, se les informará a los usuarios que se acerquen al local que este no posee información asociada en el servidor.

- 8. Modificación de Local:** Permite realizar la modificación de la información asociada a un local preexistente en el sistema, esto es, actualización de las ofertas, descuentos, altas y bajas de productos, etc.

### 5.3.4 Aplicación Cliente (Usuarios Móviles)

En esta sección se presentan las características y funcionalidades de la aplicación cliente. La misma será independiente del dispositivo móvil que utilice el usuario, y permitirá a estos interactuar con los servicios que provee el servidor. Una característica importante que debe proveer la aplicación cliente es la posibilidad de mostrar información gráfica del Centro Comercial. En otras palabras, además de mostrar la información de los locales comerciales también debe ser capaz de mostrar diferentes tipos de gráficos (planos del Centro Comercial) mostrando en ellos caminos desde la posición actual de una consulta hasta el punto de interés del usuario, por ejemplo la salida de emergencia mas cercana. Otra característica importante de la aplicación cliente es la implementación de funcionalidades para el soporte de sensibilidad al contexto del usuario. Esto será útil en el momento en que el usuario cambie de posición dentro de Centro Comercial dado que en función de esta posición el servidor enviará la información del local más próximo.

A continuación se detallan las funcionalidades que brindará la aplicación cliente:

- 1. Registro de Usuario en el Servidor:** Esta operación permite ingresar los datos que identifican al usuario, para luego ser enviados al servidor en donde serán registrados. De esta manera, el usuario ingresará también información acerca de los temas de su interés, los cuales permitirán realizar una asociación con los productos que desea adquirir y, en consecuencia, con los locales que posean los mismos. Estos temas estarán desglosados en un conjunto de ítems: Deportes, Salud, Belleza, Moda, Entretenimientos, Tecnología, etc., en función de los locales que componen el Centro Comercial.
- 2. Modificación de datos de Usuario:** Esta operación permite modificar los datos previamente registrados con la operación 1, con el fin de mantenerlos actualizados.
- 3. Envío de contexto del usuario al servidor:** Básicamente esta información consta de la posición del usuario y las características (resolución, capacidades, etc.) de su dispositivo móvil. Dichas características serán almacenadas en el servidor en el momento en que el usuario se registre la primera vez. De esta forma se ahorra espacio en el envío de datos desde la aplicación cliente hacia el servidor cada vez que el usuario cambia de posición.
- 4. Interfaz para visualización de información de un local:** Esta característica de la aplicación permite presentar la información (ofertas, descuentos, promociones, novedades, links, etc.) de los locales registrados en el sistema.

5. **Carrito Virtual de adquisición de productos:** Esta característica de la aplicación permite visualizar los productos que el usuario adquiere a medida que recorre el Centro Comercial.
6. **Alta de Producto en Carrito Virtual:** Esta operación permite incorporar un nuevo producto en el carrito del usuario.
7. **Baja de Producto en Carrito Virtual:** Esta operación permite realizar la eliminación de un producto que se encuentra en el carrito del usuario.
8. **Confirmación de adquisición de productos (checkout):** Esta operación le permite establecer al usuario la compra concreta de los productos previamente seleccionados. Para esto el usuario tendrá dos variantes de compra real:
  - a. **Efectivizar la compra en el Centro Comercial:** esto es, el usuario, una vez que finalice su recorrido del Centro Comercial, deberá presentarse en el "*Local de entrega de productos*", preestablecido por el Centro Comercial, en donde podrá visualizar y probar los productos seleccionados. Luego, si esta de acuerdo con sus calidades y características, podrá realizar la compra real mediante el pago (efectivo, tarjeta de crédito, etc.) de los mismos.
  - b. **Compra con modalidad "envío a domicilio":** este tipo de compra, le permite al usuario recibir los productos seleccionados en su domicilio y efectuar el pago en ese momento (contrareembolso). O bien el pago en esta modalidad se podrá realizar con tarjeta de crédito en el mismo momento en que se encarga el producto mediante la utilización de su dispositivo móvil (m-commerce [6]).
9. **Baja de Usuario:** Esta operación permite eliminar la registración del sistema de un usuario móvil. El servidor verificará que el usuario no posea compras pendientes, en cuyo caso se registrará una eliminación lógica del usuario, hasta tanto la compra no esté finalizada.
10. **Búsquedas desde la posición actual del usuario:** Esta operación permite realizar diferentes tipos de búsqueda desde la aplicación cliente:
  - a. **Búsqueda de un lugar particular:** Estos son, Sanitarios, Patio de Comidas, salidas de emergencia, Seguridad del Centro Comercial, Locales, etc.
  - b. **Búsqueda de un tipo de mercadería:** Con esta operación el usuario podrá ver el conjunto de locales que poseen algún tipo de mercadería de su interés.

Ambos tipos de búsquedas podrán visualizarse de dos formas diferentes:

1. **Gráficamente:** Mediante este tipo de visualización el usuario observará en la pantalla de su dispositivo móvil el plano del Centro Comercial con el/los caminos de la búsqueda realizada que representarán la información de los Link Físicos obtenidos (Physical Hipermedia [1]). Cabe aclarar que la resolución gráfica de los planos enviados al dispositivo móvil dependerá sensiblemente de la capacidad de visualización del dispositivo.
2. **Textualmente:** Mediante este tipo de visualización el usuario observará la información de la búsqueda realizada en la pantalla de su dispositivo móvil en formato textual. Dicha información incluirá los detalles de cómo llegar hasta el punto en el Centro Comercial al que se desea acceder.

#### **11. Información adicional de producto:**

Con esta operación podremos obtener información extra de un determinado producto tal como: origen, fabricación, garantía, ventajas y desventajas que hayan realizado otros usuarios que han adquirido dicho producto, etc.

Toda esta información adicional podrá ser accedida de manera muy simple para el usuario, mediante el acceso a distintos Links Digitales (Physical Hipermedia [1]) que se presentarán en la aplicación en función del producto seleccionado.

#### **12. Información de locales que comercializan un mismo producto:**

Esta operación permitirá obtener un conjunto de Link Físicos (Physical Hipermedia [1]) relacionados con un producto perteneciente a un local determinado. La información se presentará en forma gráfica o textual en un plano en el que se podrá observar el contenido de cada uno de los Link Físicos, los cuales contendrán los distintos caminos que un usuario puede seguir para llegar a otros locales que comercialicen el mismo producto.

#### **13. Registro de la ubicación del automóvil de un cliente dentro del estacionamiento del Centro Comercial:**

Esta operación es básicamente muy similar a la funcionalidad que permite que un usuario sea detectado en la cercanía de un local. Es decir, cuando un usuario registrado en el sistema, ingrese con su automóvil al parque de estacionamiento del Centro Comercial, el sistema registrará la ubicación exacta del automóvil (mediante dispositivos de detección de usuarios ubicados en cada uno de los sectores del estacionamiento). De esta forma el cliente no necesitaría recordar la posición en la cual estacionó su automóvil, ya que el sistema lo hace por él. Por último cuando el cliente desee retirarse del Centro Comercial y dirigirse hacia su automóvil para retirarse, lo único que debería hacer es solicitarle al sistema que le informe, mediante texto o gráficamente el camino a seguir desde su posición actual hasta el lugar de estacionamiento.

## 5.4 Alternativas de conectividad

En esta sección se presentan las diferentes alternativas tecnológicas para la conectividad entre los usuarios móviles, los locales y el servidor.

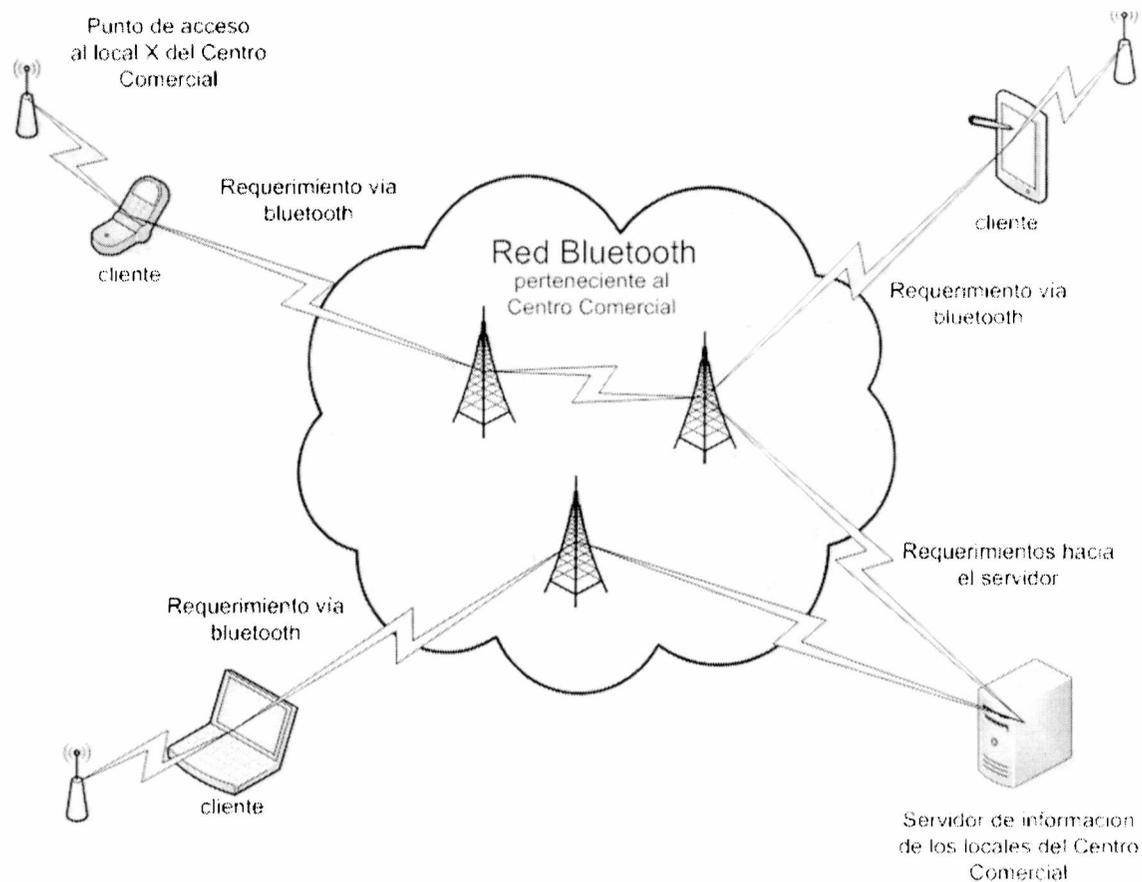
El sistema está compuesto por dos tipos de conectividades:

- *(1) Entre los dispositivos móviles y los locales del Centro Comercial:*  
Esta conexión se establecerá cada vez que un dispositivo móvil detecte algún sensor de los locales del Centro Comercial. Tal mecanismo de auto detección podría ser provisto por la tecnología bluetooth [25] la cual en su especificación indica la posibilidad de realizar este tipo de tareas.
- *(2) Entre los dispositivos móviles y el Servidor:*  
Esta clase de conexión se establecerá cuando algún dispositivo móvil solicite información al servidor, inmediatamente después de haber establecido una conexión del primer tipo.

### 5.4.1 Conectividad Cliente/Servidor vía Bluetooth

Bluetooth [25] proporciona una vía de interconexión inalámbrica entre diversos aparatos que tengan dentro de sí esta tecnología, como celulares, computadoras de mano (Palm, Pocket PC) , cámaras, computadoras portátiles, impresoras o cualquier otro tipo de dispositivos, usando por supuesto una conexión segura de radio de muy corto alcance. El alcance que logran tener estos dispositivos es de 10 metros. Para mejorar la comunicación es recomendable que nada físico (como una pared) se interponga.

Entonces una de las posibilidades, para resolver el problema de conectividad entre los usuarios móviles y el servidor, es el uso de esta tecnología. La **Figura 5.3** presenta un esquema de comunicación cliente/servidor mediante bluetooth.



**Figura 5.3: Conectividad cliente/servidor via bluetooth**

Los requisitos para proveer este tipo de conectividad son:

- **Arquitectura de Hardware:**

El hardware que compone el dispositivo Bluetooth esta compuesto por dos partes. Un dispositivo de radio, encargado de modular y transmitir la señal; y un controlador digital. El controlador digital esta compuesto por una CPU, por un procesador de señales digitales (DSP - Digital Signal Processor) llamado Link Controller (o controlador de Enlace) y de los interfaces con el dispositivo anfitrión.

El LC o Link Controller está encargado de hacer el procesamiento de la banda base y del manejo de los protocolos ARQ y FEC de capa física. Además, se encarga de las funciones de transferencia (tanto asíncrona como síncrona), codificación de Audio y encriptación de datos.

El CPU del dispositivo se encarga de atender las instrucciones relacionadas con Bluetooth del dispositivo anfitrión, para así simplificar su operación. Para ello, sobre el CPU corre un software denominado Link Manager que tiene la función de comunicarse con otros dispositivos por medio del protocolo LMP.

Entre las tareas realizadas por el LC y el Link Manager, destacan las siguientes: - Envío y Recepción de Datos. - Empaginamiento y Peticiones. - Determinación de Conexiones. - Autenticación. - Negociación y determinación de tipos de enlace. - Determinación del tipo

de cuerpo de cada paquete. - Ubicación del dispositivo en modo sniff o hold.

- ***Arquitectura de Software:***

Buscando ampliar la compatibilidad de los dispositivos Bluetooth, los nuevos dispositivos que se agregan al estándar utilizan como interfaz entre el dispositivo anfitrión (Pc, teléfono celular, etc.) y el dispositivo Bluetooth como tal (chip Bluetooth) una interfaz denominada HCI (Host Controller Interface).

Los protocolos son una forma consensuada en la que los dispositivos intercambian información. Para cada tipo de red incluida la especificación Bluetooth existe un conjunto de protocolos o reglas que definen exactamente como se pasan los mensajes por el enlace.

- ***Soporte para los enlaces cliente/servidor:***

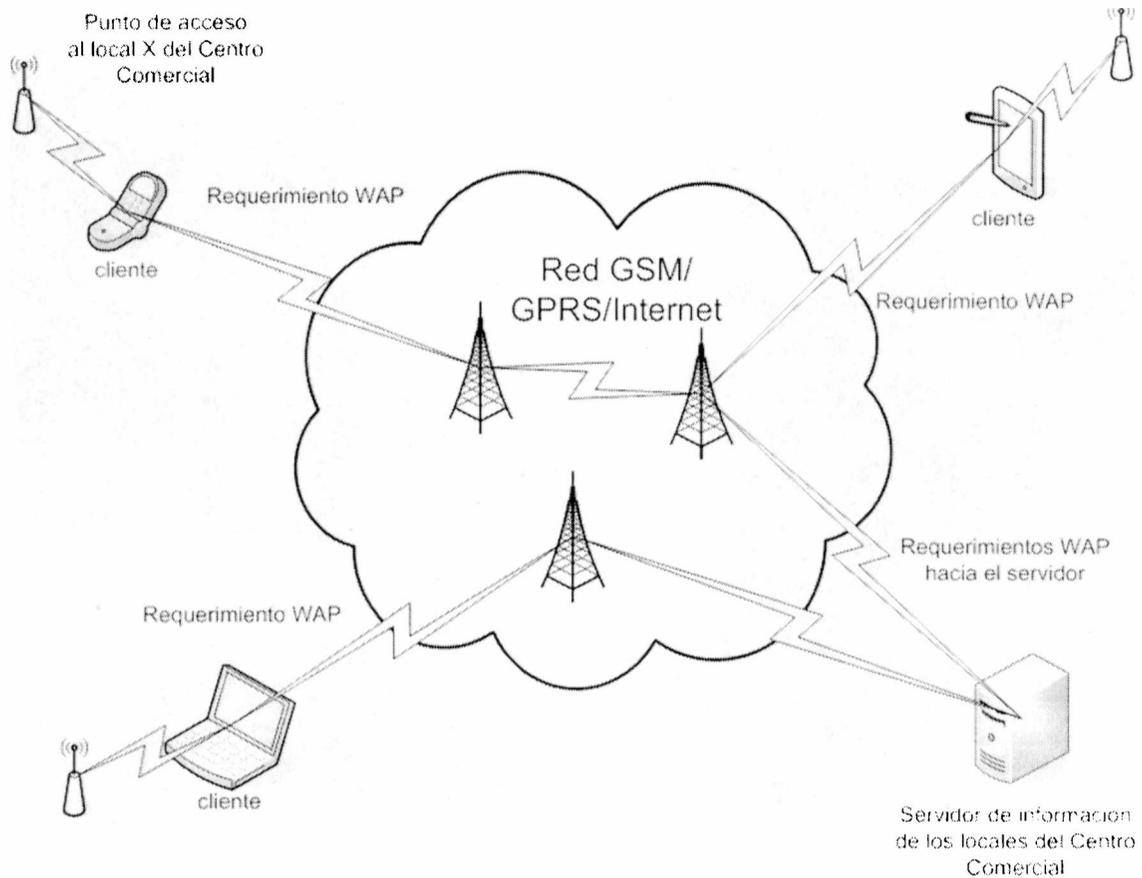
Para poder utilizar esta tecnología en el Centro Comercial sería necesario el armado de antenas, las cuales permitirían establecer comunicaciones entre los clientes y el servidor. La ventaja de utilizar esta tecnología radica en que el Centro Comercial tendría pleno control en la red de comunicación. En la sección siguiente se analizara el caso contrario, es decir, el control queda a cargo de otras empresas de comunicación.

#### **5.4.2 Conectividad Cliente/Servidor vía GPRS**

Otra de las posibilidades para resolver el problema de conectividad es el uso de la tecnología GPRS (General Packet Radio Service). GPRS es un servicio de transferencia de datos para dispositivos móviles disponible para usuarios bajo redes GSM.

A diferencia de la tecnología bluetooth, utilizando esta tecnología para establecer las comunicaciones, el Centro Comercial puede delegar el mantenimiento de la red de comunicación a las empresas propietarias de las redes. Esto se debe a que se podría hacer uso de las ya conocidas redes GSM (CTI [71], Movistar [72], Personal [73], etc.) existentes en el mercado.

La **Figura 5.4** muestra un esquema de comunicación cliente/servidor vía GPRS.



**Figura 5.4: Conectividad cliente/servidor vía GPRS**

Los requisitos para proveer este tipo de conectividad son:

- **Existencia de red GSM:**

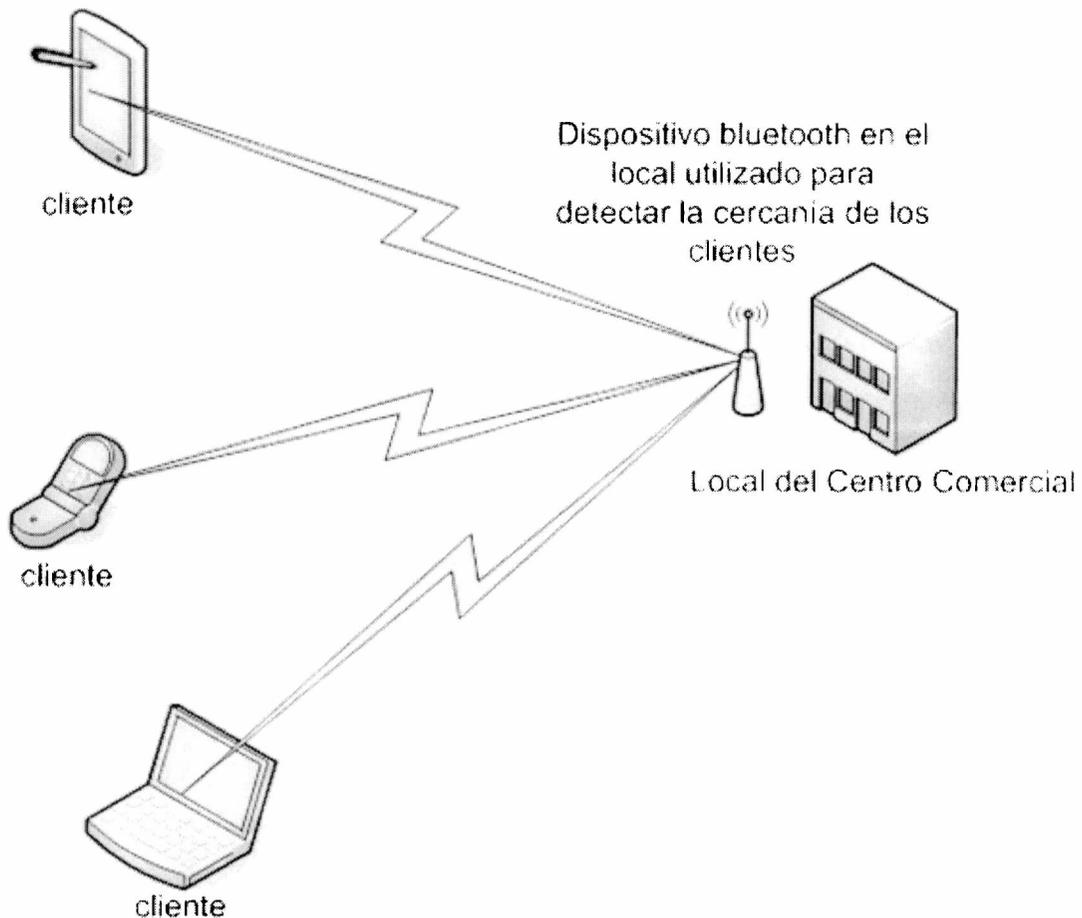
Para poder utilizar la tecnología GPRS, es necesaria la existencia de una red GSM. No necesariamente el Centro Comercial debería armar una nueva red GSM, es decir, podría hacer uso de redes existentes pudiendo de esta manera independizarse del mantenimiento de la misma.

### 5.4.3 Conectividad Cliente/Locales

Para establecer la comunicación entre la aplicación cliente y los locales se podría hacer uso nuevamente de la tecnología bluetooth.

Basándonos en la definición de Bluetooth este nos provee un método adecuado para que los dispositivos móviles detecten automáticamente la cercanía de los locales o la inversa, es decir, que los locales detecten la cercanía de los clientes. Esta variabilidad de roles es un parámetro que se podría configurar en el momento de montar el sistema.

La **Figura 5.5** muestra en esquema de comunicación entre los clientes y los locales.



**Figura 5.5: Conectividad clientes/locales vía Bluetooth**

Habiendo mostrado las diferentes tecnologías para establecer la comunicación entre los clientes y el servidor, y entre los clientes y los locales, ahora podemos establecer básicamente como sería el flujo de la información:

1. Un usuario entra al sistema.
2. La aplicación cliente del usuario detecta la cercanía de un local.
3. El dispositivo bluetooth del local le envía a la aplicación cliente una identificación del mismo.
4. La aplicación cliente, habiendo recibido la identificación, solicita al servidor información sobre el local con dicha identificación (vía bluetooth o GPRS).
5. El servidor procesa el requerimiento y envía la información del local al dispositivo móvil que solicitó la información.
6. Vuelve a 2 o pasa a 7.
7. El usuario se retira del Centro Comercial.

# Capítulo 6

## Diseño

### 6.1 Introducción

En este capítulo se presenta el modelo realizado en función de las características del sistema descritas en la sección anterior. Este modelo involucra las abstracciones de los Usuario, Dispositivos, Tipo de Dispositivos, Consola del Servidor, Locales, etc. Cada una de las mismas es representada con un diagrama bajo la estandarización UML [55].

### 6.2 Descripción del modelo

A continuación en esta sección son definidos cada uno de los modelos que representan a los diferentes módulos que componen el sistema descrito en el capítulo anterior. Primeramente se hace una presentación del modelo de acciones del servidor, luego de los usuarios móviles y los dispositivos, a continuación el módulo de compras, luego el de locales y productos y para finalizar es presentado y analizado en profundidad el diseño de hipermedia física.

#### 6.2.1 Descripción de los módulos del Servidor

En esta sección presentamos las clases que modelan en su conjunto al Servidor del sistema. En la siguiente **Figura 6.1** se muestran las clases más relevantes.

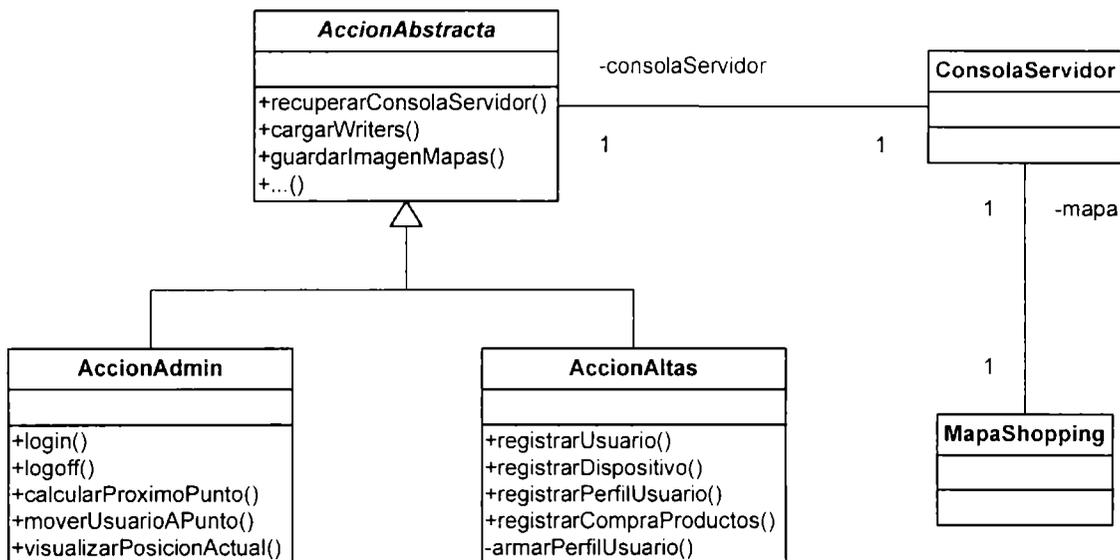
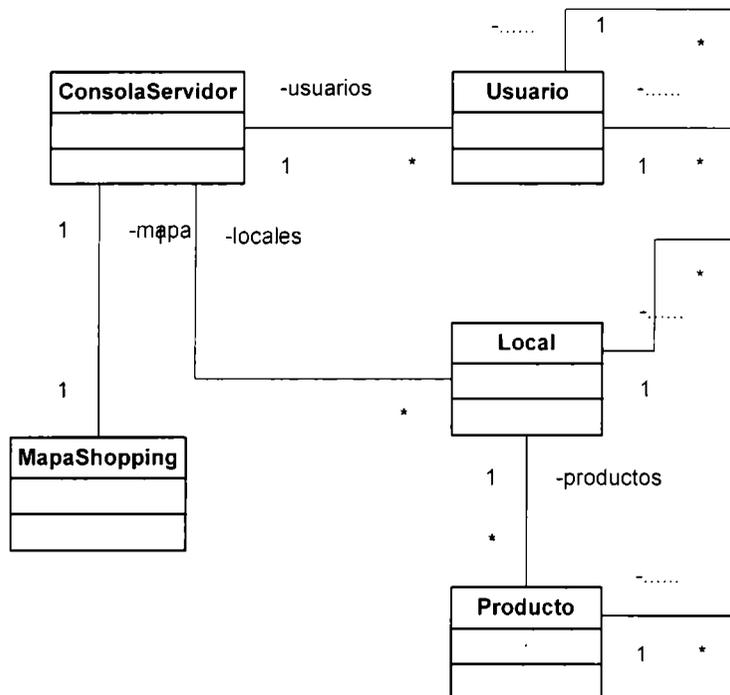


Figura 6.1: Diagrama de clases – Módulo de acciones del servidor

Las acciones son las responsables de atender las peticiones de los clientes, algunas peticiones válidas son: login, logout, registrar un nuevo usuario, registrar un dispositivo para un usuario, etc.

Por otro lado las acciones tienen conocimiento de una instancia de la clase `ConsolaServidor` la cual es la encargada de administrar todo lo referido a la administración de los datos y a la visualización de los mismos, delegando esta última tarea en la clase `ConsolaServidorUI` (la misma no se presenta en esta figura pues representa todo lo relacionado con la funcionalidad visual de la `ConsolaServidor`).

La siguiente **Figura 6.2** muestra más en detalle las relaciones de la clase `ConsolaServidor`.



**Figura 6.2: Diagrama de clases – Relaciones de la `ConsolaServidor`**

Queda claro que el diagrama anterior posee más relaciones que no se están mostrando, esto es así ya que la idea es bosquejar las relaciones que posee la consola con las demás clases del sistema y como desde ella se puede acceder a toda la información. En las siguientes secciones se presentarán los diagramas en un nivel más detallado.

## 6.2.2 Descripción de los modelos de Usuarios Móviles y Dispositivos

En esta sección describiremos el modelo de Usuarios junto con el de Dispositivos Móviles. A continuación se presenta el diagrama de clases de dichos modelos.

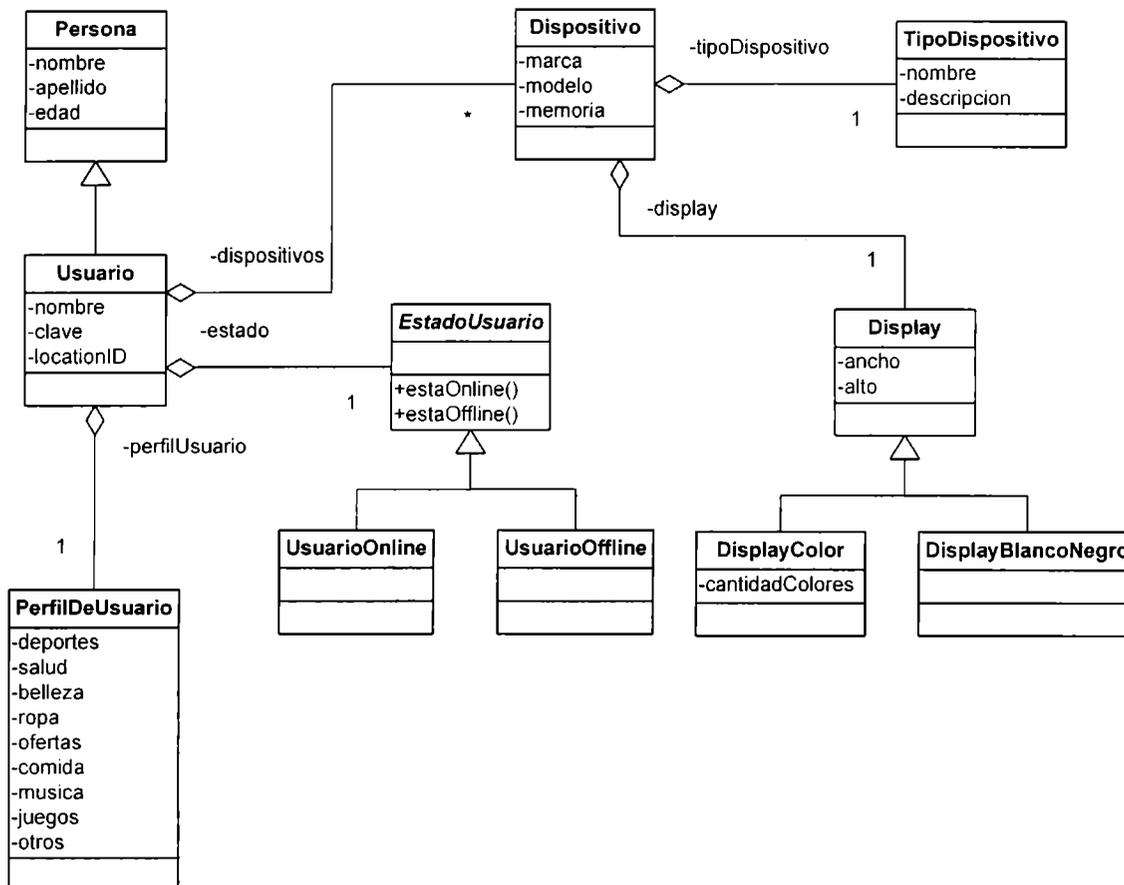


Figura 6.3: Diagrama de clases – Módulo de Usuarios y Dispositivos Móviles

- **Persona:** La clase “Persona” permite representar los datos básicos de las personas que se registran en el sistema.
- **Usuario:** Esta clase nos permite modelar la abstracción de un usuario conectado al sistema. Posee relaciones con la clase “Dispositivo” (representa a los dispositivos móviles de un usuario), “EstadoUsuario” permite determinar el estado de un usuario (conectado o desconectado), y “PerfilUsuario” representa los temas que son de interés para un usuario (deportes, salud, etc.).
- **Dispositivo:** Tal como se dijo en el punto anterior, esta clase modela los dispositivos móviles de los usuarios.
- **EstadoUsuario, UsuarioOnline y UsuarioOffline:** Estas clases permiten determinar el estado de un usuario dentro del sistema. En este punto hacemos hincapié en que fue usado el patrón de diseño “State”, (para más información

referirse a [69]), el cual en su propósito específica que permite a un objeto alterar su comportamiento según el estado interno en que se encuentre.

- **TipoDispositivo:** Esta clase modela los diferentes tipos de dispositivos móviles, por ejemplo: teléfonos celulares, palm's/PDA's, etc.
- **Display, DisplayColor y DisplayBlancoNegro:** Este conjunto de clases abstraen el concepto de los dos tipos de display que puede tener un dispositivo móvil, es decir, un display color o uno blanco y negro.

### 6.2.3 Descripción del modelo de compras

La Figura 6.4 muestra el diagrama de clases del módulo de compras.

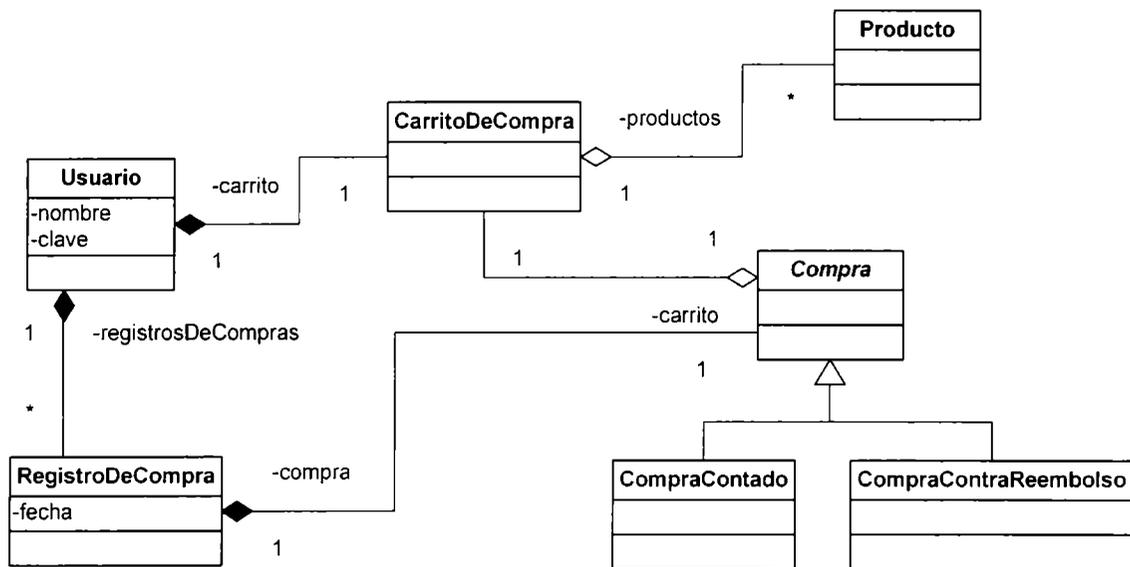


Figura 6.4: Diagrama de clases – Módulo Compras

- **RegistroDeCompra:** Esta clase permite registrar la fecha en que es efectuada una compra por un cliente. Por otro lado tiene conocimiento a la compra en sí misma, la cual es descripta a continuación.
- **Compra, CompraContado y CompraContraReembolso:** Estas clases abstraen el concepto de compra con sus diferentes tipos que pueden tener: contado o contrareembolso, básicamente poseen la información de los productos que compro un usuario en una fecha determinada. El conocimiento a los productos es por medio del carrito de compra virtual que el usuario va “llenando” a medida que camina por el centro comercial.
- **CarritoDeCompra:** Esta clase simula el comportamiento de un carro de compras, como podría ser el de un supermercado, es decir, permite guardar productos dentro de él, para luego ser abonados y retirados.

## 6.2.4 Descripción del modelo de locales y productos

El diagrama de clases presentado a continuación permite administrar toda la información concerniente a los locales del centro comercial.

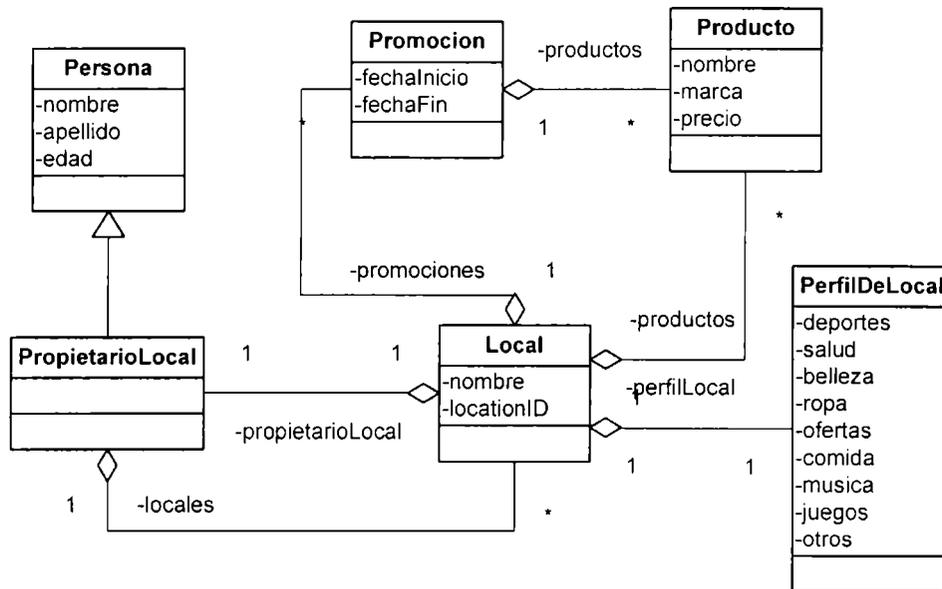


Figura 6.5: Diagrama de clases – Módulo Locales y Productos

- **Propietario:** Esta clase permite representar los datos de un propietario de algún local del centro comercial.
- **Local:** Esta clase permite representar la información de un local del centro comercial. Estos datos son: nombre, propietario del local, perfil del local, productos y promociones.
- **Promoción:** Representa las promociones que pueden llegar a ofertar los locales del centro comercial, para cada promoción se tiene: una fecha de inicio y una fecha de fin, más todos los productos asociados a esa promoción.
- **PerfilDeLocal:** Representa los diferentes tipos de productos que ofrece un local (deportes, salud, etc.).

## 6.2.5 Diseño de Hipermedia Física

En el siguiente diagrama presentamos el modelo que diseñamos para representar los conceptos de Hipermedia Física mencionados en el Capítulo 4:

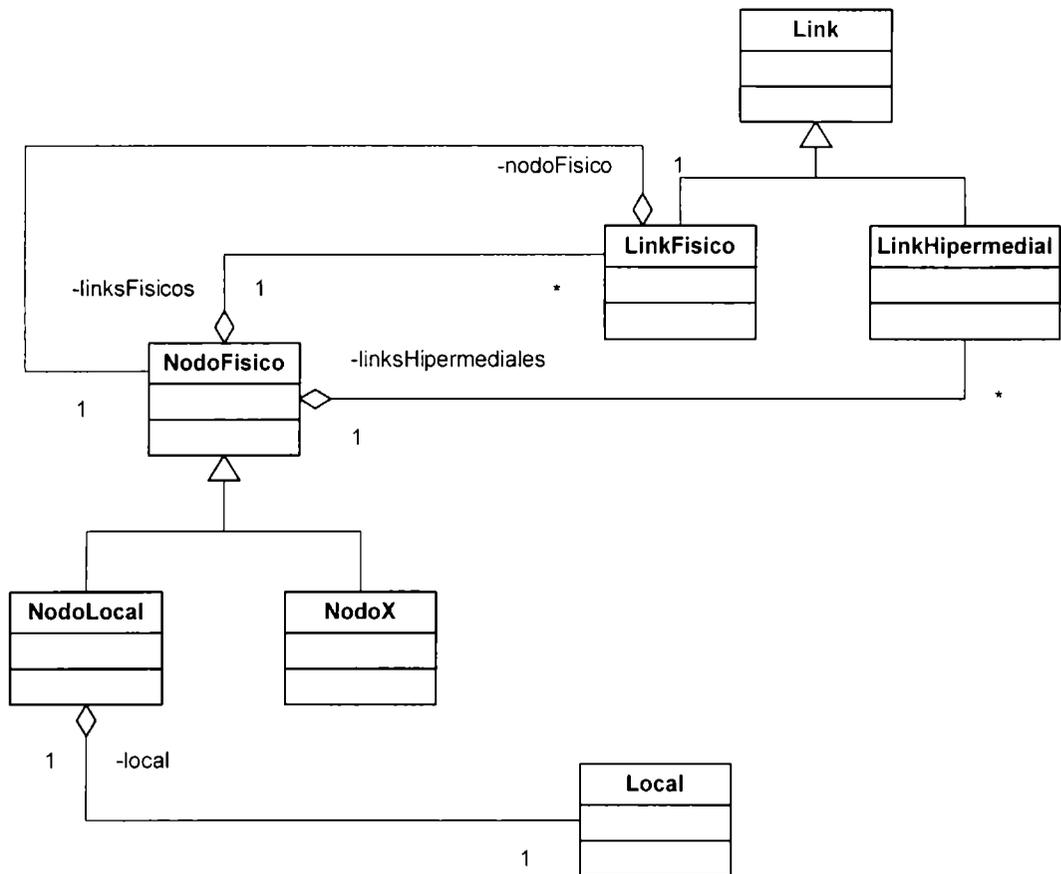


Figura 6.7: Diagrama de clases - Módulo de Hipermedia Física

### 6.2.5.1 Modelo de Nodo Físico

La clase **NodoFísico** representa el concepto definido anteriormente de un nodo físico, el cuál describe la abstracción de una entidad con características geográficas (locales, baños, mesas de ayuda, etc.). Dicha clase, esta compuesta por un conjunto de links físicos e hipermediales, que simbolizan la relaciones entre los nodos. Estas relaciones pueden ser de los siguientes tipos:

- Relación de Adyacencia: Los locales que se encuentren en forma contigua poseerán este tipo de relación.
- Relación de Cercanía: Este tipo de relación es aquella que agrupa un conjunto de nodos que se encuentran físicamente próximos en un radio determinado.
- Relación de Venta de Producto: Esta relación agrupa a todos los locales del Centro Comercial que realicen la venta de un mismo producto.

Esta clase posee un conjunto de subclases las cuales representan a las diferentes entidades que se encuentran en un Centro Comercial: Locales, Baños, etc.

### 6.2.5.2 Modelo de Link Físico

La clase LinkFísico presentada en la **Figura 6.7**, abstrae el concepto de las diferentes relaciones (adyacencia, cercanía, etc.) entre los Nodos Físicos. Un Nodo Físico (que representaría el origen de la relación) posee una colección de Links Físicos y cada uno de estos conoce a los Nodos Físicos (representando el destino de la relación) que se relacionan con el primer nodo. En otras palabras, un Link Físico representa una vía de comunicación entre dos nodos.

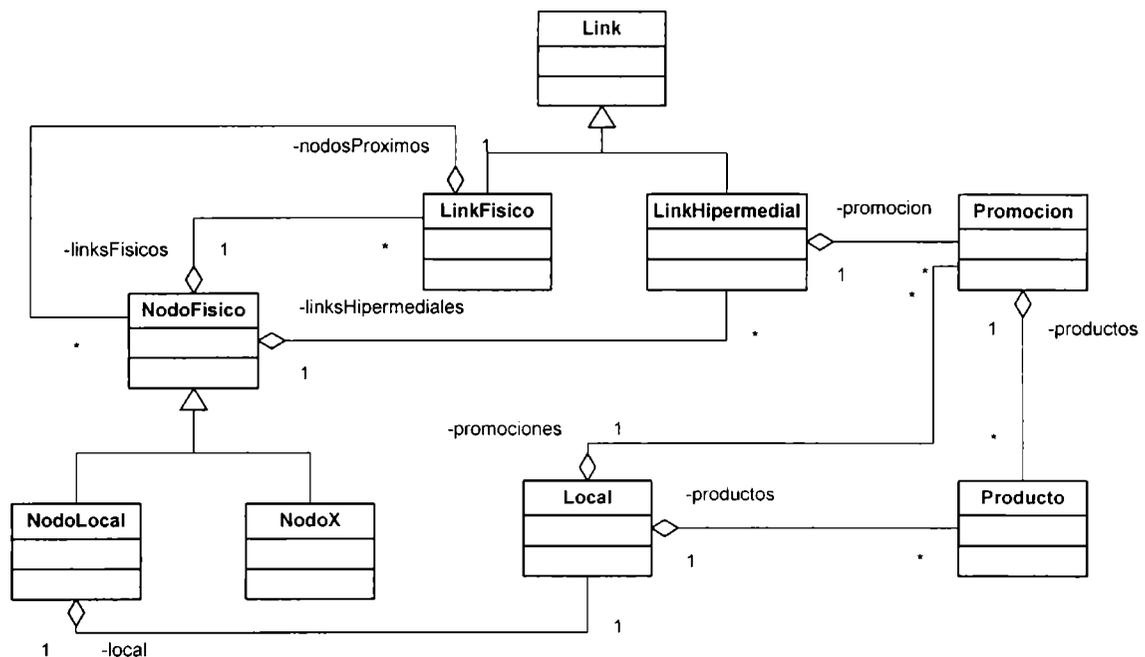
De esta manera se genera el grafo de relaciones que modela la comunicación entre los nodos que se encuentran en el Centro Comercial.

### 6.2.5.3 Modelo de Link Hipermedial

La clase LinkHipermedial presentada en la **Figura 6.7**, abstrae el concepto de información adicional que pueden poseer los Nodos Físicos (información de fabricación de productos, comentarios y sugerencias de otros usuarios, links a sitios Web relacionados, etc.)

### 6.2.6 Relaciones entre el modelo de Hipermedia Física y el modelo del dominio

La **Figura 6.8** muestra las relaciones existentes entre las clases que modelan el concepto de Hipermedia Física y las clases del dominio presentadas en las secciones anteriores.



**Figura 6.8: Diagrama de clases – Relaciones entre Hipermedia Física y las Clases del dominio**

# Capítulo 7

## Implementación

### 7.1 Introducción

En este capítulo se explica detalladamente la implementación del prototipo que se ha realizado. Un prototipo es una versión reducida en funcionalidad de un sistema, es por esto que en la primer sección son definidos el objetivo y el alcance del prototipo, y luego en las siguientes secciones se presenta la implementación de cada módulo del mismo junto con las tecnologías utilizadas para su implementación.

### 7.2 Objetivo y alcance del prototipo

El objetivo del prototipo implementado es mostrar algunas de las funcionalidades relacionadas con la Hipermedia Física y el comercio móvil.

El alcance del prototipo ha sido reducido sustancialmente con respecto al sistema real, a causa de las limitaciones en el tiempo de finalización de este trabajo.

#### 7.2.1 Funcionalidad provista en el Servidor

- **Visualización de usuarios conectados en mapa:**  
Esta funcionalidad permite visualizar a los usuarios conectados al sistema en el mapa del centro comercial.
- **Historial de camino de usuario:**  
Esta función permite visualizar sobre el mapa el camino que ha sido realizado por un determinado usuario desde el momento que ingreso al centro comercial.
- **Visualización de layers de información:**  
Los layers son las diferentes capas de información que pueden ser mostradas en el mapa del centro comercial. Tales capas de información son:
  - Puntos de entrada al centro comercial.
  - Locales del centro comercial.
  - Puntos de información, son aquellas zonas en donde se puede encontrar información sobre el centro comercial.
  - Caminos entre los locales.
  - Salidas de emergencia.
  - Seguridad, zonas en donde se encuentra el personal de seguridad.
  - Servicios (baños, etc.)
  - Teléfonos públicos.

- **Herramientas para visualización del mapa:**  
Para lograr una correcta visualización de las capas de información del mapa el servidor provee las siguientes herramientas de manipulación gráfica:
  - Zoom In.
  - Zoom out.
  - Zoom del layer activo.
  - Pan – Mover (Norte, Sur, Este, Oeste).
  - Selección y visualización de información de componentes del mapa (locales, entradas, puntos de información, etc.).
  - Funcionalidad para cambiar las propiedades de los layers (colores, fuentes, tamaños, etc.).
- **Búsqueda de caminos:**  
Una de las funcionalidades más importantes para mostrar en el prototipo es la posibilidad de realizar búsquedas entre los locales y desde los dispositivos móviles que se encuentren conectados al sistema.
- **Visualización de locales y productos registrados en el sistema**  
Esta funcionalidad permite visualizar todos los datos registrados en el sistema de los clientes junto con sus dispositivos móviles y la información de los locales y los productos que estos comercializan.

### 7.2.2 Funcionalidad provista en el Cliente

- **Registro de usuario:**  
Esta operación permite, desde la aplicación cliente, registrar a un nuevo usuario móvil en el sistema.
- **Login de usuario:**  
Esta operación permite registrar la entrada al centro comercial de un usuario previamente registrado en el sistema.
- **Visualización de información de locales:**  
Con esta funcionalidad se posibilita la visualización en el dispositivo móvil de los productos en venta o en oferta de un local.
- **Compra virtual de productos:**  
Esta operación permite, dado un conjunto de productos en venta de un local, seleccionar aquellos productos que son de interés del usuario y agregarlos a su carrito de compras virtual.
- **Logout de usuario:**  
Esta operación le permite a un usuario salir del sistema del centro comercial.

## 7.3 Implementación del Servidor

El servidor del sistema es una de las componentes más importantes dado que sobre él se realiza la mayor cantidad de procesamiento. Mas adelante veremos que en la implementación del Cliente solo se encuentra la lógica de visualización de los datos, mientras que todo el procesamiento requerido por las peticiones de los clientes es implementado en el Servidor.

### 7.3.1 Herramientas utilizadas para la implementación

En esta sección se presentan los frameworks y tecnologías que fueron utilizadas para realizar la implementación del servidor:

- **Lenguaje de programación:**

El lenguaje seleccionado para realizar la implementación es Java [56]. Esto se debe a varios factores:

- **Orientado a objetos:** Java fue diseñado como un lenguaje orientado a objetos desde el principio. Los objetos agrupan en estructuras encapsuladas tanto sus datos como los métodos (o funciones) que manipulan esos datos. La tendencia del futuro, a la que Java se suma, apunta hacia la programación orientada a objetos, especialmente en entornos cada vez más complejos y basados en red.
- **Distribuido:** Java proporciona una colección de clases para su uso en aplicaciones de red, que permiten abrir sockets y establecer y aceptar conexiones con servidores o clientes remotos, facilitando así la creación de aplicaciones distribuidas.
- **Interpretado y compilado:** Java es compilado, en la medida en que su código fuente se transforma en una especie de código máquina, los bytecodes, semejantes a las instrucciones de ensamblador. Por otra parte, es interpretado, ya que los bytecodes se pueden ejecutar directamente sobre cualquier máquina a la cual se hayan portado el intérprete y el sistema de ejecución en tiempo real (runtime).
- **Robusto:** Java fue diseñado para crear software altamente fiable. Para ello proporciona numerosas comprobaciones en compilación y en tiempo de ejecución. Sus características de memoria liberan a los programadores de una familia entera de errores (la aritmética de punteros), ya que se ha prescindido por completo los punteros, y la recolección de basura elimina la necesidad de liberación explícita de memoria.
- **Seguro:** Dada la naturaleza distribuida de Java, donde los applets se bajan desde cualquier punto de la Red, la seguridad se impuso como una necesidad de vital importancia. A nadie le gustaría ejecutar en su ordenador programas con acceso total a su sistema, procedentes de fuentes desconocidas. De esta formase implementaron barreras de seguridad en el lenguaje y en el sistema de ejecución en tiempo real.
- **No sensible a la arquitectura:** Java está diseñado para soportar aplicaciones que serán ejecutadas en los más variados entornos de red, desde Unix a Windows XP, pasando por Mac y estaciones de trabajo, sobre arquitecturas distintas y con sistemas operativos

diversos. Para transformar requisitos de ejecución tan variados, el compilador de Java genera bytecodes: un formato intermedio indiferente a la arquitectura diseñada para transportar el código eficientemente a múltiples plataformas hardware y software. El resto de problemas los soluciona el intérprete de Java.

- **Portable:** La no sensibilidad a la arquitectura representa sólo una parte de su portabilidad. Además, Java especifica los tamaños de sus tipos de datos básicos y el comportamiento de sus operadores aritméticos, de manera que los programas son iguales en todas las plataformas.

Estas dos últimas características se conocen como la *Máquina Virtual Java* (JVM).

- **Multi-Threading:** Hoy en día las aplicaciones que solo pueden ejecutar una acción a vez se totalmente limitadas. Java soporta sincronización de múltiples hilos (threads) de ejecución (*multithreading*) a nivel de lenguaje, especialmente útiles en la creación de aplicaciones de red distribuidas. Así, mientras un hilo se encarga de la comunicación, otro puede interactuar con el usuario mientras otro presenta una animación en pantalla y otro realiza cálculos.
- **Dinámico:** El lenguaje Java y su sistema de ejecución en tiempo real son dinámicos en la fase de enlazado. Las clases sólo se enlazan a medida que son necesitadas. Se pueden enlazar nuevos módulos de código bajo demanda, procedente de fuentes muy variadas, incluso desde la Red.

- **Entorno de desarrollo:**

El entorno de desarrollo seleccionado para realizar la implementación es Eclipse. Eclipse es un proyecto de desarrollo de software *Open Source* dedicado a obtener un producto de alta calidad para el desarrollo de herramientas integradas.

La elección de este ambiente de desarrollo se debió a su practicidad para agregar nueva funcionalidad al entorno mediante la instalación de plugins.

- **ESRI – MapObjects Edición Java:**

MapObjects Java Edition [57] es un conjunto de componentes desarrollados en su totalidad en Java, que permite a los desarrolladores construir aplicaciones GIS multiplataforma que satisfagan las necesidades concretas de cada usuario.

Estos componentes, se agrupan en librerías, y son los mismos componentes que se han utilizado para desarrollar las aplicaciones cliente de ArcIMS [58] (Author, Administrator, Designer, Java Viewer y ArcExplorer Java Edition). Para más información dirigirse al Apéndice C.

- **JFC (Java Foundation Classes) – Swing:**

JFC es la abreviatura de Java Foundation Classes, que comprende un grupo de características para ayudar a construir interfaces gráficos de usuario (GUIs). *Swing* es la evolución de AWT y proporciona un

conjunto completo de Componentes, todos ellos *lightweight*, es decir, ya no se usan componentes "peer" dependientes del sistema operativo tal como lo hacia AWT.

En nuestro prototipo hemos utilizado las componentes Swing para implementar todo lo concerniente a los aspectos visuales del servidor.

- **Arquitectura Cliente-Servidor:**

La arquitectura cliente-servidor también llamado modelo cliente-servidor o servidor-cliente es una forma de dividir y especializar programas y equipos de cómputo a fin de que la tarea que cada uno de ellos realiza se efectúe con la mayor eficiencia, y permita simplificar las actualizaciones y mantenimiento del sistema.

En esta arquitectura la capacidad de proceso está repartida entre el servidor y los clientes.

En la funcionalidad de un programa distribuido se pueden distinguir 3 capas o niveles:

1. Manejador de Base de Datos (Nivel de almacenamiento),
2. Procesador de aplicaciones o reglas del negocio (Nivel lógico) e
3. Interface del usuario (Nivel de presentación)

En una arquitectura monolítica no hay distribución; los tres niveles tienen lugar en el mismo equipo. En un comienzo, los mainframes concentraban la funcionalidad de almacenamiento (#1) y lógica (#2) y a ellos se conectaban terminales tontas, posiblemente ubicadas en sitios remotos. En el modelo cliente-servidor, en cambio, el trabajo se reparte entre dos ordenadores. De acuerdo con la distribución de la lógica de la aplicación hay dos posibilidades:

1. Cliente delgado: si el cliente solo se hace cargo de la presentación.
2. Cliente pesado: si el cliente asume también la lógica del negocio.

En la actualidad se suele hablar de arquitectura de tres niveles, donde la capa de almacenamiento y la de aplicación se ubican en (al menos) dos servidores diferentes, conocidos como servidores de datos y servidores de aplicaciones.

En la **Figura 7.1**, presentada a continuación, se ilustra una arquitectura Cliente/Servidor con la utilización de Clientes Móviles, reflejando la interacción en dicha arquitectura mediante la utilización de distintas tecnologías de red.

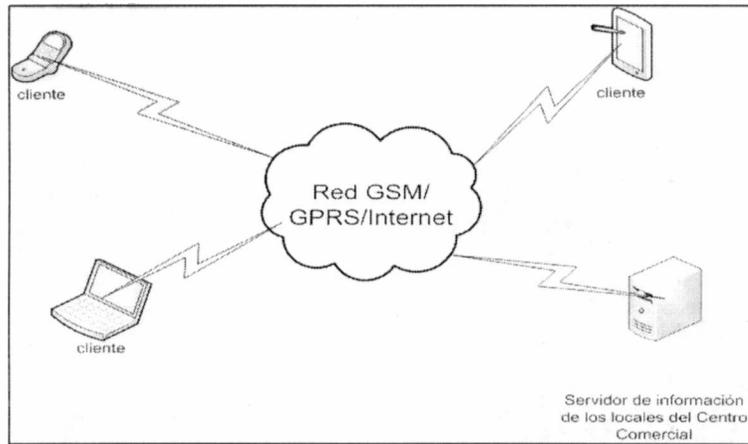


Figura 7.1: Arquitectura Cliente/Servidor con clientes móviles.

- **Servidor de Aplicaciones Tomcat de Apache:**

Tomcat es un contenedor de Servlets con un entorno JSP. Un contenedor de Servlets es un shell de ejecución que maneja e invoca servlets por cuenta del usuario.

En la implementación no fue utilizado un entorno JSP, dado que los clientes del sistema son los usuarios móviles implementados mediante la tecnología J2ME. Pero la posibilidad de realizar peticiones vía el protocolo HTTP [70] fue un punto a favor para determinar la elección de un servidor que supiera responder a peticiones HTTP tal como lo permite el servidor Tomcat.

### 7.3.2 Implementación de la Consola del Servidor

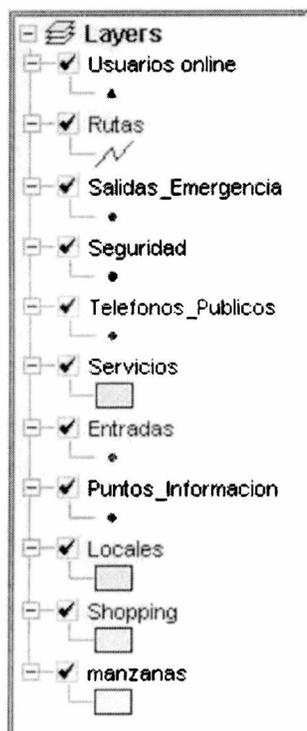
En esta sección se presentarán las pantallas que conforman a la consola del servidor. Explicando para cada una de ellas su funcionalidad dentro del sistema.



Figura 7.2: Consola del Servidor

En la **Figura 7.2** se presenta la ventana de la Consola del servidor, en esta se pueden apreciar cinco áreas bien definidas. A continuación se detallarán cada una de estas áreas:

### 1. Selección de Layers:



**Figura 7.3:** Selección Layers

Esta estructura en forma de árbol posee información de cada uno de los layers que componen el mapa del Centro Comercial.

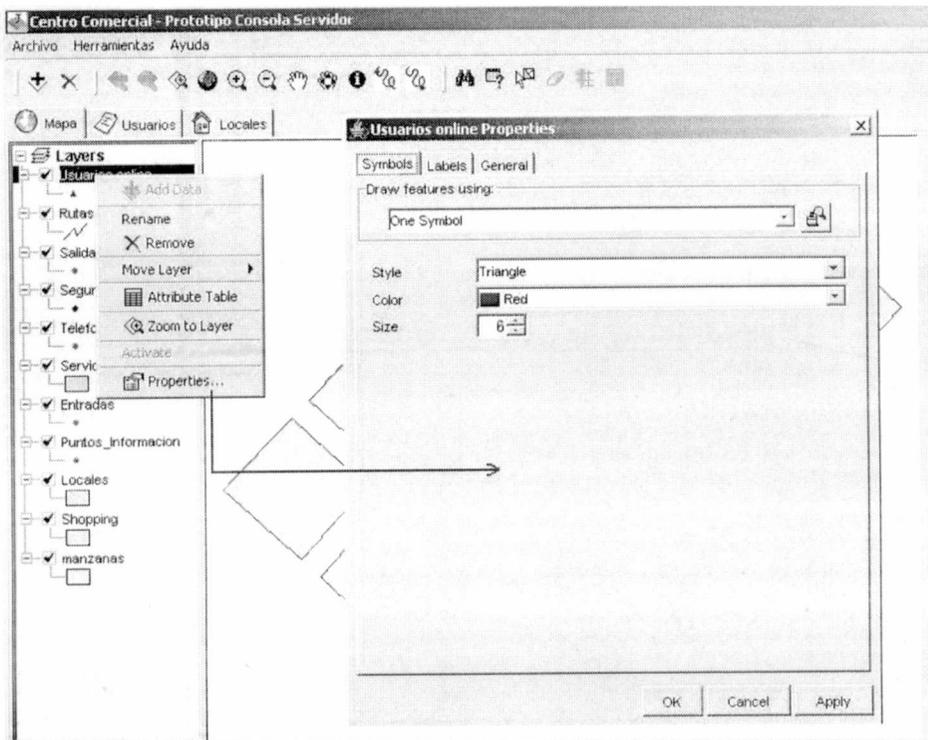
El texto mostrado en el árbol es el nombre de los layers y la imagen que se muestra por debajo del texto representa el tipo de información geométrica que se ha utilizado para representar al layer. Estas representaciones pueden ser de los siguientes tipos:

- *Puntos*: Los ejemplos de este tipo de geometría son los usuarios, las salidas de emergencia, la seguridad, los teléfonos públicos, etc.
- *Polígonos*: Sirven para representar objetos geométricos tales como los locales del centro comercial, los servicios, etc.
- *Líneas*: El layer Rutas está representado por líneas, es decir, los diferentes caminos que unen a los locales del centro comercial.

#### 1.1 Propiedades de los Layers:

Cada layer posee un conjunto de propiedades, tales como: color, tipo de letra, etc., las cuales pueden ser modificadas en tiempo de ejecución con la herramienta de “Propiedades del Layer” implementada para tal fin.

En la **Figura 7.4** se puede apreciar la vista de la ventana para la modificación de las propiedades mencionadas en el párrafo anterior:



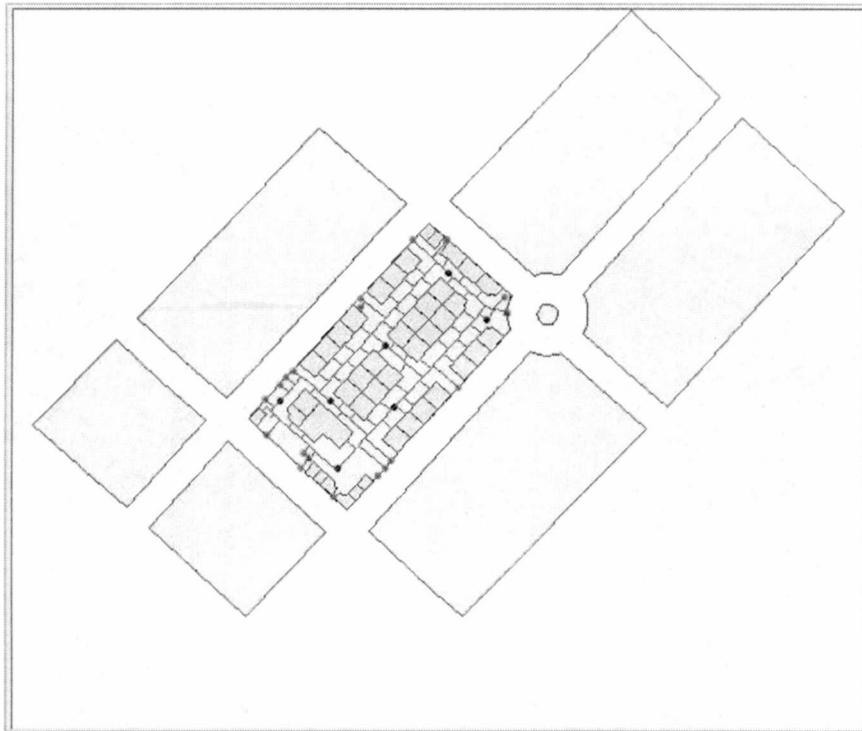
**Figura 7.4: Propiedades de los layers**

Como se puede apreciar en la **Figura 7.4** existen más operaciones para realizar sobre un layer: “Rename”, permite renombrar a un layer; “Remove”, permite eliminar del mapa el layer seleccionado; “Move Layer”, permite mover el layer mas arriba o mas abajo según se desee; “Zoom to layer”, establece el zoom de manera que el layer seleccionado sea visualizado de manera completa dentro de la pantalla de visualización del mapa.

Otra funcionalidad provista por la estructura de visualización de layers es la posibilidad de habilitarlos/deshabilitarlos, es decir, hacer visible un layer o no sobre el mapa.

## **2. Pantalla de visualización del mapa:**

Esta pantalla permite visualizar todos los datos geométricos de los layers. Cada layer es superpuesto sobre otro layer para armar en conjunto la representación del mapa del centro comercial.



**Figura 7.5: Pantalla de visualización del mapa**

### 3. Listado de usuarios y locales:



**Figura 7.6: Lista de Usuarios**

Estas dos componentes muestran el listado de usuarios (nombre completo de la persona y el nombre de usuario) conectados al sistema y los locales (nombre del local y el nombre completo del propietario del local) del centro comercial.

Al seleccionar un usuario o un local de los listados el zoom de la ventana del mapa es actualizado con respecto a la ubicación geográfica del objeto seleccionado.

Asociado a cada usuario conectado al sistema existe un historial del camino que lleva recorrido desde el momento en que ingreso al sistema hasta el momento actual de la consulta. Para acceder a esta operación de debe activar el menú popup sobre algún usuario y luego seleccionar la opción "Historial camino". La operación "Limpiar camino" borra del mapa el historial del camino.

#### 4. Barra de Herramientas:



Figura 7.8: Barra de Herramientas

La Figura 7.8 superior muestra el conjunto de operaciones que ofrece la barra de herramientas. A continuación se presenta una descripción breve de cada una de estas:

- **+ Agregar Layer:** Permite agregar un nuevo layer al mapa.
- **× Eliminar Layer:** Permite eliminar un layer (previamente seleccionado) del mapa del centro comercial.
- **↶ Zoom previo:** Permite volver al zoom previo luego de cambiar el zoom.
- **↷ Zoom posterior:** Permite avanzar al zoom posterior luego de cambiar el zoom.
- **📍 Zoom layer actual:** establece el zoom de manera que el layer seleccionado sea visualizado de manera completa dentro de la pantalla de visualización del mapa.
- **🌐 Zoom por defecto:** establece el zoom en función del sistema de coordenadas del sistema.
- **🔍 Zoom in:** permite acercar la visualización del mapa mediante el uso del mouse.
- **🔍 Zoom out:** permite alejar la visualización del mapa mediante el uso del mouse.
- **🖱️ Pan:** permite desplazar la visualización del mapa mediante el uso del mouse.
- **🌐 Pan en Dirección:** permite desplazar la visualización del mapa hacia el Norte, Sur, Este u Oeste.
- **ℹ️ Información:** permite visualizar información de un objeto seleccionado en el mapa.
- **📍 Selección punto origen de búsqueda:** permite seleccionar un punto de origen para realizar la búsqueda entre locales.
- **📍 Selección punto destino de búsqueda:** permite seleccionar un punto de destino para realizar la búsqueda entre locales.

-  **Buscar:** Permite hacer búsquedas sobre los layers del mapa (solo por ID).
-  **Constructor de Consultas:** Permite generar visualmente consultas a la base de datos geográfica.
-  **Selección de objetos:** Permite seleccionar un conjunto de objetos del layer seleccionado. Las opciones de selección son: Rectángulo, Circulo, Línea y Polígono.
-  **Limpiar selección:** Permite limpiar la selección realizada con la operación previamente descrita.
-  **Buffer:** Permite guardar en memoria temporal el conjunto de objetos seleccionados.
-  **Atributos de la tabla:** Permite visualizar los atributos de un conjunto de objetos seleccionados almacenados en la base de datos.

### 7.3.3 Implementación del algoritmo de búsqueda entre locales

Una de las funcionalidades más importantes para este trabajo es la posibilidad de realizar búsquedas entre locales del centro comercial. Esta funcionalidad es imprescindible dado que en ella se basan los conceptos previamente presentados de la Hipermedia Física.

El campo de estudio de algoritmos de búsqueda ha sido ampliamente estudiado en las últimas décadas. Según [59] un algoritmo de búsqueda es aquel que toma un problema como entrada y retorna una solución al problema, usualmente luego de haber evaluado un conjunto de posibles soluciones. El conjunto de posibles soluciones para un problema es llamado el “espacio de búsqueda”. Los algoritmos de búsqueda por fuerza bruta [60] son de uso simple y generalmente los métodos de búsqueda que utilizan sobre el “espacio de búsqueda” son los más intuitivos en contraposición a estos métodos existen algoritmos optimizados (mediante heurísticas) que tratan de reducir el tiempo requerido para realizar las búsquedas.

#### 7.3.3.1 Algoritmos de búsqueda sobre grafos

Dentro del amplio espectro de algoritmos de búsqueda (búsquedas sobre árboles, listas, arreglos) se encuentran los algoritmos de búsqueda sobre grafos [61]. A continuación detallaremos algunos de estos tipos de algoritmos que han sido ampliamente investigados.

- **Algoritmo de Dijkstra:**  
También llamado algoritmo de caminos mínimos, es un algoritmo para la determinación de todos los caminos más cortos, comenzando la búsqueda en un vértice origen de un conjunto de vértices de un grafo dirigido con peso en cada arista.
- **Algoritmo de Kruskal:**  
El algoritmo de Kruskal es un algoritmo de la teoría de grafos para encontrar un árbol expandido mínimo [62] en un grafo conexo y valuado.

Es decir, busca un subconjunto de aristas que, formando un árbol, incluyen todos los vértices y donde el peso total de todas las aristas del árbol es el mínimo. Si el grafo no es conexo, entonces busca un bosque expandido mínimo (un *árbol expandido mínimo* para cada componente conexa).

- **Algoritmo de Prim:**

El algoritmo de Prim es un algoritmo de la teoría de los grafos para encontrar un árbol de expansión mínimo en un grafo conectado y valuado. En otras palabras, el algoritmo encuentra un subconjunto de aristas que forman un árbol con todos los vértices, en donde el peso total de todas las aristas en el árbol es el mínimo posible. Si el grafo no está conectado, entonces el algoritmo encontrará el árbol de expansión mínimo para uno de los componentes conectados. El algoritmo fue diseñado en 1930 por el matemático Vojtech Jarník y luego de manera independiente por el científico computacional Robert C. Prim en 1957 y redescubierto por Dijkstra en 1959. Por esta razón el algoritmo es también conocido como *algoritmo DJP* o *algoritmo de Jarník*.

- **Algoritmo de búsqueda en profundidad / Depth-first search (DFS):**

Este algoritmo permite recorrer todos los nodos de un grafo o árbol de manera ordenada, pero no uniforme. Su manera de funcionar se basa en ir expandiendo cada una de los nodos que va localizando, de manera recursiva, recorriendo todos los nodos de un camino concreto. Cuando ya no quedan más nodos por visitar en este camino, regresa hacia atrás (backtracking), de tal manera que comienza el mismo proceso con cada uno de los hermanos del nodo ya procesado.

- **Algoritmo de búsqueda en anchura / Breadth-first search (BFS):**

Este es un algoritmo para recorrer o buscar elementos en un grafo (usado frecuentemente sobre árboles). Intuitivamente, se comienza en la raíz (eligiendo algún nodo como elemento raíz en el caso de un grafo) y se exploran todos los vecinos de este nodo. A continuación para cada uno de los vecinos se exploran sus respectivos vecinos, y así hasta que se recorra todo el árbol.

Formalmente, BFS es un algoritmo de *búsqueda sin información*, que expande y examina todos los nodos de un árbol sistemáticamente para buscar una solución. No usa ninguna heurística.

Si la búsqueda la estamos realizando sobre un grafo que incluya ciclos, debemos añadir alguna forma de comprobar, para cada nodo, si ha sido visitado y/o expandido.

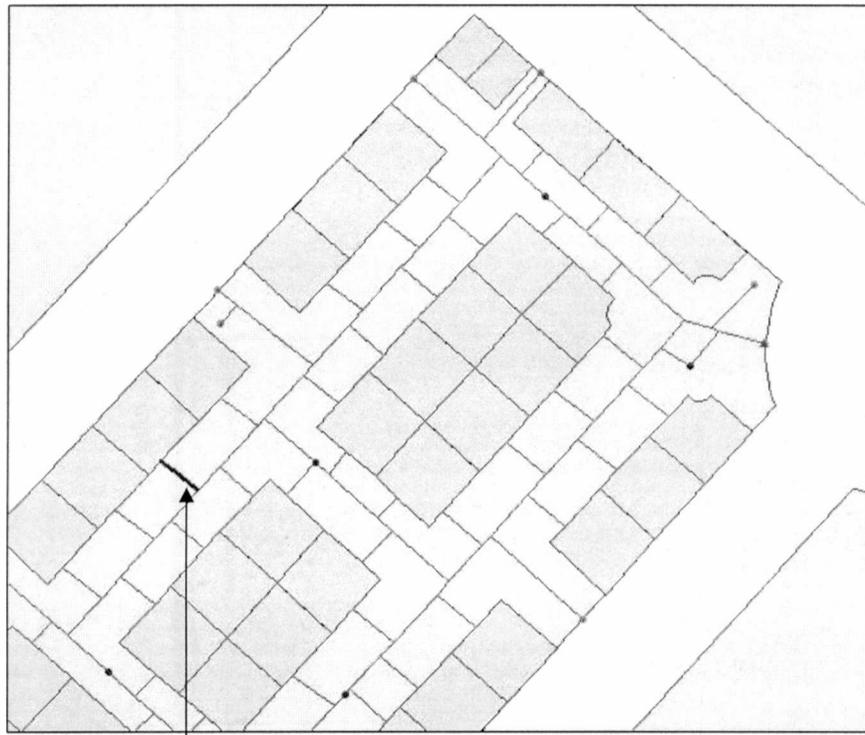
En la implementación del prototipo fue utilizada una versión modificada del algoritmo BFS, en la cual se mantiene un historial de tramos visitados (dado que el grafo es cíclico). Dicho historial permite evitar los ciclos dentro del grafo. La otra modificación realizada fue mantener un costo asociado a cada camino encontrado entre el punto origen y el punto destino, de esta forma aseguramos que el algoritmo encuentra el camino de costo mínimo optimizando el tiempo de búsqueda, dado que no visita aquellos nodos cuya visita genera un costo mayor que algún costo de un camino ya encontrado.

### 7.3.3.2 Haciendo búsquedas sobre el prototipo

En esta sección se mostrará la funcionalidad que permite realizar búsquedas de caminos entre diferentes locales del Centro Comercial.

Si bien es factible realizar la implementación de diferentes algoritmos de búsqueda, por una cuestión de tiempos nos limitamos a realizar la implementación de la búsqueda de mejor camino, donde mejor camino se entiende como el más corto en distancia desde un local a otro.

En la **Figura 7.9** se muestra el momento en que es seleccionado el punto origen para realizar alguna búsqueda en el sistema. Una vez seleccionado el punto, el sistema selecciona automáticamente la línea más próxima que lo contiene, dicha línea es repintada de color azul para confirmar la selección realizada por el usuario.



**Figura 7.9:** Selección punto origen.

Tramo seleccionado como punto de partida para la búsqueda.

Ahora, en la figura que se presenta a continuación se muestra el momento en que es seleccionado el punto destino para realizar la búsqueda en el sistema. Una vez seleccionado el punto comienza a realizarse la búsqueda automáticamente. El tiempo requerido para computar la búsqueda dependerá de la distancia de los tramos dentro del mapa, es decir, a mayor distancia entre los tramos mayor es el tiempo requerido.



**Figura 7.10: Selección punto destino.**

Tramo seleccionado como punto destino para la búsqueda.

Una vez que el sistema ha finalizado el cómputo de la búsqueda el resultado de la misma es presentado en el mapa pintando todos los tramos que componen dicho resultado.

La **Figura 7.11** muestra el resultado de la ejecución de la búsqueda seleccionando como punto origen y destino a los puntos de las dos figuras anteriores.

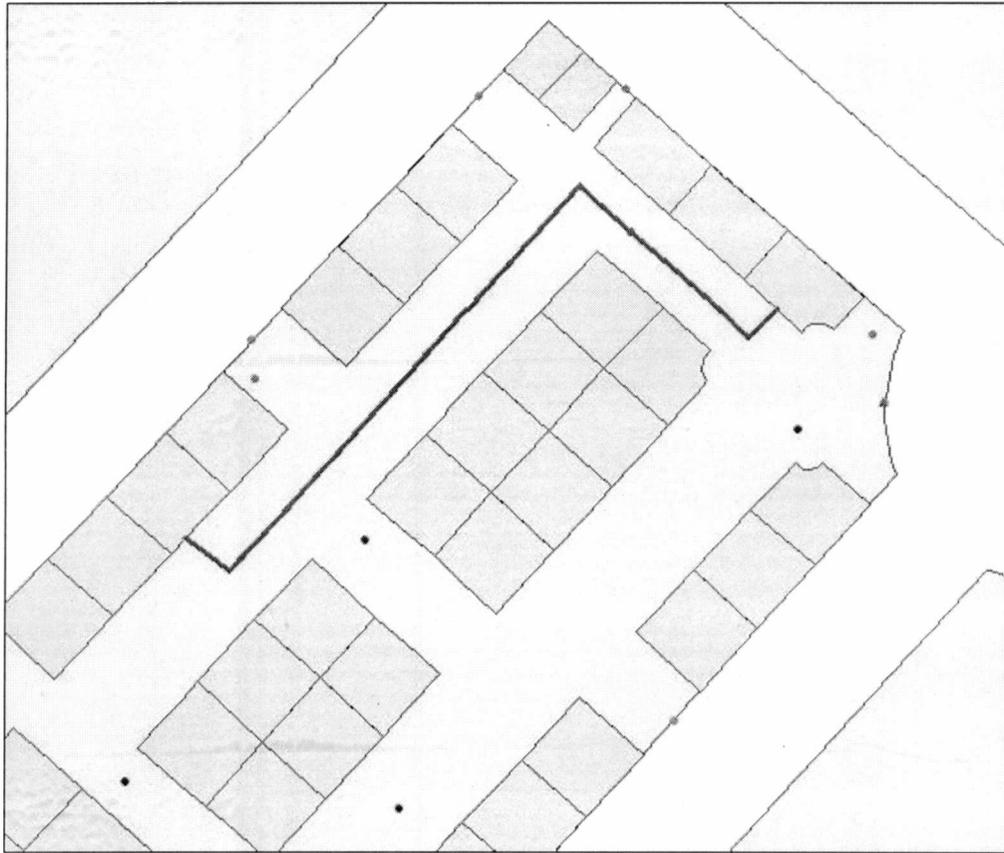


Figura 7.11: Resultado de la búsqueda del Camino

#### 7.3.4 Usuarios del sistema

Otra de las características que provee el prototipo es la posibilidad de visualizar los datos de los usuarios registrados en el sistema. Para cada uno de estos se pueden ver los siguientes datos:

- Nombre de usuario.
- Nombre personal.
- Apellido.
- Edad.
- Datos del perfil de usuario:
  - Deportes.
  - Comida.
  - Ofertas.
  - Salud.
  - Música.
  - Belleza.
  - Juegos.
  - Ropa.
  - Otros.
- Dispositivos. Para cada dispositivo que posee el usuario se tiene la siguiente información:
  - Marca.
  - Modelo.
  - Cantidad de memoria.

- Tipo de Display.
- Tipo de dispositivo.

En la **Figura 7.12** se presenta la interfase que contiene la información detallada anteriormente.

The screenshot shows a web application interface with a navigation bar at the top containing 'Mapa', 'Usuarios', and 'Locales'. The main content area is divided into several sections:

- Usuarios:** A list of users with 'Miguel Sotomayor (msoto)' selected.
- Datos del usuario seleccionado:** A form displaying user details:
  - Nombre Usuario: msoto
  - Nombre: Miguel
  - Apellido: Sotomayor
  - Edad: 42
- Perfil Usuario:** A set of checkboxes for user preferences:
  - Deportes
  - Comida
  - Ofertas
  - Salud
  - Música
  - Belleza
  - Juegos
  - Ropa
  - Otros
- Dispositivos:** A list of devices with 'Alcatel AL992' selected.
- Device Details:** A form displaying device specifications:
  - Marca: Alcatel
  - Modelo: AL992
  - Memoria: 8MB
  - Display: 128x128 - 65000 Colores
  - Tipo Dispositivo: Celular

**Figura 7.12: Información del usuario.**

### 7.3.5 Locales del Centro Comercial

También es posible, desde la consola de servidor, visualizar la información registrada de los locales del sistema. Para cada uno de ellos se encuentra almacenada la siguiente información:

- Nombre del local.
- Nombre del propietario.
- Apellido del propietario.
- Datos del perfil del local: estos pueden ser,
  - Deportes.
  - Comida.
  - Ofertas.
  - Salud.
  - Música.
  - Belleza.
  - Juegos.
  - Ropa.
  - Otros.
- Productos en venta. Para cada uno de estos se tiene la siguiente información:
  - Marca.
  - Precio (o costo).

En la siguiente figura se presenta la interfase que contiene la información de los locales detallada anteriormente.

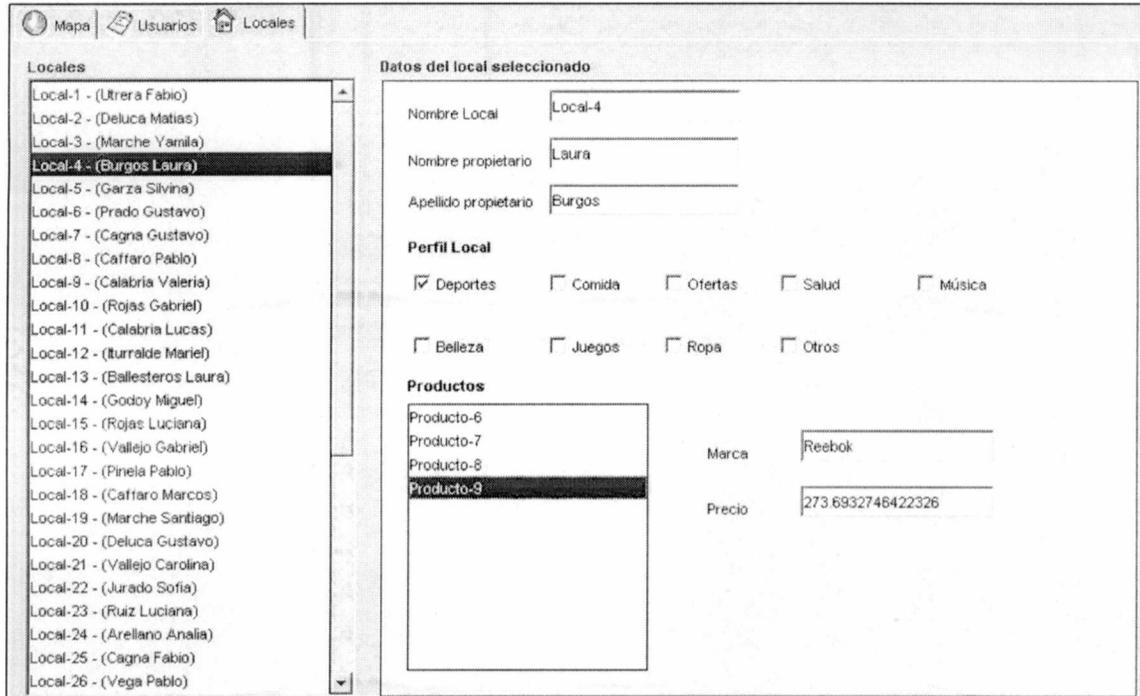


Figura 7.13: Información de los locales.

### 7.3.6 Persistencia de los datos

A la hora de persistir la información que administra el sistema, como una primera opción utilizamos Hibernate [66], el cual básicamente permite persistir objetos en bases de datos relacionales de una manera relativamente simple. Si bien es un framework muy robusto resulto ser contraproducente para realizar la implementación del prototipo por cuestiones de configuración, es por esto que se optó por un método mucho mas simple que es la persistencia mediante la serialización de objetos a XML [68].

Para realizar la serialización de los objetos fue utilizada la librería XStream [67] la cual provee un conjunto de clases que engloban el comportamiento requerido para la codificación de objetos a XML y la decodificación de código XML a objetos del modelo.

La decisión de utilizar XStream como herramienta para implementar la persistencia de debió a estos motivos:

- ***Simplicidad y sencillez en su utilización.***
- ***No se necesitan archivos de mapeos:*** La mayoría de los objetos pueden ser serializados sin la necesidad de especificar un archivo de mapeo.

- **Performance:** La velocidad y el poco consumo de memoria son parte crucial de la librería, gracias a esto es posible serializar grandes grafos de objetos.
- **XML limpio:** Toda aquella información que puede ser recuperada por reflexión no es duplicada en el código XML.
- **No requiere que los objetos del modelo sean modificados:** Serializa los atributos internos de una clase, incluyendo los privados y los finales. Soporta clases no públicas y clases internas. No requiere que las clases posean un constructor por defecto.
- **Soporte para grafo de objeto completo:** Las referencias duplicadas encontradas en el modelo de objetos son mantenidas. Soporta referencias circulares.
- **Integración con otras APIs XML:** Mediante la implementación de una interface java, XStream puede serializar directamente desde y hacia cualquier estructura de árbol (no únicamente XML).
- **Estrategias de conversión configurables:** Se pueden definir o modificar las estrategias existentes especificando como se desea representar ciertos tipos de datos.
- **Mensajes de error:** Cuando ocurre una excepción debido a un XML con errores sintácticos, XStream provee un diagnostico detallado para ayudar a resolver el problema.

### 7.3.6.1 Diseño e Implementación del modulo de Persistencia

La **Figura 7.14** muestra el diseño de clases del modulo de persistencia. En total son tres clases: AdminPersistenciaObjetos, AdminPersistenciaXML y ObjetosRaiz. Las demás clases mostradas en el diagrama están incluidas para una mejor comprensión del modelo.

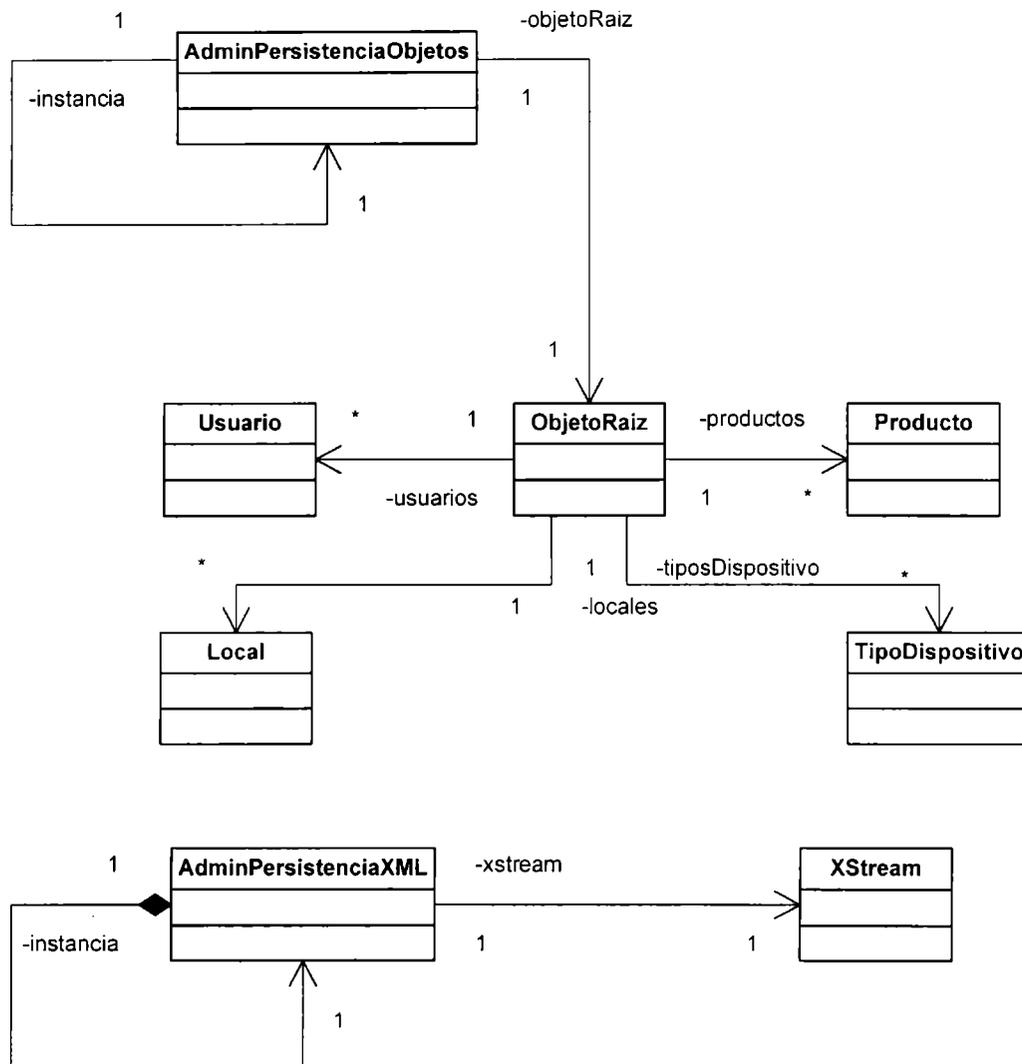


Figura 7.14: Diagrama de clases del módulo de persistencia

Para una mejor comprensión del módulo a continuación se presenta una descripción de cada una de las clases que componen el diseño.

- **Clase *AdminPersistenciaObjetos*:** Esta clase hace uso del patrón de diseño singleton [69], el cual permite restringir la instanciación de una clase a un único o eventualmente muy pocos objetos. Resulta muy útil cuando por ejemplo se necesita un único objeto que coordina acciones en un sistema tal como es el caso de la clase que se está modelando. Esta clase es la encargada de establecer una capa entre las acciones del servidor y los objetos a persistir. En otras palabras, cada vez que el servidor quiere guardar, eliminar o actualizar algún objeto delega esta responsabilidad a la clase *AdminPersistenciaObjetos*. Por otro lado posee conocimiento a una instancia de *ObjetoRaiz* cuya funcionalidad será explicada a continuación.
- **Clase *ObjetoRaiz*:** La existencia de este objeto surge a causa de que si se quisiese guardar cada uno de los objetos del modelo por separado se tendría un archivo XML por cada uno de estos, entonces

para poder tener un solo archivo en donde se encuentre toda la información del sistema se creó la clase ObjetoRaiz, la cual posee las colecciones de todos los usuarios, locales, tipos de dispositivo y productos del sistema. Aquellas clases del modelo que no estén en colecciones dentro del ObjetoRaiz son almacenadas igualmente dado que XStream permite almacenar los objetos con todo su grafo de referencia a instancias de otras clases. Por ejemplo, los locales poseen propietarios, los cuales no se encuentran almacenados en una colección del ObjetoRaiz pero sí son almacenados como referencias de la clase Local. Entonces para recuperar todos los propietarios de los locales primero se debe recuperar los locales y luego acceder a los propietarios.

- **Clase AdminPersistenciaXML:** Esta clase es la encargada de manejar la lógica de bajo nivel en cuanto a la persistencia, esto es, acceso al sistema de archivos e interactuar con las clases de las librerías de XStream. Básicamente, el proceso de serialización y el armado de los objetos es delegado a una instancia de la clase XStream, mientras que el guardado del archivo XML en el sistema de archivos queda a cargo de la clase AdminPersistenciaXML. Por otro lado esta clase también es la responsable de cargar en memoria las instancias de los objetos previamente guardados en el sistema cada vez que se inicia el servidor.

## 7.4 Implementación del Cliente

La implementación de la aplicación cliente esta basada en el concepto de “cliente delgado”, esto es, el cliente solo se hace cargo de la presentación. La lógica de negocios y manejo de persistencia quedan a cargo del servidor.

En esta sección del capítulo se explicará cómo fue realizada la implementación, presentando las herramientas y las tecnologías que fueron utilizadas.

### 7.4.1 Herramientas utilizadas para la implementación

En esta sección se presentan las diferentes tecnologías que fueron utilizadas para realizar la implementación de la aplicación cliente.

- **Lenguaje de programación:**  
El lenguaje seleccionado para realizar la implementación es Java [56] en su versión Java 2 Micro Edition (J2ME) [29]. Esta plataforma es una colección de APIs en Java orientadas a productos de consumo como PDAs, teléfonos móviles o electrodomésticos.
- **Entorno de desarrollo:**  
El entorno de desarrollo seleccionado para realizar la implementación es el mismo que el que fue utilizado para la implementación del servidor (Eclipse). Tal como se explico en la sección del servidor, una las razones para utilizar el entorno Eclipse es la facilidad que tiene para agregar nueva funcionalidad mediante la instalación de plugins.
- **EclipseMe - Plugin de Eclipse para desarrollo de MIDlets J2ME:**  
EclipseMe [64] es un plugin de Eclipse para el desarrollo de MIDlets J2ME. EclipseME se encarga del trabajo pesado de conectar la Wireless Toolkits con el ambiente de desarrollo Eclipse, permitiendo al usuario enfocarse en la tarea de desarrollo de la aplicación y no preocuparse por detalles finos de configuración requeridos por J2ME.

### 7.4.2 Implementación de la Aplicación Cliente

Una de las limitaciones que encontramos a la hora de realizar la implementación es la dificultad que existe para poder realizar las pruebas del trabajo implementado en dispositivos móviles reales. Para resolver este problema recurrimos al entorno de desarrollo, testeo y simulación que nos provee la Sun Java Wireless Toolkit [65]. Esta herramienta de desarrollo, la cual puede ser integrada al entorno Eclipse, incluye ambientes de simulación, optimización de parámetros, documentación, y ejemplos.

### 7.4.2.1 Funcionalidades de la Aplicación Cliente

A continuación se detallará cada una de las funcionalidades junto con las diferentes interfaces que integran la aplicación cliente.

#### 1. Registro de usuario:

Esta función permite registrar a un nuevo usuario en el sistema. La siguiente figura es la interface de login al sistema.



Figura 7.15: Pantalla de inicio del Usuario.

En esta interface el usuario puede acceder a las siguientes dos opciones, en el caso de que ya se encuentre registrado puede acceder a la opción de login y en caso contrario puede registrarse como un nuevo usuario en el sistema.

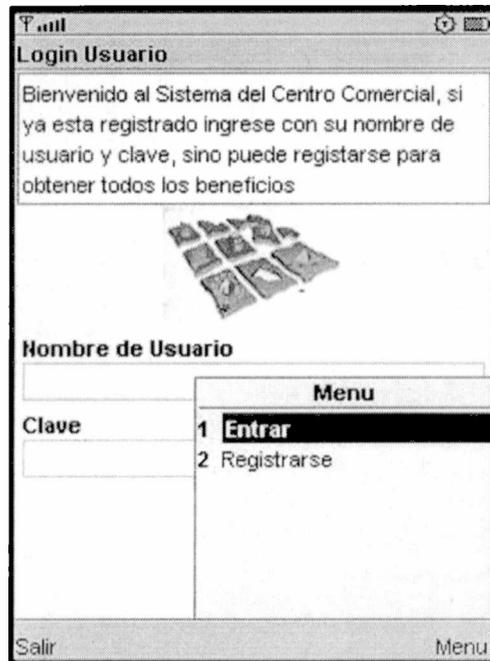


Figura 7.16: Menú de Login o Registro del Usuario.

Si un cliente no se encuentra registrado y desea ingresar al sistema accediendo a la opción 2 del Menú puede ingresar todos sus datos para ser registrados, la siguiente figura presenta la primer interfáce para el ingreso de los datos del usuario.

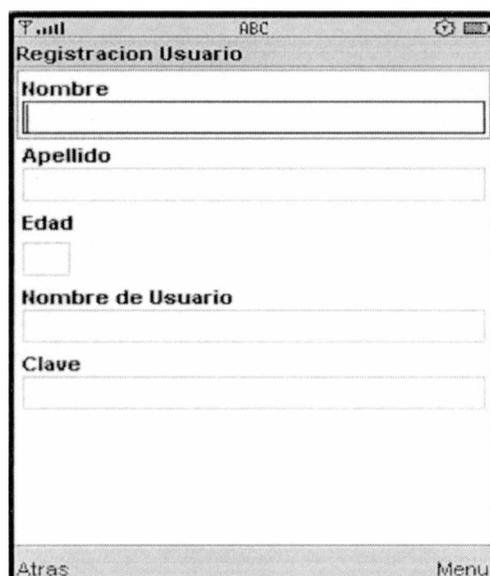


Figura 7.17: Registro del Usuario.

Tal como se puede apreciar en la figura anterior, los datos requeridos para registrar un nuevo usuario son los siguientes:

- Nombre.
- Apellido.
- Edad.
- Nombre de usuario (para el posterior acceso al sistema).
- Clave (para el posterior acceso al sistema).

Una vez que el usuario carga los datos, el siguiente paso es registrar el o los dispositivos que posea. Para efectuar esta registraci3n la aplicaci3n cliente provee la siguiente interface

The screenshot shows a mobile application interface titled "Registracion Dispositivo". It contains several input fields and radio buttons for device registration. The "Tipo de dispositivo" section has three radio buttons: "Celular" (selected), "Palm", and "Notebook". Below this are text input fields for "Marca" and "Modelo". There are also input fields for "Memoria (en MB)", "Alto display", and "Ancho display". The "Display Color" section has two radio buttons: "Si" (selected) and "No". At the bottom of the form is an input field for "Cantidad de colores". The bottom navigation bar contains three buttons: "Atras", a central up/down arrow, and "Registrar Dispos".

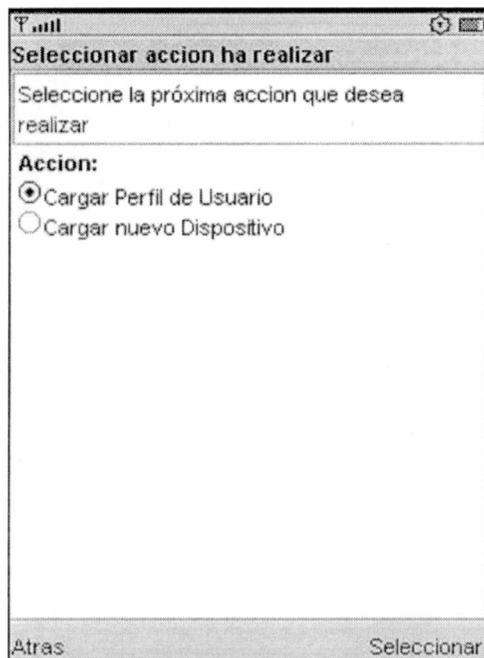
Figura 7.18: Registro de dispositivos del Usuario.

Para proceder con la registraci3n de un dispositivo los datos requeridos son los siguientes:

- Tipo de dispositivo: Celular, Palm o Notebook.
- Marca, por ejemplo: Motorola, Nokia, etc.
- Modelo, por ejemplo: V180, V300, etc.
- Cantidad de memoria.
- Alto y ancho del display.
- Tipo de display: color o blanco y negro.
- Y por 3ltimo la cantidad de colores que puede representar el display (en el caso de que el tipo de display sea color).

Una vez que un dispositivo es registrado el usuario puede optar entre registrar un nuevo dispositivo (en caso de que posea m3s de un dispositivo) o registrar su perfil de usuario.

La siguiente interface muestra las opciones que puede seleccionar el usuario.



**Figura 7.19: Opciones de carga para el Usuario.**

En el caso de que el usuario desee registrar un nuevo dispositivo, los pasos a seguir son los mismos que los explicados anteriormente. En cambio si desea registrar su perfil de usuario, es decir, aquellas cosas o temas que sean de su interés, los datos a registrar pueden ser apreciados en la siguiente figura.



**Figura 7.20: Registro de temas de interés.**

Cuando todos los datos requeridos para el alta de un nuevo usuario han sido cargados, el sistema almacena toda la información para su utilización en los posteriores logeos del usuario.

## 2. Login de usuario:

Esta función permite que un usuario, previamente registrado, ingrese al sistema del Centro Comercial.

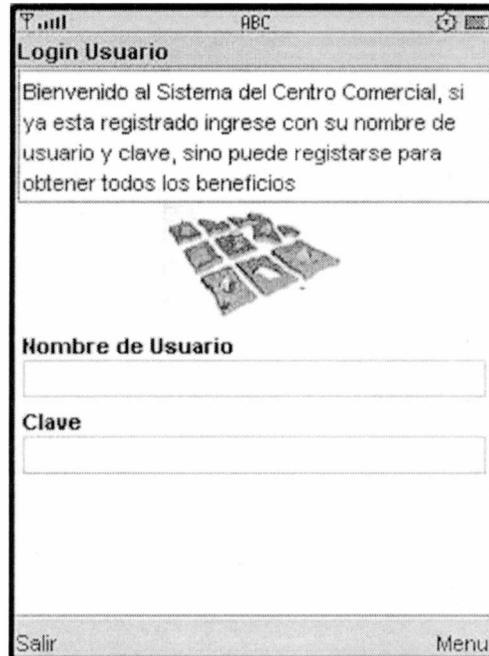


Figura 7.21: Login del Usuario.

### 7.4.3 Simulación del movimiento del usuario

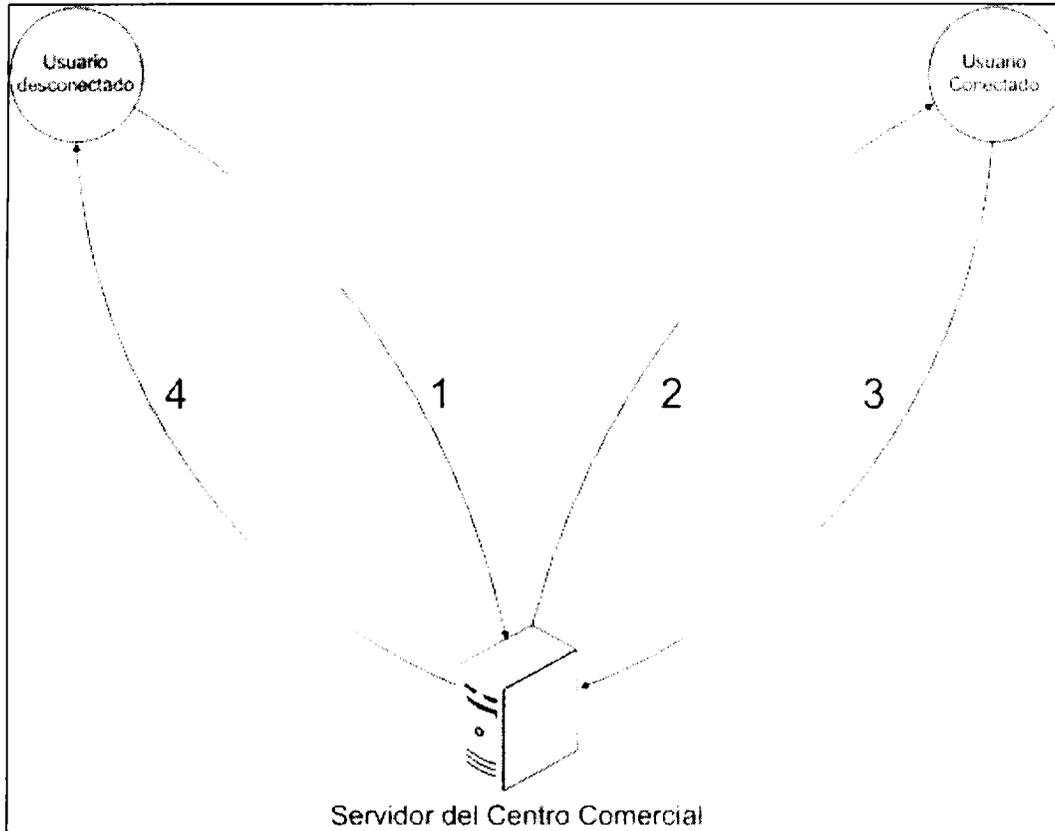
Una de las acciones más habituales que realiza una persona dentro de un Centro Comercial es caminar por el mismo observando las vidrieras de los diferentes locales, como así también la acción de efectuar compras cuando ingresa a un local.

Podemos abstraer el proceso, simplificándolo, y dividirlo en las siguientes instancias:

1. Caminar: observando la vidriera de los locales (deteniéndose un tiempo x)
2. Entrar a algún local.
3. Comprar algún/algunos producto/s del local.
4. Salir del local.
5. Volver a 1 o pasar a 6.
6. Salir del Centro Comercial.

Dado que el prototipo no interactuara con usuarios reales fue necesario realizar la implementación de un proceso que simule los ítems, 1, 2, 4 y 5. Los ítems 3 y 6 pueden ser ejecutados desde la aplicación cliente, es por esto que no requieren ser simulados mediante un proceso.

A continuación se presenta un grafico que intenta esclarecer el proceso que simula el movimiento de un usuario dentro del Centro Comercial.



**Figura 7.22:** Interacción entre los usuarios y el servidor durante la simulación del movimiento.

Como se puede apreciar en la **Figura 7.22** el proceso de simulación consta de cuatro ítems:

1. El usuario ingresa al Centro Comercial y envía al servidor sus datos de login. El Servidor recibe los datos y verifica su existencia en el sistema.
2. En caso afirmativo se genera un punto de acceso aleatorio y este es enviado a la Aplicación Cliente. Este punto representaría una de las puertas de acceso al Centro Comercial. El usuario pasa de estar “Desconectado” a “Conectado”.
3. La simulación seguirá ejecutándose hasta que desde la Aplicación Cliente se seleccione la opción “Salir del Sistema”.

La determinación de los siguientes puntos hacia donde debe avanzar el usuario queda a cargo del Servidor el cual recibe una identificación de un punto X y calcula, mediante consultas espaciales, los puntos adyacentes aún no visitados, de este conjunto de puntos se selecciona uno al azar y es enviado a la Aplicación Cliente. En el caso de que uno de estos puntos tuviese adyacencia con un local la información de dicho local es enviada a la Aplicación Cliente para ser presentada al usuario, siempre y cuando el perfil del usuario concuerde con el perfil de los productos que ofrece el local.

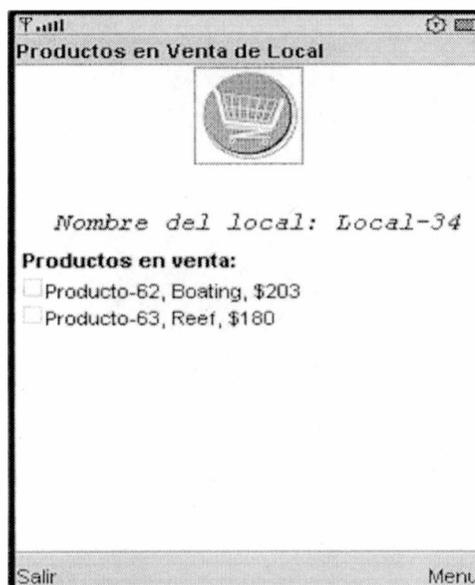
- Desde la Aplicación Cliente se selecciona la opción “Salir del Sistema” esto genera una desconexión con el servidor y el usuario pasa al estado “Desconectado”.

#### 7.4.4 Visualización de productos en venta

En esta sección se describe la implementación del proceso de visualización de productos en venta de un local. El proceso de simulación, definido en la sección anterior, establecerá una comunicación con la Aplicación Cliente cada vez que el la posición del usuario se encuentre cercana a un local cuyo perfil coincida con el perfil del usuario.

La **Figura 7.23** presenta la interfase de visualización de los productos.

En primer término se muestra el nombre del local y luego los productos que este ofrece, con la información de su nombre, marca y costo asociado:



**Figura 7.23: Productos en venta del local “Local-34”**

La interfase provee 3 diferentes funcionalidades, en la **Figura 7.24** se pueden apreciar cada una de estas:



**Figura 7.24: Productos en venta / Opciones del Menú**

Descripción de las opciones del menú:

**1. Continuar Simulación:** Esta operación permite continuar con el proceso de simulación de movimiento del usuario. La Figura 7.25 presenta la interfase “Simulación en Proceso...”. Desde esta interfase la única opción para seleccionar es “Salir” del sistema.

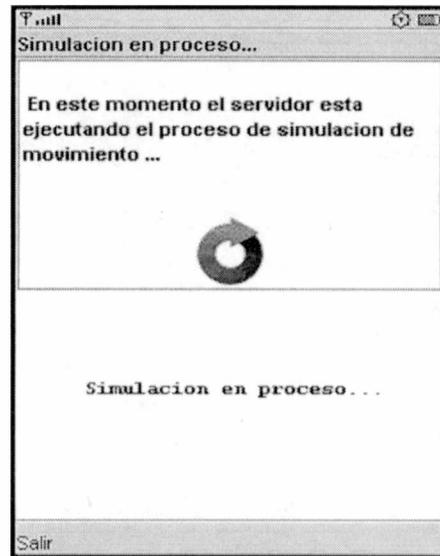


Figura 7.25: Simulación en Proceso

**2. Comprar Productos:** Esta operación permite registrar los productos seleccionados, como productos con compra pendiente. Cuando el usuario desee salir del sistema puede efectuar la compra real de los mismos ejecutando una operación de “checkout”. La interfase siguiente a esta operación es la de “Simulación en Proceso”.

**3. Visualizar Posición actual:** Esta operación permite enviar una petición al servidor acerca de la posición actual del usuario que envía la solicitud. La respuesta del servidor es una imagen, con un zoom adecuado para la visualización en el dispositivo. En este punto entra en juego el concepto de personalización ya que en función de los datos previamente registrados por el usuario (tipos de dispositivos, display, memoria, etc.) la imagen generada dependerá de estos parámetros. La Figura 7.26 presenta la interfase de visualización de posición actual de usuario.

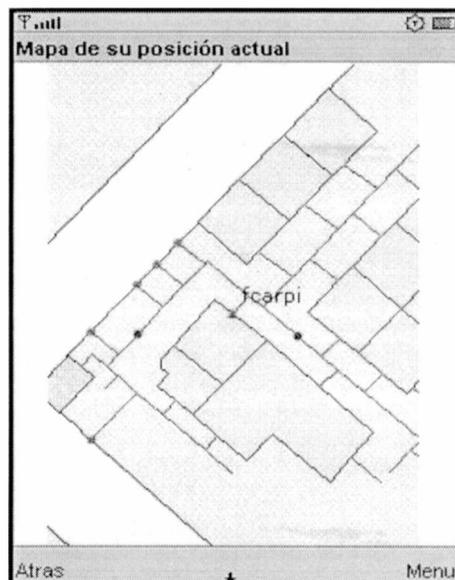


Figura 7.26: Posición actual en el Centro Comercial

# Capítulo 8

## Conclusiones

Como capítulo final de este trabajo, describiremos aquí algunas conclusiones sobre el trabajo realizado junto a la enumeración de ciertas tareas a efectuar en un futuro cercano como forma de continuar con todo lo desarrollado en el transcurso de éste proyecto.

### 8.1 Conclusiones

A continuación se enumeran una serie de ítems que surgen como conclusiones luego de estos meses de trabajo:

- En esta tesis hemos abordado un tema de investigación en proceso de crecimiento. Actualmente son pocos los sistemas que aplican los conceptos de la Hipermedia Física. Este trabajo no solo presenta a la HP dentro de su marco teórico sino que también la acerca a un dominio práctico, como es el caso del Centro Comercial.
- La implementación del prototipo deja en claro que un sistema basado en las tecnologías de Hipermedia Física, M-Commerce, etc., puede ser un producto de interés comercial en el mundo actual ampliamente comunicado por redes.
- Una característica que juega en contra a la hora de implementar sistemas de este tipo, es que actualmente en países en desarrollo, como es el caso de Argentina, la tecnología que se requiere para su implementación aún posee un costo elevado. Por ejemplo, los celulares con soporte para bluetooth tienen un costo elevado y recién ahora están comenzando a ser vendidos masivamente en el país.
- Como consecuencia directa del punto anterior nos vimos imposibilitados de implementar un prototipo que no realice una simulación y que utilice dispositivos de hardware reales, lo que nos hubiese generado una mayor complejidad en la implementación del prototipo, pero que por otro lado se podría haber apreciado con mayor claridad la utilidad del sistema y se podrían haber ampliado la perspectiva de la implementación del sistema.
- Las características de adaptatividad y personalización son claves para el éxito de un sistema de hipermedia física y comercio electrónico, más aún cuando en el dominio utilizado se identifican los gustos personales de los clientes.
- El uso de las tecnologías que fueron utilizadas para realizar este trabajo (J2ME, Tomcat, etc.) redujo la complejidad del proceso de desarrollo del sistema, ya que en su implementación se pudo obviar todos los detalles de comunicación entre los dispositivos.

- La utilización del paradigma de objetos para realizar el diseño es ideal para representar la información del mundo real. Por otro lado, el diseño de hipermedia física esta íntimamente ligado a entidades concretas, tales como locales, productos, usuarios, etc. Por lo tanto, la modelización en objetos simplifica el desarrollo de un sistema de hipermedia física.
- Las diversas estrategias mostradas con las que un usuario puede navegar en un sistema de Hipermedia Física, muestran con claridad la adaptabilidad que la Hipermedia Física provee, en forma clara y sencilla, ante las necesidades de un usuario en función del dominio sobre el que se haya implementado el sistema.
- El sistema de Hipermedia Física aplicada en un Centro Comercial brinda la posibilidad de que cada usuario que este utilizando el mismo, ingrese comentarios con ventajas y desventajas sobre los productos y locales en los que realizando sus compras. Estos comentarios pueden ser vistos por los demás usuarios quienes pueden tenerlos en cuenta a la hora de tomar una decisión sobre donde y como comprar un determinado producto, y así realizar sus compras en una forma más segura al tener mayor información a su alcance. Además, estas ventajas y desventajas generaran una sana competencia entre los distintos comercios del Centro Comercial en post de lograr una mejor atención al cliente y brindándole el mejor producto posible.

## 8.2 Trabajo Futuro

- Analizar nuevos dominios de aplicación de la Hipermedia Física, tales como: Hospitales, Transportes, Fuerzas Armadas, etc.
- Concluir el diseño y la implementación del sistema, lo cual posiblemente produciría un interesante producto comercial.
- Desarrollar un sistema de calificaciones positivas y negativas para los distintos locales del Centro Comercial, en el cuál los usuarios ingresan dichas calificaciones en función de los productos que adquieran y en función de las mismas premiar a los locales con mejor atención y con mayores ventas. Dicho sistema sería un modulo a acoplarse como parte del Sistema del Centro Comercial.
- Diseñar e implementar un Framework de Estrategias para la aplicación de la Hipermedia Física en distintos contextos, que permita brindar distintos tipos de navegabilidad en función al usuario y del dominio en cuestión.
- Investigar en mayor profundidad las posibles relaciones existentes entre Hipermedia Física y Comercio Móvil, analizando las ventajas de la utilización de ambos conceptos integrados.

# Apéndice A

## Map Object-Java

En la implementación fue utilizado Map Object-java [57], este conjunto de librerías permitió representar mapas georeferenciados en la aplicación.

Map object-java es una suite de componentes desarrollados en java para crear aplicaciones GIS, esta suite esta compuesta por una colección de librerías (archivos jar). Estas librerías proveen gran cantidad de recursos GIS y componentes para visualización de mapas.

Los componentes que provee Map Object pueden ser divididos en las siguientes categorías:

- Objetos de visualización.
- Objetos de contenido.
- Objetos de acceso a datos.

### A.1 Objetos de visualización

Esta sección describe los componentes de visualización utilizados para la implementación del prototipo.

- **Mapa**

El mapa (Map) provee una forma de visualizar los datos espaciales. Cada tipo de datos es visualizado en un layer separado. Usualmente los layers mostrados se obtienen desde un origen (por ejemplo desde un path de disco), otros layers a visualizar pueden ser creados por el usuario y su repintado es dinámico, luego de su creación pueden ser guardados en disco.

La **Figura A.1** muestra la componente de visualización map, con un layer cargado de disco desde un archivo con formato shapeFile de Esri (representa el mapa del Centro Comercial) y con layers creados por el usuario (puntos que pueden visualizarse en el mapa).

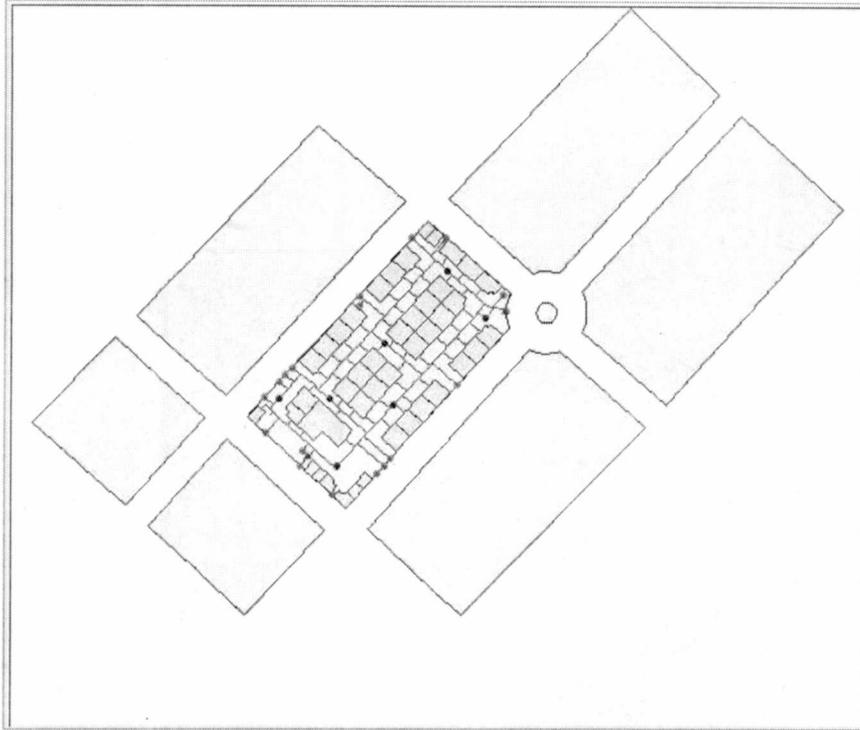


Figura A.1: Visualización de un mapa en la componente Map

- **Tabla de contenidos**

La tabla de contenidos (TreeToc), provee una forma de visualizar los layer existentes en un mapa. Cada layer es representado por un nodo con una leyenda que contiene un color con el que se muestra el layer, un título y un checkbox, que indica la visibilidad. Para que el TreeToc reconozca sobre que mapa debe mostrar los layers debe enviársele el mensaje setMap(aMap).

La Figura A.2 muestra una tabla de contenido, en donde pueden visualizarse cuatro layers: Usuarios online, Rutas, Salidas de Emergencia y Seguridad, todos seleccionados para ser visualizados.

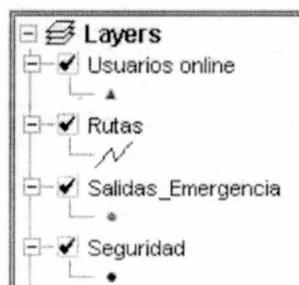


Figura A.2: Componente TreeToc

- **Layers dinámicos**

Un layer dinámico (AcetateLayer) es una componente transparente la cual es utilizada para crear layers que pueden encontrarse en continuo movimiento. Un ejemplo de uso es representar usuarios en movimiento. En un AcetateLayer pueden agregarse, removerse, o cambiarse elementos dinámicamente.

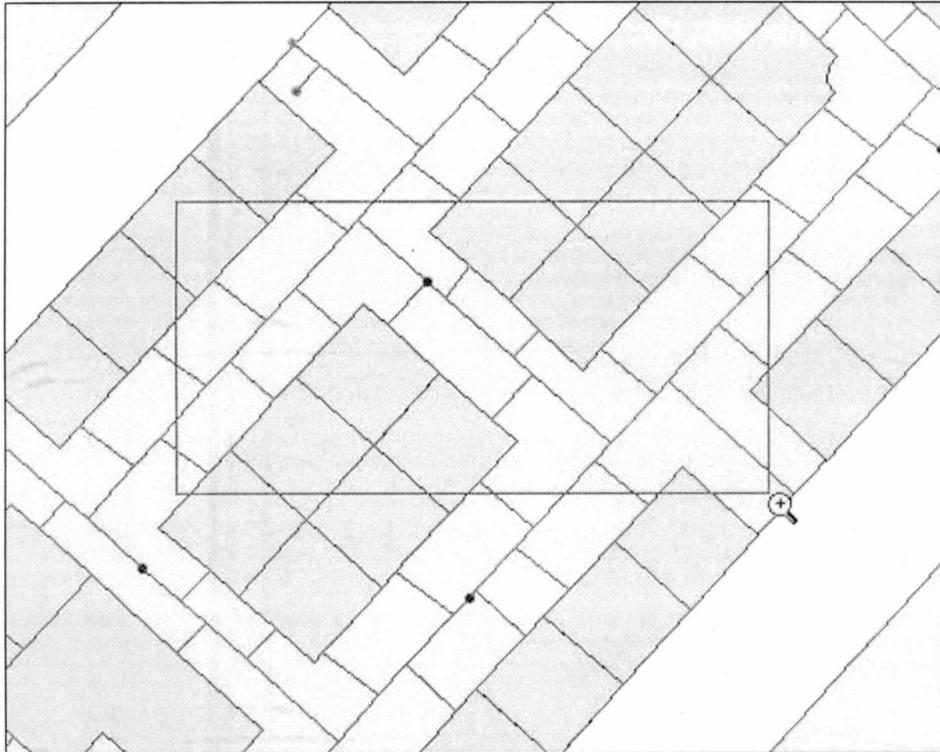
La diferencia básica entre un layer estático y un Acetatelayer, es que este último no necesita un repintado para mostrar cualquier actualización.

Cabe destacar que un mapa puede contener todos los AcetateLayer que sean necesarios.

- **Herramientas**

Las herramientas (Tools) permiten manejar eventos del mouse sobre el mapa. Solo una herramienta puede estar seleccionada a la vez. Un ejemplo de herramienta es la de zoom, que permite ver el mapa con mayor detalle.

La **Figura A.3** muestra el funcionamiento de la herramienta de zoom.



**Figura A.3: Ejemplo de funcionamiento de herramienta de zoom**



## A.2 Objetos de contenido

Dada la extensa cantidad de objetos de contenido disponibles en las librerías de MapObject-Java, en esta sección solo serán mencionados aquellos utilizados para la realización del prototipo.

- **Mapa**

Así como existe el objeto Map para visualización, existe el objeto Map de contenido que simboliza un mapa. Es decir el mapa de visualización es una vista del mapa de contenido.

Para definir la abstracción de un mapa se definen un conjunto de interfaces para promover el reuso de clases existentes, tanto para aplicaciones Client-Side como Server-Side. Las interfaces trabajan como un puente para compartir un protocolo común independientemente de cómo sea realizada la operación.

- **Display**

Un objeto Map de visualización tiene asociado un objeto DisplayManager que organiza la forma en que los layers son visualizados en el mapa. Un Display básicamente muestra tres tipos de objetos: figuras (shapes), selecciones y nombres. Un DisplayManager también es el encargado de crear el área que debe mostrarse, esto se hace con la clase DisplayArea. Un DisplayArea representa un área particular que debe visualizarse del mapa total que se esta visualizando.

- **Layers**

Un mapa esta compuesto por un conjunto de objetos layer que se organizan y manipulan usando un LayerSet. Esta componente es quien realmente contiene y ordena layers y cualquier clase de layer puede ser agregado a un LayerSet.

Así como existen layers simples existen GroupLayers que pueden ser usados para agrupar layers existentes en una estructura de árbol.

- **FeatureLayer**

Un FeatureLayer necesariamente debe mostrar las características y/o los nombres que se encuentran asociados a una FeatureClass, quien realmente contiene los elementos del layer. Existe un renderer separado (explicado posteriormente) para dibujar shapes (figuras que se muestran en el layer), selecciones (figuras seleccionadas) y los labels (textos asociados al layer).

Cuando se debe dibujar un elemento el FeatureLayer colabora con el DisplayManager para invocar el renderer apropiado.

- **Renderers**

Los renderers son los que tiene la responsabilidad de saber como dibujar un objeto contenido en un layer. Aunque los renderers son principalmente usados por FeatureLayer para mostrar componentes, estos también pueden ser usados por otros tipos de objetos como por ejemplo RubberBands.

Los renderers estándares son utilizados por un FeatureLayer para dibujar todas sus componentes de manera adecuada.

Existen varias subclases de renderers, por ejemplo GroupRenderer maneja varios Renderers, y tienen una propiedad que permite escoger cual renderer utilizar en cada componente del layer, ScaleDependentRenderer provee dependencias de escalas para un elemento, SimpleRenderer provee una forma de dibujar todas las componentes usando un solo símbolo.

También están implementados los LabelRenderer, que manejan símbolos para los textos del mapa y permiten dibujar uno o más campos. Si más de un campo es especificado, entonces el carácter blanco es usado para concatenar los valores.

- **Símbolos**

Un símbolo (Symbol) dibuja la geometría de características en un objeto grafico2D que es asociado con un visualizador de mapa. Cada Renderer tiene asociado al menos un Symbol que es el que va a utilizar para dibujar la componente de layer.

Varias implementaciones de Symbol se encuentran disponibles, donde cada una produce diferentes efectos visuales. Por ejemplo, un SimpleFillSymbol puede ser usado para rellenar polígonos con colores sólidos, y un RasterFillSymbol puede ser usado para dibujar polígonos rellenos con un patrón obtenido desde una imagen.

Los TextSymbols operan de forma similar a un símbolo para dibujar componentes de layer, pero estos solo sirven para dibujar labels dentro de un mapa.

Los GroupSymbol pueden ser usados para crear efectos de repintado compuesto para un componente.

- **Layers gráficos**

Un layer gráfico (GraphicLayer) es un contenedor que puede ser usado para mostrar diferentes clases de geometrías y nombres en un mismo layer. Este contrasta con el FeatureLayer porque el FeatureLayer solo puede mostrar un tipo de geometría (punto línea o polígono). Cualquier elemento de la clase Element puede ser agregado al GraphicLayer.

Un GraphicLayer mantiene una lista de elementos que pueden ser seleccionados o buscados. Este provee métodos para cambiar el orden en que se muestran los elementos, tal como enviarlos al frente o enviar al fondo, y moverlos o removerlos.

Un FeatureGraphicLayer provee soporte adicional para mostrar varios elementos a la vez, así como también localizar elementos usando criterios espaciales.

- **Geometry**

FeatureGeometry es la geometría asociada con componentes. Las tres familias de tipos asociadas a FeatureGeometry son Polígono, polilínea y multi puntos. Por razones de performances, el tipo punto no fue incluido en Map object.

Las FeatureGeometry están compuestas por otros elementos: un multipunto (MultiPoint) que contiene una colección de puntos (PointCollection), una polilínea (Polilyne) que contiene cero o más líneas (Paths), y un polígono (Polygon) que contiene elementos que lo forman (Rings). A su vez Rings y Paths contiene una colección de puntos (PointCollection).

Para facilitar la forma de repintado las FeatureGeometry extiende java.awt.Shape.

### A.3 Objetos de acceso a datos

Esta sección introduce las componentes que pueden ser utilizadas para el efectuar el acceso a datos.

- **Content API**

La Content API define una interfaz para acceso a datos GIS, desde diversos orígenes. Provee una conexión que brinda una forma transparente de acceder a orígenes de datos. Esta información es utilizada para crear un Source, quien maneja una conexión a recursos.

- **Conjunto de datos de layer**

Un conjunto de datos de layer (DataSetLayer) es el que realmente contiene todos los elementos de un layer (Element).

Desde un LayerSource se pueden obtener DataSetLayer. Dependiendo el tipo de LayerSource varían los DataSetLayer. De esta forma para un FeatureLayer su Dataset es un FeatureClass y para un ImageLayer su Dataset es ImageClass.

La interfaz de LayerSource permite acceder de forma transparente a los elementos del layer sin tener que llegar al Dataset, solo se debe conocer la existencia del Dataset si se desea agregarle algún comportamiento extra al Dataset.

- **Consultas**

Cuando se hace una operación de búsqueda sobre un layer, se debe indicar como se va realizar esa búsqueda, para esto es que existen las consultas (Queries).

Para seleccionar o buscar, se debe crear un filtro del Query (QueryFilter). Un QueryFilter debe contener una lista de campos que describen los valores que se quieren obtener desde la búsqueda, y opcionalmente puede tener una cláusula para selección de elementos (la cláusula de selección funciona como lo hace el

where de SQL). Si además se quiere restringir que la búsqueda se haga en solo una porción del layer se debe crear un filtro espacial (SpatialFilter).

- **Datos**

Cuando se hace una búsqueda sobre algún layer el resultado es un Cursor que contiene datos. Dependiendo el tipo del layer será el tipo de datos obtenido. Por ejemplo un Featurelayer retorna componentes y un ImageLayer retorna imágenes.

- **Conexión con archivos**

Es una implementación para una conexión con el file system. A partir de esta una aplicación puede fácilmente usar un objeto conexión para conectarse con uno o varios archivos. Para la implementación de la conexión con el file system se provee un FileHandler que provee una forma genérica de obtener el contenido de una FileFolderConnection. Además existe un jerarquía de handlers dependiendo el tipo de archivos con el que se desea trabajar, por ejemplo con un ShapeFileHandler para un archivo de formato shapefile de ESRI (.shp).

- **Índices Espaciales**

Los índice espaciales (SpatialIndex) facilitan las búsqueda de datos espaciales en memoria, de forma de hacerla más eficiente. Para indexación provee MutableTree y Tree, y los nodos que pueden ser agregados en el árbol deben ser subclases de SpatialNode y si se busca recorrer un SpatialTree se debe crear un NodeVisitor.

- **Cache**

La cache (MemoryCache) provee una forma de almacenar datos identificables en memoria para aumentar la velocidad de acceso a ellos. La cache tiene un número fijo de elementos identificables que pueden ser almacenados.

# Apéndice B

## Bluetooth

Según Pastrana [24] las comunicaciones inalámbricas están presentes en muchas de nuestras actividades diarias y su uso ha llegado a ser tan común, que perdemos la percepción de lo útil y a veces indispensable que pueden llegar a ser. Las redes celulares para transmitir voz y datos han surgido para proveer la movilidad y disponibilidad de la comunicación que el ritmo acelerado de vida de las grandes urbes exige. La utilización de sensores infrarrojos y de radiofrecuencia provee la comodidad de controlar y operar a distancia aparatos electrónicos volviendo más sencillo nuestro quehacer diario.

Asimismo, la creación de estándares de comunicaciones inalámbricas en las redes de transmisión de datos ha abierto oportunidades de desarrollo de estas tecnologías, aprovechando la utilización de interfaces aéreas operadas bajo frecuencias no licenciadas.

Bluetooth forma parte de las tecnologías creadas para proveer comunicación inalámbrica en áreas de uso personal. Sin embargo, su uso va más allá de la eliminación de cables, ya que es lo suficientemente flexible para permitir la creación de aplicaciones que abren un mundo con límite en la imaginación.

### **B.1 ¿Qué es Bluetooth?**

Es una tecnología desarrollada por Ericsson en 1994, que hace factible la conectividad inalámbrica entre dispositivos a corta distancia, éstos pueden llegar a formar redes con diversos equipos de comunicación: computadoras móviles, radiolocalizadores, teléfonos celulares, PDAs, e, inclusive, electrodomésticos.

La especificación del estándar Bluetooth [25] se compone de dos capítulos, uno de ellos describe las especificaciones técnicas principales, mientras que el otro define perfiles específicos para aplicaciones, estos últimos aseguran la interoperabilidad de dispositivos Bluetooth entre fabricantes. Algunos de estos perfiles son el de acceso genérico, identificación de servicio, puerto serial, acceso a LAN sincronización y el de dispositivo de información móvil (MIDP [26]).

### **B.2 Características**

- Tecnología inalámbrica. Reemplaza la conexión por cables en distancias que no exceden los 10 metros (esto es variable dependiendo de la potencia de la antena emisora), alcanzando velocidades del rango de 1Mbps.
- Comunicación automática. La estructura de los protocolos que lo forman favorece la comunicación automática sin necesidad de que el usuario la inicie.
- Bajo consumo de potencia. Lo pequeño de los dispositivos y su portabilidad requieren de un uso adecuado de la energía, el cual provee esta tecnología.
- Bajo costo. Los dispositivos de comunicación que soporta pueden experimentar un incremento en su costo no mayor a 20 dólares con tendencia a bajar. Asimismo, su operación se efectúa bajo una banda de frecuencias no licenciada (2.4GHZ), lo que ayuda a su bajo costo.

- Integración de servicios. Puede soportar transmisiones de voz y datos de manera simultánea.
- Transmisión omnidireccional. Debido a que basa su comunicación en radiofrecuencia, no requiere línea de vista y permite configuraciones punto-multipunto.
- Seguridad. Utiliza Spread Spectrum Frequency Hopping como técnica de multiplexaje, lo que disminuye el riesgo de que las comunicaciones sean interceptadas o presenten interferencia con otras aplicaciones. Provee también especificaciones para autenticar dispositivos que intenten conectarse a la red Bluetooth, así como cifrado en el manejo de llaves para proteger la información.
- Establecimiento de redes. Tiene la característica de formar redes en una topología donde un dispositivo hace las veces de maestro y hasta siete más operando como esclavos. Esta configuración se conoce como piconet. Un grupo de piconets, no más de diez, es referido como Scatternet.

### **B.3 Bluetooth vs. Infrarrojo**

Ambos protocolos especifican una comunicación inalámbrica a corta distancia, hay quienes suponen que Bluetooth podría sustituir las aplicaciones de infrarrojo por las claras ventajas que provee, las cuales se deducen de sus propias características.

Debido a la similitud de aplicaciones, se considera importante delimitar las ventajas entre una y otra tecnología.

- El infrarrojo requiere de una comunicación lineal entre transmisor y receptor, lo que hace imprescindible la línea de vista para su efectiva transmisión.
- Las frecuencias de la banda del infrarrojo no permiten la penetración a través de paredes, dándole una importante ventaja a la radiofrecuencia que opera Bluetooth.
- La comunicación con infrarrojo siempre será uno a uno, dejando de lado las configuraciones punto-multipunto.
- Bluetooth permite la generación de redes.

### **B.4 Aplicaciones con Java**

Bluetooth por sí mismo ofrece aplicaciones nativas de las cuales, las más comunes, permiten la transferencia de archivos entre dispositivos móviles, la comunicación de voz con dispositivos manos libres, la conectividad de equipos periféricos como teclados, impresoras, monitores, etc.; y el control de electrodomésticos como refrigeradores y hornos de microondas.

De acuerdo con [www.bluetooth.com](http://www.bluetooth.com), actualmente existe una amplia gama de productos comerciales que abarcan áreas como audio y video, dispositivos periféricos, dispositivos médicos, equipo de oficina y cómputo, dispositivos y accesorios portátiles de comunicación, aparatos de medición y juegos, entre otros.

Sin embargo, la integración de Bluetooth con otras herramientas, como lo es Java 2 Micro Edition (J2ME [29]), abre una ventana inmensa de posibilidades para la creación de aplicaciones que pueden ser hechas más a la medida de los usuarios.

El desarrollo de aplicaciones bajo la especificación de Java JSR 82 [30] con J2ME permite el acceso y control sobre dispositivos que soporten Bluetooth.

Las interfaces generadas para las aplicaciones a través de Java (APIs Application Programming Interfaces) son conocidas como Midlets [31] que, entre otras cosas, posibilitan el registro y descubrimiento de servicios, descubrimiento de dispositivos,

establecimiento de canales de comunicación, además de envío y recepción de datos (no incluyen voz).

Las aplicaciones de Java para Bluetooth utilizan dos paquetes esenciales:

- `javax.bluetooth`, que conforma las especificaciones básicas.
- `javax.obex`, a través del cual se realiza el intercambio de objetos (transferencia de datos) entre dispositivos.

Para la programación de una API, se deben contemplar cinco funciones principales, basadas en la operación del protocolo Bluetooth:

1. Inicializar los parámetros de comunicación tales como la velocidad de transmisión, el puerto de comunicación y el establecimiento del modo de descubrimiento de dispositivos.
2. Establecer la definición de los dispositivos para etiquetarlos como locales o remotos.
3. Llevar a cabo el descubrimiento de los dispositivos en la red.
4. Realizar el registro y descubrimiento de los servicios disponibles en la red.
5. Comunicación.

# Apéndice C

## Java 2 Micro Edition (J2ME)

La plataforma J2ME es una familia de especificaciones que definen varias versiones minimizadas de la plataforma Java 2; estas versiones minimizadas pueden ser usadas para programar en dispositivos electrónicos; desde teléfonos celulares, en PDAs, hasta en tarjetas inteligentes, etc. Estos dispositivos presentan en común que no disponen de abundante memoria ni mucha potencia en el procesamiento, ni tampoco necesitan de todo el soporte que brinda el J2SE, (la plataforma estándar de Java usada en sistemas de escritorio y servidor)

### C.1 ¿Qué es la plataforma J2ME?

A principios de los 90, Sun Microsystems creó un nuevo lenguaje de programación llamado Oak como parte de un proyecto de investigación para construir productos electrónicos que dependan principalmente del software. El primer prototipo para Oak fue un controlador portable llamado Star7, un pequeño dispositivo handheld con una pantalla touchscreen LCD que tenía incorporado soporte a redes inalámbricas y comunicaciones infrarrojas. Este dispositivo podría ser usado como control remoto para televisores o VCR y como guía de programas electrónicos, e incluso tenía algunas funciones que ahora son asociadas a los PDAs, como agenda de citas. El software para este tipo de dispositivos necesitaba ser extremadamente confiable y no debía hacer excesivo uso de memoria ni requerir demasiada potencia en el procesador. Oak fue desarrollado como resultado de la experiencia del equipo de desarrollo con el lenguaje C++, el cual, a pesar de tener muchas grandes características, demostró que era un lenguaje complejo y ocasionaba que los programadores comentan fácilmente errores y eso afectaba la confiabilidad del software.

Oak fue diseñado para quitar o reducir la posibilidad de que los programadores comentan errores, ¿cómo? detectando la mayoría de errores en tiempo de compilación y quitando algunas de las características del lenguaje C++ (como punteros y la administración de memoria controlada por el programador) que eran los problemas más comunes.

Desafortunadamente, el mercado para el tipo de dispositivos que el nuevo lenguaje fue creado no se desarrolló tanto como Sun Microsystems esperaba, y al final ningún dispositivo basado en Oak fue vendido a los clientes. Sin embargo, al mismo tiempo, el inicio del conocimiento público de Internet produjo un mercado para el software de navegación para Internet (los navegadores Web). En respuesta a esto, Sun Microsystems renombró el lenguaje de programación Oak a Java y lo usó para desarrollar un navegador multiplataforma llamado HotJava. También le dio la licencia de Java a Netscape, quienes lo incorporaron en su navegador que por ese entonces era el más popular en el mercado, luego fueron incorporados los Java applets.

En un par de años, las capacidades multiplataforma del lenguaje de programación Java y su potencia como plataforma de desarrollo para aplicaciones que podían ser escritas una vez y ejecutadas en diversos sistemas Windows y Unix, había despertado el interés de usuarios finales, porque vieron en ella una manera de reducir los costos del desarrollo de software.

Con el objetivo de conocer las necesidades de los experimentados desarrolladores en Windows y Motif/X-Windows para crear aplicaciones para usuarios finales sofisticados acostumbrados a usar interfaces ricas, Sun Microsystems rápidamente expandió el alcance y tamaño de la plataforma Java. Esta plataforma extendida incluyó un conjunto más complejo de librerías de interfaces de usuario que aquellos que usaran para construir applets, además con un conjunto de características de computación distribuida y seguridad mejorada.

Con el tiempo Sun Microsystems liberó la primera versión de la plataforma Java 2, había sido necesario dividirla en varias piezas. La funcionalidad principal, estimado como el mínimo soporte requerido para cualquier ambiente Java, estaba empaquetada en el Java 2 Standard Edition (J2SE).

Muchos paquetes opcionales pueden ser agregados al J2SE para satisfacer requerimientos específicos para aplicaciones particulares, como extensiones seguras de sockets que permitan el comercio electrónico. Sun Microsystems también respondió al incremento del interés de usar Java para el desarrollo a un nivel empresarial, y ambientes de servidores de aplicaciones con la plataforma Java 2 Enterprise Edition (J2EE), el cual incorpora nuevas tecnologías como servlets, Enterprise JavaBeans, JavaServer pages, etc.

Como la mayoría de software, los requerimientos de recursos de Java tienen un incremento con cada nueva versión que aparece. A pesar que Java tiene sus raíces en el software para productos electrónicos pequeños, J2SE requiere mucha más memoria y potencia en el procesador para que sea una solución viable en el mercado.

Irónicamente, mientras Sun Microsystems estaba desarrollando Java para Internet y para la programación comercial, la demanda empezó a crecer en los dispositivos pequeños e incluso en tarjetas inteligentes, retornando Java a sus raíces.

Sun Microsystems respondió a esta demanda creando varias plataformas Java con funcionalidades reducidas, cada una hecha a la medida de un segmento vertical y específico del mercado.

Estas plataformas reducidas están todas basadas en el JDK 1.1, el predecesor de la plataforma Java 2, y cada una tiene una estrategia diferente al problema de reducir la plataforma para acomodarla a los recursos disponibles. Por lo tanto, cada una de estas plataformas de funcionalidad reducida representa una solución ad-hoc al problema. Por ello es que aparece la plataforma J2ME, para reemplazar todas esas plataformas reducidas basadas en el JDK 1.1 y crear una sola solución basada en Java 2.

En conclusión, J2ME es la versión de Java orientada a los dispositivos móviles. Debido a que los dispositivos móviles tienen una potencia de cálculo baja e interfaces de usuario pobres, es necesaria una versión específica de Java destinada a estos dispositivos, ya que el resto de versiones de Java, J2SE o J2EE, no encajan dentro de este esquema. J2ME es por tanto, una versión “reducida” de J2SE.

## **C.2 Configuración**

La configuración es un mínimo grupo de APIs (Application Program Interface), útiles para desarrollar las aplicaciones destinadas a un amplio rango de dispositivos. La configuración estándar para los dispositivos inalámbricos es conocida como CLDC (*Connected Limited Device Configuration*). El CLDC proporciona un nivel mínimo de funcionalidades para desarrollar aplicaciones para un determinado conjunto de dispositivos inalámbricos. Se puede decir que CLDC es el conjunto de clases esenciales para construir aplicaciones. Hoy por hoy, sólo tenemos una configuración, pero es de

esperar que en el futuro aparezcan distintas configuraciones orientadas a determinados grupos de dispositivos. Los requisitos mínimos de hardware que contempla CLDC son:

- 160KB de memoria disponible para Java.
- Procesador de 16 bits o Consumo bajo de batería.
- Conexión a red.

Los dispositivos que claramente encajan dentro de este grupo, son los teléfonos móviles, los PDA (*Personal Digital Assintant*), los “*Pocket PC*”...

En cuanto a los requisitos de memoria, según CLDC, los 160KB se utilizan de la siguiente forma:

- 128KB de memoria no volátil para la máquina virtual Java y para las librerías del API de CLDC.
- 32KB de memoria volátil, para sistema de ejecución (*Java Runtime System*).

En cuanto a las limitaciones impuestas por CLDC, tenemos por ejemplo las operaciones en coma flotante. CLDC no proporciona soporte para matemática en coma flotante. Otra limitación es la eliminación del método *Object.finalize*. Este método es invocado cuando un objeto es eliminado de la memoria, para optimizar los recursos. También se limita el manejo de las excepciones. Es complicado definir una serie de clases de error estándar, que se ajuste a todos los dispositivos contemplados dentro de CLDC. La solución es soportar un grupo limitado de clases de error y permitir que el API específico de cada dispositivo defina su propio conjunto de errores y excepciones.

La seguridad dentro de CLDC es sencilla, sigue el famoso modelo *sandbox*. Las líneas básicas del modelo de seguridad *sandbox* en CLDC son:

- Los ficheros de clases, deben ser verificados como aplicaciones válidas.
- Sólo las APIs predefinidas dentro de CLDC están disponibles.
- No se permite cargadores de clases definidos por el usuario.
- Sólo las capacidades nativas proporcionadas por CLDC son accesibles.

### C.3 Perfiles

En la arquitectura de J2ME, por encima de la configuración, tenemos el perfil (*profile*). El perfil es un grupo más específico de APIs, desde el punto de vista del dispositivo. Es decir, la configuración se ajusta a una familia de dispositivos, y el perfil se orienta hacia un grupo determinado de dispositivos dentro de dicha familia. El perfil, añade funcionalidades adicionales a las proporcionadas por la configuración.

La especificación MIDP (*Mobile Information Device Profile*), describe un dispositivo MIDP como un dispositivo, pequeño, de recursos limitados, móvil y con una conexión “inalámbrica”.

### C.4 MIDLet

Las aplicaciones J2ME desarrolladas bajo la especificación MIDP, se denominan MIDLets. Las clases de un MIDLet, son almacenadas en *bytecodes* java, dentro de un fichero *.class*. Estas clases, deben ser verificadas antes de su “puesta en marcha”, para garantizar que no realizan ninguna operación no permitida. Este preverificación, se debe hacer debido a las limitaciones de la máquina virtual usada en estos dispositivos. Esta máquina virtual se denomina KVM. Para mantener esta máquina virtual lo más sencilla y pequeña posible, se elimina esta verificación, y se realiza antes de la entrada en producción. La preverificación se realiza después de la compilación, y el resultado es una nueva clase, lista para ser puesta en producción.

Los MIDLets, son empaquetados en ficheros “.jar”. Se requiere alguna **información extra**, para la puesta en marcha de las aplicaciones. Esta información se **almacena** en el fichero de “manifiesto”, que va incluido en el fichero “.jar” y en un **fichero descriptor**, con extensión “.jad”. Un fichero “.jar” típico, por tanto, se compondrá **de**:

- **Clases del MIDLet.**
- **Clases de soporte.**
- **Recursos (imágenes, sonidos...).**
- **Manifiesto (fichero “.mf”).**
- **Descriptor (fichero “.jad”).**

Un fichero “.jar” puede contener varios MIDLets. Esta colección de MIDLets, se suele llamar “MIDLet Suite”. Esta unión de varios MIDLets en una distribución, permite compartir recursos (imágenes, sonidos...), y por tanto optimizar los recursos del dispositivo.

# Apéndice D

## General Packet Radio Service (GPRS)

### D.1 Introducción

La red GSM prevé unos servicios de transmisión de datos desde la fase inicial (fase 1). Sin embargo, se trata de servicios con modalidad de transferencia por conmutación del circuito, es decir, donde la red, una vez establecida la conexión física de cabo a rabo entre dos usuarios, dedica los recursos propios hasta que no es solicitado expresamente el establecimiento de la conexión, independientemente del hecho de que los dos usuarios se intercambien datos durante todo el tiempo de conexión. Esta modalidad de transferencia es óptima sólo en el caso en que los dos usuarios tengan que intercambiarse una cantidad significativa de datos (transferencia de ficheros o archivos); resulta ineficiente en cuanto los datos a intercambiarse son de pequeña entidad o bien, en el caso más frecuente, el tráfico de datos es de tipo interactivo o transitorio, es decir, el tiempo de uso efectivo de los recursos de la red supone sólo una parte con respecto al tiempo total de conexión (como, por ejemplo, la navegación en Internet a través de la WorldWideWeb). Es decir, se crea el mismo problema para el GSM que para la PSTN (Public Switshed Telephone Network) hace unos años: prever una modalidad de transferencia por paquetes de datos, en la que los datos de los usuarios, contenidos en entidades de protocolo autosuficientes con indicación del remitente y del destinatario, pueden ser transportados por la propia red sin necesidad de una estrecha asociación con un circuito físico. Ya se ha dado un paso intermedio en esa dirección con el GSM de fase 2, previendo servicios con acceso a las puertas pertinentes de la red PSPDN (Public Switched Packet Data Network). Sin embargo, siempre es necesario establecer una conexión física (por conmutación del circuito) en la red de radio, incluso cuando se accede a un canal virtual de la red de paquetes. El resultado de ello es que el recurso de radio es igualmente infrautilizado y el usuario ocupa un canal de tráfico (por cuyo uso tendrá que pagar presumiblemente por el tiempo empleado), para conectarse a otra red en la cual, sin embargo, la información no viaja a un rendimiento fijo (y el transporte relativo se suele pagar en base al volumen de datos transportados).

Con el sistema GPRS (General Packet Radio Service), introducido por ETSI (European Telecommunication Standard Institute) para la fase 2+ del sistema GSM, el acceso a la red de paquetes se lleva al nivel del usuario del móvil a través de protocolos como los TCP/IP (Transmission Control Protocol), X.25, y CLNP (Connectionless Network Protocol), sin ninguna otra necesidad de utilizar conexiones intermedias por conmutación del circuito. Al contrario que el servicio de transferencia de datos con modalidad de conmutación de circuito, en el que cada conexión establecida se dedica sólo al usuario que la ha solicitado, el servicio GPRS permite la transmisión de paquetes en modalidad link by link, es decir, los paquetes de información se encaminan en fases separadas a través de los diversos nodos de soporte del servicio, denominados GSN (Gateway Support Node). Por ejemplo, una vez que un paquete ha sido transmitido por el interfaz de radio (Um), se vuelven a liberar los recursos Um, que así pueden ser utilizados por algún otro usuario y el paquete se vuelve a enviar sucesivamente de nodo a nodo hacia su destino. En los servicios GSM los recursos son gestionados según la modalidad "resource reservation", o sea, se emplean hasta el mismo momento en que la

petición de servicio no se ha llevado a término. En el GPRS, sin embargo, se adopta la técnica del context reservation, es decir, se tiende a preservar las informaciones necesarias para soportar ya sea las peticiones de servicio de forma activa o las que se encuentran momentáneamente en espera. Por tanto, los recursos de radio se ocupan, en efecto, sólo cuando hay necesidad de enviar o recibir datos. Los mismos recursos de radio de una celda se dividen así entre todas las estaciones móviles (MS), aumentando notablemente la eficacia del sistema. El servicio GPRS, por tanto, está dirigido a aplicaciones que tienen las siguientes características:

- Transmisión poco frecuente de pequeñas o grandes cantidades de datos (por ejemplo, aplicaciones interactivas).
- Transmisión intermitente de tráfico de datos bursty (por ejemplo, aplicaciones en las que el tiempo medio entre dos transacciones consecutivas es de duración superior a la duración media de una única transacción.).

## **D.2 Protocolo**

El protocolo GPRS es un protocolo de nivel tres, transparente para todas las entidades de red comprendidas entre el dispositivo móvil MT y el nodo GSN al que el móvil está, lógicamente, conectado; las entidades entre las que se establece una conexión a este nivel están, de hecho, localizadas en el dispositivo móvil MT y en el nodo GSN. Este protocolo soporta tanto el intercambio de informaciones de control como de paquetes PDP-PDU (Packet Data Protocol - Protocol Data Unit) entre el móvil y el nodo al que éste está conectado (los PDP-PDU son, de hecho, encapsulados en las tramas GPRS).

## Referencias bibliográficas

1. "HyperReal: A Hypermedia Model for Mixed Reality". Luis Romero, Nuno Correia. Computer Science Department, New University of Lisbon.
2. "The Interactive Library Building". Kaj Grønbaek, Professor Department of Computer Science, University of Aarhus.
3. Model View Controller - <http://java.sun.com/blueprints/patterns/MVC.html>  
[http://en.wikipedia.org/wiki/Model\\_view\\_controller](http://en.wikipedia.org/wiki/Model_view_controller)
4. Pervasive Computing -  
[http://www.webopedia.com/TERM/P/pervasive\\_computing.html](http://www.webopedia.com/TERM/P/pervasive_computing.html)  
[http://en.wikipedia.org/wiki/Pervasive\\_computing](http://en.wikipedia.org/wiki/Pervasive_computing)
5. Hipermedia – <http://www.webopedia.com/TERM/h/hypermedia.html>  
<http://en.wikipedia.org/wiki/Hypermedia>
6. M-Commerce - <http://en.wikipedia.org/wiki/M-commerce>
7. e-Commerce - <http://en.wikipedia.org/wiki/E-commerce>
8. "Navigational Hypertext Models For Physical Hypermedia Environments". David E. Millard, David C. De Roure, Danus T. Michaelides, Mark K. hompson, Mark J. Weal, University of Southampton, UK.
9. "Modeling Context Information in Pervasive Computing Systems". Karen Henricksen, Jadwiga Indulska, and Andry Rakotonirainy. School of Information Technology and Electrical Engineering The University of Queensland St Lucia QLD 4072 Australia.
10. Pervasive Computing: Vision and Challenges. M. Satyanarayanan School of Computer Science Carnegie Mellon University.
11. <http://www.forrester.com>
12. "Hot Topics: Ubiquitous Computing" IEEE Computer, Mark Weiser, October 1993.
13. "Computación Ubicua", Antonio Berlanga, Centro de Difusión de Tecnologías ETSIT-UPM
14. "Mobile Computing: Characteristics, Business Benefits, and the Mobile Framework", James Bryan Zimmerman, University of Maryland European Division - Bowie State INSS 690 CC, April 2, 1999.
15. "Introduction to Mobile Computing", Sandeep Jain, Computer Science Department at Purdue University.

16. “WAP Forum” - <http://www.wapforum.org>
17. “Mobile Access” - <http://www.w3.org/Mobile/>
18. “Workgroup for Electronic Data Interchange Home Page” -  
<http://www.wedi.org>
19. “Comercio electrónico” - <http://en.wikipedia.org/wiki/Ecommerce>
20. Ventas online - <http://www.amazon.com>
21. Google – <http://www.google.com>
22. eBay - <http://www.ebay.com>
23. Paypal - <http://www.paypal.com>
24. “Bluetooth más que una conexión inalámbrica”, Lourdes Velázquez Pastrana, Directora de Telecomunicaciones de la Dirección General de Servicios de Cómputo Académico (DGSCA), Universidad Nacional Autónoma de México.
25. Estandar Bluetooth - <http://www.bluetooth.com>
26. Mobile Information Device Profile MIDP - <http://java.sun.com/products/midp/>
27. IEEE - <http://www.ieee.org>
28. <http://www.ieee802.org/15/>
29. Java2 Micro Edition - <http://java.sun.com/j2me/>
30. JSR 82: Java™ APIs for Bluetooth - <http://www.jcp.org/en/jsr/detail?id=82>
31. The Magic of Midlets: A Pocket Guide to the Java 2 Platform, Micro Edition (J2ME) Technology, Martin Hardee,  
<http://java.sun.com/features/2001/06/golden.j2me.html>
32. Sistemas de Tercera Generación - <http://es.wikipedia.org/wiki/3G>
33. Sistema Móvil Universal de Telecomunicaciones -  
<http://es.wikipedia.org/wiki/UMTS>
34. Sistema de Mensajes Cortos -  
[http://en.wikipedia.org/wiki/Short\\_message\\_service](http://en.wikipedia.org/wiki/Short_message_service)
35. <http://www.mobipay.es/>
36. Nokia - <http://www.nokia.es/>
37. Sony-Ericsson - <http://www.sonyericsson.com>
38. Motorola - <http://www.motorola.com>
39. “Monedero electrónico”, Nokia -  
[http://www.nokia.es/telefonos/tecnologias/comercio\\_monedero\\_id0204.jsp](http://www.nokia.es/telefonos/tecnologias/comercio_monedero_id0204.jsp)

40. Teléfonos móviles Nokia con soporte de “Monedero Electrónico” - [http://www.nokia.es/telefonos/tecnologias/comercio\\_telefonos\\_id0204.jsp](http://www.nokia.es/telefonos/tecnologias/comercio_telefonos_id0204.jsp)
41. Díaz, P., Catenazzi, N., Aedo, I. (1996): "De la Multimedia a la Hipermedia". RA-MA Editores, Madrid. 1996.
42. Berners-Lee, T., Cailliau, R., Luotonen, A., Nielsen, H., Secret, A. (1994): "The World Wide Web". Communications of the ACM, Vol. 37, Nº 8. Agosto 1994. pp 76-82
43. DRAE (1992): Diccionario de la Real Academia de la Lengua Española. Madrid. Edición 1992
44. Parunak, H. (1991): "Don't link me in: set based hypermedia for taxonomic reasoning". Proceedings 3th ACM Conference on Hypertext and Hypermedia - Hypertext'91. ACM Press, New York, pp. 233-242.
45. “Modeling Physical Hypermedia Applications”  
Silvia Gordillo, Gustavo Rossi, Fernando Lyardet
46. Physical Hypermedia as a Formalization of Ubiquitous Information Systems  
David E. Millard, David C. De Roure, Danus T. Michaelides,  
Mark K. Thompson, Mark J. Weal, Marek Bell, Matthew Chalmers
47. K. Gronbaek, J. Kristensen, M. Eriksen: “Physical Hypermedia: Organizing Collections of Mixed Physical and Digital Material”. Proceedings of the 14th ACM International Conference of Hypertext and Hypermedia (Hypertext 2003), pp 10-19, ACM Press
48. Abowd, G.D., Brumitt, B., Shafer, S. (Eds.): Proceedings of Ubicomp 2001: Ubiquitous Computing Third International Conference Atlanta, Georgia, USA, September 30 – October 2, 2001, Proceedings. LNCS 2201. Springer Verlag.
49. “Disseminating Active Map Information to Mobile Host”. Bill N. Schilit, Columbia Science Department; Marvin M. Theimer, Xerox Palo Alto Research Center.
50. “The Stick-e Document: A Framework For Creating Context-aware Applications” Brown, P.J., , In the Proceedings of the Electronic Publishing, paginas 259-272, Laxenburg, Austria, IFIP. Septiembre 1996.
51. “Towards A Better Understanding of Context and Context-Awareness. In the Workshop on the What, Who, Where, When and How of Context-

Awareness”, Dey, A.K., and Abowd, G.D., , affiliated with the 2000 ACM Conference on Human Factors in Computer Systems (CHI 2000), The Hague, Netherlands. 1-6 Abril del 2000.

52. “All about GPS”, <http://www.trimble.com/gps/>
53. “GPS Fleet Solutions”, <http://www.gpsfleetsolutions.com/>
54. Thomas, P. Gellersen, H.W. (Eds.): Proceedings of Handheld and Ubiquitous Computing Second International Symposium, HUC 2000, Bristol, UK, September 2000. LNCS 1927, Springer Verlag.
55. Unified Modeling Language (UML) - <http://www.uml.org/>
56. Sun JAVA - <http://java.sun.com/>
57. ESRI, MapObjects Java Edition,  
<http://www.esri.com/software/mojava/index.html>
58. ESRI, ArcIMS, <http://www.esri.com/arcims>
59. Algoritmo de Búsqueda, [http://en.wikipedia.org/wiki/Search\\_algorithm](http://en.wikipedia.org/wiki/Search_algorithm)
60. Algoritmos de búsqueda por “fuerza bruta”,  
[http://en.wikipedia.org/wiki/Brute-force\\_search](http://en.wikipedia.org/wiki/Brute-force_search)
61. Teoría de Grafos, [http://en.wikipedia.org/wiki/Graph\\_theory](http://en.wikipedia.org/wiki/Graph_theory)
62. Definición - Árbol expandido mínimo  
[http://en.wikipedia.org/wiki/Minimum\\_spanning\\_tree](http://en.wikipedia.org/wiki/Minimum_spanning_tree)
63. “The HyperMuseum” Stephen Jones 387 Riley St Surry Hills NSW 2010 Australia, [sjones@culture.com.au](mailto:sjones@culture.com.au)
64. Plugin EclipseMe - <http://eclipseme.org/>
65. Sun Java Wireless Toolkit -  
<http://java.sun.com/products/sjwtoolkit/index.html>.
66. Hibernate, Framework para persistencia de objetos –  
<http://www.hibernate.org>
67. XStream, Persistencia de objetos en XML - <http://xstream.codehaus.org/>
68. XML, Extensible Markup Languaje - <http://www.w3.org/XML/>
69. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, “Design Patterns: Elements of Reusable Object-Oriented Software”.
70. HTTP, Hypertext Transfer Protocol, <http://www.w3.org/Protocols/>
71. CTI Móvil - <http://www.cti.com.ar/>
72. Movistar - <http://www.movistar.com.ar/>
73. Personal - <http://www.personal.com.ar/>



BIBLIOTECA  
FAC. DE INFORMÁTICA  
U.N.L.P.

DONACION... FACULTAD ..... 7 5  
\$..... 00125  
Fecha..... 12-3-08  
Inv. E..... Inv. B. 003108

TES  
05/25  
DIF-03108  
SALA



UNIVERSIDAD NACIONAL DE LA PLATA  
Biblioteca  
50 y 120 La Plata  
catálogo: info.unlp.edu.ar  
biblioteca@info.unlp.edu.ar



DIF-03108