



CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS  
DEL INSTITUTO POLITÉCNICO NACIONAL

Unidad Zacatenco

**Departamento de Computación**

**Plataforma para aplicaciones contextuales  
multiusuario en entornos nómadas**

Tesis que presenta

**Herón Arzáquel Anzures Reyes**

para obtener el Grado de

**Maestro en Ciencias**

**en Computación**

Directores de Tesis

**Dra. Sonia Guadalupe Mendoza Chapa**

**Dr. Adriano de Luca Pennacchia**

México, D.F.

Noviembre 2011



## Resumen

Una creciente tendencia en la forma de presentar información a los usuarios es mediante su despliegue dinámico en pantallas públicas localizadas en puntos clave dentro del ambiente, en tiendas departamentales, aeropuertos, museos, restaurantes y hasta en las paredes exteriores de edificios altos. Estos despliegues públicos normalmente difunden información contextual al público, y aunque algunos de ellos tienen soporte para interacción mediante pantallas táctiles, la misma información es siempre desplegada a todos los usuarios en un momento dado.

Por otro lado, la tecnología móvil ha sido aceptada a tal grado que la mayor parte de las personas hacen uso de teléfonos celulares o PDAs en su vida diaria. Los dispositivos móviles han incrementado tanto su número como su poder, de modo que comienza a ser común verlos como herramientas para conseguir capacidades previamente inalcanzables. Sorprendentemente se han hecho muy pocas investigaciones que combinan ambos enfoques. En el presente protocolo de tesis, se propone una nueva forma de llevar contenido a los usuarios, a través de una plataforma ubicua, colaborativa, sensible al contexto, accesible desde dispositivos móviles y con soporte para despliegues públicos replicables. Mediante esta plataforma los usuarios podrán participar a través de sus dispositivos móviles en los eventos contextuales proporcionados por los proveedores que se encuentren en el área como juegos multi-usuario y encuestas automáticas.

**Palabras clave:** cómputo ubicuo, despliegues públicos, dispositivos móviles, plataforma de juegos multi-usuario.



## Abstract

Public screens can now be found in many places, from shopping malls and airports to museums and restaurants, and even in the outside walls of tall buildings. These public screens usually convey contextual information to the public, and even though some of them support interactivity via touchscreen, the same information is always displayed to all its users regardless of their interests or preferences.

On the other hand, mobile devices in general have shown great success by substantially increasing their numbers and their power. As such it is common to start viewing them as tools to achieve previously unattainable capabilities. In this thesis we engage the idea of using mobile devices to allow multiple users to simultaneously interact with public screens. As a matter of motivation, we consider a few potential applications that range from the marketing to the entertaining industry. We discuss some key characteristics that are desired on a platform that supports this kind of interaction. Finally we present PACMEN, a platform that supports the development of applications oriented to allow multiple users to simultaneously interact with public screens using mobile devices.

**Keywords:** ubiquitous computing, public displays, mobile devices, multiplayer gaming platform.



# Agradecimientos

**Al CONACyT (Consejo Nacional de Ciencia y Tecnología):**

*Ya que sin el apoyo económico proporcionado por esta institución, la presente tesis de maestría muy probablemente no habría podido completarse.*

**Al CINVESTAV:**

*Por haberme ofrecido la oportunidad de cumplir mis sueños de formar parte de una magnífica institución y convivir con personas tan maravillosas como las que aquí se reúnen.*

**A la Dra. Sonia:**

*Por brindarme su asesoría en este trabajo de tesis. Por su dedicación y paciencia durante cada etapa del proyecto.*

**Al Dr. Adriano de Luca:**

*Por aceptar ser mi co-director de tesis y brindar su apoyo incondicional.*

**A mis padres:**

*Por su deseo inagotable e inamovible de verme ser feliz.*

**A mi hermana:**

*Por el lazo de confianza que durante esta etapa se forjó y que espero que nunca se rompa. Por escucharme, burlarse de mí y hacerme reír.*

**A mis amigos del cinves:**

*Gracias a ustedes frecuentemente olvidaba que entre mis objetivos estaba el de obtener un grado y que en lugar de eso el propósito era el de reunirnos para compartir, pues convivir con ustedes fue una experiencia más enriquecedora que cualquier asignatura que haya o hubiese podido cursar.*

**A todos los profesores del departamento de computación:**

*Lástima que no existe horario ni mente en los que quepan todos los cursos impartidos en el departamento, pues de otra forma los habría querido cursar todos. Gracias por compartir su conocimiento y experiencia.*

**A Sofy:**

*Un pilar muy importante en el departamento. Por su paciencia, amabilidad y dedicación.*



# Índice general

<b>Índice de figuras</b>	<b>ix</b>
<b>1 Introducción</b>	<b>1</b>
1.1 Contexto de Investigación . . . . .	1
1.2 Planteamiento del problema . . . . .	3
1.3 Objetivos del proyecto . . . . .	7
1.3.1 Objetivo general . . . . .	7
1.3.2 Objetivos particulares . . . . .	8
1.4 Organización de la tesis . . . . .	9
<b>2 Estado del arte</b>	<b>11</b>
2.1 Interacción con despliegues públicos . . . . .	11
2.2 Mecanismos de difusión . . . . .	17
<b>3 Análisis y diseño de una plataforma genérica</b>	<b>21</b>
3.1 Características clave . . . . .	22
3.2 Análisis de una plataforma genérica . . . . .	24
3.2.1 Alcances del proyecto . . . . .	26
3.2.2 Usuarios finales y dispositivos . . . . .	26
3.3 Diseño de una plataforma genérica . . . . .	27
3.3.1 Componentes . . . . .	27
3.3.2 Interfaz gráfica de usuario . . . . .	34

<b>4</b>	<b>La plataforma PACMEN</b>	<b>37</b>
4.1	Implementación de la plataforma PACMEN . . . . .	37
4.1.1	Arquitectura básica . . . . .	37
4.1.2	Contenido multimedia . . . . .	38
4.1.3	Servicios proporcionados por el servidor . . . . .	39
4.1.4	Descarga y ejecución de aplicaciones contextuales . . . . .	39
4.1.5	Manejo de comunicaciones . . . . .	42
4.1.6	Esquema de mensajería <i>Publish/Subscribe</i> . . . . .	42
4.1.7	Implementación del esquema <i>publish/subscribe</i> en la plataforma PACMEN . . . . .	45
4.2	Aplicacion de prueba . . . . .	48
4.2.1	Pacmen: The Game . . . . .	48
4.2.2	Resultados . . . . .	50
<b>5</b>	<b>Conclusiones y trabajo futuro</b>	<b>53</b>
5.1	Conclusiones . . . . .	54
5.1.1	Ventajas adquiridas por los desarrolladores . . . . .	54
5.1.2	Ventajas adquiridas por los usuarios . . . . .	55
5.2	Trabajo futuro . . . . .	56
	<b>Bibliografía</b>	<b>60</b>

# Índice de figuras

1.1	Dimensiones de espacio y tiempo en TCAO . . . . .	2
1.2	Contexto de investigación . . . . .	3
1.3	Mecanismo de interacción propuesto en el presente documento de tesis . . . . .	7
1.4	Organización del documento de tesis . . . . .	10
2.1	BlueBoard usa una pantalla táctil como mecanismo de interacción. . . . .	12
2.2	Dynamo: al usar teclados y ratones para interactuar con el despliegue se permite un mayor número de usuarios. . . . .	12
2.3	Ubicicero en una PDA con un lector RFID. . . . .	13
2.4	Dispositivo ViewPort del sistema Hello.Wall. . . . .	13
2.5	Ejemplo de interacción con el sistema Hello.Wall: arriba) el usuario contesta una serie de preguntas en su dispositivo ViewPort, abajo) el despliegue público reacciona ante cada respuesta recibida. . . . .	14
2.6	Escenario de ContentCascade: un usuario descarga del despliegue público información hacia su teléfono celular. . . . .	15
2.7	Mediante un teléfono celular se toma una fotografía de una etiqueta visual. Al identificar la etiqueta, el sistema regresa información relevante al teléfono móvil: a) la etiqueta visual se encuentra en un despliegue público, b) la etiqueta visual se encuentra en una superficie de papel. . . . .	16
2.8	Rotación de un elemento del despliegue público usando la cámara de un teléfono celular como medio de interacción . . . . .	17
2.9	Ejemplos de aplicaciones soportadas por la plataforma genérica de Paek et al.: a) navegador Web compartido, b) votaciones, c) mascotas virtuales. . . . .	18
2.10	Polar Defense: la interacción entre teléfonos celulares y un despliegue público se hace a través de mensajes SMS. . . . .	19
2.11	En el sistema MobiLenin se considera la opción de que el restaurante sea el responsable de distribuir la aplicación entre sus comensales. . . . .	19

2.12	En el sistema Publix el usuario debe mandar una fotografía en negro al despliegue público para adquirir la aplicación a través de la cual se llevará a cabo la interacción. . . . .	19
3.1	Un navegador Web visto como una plataforma genérica . . . . .	23
3.2	Experiencia multimedia: sonidos, animaciones e imágenes . . . . .	23
3.3	Interacción simultánea de múltiples usuarios con un despliegue público	24
3.4	Hardware adicional requerido por UbiCicero: lector de RFID [5] . . . .	25
3.5	Tres componentes principales: el servidor de PACMEN, el despliegue y uno o más clientes . . . . .	27
3.6	Los tres servicios principales del servidor de PACMEN . . . . .	28
3.7	El servicio de la aplicación contextual . . . . .	29
3.8	Servicio de comunicaciones . . . . .	30
3.9	Componentes del cliente . . . . .	31
3.10	La aplicación SWFLoader . . . . .	32
3.11	Componentes de la aplicación contextual . . . . .	33
3.12	Comunicación entre los componentes de la aplicación contextual . . . .	34
4.1	La arquitectura de red de PACMEN . . . . .	38
4.2	Los tres servicios principales del servidor, el tipo de implementación y el puerto correspondiente . . . . .	39
4.3	Cálculo del número del servidor . . . . .	40
4.4	Creación de nuevas conexiones con clientes . . . . .	45
4.5	Registro de nuevos clientes y suscripción a tópicos . . . . .	47
4.6	Captura de pantalla de Pacmen: The Game en el despliegue público . . . .	49
4.7	Captura de pantalla de Pacmen: The Game en el dispositivo del cliente	49

# Capítulo 1

## Introducción

### 1.1 Contexto de Investigación

La presente tesis de maestría se inscribe en el campo de investigación denominado Trabajo Cooperativo Asistido por Computadora (TCAC), el cual estudia tanto los aspectos sociales de las actividades individuales y colectivas, como los aspectos tecnológicos de la información en aras de construir sistemas que dan soporte a la colaboración entre personas. El interés de la presente tesis se centra en los aspectos tecnológicos de TCAC, con el fin de diseñar e implementar una plataforma que permita que un grupo de usuarios casuales interactúen entre sí a través de sus dispositivos móviles, al mismo tiempo que sus acciones se ven reflejadas en una pantalla compartida común, i.e., en un despliegue público.

Dos dimensiones son importantes en el área de TCAC para caracterizar un sistema: *espacio* y *tiempo* [1]. En la dimensión del espacio se distinguen interacciones cara a cara e interacciones remotas. En la dimensión del tiempo se observan interacciones síncronas y asíncronas. La figura 1.1 muestra los cuadrantes de TCAC de acuerdo a esta clasificación.

La plataforma propuesta se enfoca en interacciones de tipo cara a cara y síncronas, i.e., los usuarios deben estar localizados en el mismo lugar y al mismo tiempo. En esta plataforma se busca que los usuarios tengan conciencia de sus propias acciones y también de las acciones de los demás participantes a través del despliegue público. La presente propuesta también se inscribe en el dominio de investigación de Interacción Hombre-Máquina pues busca explorar una novedosa forma de interacción entre usuarios a través de dispositivos móviles y pantallas compartidas, simultáneamente.

La figura 1.2 muestra el resumen de las áreas de las ciencias computacionales que convergen en esta tesis de maestría.

		Interacción (TIEMPO)	
		Síncrona	Asíncrona
Interacción (ESPACIO)	Cara a cara	<ul style="list-style-type: none"> <li>• Colaboración espontánea</li> <li>• Reuniones formales</li> <li>• Mesas compartidas</li> <li>• Roomware</li> </ul>	<ul style="list-style-type: none"> <li>• Despliegues públicos de información</li> </ul>
	Remota	<ul style="list-style-type: none"> <li>• Audio/Video conferencia</li> <li>• Mensajero instantáneo</li> <li>• Editores multiusuario</li> </ul>	<ul style="list-style-type: none"> <li>• Correo electrónico</li> <li>• Blogging</li> <li>• Wikis</li> </ul>

Figura 1.1: Dimensiones de espacio y tiempo en TCAO

## Despliegues públicos y dispositivos móviles

Una creciente porción de la información es presentada en pantallas digitales situadas en puntos estratégicos, e.g., las pantallas de los aeropuertos que despliegan información sobre los vuelos, las carteleras digitales que muestran los horarios de las películas en algunos cines, kioskos y pizarrones de noticias digitales.

El uso de despliegues publicos digitales permite que la información sea fácilmente actualizada de forma remota y dinámica, sin embargo la mayoría de estos despliegues carecen de interactividad y los que la proveen suelen hacerlo para un usuario a la vez a través de pantallas táctiles [2]. En contraste, los dispositivos móviles, como celulares, laptops y PDAs, permiten tanto el despliegue como la retroalimentación de información, de tal manera que favorecen la comunicación, coordinación y colaboración sincronizada entre distintos usuarios.

Ambas tecnologías han sido objeto de estudio de manera independiente pero, sorprendentemente, hay poco trabajo publicado relativo a la combinación de ambos enfoques (dispositivos móviles y despliegues públicos) [3]. El estudio de la interacción entre estos dispositivos presenta una oportunidad en el desarrollo de un nuevo paradigma en interfaces de usuario en el que participantes casuales, sin previo conocimiento del sistema, pueden interactuar usando su dispositivo móvil por periodos cortos de tiempo con el servicio proporcionado por el despliegue público [4].

La presente tesis de maestría trata de cómo dotar a los despliegues públicos de interactividad multi-usuario a través de dispositivos móviles, al mismo tiempo que provee a estos últimos de contenido colaborativo sensible al contexto por medio de despliegues públicos.

Como parte del trabajo relacionado con el tema de investigación se puede men-

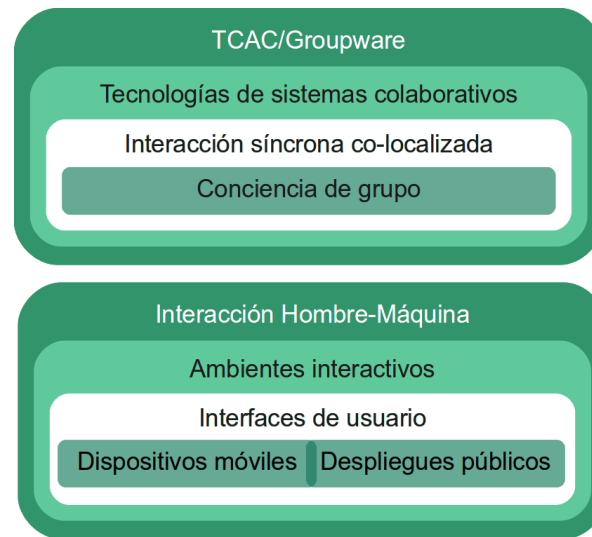


Figura 1.2: Contexto de investigación

cionar el sistema UbiCicero [5], una guía de museo multi-dispositivo sensible a la localización en la que, a través de una aplicación personalizada para dispositivos móviles, los usuarios pueden descargar información de un despliegue público, de la misma forma que se hace en el sistema Content Cascade [6], o incluso tomar fotografías de algún elemento de la pantalla pública y enviarlas a través de Bluetooth, de manera similar a la propuesta de Toye et al. [7]. En el capítulo dedicado al estado del arte se analizarán estos y otros sistemas relacionados con este trabajo de investigación.

## 1.2 Planteamiento del problema

Dos tecnologías son de interés en el tema de tesis objeto de este documento: despliegues públicos y dispositivos móviles. Ambas tecnologías han sido estudiadas de manera independiente bajo el dominio de investigación de los sistemas colaborativos, cuyo interés se centra en la forma de usar estas tecnologías para mejorar la colaboración entre los miembros de un grupo. La mayor parte de la literatura científica se enfoca en el uso de despliegues públicos como contenedores de información más que como herramientas para resolver tareas específicas [8], i.e., lo que se ha explorado hasta el momento es cómo actualizar y presentar la información que se despliega, dejando en un plano secundario a la interacción de los usuarios con la pantalla pública. Aún más, las investigaciones han favorecido la comunicación asíncrona basada en mensajes escasos e independientes. En consecuencia, muy poco trabajo se ha hecho sobre comunicaciones síncronas, las cuales son necesarias para soportar la interacción en tiempo real con los usuarios.

Por otro lado, los dispositivos móviles han presentado un gran auge, al grado que

es de esperar que las personas que potencialmente pueden interactuar con despliegues públicos también cuenten con dispositivos móviles, como teléfonos celulares. Los modelos actuales cuentan con características avanzadas que pueden propiciar la interacción entre usuarios. Ejemplos de dichas características son tecnologías inalámbricas Bluetooth y Wi-Fi, cámaras digitales y pantallas táctiles. Por lo tanto, es razonable la posibilidad de que los usuarios puedan interactuar con un despliegue público a través de sus dispositivos móviles [4].

El interés de la presente tesis de maestría es explorar esta, relativamente nueva, forma de interacción. Considere el siguiente escenario:

“Larry llega a la central de autobuses una hora antes de que salga su autobús con destino a Córdoba. Mientras espera, se acerca a un pequeño grupo de personas que se encuentran reunidas cerca de una pantalla, observa que se trata de un videojuego de carreras y decide unirse, así que saca su teléfono celular y se integra a la partida. En el despliegue público se muestra un mapa con las posiciones de todos los jugadores que están interactuando, en tanto que, cada uno de los participantes usa su celular para ver y controlar su propio vehículo en el juego. Cuando Larry escucha que empiezan a llamar a los pasajeros con destino a Veracruz, abandona la partida y se dirige al andén correspondiente. Los otros usuarios continúan jugando.

Al llegar a Córdoba se da cuenta de que está a punto de llegar a la casa de sus parientes con las manos vacías, así que se dirige al centro comercial. Casi ha encontrado todo lo que buscaba, excepto ate de membrillo. Después de pagar, decide pasar por la pantalla de atención a clientes, en la que a través de su celular puede proporcionar una lista de los productos que no encontró. Unas cuantas personas más están cerca haciendo lo mismo, mientras que algunas otras están comunicando sus quejas y sugerencias. En la pantalla de atención a clientes se muestra información relacionada, e.g., una lista con los productos que más han sido requeridos y su respectiva localización si es que el producto sí se encontraba en algún lugar del centro comercial.

Por fin llega a su destino, todos lo reciben contentos, sobre todo sus sobrinos, quienes lo estaban esperando para ir al Museo de Antropología. Al llegar al museo observa que hay una de esas aplicaciones en las que uno puede registrar las salas por las que ha pasado. Uno se acerca a las pantallas disponibles en cada sala para indicar con su celular que se ha pasado por ahí. En las pantallas se muestran las estadísticas de las salas más visitadas y las personas que casi terminan el recorrido. Después de la visita al museo todos regresan a casa. Es hora de cenar. ”

En el escenario anterior se puede observar que todas las interacciones llevadas a cabo son de tipo casual en aplicaciones contextuales, i.e., sin previo conocimiento del sistema. Se pueden identificar dos grandes factores a tomar en cuenta en el desarrollo de esta tesis: la compatibilidad y la usabilidad. La compatibilidad se refiere a la definición clara de los requisitos del sistema ya que, debido a la gran diversidad de dispositivos móviles existentes en el mercado, sería preferible utilizar herramientas multiplataforma. Por otra parte, con respecto a la usabilidad, debido a que se



pretende que el usuario pueda consumir contenido diverso y cambiante, no suena razonable pedirle que instale una aplicación diferente cada vez que quiera acceder a algo nuevo, ya que sería como tener que instalar cada página de Internet que visitamos. Sería preferible disponer de algún mecanismo que permita descargar temporalmente contenido, tal como sucede con un navegador Web.

## **Aplicaciones contextuales**

Durante el desarrollo de esta tesis, se hace un especial énfasis en un tipo de aplicaciones llamadas aplicaciones contextuales. Se les denomina así pues su funcionamiento, utilización y disponibilidad dependen de variables contextuales, en este caso, del lugar. Así pues la aplicación contenida en un despliegue público dentro de un aeropuerto no tendrá el mismo objetivo, funcionalidad, tipo de información, etc. que la contenida en un despliegue público dentro de un cine o un restaurante.

Un despliegue público en un aeropuerto probablemente muestre información sobre las llegadas y salidas de vuelos. Los usuarios, a través de sus dispositivos móviles, podrían solicitar y recibir información más detallada sobre los vuelos disponibles, comprar boletos, consultar promociones, etc.

En un cine, un despliegue público podría mostrar información general acerca de la cartelera disponible y así, mediante sus dispositivos móviles, los usuarios podrían consultar la sinopsis de alguna película, calificar o recomendar películas que hayan visto o participar en juegos y promociones por descuentos, boletos gratis o artículos de dulcería.

Un despliegue público en un restaurante podría mostrar el menú del día, recomendaciones y promociones. Los usuarios podrían utilizar sus dispositivos móviles para interactuar con el despliegue público y recibir información más detallada sobre los platillos ofrecidos (como los ingredientes, el modo de preparación o contenido nutricional) y participar votando entre varias opciones los platillos que les gustaría estuvieran disponibles la siguiente semana.

En todos los casos que se acaban de mostrar existe un despliegue público y usuarios interactuando con él por medio de dispositivos móviles, sin embargo las aplicaciones son completamente diferentes, pues dependen enteramente del contexto de utilización. El principal interés de esta tesis de maestría es la especificación, el diseño y desarrollo de una plataforma genérica que facilite por un lado la implementación de este tipo de aplicaciones contextuales y, por otro lado, que permita a los usuarios utilizar una sola aplicación instalada en sus dispositivos móviles para interactuar con todos los despliegues públicos que adopten la plataforma, liberándolos así de la carga de averiguar si su dispositivo es compatible, de instalar una nueva aplicación cada vez que interactúen con un despliegue público diferente y de manejar los riesgos que dicha instalación implicaría.

Y sobre todo se tiene el profundo interés de inundar a la comunidad de estas nuevas ideas y posibilidades. Creemos que existe un gran potencial en este camino,

y aunque muchas veces los proyectos y plataformas como la que aquí se desarrolla, por una u otra razón no tienen un impacto sobresaliente y mueren tarde o temprano, las ideas persisten y lograr que algunos ojos volteen a ver este horizonte es pues, la verdadera razón de esta tesis de maestría.

## 1.3 Objetivos del proyecto

A continuación se detallan los objetivos de la presente tesis, de los cuáles se distingue un objetivo general y varios objetivos particulares.

### 1.3.1 Objetivo general

Construir una plataforma compuesta de librerías que permita el desarrollo y la ejecución de aplicaciones multi-usuario para dispositivos móviles, con soporte de conciencia de grupo a través de un despliegue público. Estas aplicaciones harán uso de las bondades de nuevas tecnologías para dar soporte a comunicaciones síncronas y contenido enriquecido.

El método de comunicación de esta plataforma debe ser síncrono y multi-usuario, además de permitir la descarga de contenido atractivo e interactivo en el sitio de uso (cf. Figura 1.3).



Figura 1.3: Mecanismo de interacción propuesto en el presente documento de tesis

Mediante esta plataforma se pretende dar vida a nuevas e interesantes aplicaciones o enriquecer la experiencia de algunas ya existentes, como es el caso de:

- **Cine interactivo:** consiste de un despliegue público donde se proyecta una película; en determinados momentos cruciales, la película se pausa con el objetivo de realizar una votación entre los espectadores para determinar, por ejemplo, la siguiente decisión del personaje principal. A través de sus dispositivos móviles los usuarios votan por la opción que más les interese, de manera que la película va a continuar su desarrollo en función de la opción que haya adquirido más votos.

- **Videojuegos colaborativos contextuales:** se trata de videojuegos multijugador en donde cada usuario controla a su personaje a través de su dispositivo móvil. El despliegue público además de funcionar como servidor provee conciencia de grupo a través de diferentes mecanismos, e.g., un mapa a escala de todas las áreas (vista radar) o algunas estadísticas sobre los jugadores y equipos.
- **Encuestas automáticas:** encuestas de todo tipo podrían efectuarse de una forma atractiva y automática. Por ejemplo, en un restaurante podría encuestarse sobre las preferencias, opiniones y sugerencias de los comensales. Una tienda departamental podría recibir comentarios de los usuarios sobre artículos que tenían la intención de comprar pero que no encontraron.
- **Colección de estampillas virtuales por visitar algunos lugares:** las líneas aéreas o centros turísticos podrían instalar una red de estampillas virtuales, las cuales serían otorgadas a los visitantes, quienes pueden identificarse por medio de su dispositivo móvil al comunicarse con los despliegues públicos. Esta información podría entonces consultarse por Internet y compartirse en redes sociales.

### 1.3.2 Objetivos particulares

- Analizar la literatura científica, con el fin de conocer un panorama amplio sobre los trabajos relacionados con despliegues públicos y su interacción con dispositivos móviles.
- Hacer un estudio de las características de diferentes dispositivos móviles (modelos de celulares, consolas portátiles, laptops, PDAs) con el fin de determinar el tipo de dispositivos al que el proyecto estará dirigido tomando en cuenta los siguientes aspectos:
  - Tipo de comunicación a usar entre el despliegue público y los dispositivos móviles (e.g., Bluetooth o Wi-Fi).
  - Plataforma de desarrollo a utilizar (e.g., Java, Adobe Air, Flash Player).
- Precisar el mecanismo de distribución del contenido. Se tienen dos opciones: que las aplicaciones sea instalables o que sean interpretadas. La primera opción es poco flexible para interacciones casuales, pero se pueden conseguir aplicaciones más rápidas y finas que con la segunda opción, cuya ventaja es que resulta más general y puede dar soporte a un mayor número de aplicaciones.
- Definir e implementar la plataforma en términos de módulos, interfaz gráfica de usuario y mecanismos de entrada/salida.
- Desarrollar una aplicación de ejemplo que valide la funcionalidad de la plataforma propuesta.

## 1.4 Organización de la tesis

La presente tesis está estructurada en cinco capítulos (cf. Figura 1.4).

En éste, el capítulo 1, se presenta el contexto de investigación, el planteamiento del problema y objetivos del proyecto de investigación.

En el capítulo 2 se describe el estado del arte, mostrando un panorama amplio de aplicaciones y sistemas que hacen uso de la combinación de tecnologías de dispositivos móviles con despliegues públicos. Se describen sus características, funcionamiento general y limitaciones. A partir de estas características se destacan, mediante un estudio comparativo, las principales ventajas y desventajas de los sistemas analizados.

En el capítulo 3 se describe el análisis y el diseño de la plataforma. En cuanto al análisis se definen: 1) los requerimientos del proyecto, 2) el grupo de personas y los dispositivos a los que está dirigido, 3) las herramientas necesarias para el desarrollo y 4) los alcances del proyecto de investigación. En cuanto al diseño se define la arquitectura general de la plataforma y el modelo especificando: 1) la cantidad de módulos, su funcionamiento e interacción, 2) la interfaz gráfica de usuario, así como los mecanismos de entrada y salida y finalmente 3) los agentes de comunicación y transmisión.

En el capítulo 4 se describe la implementación de la plataforma, así como la implementación de la aplicación de ejemplo a través de la cual se evalúa las siguientes características: 1) la usabilidad y el desempeño de la plataforma, y 2) la compatibilidad y estabilidad. También se presentan las pruebas y resultados de la plataforma.

Finalmente, en el capítulo 5 se presenta las conclusiones, contribuciones y perspectivas futuras de este trabajo de investigación.

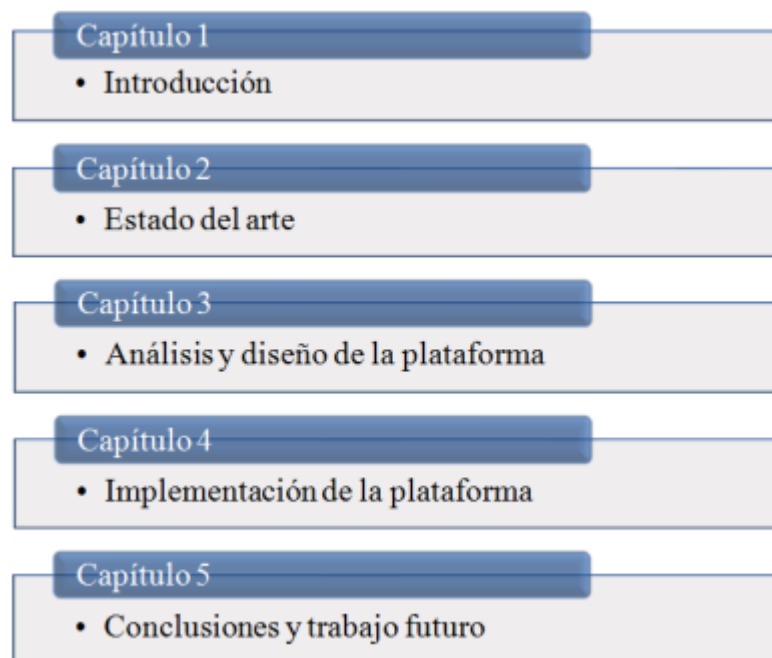


Figura 1.4: Organización del documento de tesis

# Capítulo 2

## Estado del arte

Los despliegues públicos pueden considerarse ubicuos porque son ampliamente usados para mostrar información de interés general para los usuarios. Sin embargo, varios autores han explorado el potencial de estos dispositivos para ofrecer otros mecanismos de interacción más dinámicos y enriquecidos. En este capítulo, se analizan algunos trabajos relacionados y se enfatizan los siguientes aspectos: 1) la manera en la que se lleva a cabo la interacción entre usuarios y despliegues públicos (sección 2.1) y 2) la forma en que se ponen a disposición del público las aplicaciones que aprovechan esta interacción (sección 2.2).

### 2.1 Interacción con despliegues públicos

Varios mecanismos se han propuesto para lograr una interacción más activa entre usuarios y despliegues públicos.

Una de las soluciones más intuitivas es el uso de pantallas táctiles para proporcionar a los usuarios un control sobre el contenido que desean visualizar o los servicios que desean acceder.

Community Wall [9] utiliza un despliegue táctil en el que el usuario puede tocar alguno de los elementos de su interés para obtener información detallada relacionada. Así mismo permite, entre otras acciones, comentar sobre el contenido o recibir una copia de la información, ya sea de manera impresa o vía correo electrónico.

BlueBoard [10] es otra aplicación que proporciona interacción a través de una pantalla táctil. Uno de los escenarios de uso de BlueBoard es el de una persona que camina cerca de la pantalla, se detiene a revisar su calendario y se aleja (cf. Figura 2.1).

El uso de pantallas táctiles requiere la instalación de despliegues especializados, lo que muchas veces se traduce en costos elevados. Un inconveniente mayor se refiere al número de individuos que pueden interactuar simultáneamente, ya que su uso sólo es



Figura 2.1: BlueBoard usa una pantalla táctil como mecanismo de interacción.

efectivo en aplicaciones mono-usuario o con un número reducido de usuarios<sup>1</sup>, debido a que se necesita que las personas se encuentren muy cerca del despliegue y que no interfieran unas con otras.

Para lograr que interactúe un mayor número de personas se han propuesto otros mecanismos. Brignull et al. proponen el sistema Dynamo que soporta interacción multi-usuario a través de teclados y ratones inalámbricos [11]. Con esta propuesta, los usuarios no necesitan estar muy cerca del despliegue, por lo que se permite un mayor número de participantes (cf. Figura 2.2).



Figura 2.2: Dynamo: al usar teclados y ratones para interactuar con el despliegue se permite un mayor número de usuarios.

Una propuesta parecida a la anterior es la de Ghiani et al. [5], en la cual la interacción multi-usuario se soporta a través de PDAs equipados con lectores RFID de largo alcance, en lugar de ratones y teclados inalámbricos (cf. Figura 2.3). El sistema en donde se aplica esta forma de interacción es UbiCicero, una guía de museo sensible a la localización. Un ejemplo de la interacción soportada en este sistema es el siguiente: los visitantes pueden seleccionar una obra de arte en su PDA, ante lo

<sup>1</sup>En el caso de pantallas multi-táctiles.



cual el despliegue público muestra, entre otra información relevante, la ruta más corta para llegar a ella. UbiCicero también soporta la interacción de múltiples usuarios, a través de juegos grupales: los visitantes se dividen en equipos y se les proporcionan actividades sencillas como responder preguntas de opción múltiple. Por cada acierto se asignan puntos, los cuales se visualizan en el despliegue público.



Figura 2.3: UbiCicero en una PDA con un lector RFID.

Streitz et al. proponen el dispositivo ViewPort para interactuar con el despliegue público del sistema Hello.Wall [12]. ViewPort está basado en un dispositivo PocketPC con adaptador para WLAN y pantalla táctil (cf. Figura 2.4). En su trabajo de investigación, los autores describen un esquema de interacción simultánea usando múltiples dispositivos ViewPort frente al despliegue público.

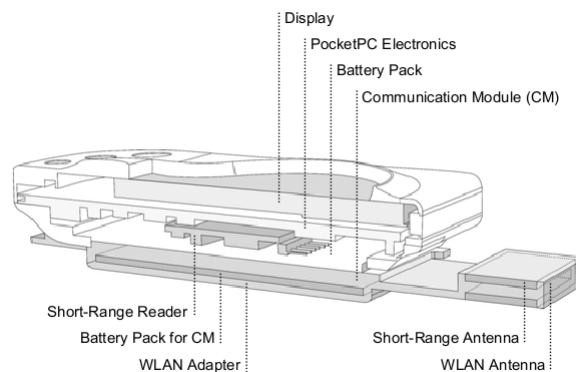


Figura 2.4: Dispositivo ViewPort del sistema Hello.Wall.

Un ejemplo de escenario de uso, proporcionado por los autores, es el uso del sistema Hello.Wall como medio para contestar encuestas o participar en votaciones: 1)

si nadie está cerca del despliegue público, sólo se muestra información sobre el estado general de la encuesta (o votación), 2) cuando un usuario es sentido, se despliegan en la pantalla del dispositivo ViewPort las preguntas que componen la encuesta para que el puedan ser contestadas, 3) cada vez que el usuario responde una pregunta, el despliegue público muestra una secuencia de imágenes para indicar que se ha recibido su respuesta, 4) si el usuario termina de responder la encuesta, se muestran imágenes en el despliegue público que le dan a entender que Hello.Wall agradece su participación (cf. Figura 2.5).

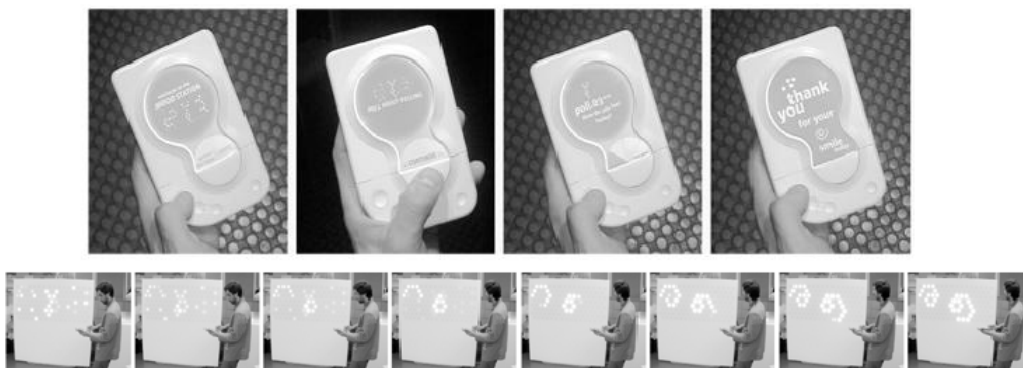


Figura 2.5: Ejemplo de interacción con el sistema Hello.Wall: arriba) el usuario contesta una serie de preguntas en su dispositivo ViewPort, abajo) el despliegue público reacciona ante cada respuesta recibida.

Un segundo escenario de uso es el de un juego de tipo memorama, en el que el usuario “voltea”, una a una, celdas individuales del despliegue público para tratar de encontrar la imagen que coincide con aquella desplegada en su dispositivo ViewPort. Múltiples usuarios pueden jugar simultáneamente.

BlueBoard, UbiCicero y Hello.Wall funcionan en ambientes controlados en donde es posible distribuir a los usuarios los dispositivos a través de los cuales se lleva a cabo la interacción con los despliegues públicos: ratones y teclados inalámbricos, PDAs equipados con lectores RFID y dispositivos creados para ese propósito en específico, como es el caso de ViewPort. En dichos ambientes se necesita conocer el número de usuarios que potencialmente pueden interactuar con el despliegue público para proporcionarles tantos dispositivos como sean necesarios, así como contar con un mecanismo para controlar las posibles pérdidas de los mismos.

Sin embargo, en ambientes nómadas no existe control sobre los usuarios potenciales. En estos casos conviene explotar el dispositivo móvil más común entre ellos, e.g., teléfono celular, para que cada usuario pueda interactuar con su propio dispositivo. Como se describe a continuación, la literatura científica reporta varias propuestas encaminadas al uso de teléfonos móviles como medio de interacción con despliegues públicos.

Raj et al. proponen el *framework* ContentCascade [6] que permite a los usuarios descargar contenido desde un despliegue público hacia su teléfono celular mediante Bluetooth. El escenario emblemático de esta aplicación consiste en permitir a los usuarios descargar *trailers* y sinápsis de películas desde *posters* digitales en cines hacia su dispositivo personal (cf. Figura 2.6).



Figura 2.6: Escenario de ContentCascade: un usuario descarga del despliegue público información hacia su teléfono celular.

Toye et al. [7] proponen también el uso de teléfonos celulares equipados con cámara fotográfica. Los usuarios toman fotografías de etiquetas visuales sobre tópicos que se encuentran en el despliegue público. Estas fotografías son enviadas después al servidor por medio de Bluetooth, el cual las procesa, las reconoce y devuelve como respuesta información más detallada sobre el tópico seleccionado. Dicha información se envía posteriormente al teléfono móvil (Figura 2.7). El mecanismo descrito es útil no solo para despliegues públicos, sino prácticamente para cualquier superficie en la que se encuentren localizadas las etiquetas visuales.

Jeon et al. describen también un mecanismo de interacción entre despliegues públicos y teléfonos celulares que utiliza la cámara digital integrada en dichos teléfonos [13]. Uno de los usos de este mecanismo es, por ejemplo, rotar un elemento del despliegue público, para lo cual se enfoca la cámara sobre el elemento a rotar, se presiona el botón de seleccionar y se hace el movimiento apropiado del teléfono celular (cf. Figura 2.8).

Paek et al. [2] proponen una plataforma genérica para dar soporte a la interacción de teléfonos móviles con despliegues públicos, a través de mecanismos de interacción asíncrona: mensajes de texto SMS y correos electrónicos a direcciones específicas. Esta plataforma toma en cuenta la participación de varios usuarios de manera simultánea. Distintos tipos de aplicaciones son soportados por esta plataforma :



Figura 2.7: Mediante un teléfono celular se toma una fotografía de una etiqueta visual. Al identificar la etiqueta, el sistema regresa información relevante al teléfono móvil: a) la etiqueta visual se encuentra en un despliegue público, b) la etiqueta visual se encuentra en una superficie de papel.

- *Aplicaciones jukebox*: donde el despliegue público se aprovecha como un recurso que debe ser compartido entre usuarios, que acceden a él por turnos. Por ejemplo, en el despliegue público puede visualizarse un navegador Web compartido, en el que los usuarios indican con su teléfono móvil la acción que quieren realizar y esperan a que su petición sea atendida (cf. Figura 2.9.a).
- *Herramientas colaborativas*: permiten a múltiples usuarios participar en una meta conjunta, e.g., encuestas, votaciones y juegos grupales (cf. Figura 2.9.b).
- *Herramientas para propiciar la interacción entre usuarios*: crean formas para iniciar la comunicación entre desconocidos, e.g.,: un ambiente de mascotas virtuales que interactúan conjuntamente en el despliegue público (cf. Figura 2.9.c).

Por otra parte, Finke et al. investigan el impacto social de la interacción entre despliegues públicos y teléfonos celulares. En su trabajo mencionan que la unión de estas dos tecnologías constituye una considerable promesa para aplicaciones cuya finalidad es el entretenimiento compartido [14]. Como parte de su investigación, desarrollaron el juego interactivo *Polar Defense* en el que varios usuarios interactúan con un despliegue público a través de mensajes SMS. Dentro de sus resultados, destacan las ventajas de usar mensajes SMS como mecanismo de interacción: son ampliamente usados y conocidos por el público en general, son confiables y anónimos, sin embargo presentan desventajas como el costo de cada mensaje y la alta latencia por lo que sólo pueden ser usados en aplicaciones específicas. *Polar Defense*, por ejemplo, es un juego en el que el usuario debe defenderse de un ataque de osos por medio de seis torres. Las coordenadas de las torres se mandan vía SMS. En el despliegue público se muestra el juego en acción y los resultados con base en las coordenadas recibidas. Aunque varios usuarios pueden mandar sus sugerencias de coordenadas para las torres, se resuelve un mensaje a la vez. En el despliegue público se mantienen las mejores puntuaciones (cf. Figura 2.10).

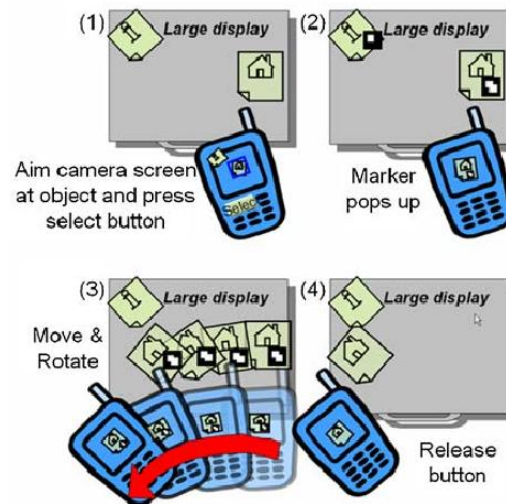


Figura 2.8: Rotación de un elemento del despliegue público usando la cámara de un teléfono celular como medio de interacción

## Conclusiones

En el presente trabajo de tesis se apoya la idea de que la interacción con despliegues públicos sea a través de dispositivos móviles de fácil acceso para los usuarios: teléfonos celulares, PDAs y computadoras portátiles. Al igual que en el trabajo de Paek et al., en el presente trabajo también se propone una plataforma genérica. Sin embargo, se hace uso de Wi-Fi como medio de comunicación, en vez de mensajes asíncronos de texto, con el fin de lograr mayor flexibilidad, mejor tiempo de respuesta y aplicaciones enriquecidas.

## 2.2 Mecanismos de difusión

Una vez que se ha considerado usar dispositivos que están al alcance de los usuarios, el siguiente problema que se tiene que tomar en cuenta es cómo poner a disposición del público las aplicaciones desarrolladas.

Una opción es que dichas aplicaciones se distribuyan en el ambiente en el que han de ser utilizadas. MobiLenin [15] es un sistema en el que los usuarios pueden votar a través de sus dispositivos móviles por el video que quieren visualizar en el despliegue público. El requisito es que deben instalar la aplicación correspondiente en su celular. MobiLenin está diseñado para usarse en restaurantes y *clubs*. La idea de sus autores es que la aplicación sea distribuida directamente por el restaurante en donde es utilizada y que se incentive a la audiencia a adquirir la aplicación con la posibilidad de ganar premios al participar en la elección de los videos a mostrar (cf. Figura 2.11). Esta



Figura 2.9: Ejemplos de aplicaciones soportadas por la plataforma genérica de Paek et al.: a) navegador Web compartido, b) votaciones, c) mascotas virtuales.

opción no es viable para ambientes nómadas, puesto que las interacciones en estos ambientes son casuales, mas no localizadas en algún punto específico.

Otra propuesta es la proporcionada por Ventura et al. en su sistema Publix [16], en el que los usuarios interactúan con anuncios publicitarios a través de sus teléfonos celulares para obtener información del producto anunciado, promociones, imágenes, juegos, etc. Los autores, al darse cuenta de que las camaras digitales son muy comunes en los teléfonos móviles, idearon una forma de distribuir la aplicación en la que el usuario debe tomar una foto completamente en negro y enviarla vía Bluetooth, al despliegue público, el cual como respuesta proporciona el archivo de la aplicación listo para ser instalado en el teléfono celular del usuario (cf. Figura 2.12).

## Conclusiones

La desventaja de las propuestas mencionadas es que se obliga al usuario a instalar cada una de las aplicaciones disponibles. El usuario podría sentirse incómodo con esta situación ante la gran cantidad de virus y códigos maliciosos existentes. Por otro lado, las interacciones con los despliegues públicos son casuales, así que es inadecuado obligar al usuario a instalar aplicaciones que sólo serán utilizadas brevemente.

En este trabajo se propone que el usuario cuente con una sola aplicación instalada en su teléfono celular, a partir de la cual sea posible interactuar con todas las diversas aplicaciones disponibles en entornos nómadas. La analogía es la de un navegador Web, a través del cual el usuario puede entrar a todas las páginas disponibles en Internet.





Figura 2.10: Polar Defense: la interacción entre teléfonos celulares y un despliegue público se hace a través de mensajes SMS.

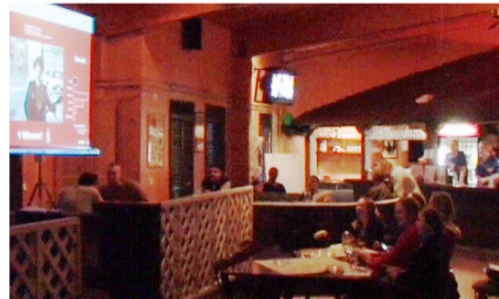


Figura 2.11: En el sistema MobiLenin se considera la opción de que el restaurante sea el responsable de distribuir la aplicación entre sus comensales.



Figura 2.12: En el sistema Publix el usuario debe mandar una fotografía en negro al despliegue público para adquirir la aplicación a través de la cual se llevará a cabo la interacción.





# Capítulo 3

## Análisis y diseño de una plataforma genérica

Del estado del arte, podemos formarnos una noción de las posibilidades que se presentan al hacer uso de dispositivos móviles para interactuar con despliegues públicos. Esta tesis trata de sintetizar las principales características deseadas de una plataforma genérica, que permita el desarrollo de aplicaciones que aprovechen la interacción entre dispositivos móviles y despliegues públicos. En este capítulo, se discute el análisis y diseño de una plataforma genérica a la que llamamos PACMEN (Plataforma para Aplicaciones Contextuales multi-usuario en Entornos Nómadas) la cual trata de cumplir con algunas características que consideramos fundamentales para propiciar un entorno amigable e intuitivo para los usuarios finales. Estas características fundamentales son también enlistadas y detalladas más adelante en este capítulo.

Existen varios aspectos a tomar en cuenta durante el diseño de una plataforma de este estilo ¿Qué tipo de dispositivos debe soportar? ¿Qué tipo de tecnología para la comunicación debe utilizar? ¿De qué manera los dispositivos descubren despliegues públicos a los cuales conectarse? ¿Cuántos usuarios deberían poder interactuar de manera simultánea? Y aunque la respuesta a la mayoría de estas preguntas puede ser un elaborado “Pues depende...”, creemos que existen ciertas características fundamentales, que le darían a la plataforma la usabilidad mínima necesaria para tener una oportunidad de ser aceptada por la comunidad de usuarios en general.

Sin embargo, dado que el tipo de aplicaciones que podrían hacer uso de las interacciones entre dispositivos móviles y despliegues públicos es muy variado, es difícil imaginar que una sola plataforma genérica sea la más adecuada y eficiente para todos los propósitos. Este hecho se puede ver como la misma razón que explica la existencia de distintos lenguajes de programación de propósito general, pues algunos tienen ventajas sobre otros dependiendo de las prioridades del proyecto o las habilidades del desarrollador. Es por ésto que, cuando hablamos de características deseadas en una plataforma genérica, es necesario especificar las prioridades que hacen que estas características sean importantes.

En esta tesis, decidimos dar prioridad a los rubros de mercadotecnia, entretenimiento y difusión, i.e., juegos multi-jugador, encuestas públicas, promoción de productos y servicios, etc. Ésto no quiere decir que aplicaciones pertenecientes a otros dominios no puedan desarrollarse exitosamente en la plataforma PACMEN. Tampoco significa que PACMEN sea la única o mejor opción para aplicaciones en los rubros antes mencionados. Lo que quiere decir es que, cada vez que se nos presentó una disyuntiva ya sea tecnológica, algorítmica o conceptual durante el diseño y desarrollo de dicha plataforma, consideramos estas prioridades para tomar una decisión.

## 3.1 Características clave

A continuación, presentamos las características clave que consideramos recomendables para obtener una buena usabilidad y un desempeño satisfactorio en una plataforma de este tipo.

### Genérica

Una importante característica concierne a la capacidad de ejecutar una gran variedad de aplicaciones sin necesidad de instalarlas o compilarlas, tal como lo hace un navegador Web (cf. Figura 3.1) que puede desplegar una gran variedad de páginas sin necesidad de instalar cada una de ellas previamente. De esta forma, cuando el usuario encuentre un despliegue público con el que quisiera interactuar, sólo tendrá que establecer comunicación con el despliegue y descargar la aplicación ofrecida por este, la cual se ejecutará automáticamente sin necesidad de ser instalada y mucho menos de ser compilada.

### Responsiva

En el estado actual de la tecnología, es cada vez más normal que las personas busquen una alta velocidad de respuesta en los sistemas y dispositivos. Aunque sigue siendo un aspecto relativo la cantidad de tiempo que los usuarios consideren aceptable esperar, algo es seguro... mientras menos tiempo esperen, es mejor. Dependiendo de la aplicación en cuestión, los tiempos de espera pueden variar considerablemente: mientras que en un videojuego multi-jugador una espera de 2 segundos entre la acción de un jugador y la reacción del sistema puede considerarse una latencia inaceptable, la descarga de un artículo completo con imágenes puede ser una acción que tiene una tolerancia mayor.

### Multimedia

El texto es bueno, pero una imagen dice más que mil palabras. Para llevar al usuario una experiencia atractiva y enriquecida, es necesario un conjunto de soportes como sonidos, animaciones e imágenes (cf. Figura 3.2).

### Multi-usuario

No sólo es necesario que múltiples usuarios puedan interactuar con el despliegue público de manera simultánea, sino que también puedan hacerlo tranquila

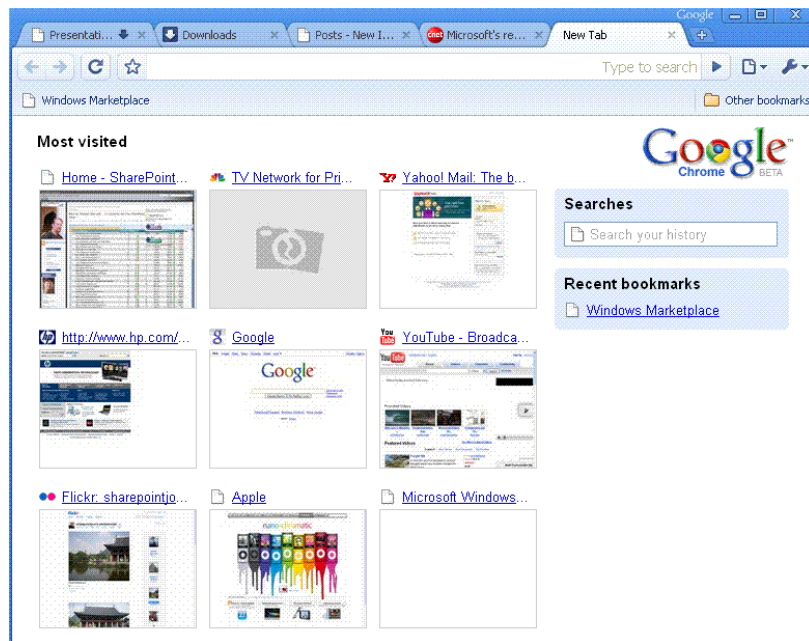


Figura 3.1: Un navegador Web visto como una plataforma genérica



Figura 3.2: Experiencia multimedia: sonidos, animaciones e imágenes



Figura 3.3: Interacción simultánea de múltiples usuarios con un despliegue público

y fluidamente, sin preocuparse por interrumpir o ser interrumpidos por otros usuarios. Para lograr este fin, los usuarios deben ser capaces de comenzar y terminar la comunicación con el despliegue en cualquier momento, sin que estas acciones afecten las actividades de otros usuarios del sistema (cf. Figura 3.3).

### **Independiente**

Si una plataforma requiere software adicional o, peor aun, hardware adicional (cf. Figura 3.4), menos usuarios estarán interesados en darle una oportunidad, en consecuencia la plataforma podría estar destinada a fracasar y ser olvidada. Por esta razón es importante hacer un ejercicio de valoración del tipo de herramientas que resulte conveniente utilizar.

## **3.2 Análisis de una plataforma genérica**

Tomando en cuenta las características descritas en la sección 3.1, se procede a analizar la factibilidad de una plataforma con esas propiedades:

- Que facilite la interacción con despliegues públicos a través de dispositivos móviles.
- Que tenga capacidades multimedia para hacer la interacción más amigable, accesible y atractiva.
- Que pueda ser utilizada para una gran variedad de propósitos y tipos de aplicaciones contextuales.
- Que sea responsiva, pues en esta época los usuarios requieren tiempos de respuesta cortos y algunas aplicaciones como los videojuegos no son usables ni disfrutables en lo absoluto, si el sistema muestra una latencia considerable.
- Que no requiera hardware adicional.



Figura 3.4: Hardware adicional requerido por UbiCicero: lector de RFID [5]

### 3.2.1 Alcances del proyecto

Tomando en cuenta que el objetivo final de esta tesis es permitir la interacción entre dispositivos móviles y despliegues públicos, para así abrir paso a un nuevo mundo de posibilidades respecto al uso de aplicaciones contextuales en aeropuertos, centros comerciales, cines, restaurantes, etc., es necesario definir los alcances del proyecto, pues es de imaginarse que, en un área de investigación como esta, las oportunidades de avance son muchas en múltiples aspectos, pero no es posible cubrirlas todas en un primer intento.

Este trabajo tiene sus bases en los resultados obtenidos de otros proyectos en el estado del arte y fundamenta su desarrollo en prácticas que permitan una fuerte estructura básica, dejando gran parte de los detalles específicos a los desarrolladores de las aplicaciones contextuales para que ellos decidan, con base en sus preferencias o necesidades, la estructura que más les convenga utilizar encima de la plataforma.

La plataforma provee la funcionalidad necesaria para llevar a cabo una interacción fluida entre los dispositivos móviles y el despliegue público, sin embargo no se encarga de hacer ningún control acerca de éste, cualquier tipo de filtro que se desee imponer sobre la comunicación de los usuarios deberá ser desarrollada directamente sobre la aplicación contextual; lo mismo sucede con sistemas criptográficos que se deseen implementar para el manejo de comunicaciones seguras.

Tampoco se proveen herramientas para el manejo de información sensible por lo que, en caso de ser necesario, los esquemas tendrán que ser implementados directamente sobre la aplicación contextual.

### 3.2.2 Usuarios finales y dispositivos

Los usuarios finales se consideran clientes casuales de las prestaciones que otorga el proveedor de servicios, el cual puede ser cualquier entidad como una línea aérea, un restaurante, una cadena de cine o una empresa de mercadotecnia y publicidad que utilice un despliegue público para servir una aplicación contextual a sus clientes. Por lo tanto, se considera que los usuarios no necesariamente tienen conocimiento acerca del funcionamiento de la aplicación, de los servicios provistos o del uso de las características especiales de sus dispositivos móviles. Se aconseja pues a los desarrolladores, que utilicen esta plataforma para construir sus aplicaciones, que tomen en cuenta esos puntos e implementen la funcionalidad acorde a ello.

No se debe esperar que los usuarios finales tengan conocimiento acerca del momento en que deben iniciar o terminar la comunicación ni de cada una de las implicaciones de sus acciones, así como tampoco de que sean siempre bien intencionados. Así pues, se recomienda precaución a la hora de manejar la entrada de los usuarios, ya que quizá se requieran filtros de spam, de lenguaje soez, de ataques de negación de servicio, entre otros.

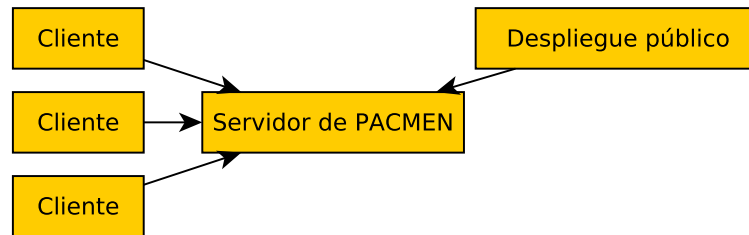


Figura 3.5: Tres componentes principales: el servidor de PACMEN, el despliegue y uno o más clientes

Los dispositivos móviles para los cuales la plataforma está orientada son todos aquellos que tengan capacidades *touchscreen*, que soporten comunicaciones sobre Wi-Fi y que puedan ejecutar Flash Player 10. Cualquier otra característica no debe darse por hecho, e.g., uno no debe esperar que el dispositivo tenga teclado o botones direccionales, sin embargo se recomienda que las aplicaciones den soporte adicional para este tipo interfaces en caso de estar disponibles en el dispositivo del usuario.

### 3.3 Diseño de una plataforma genérica

Existen varias aplicaciones que pueden aprovechar el uso de dispositivos móviles para la interacción con despliegues públicos, algunas de las cuales ya han sido revisadas en el capítulo del estado del arte. Lo que busca esta tesis es la creación de una plataforma genérica, la cuál pueda ser usada para implementar distintas aplicaciones contextuales sin tener que desarrollar cada una desde cero. A continuación, se detalla el diseño de esta plataforma.

#### 3.3.1 Componentes

La plataforma PACMEN se divide en varios componentes abstractos, cada uno con una funcionalidad y un propósito específicos, que van desde el componente encargado de capturar los eventos del usuario, hasta el componente responsable del manejo de permisos para la comunicación a través de sockets. A continuación, presentamos una serie de diagramas a distintos niveles de granularidad sobre el funcionamiento de la plataforma con base en sus componentes.

Al nivel más alto se identifican tres componentes: el servidor de PACMEN, el despliegue público y uno o más clientes, como se ve en la Figura 3.5.

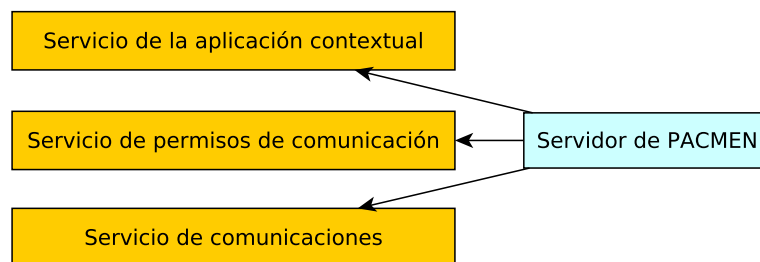


Figura 3.6: Los tres servicios principales del servidor de PACMEN

### Servidor de PACMEN

Desglosamos primero al servidor de PACMEN en sus componentes primarios. Este servidor ofrece tres servicios independientes que se encargan de distintas funcionalidades cada uno. Se considera que son independientes pues cada uno se encuentra en un contenedor distinto y no existe comunicación directa entre ellos, i.e., ninguno de estos servicios sabe de la existencia de los otros.

Como se observa en la Figura 4.2, estos tres servicios son: el servicio de la aplicación contextual, el servicio de permisos de comunicación y el servicio de comunicaciones, los cuales se describen a continuación.

- *Servicio de la aplicación contextual*: como se describió anteriormente, para que el usuario comience a utilizar la aplicación contextual en aras de interactuar con el despliegue público, el usuario debe primero contactar al servidor de PACMEN y comenzar la descarga de la aplicación contextual, la cual será ejecutada directamente en su dispositivo. Así pues, la aplicación contextual a ser descargada por el usuario es proveida a través de este servicio del servidor de PACMEN.

Se compone de un servidor Web, el cuál está disponible en un puerto independiente a los demás servicios provistos por la plataforma. El usuario accede a este servicio a través de una aplicación instalada en su dispositivo móvil, llamada SWFLoader, la cuál detallaremos más adelante. SWFLoader hace las veces de un navegador Web y solicita al usuario el número del servidor de PACMEN, el cuál no es más que una versión compacta de la dirección IP del servidor de PACMEN dentro de la red privada.

Como cualquier otro servidor Web, el servicio de la aplicación contextual debe estar disponible para varios usuarios de manera simultánea. En la plataforma PACMEN, este servicio es provisto por un servidor Web Apache Tomcat, aunque bien podría usarse cualquier otro servidor Web o implementarse una versión minimalista y optimizada, pues en realidad no se utilizan todas las capacidades provistas por un servidor Web profesional.



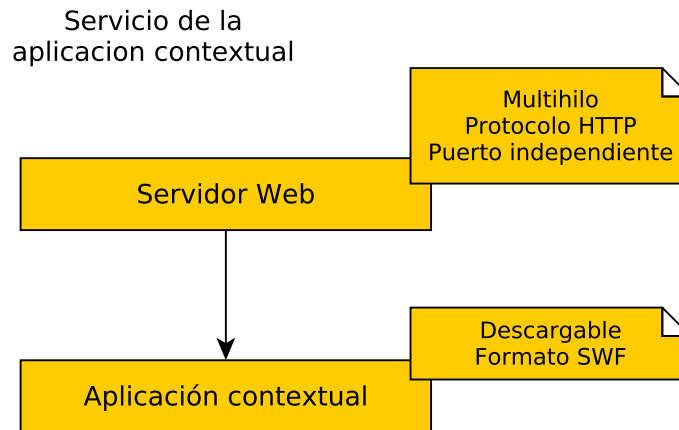


Figura 3.7: El servicio de la aplicación contextual

SWFLoader contacta al servicio de la aplicación contextual y, a través de una petición GET del protocolo HTTP, comienza a descargar la aplicación contextual. Una vez descargada, se ejecuta dentro del dispositivo del cliente y solicita permisos de comunicación con sockets TCP al servicio correspondiente.

- *Servicio de permisos de comunicación:* el contenido para Flash Player es distribuido por la red y es alojado directamente por los proveedores de dicho contenido. Además, el contenido es ampliamente utilizado porque presenta varias características especiales y provee una experiencia más enriquecida que la obtenida al utilizar simplemente HTML. Una de esas características especiales es la habilidad de crear sockets TCP para poder intercambiar información con los servidores. Desde el punto de vista de un administrador de red, la idea de que el contenido en Internet pueda crear conexiones con sockets hacia servidores internos es hasta cierto punto preocupante. Es por ello que Flash Player requiere permisos del servidor de contenido antes de poder establecer comunicación a través de sockets TCP. El encargado de otorgarlos es el servicio de permisos de comunicación, cuya implementación se encuentra detallada en la sección 4.1.
- *Servicio de comunicaciones:* una vez que la aplicación contextual ha sido descargada al dispositivo del cliente y ha adquirido los permisos para iniciar comunicación con el servidor de PACMEN a través de sockets TCP, comienza la interacción. La aplicación contextual que se ejecuta en el dispositivo del cliente captura los eventos del usuario y los comunica al servicio de comunicaciones, el cual a su vez envía información relevante a cada uno de los clientes que se encuentran conectados. Esta información puede ser específica para un usuario en particular, en respuesta quizá a sus acciones y peticiones o bien puede ser información de interés para todos los usuarios conectados.

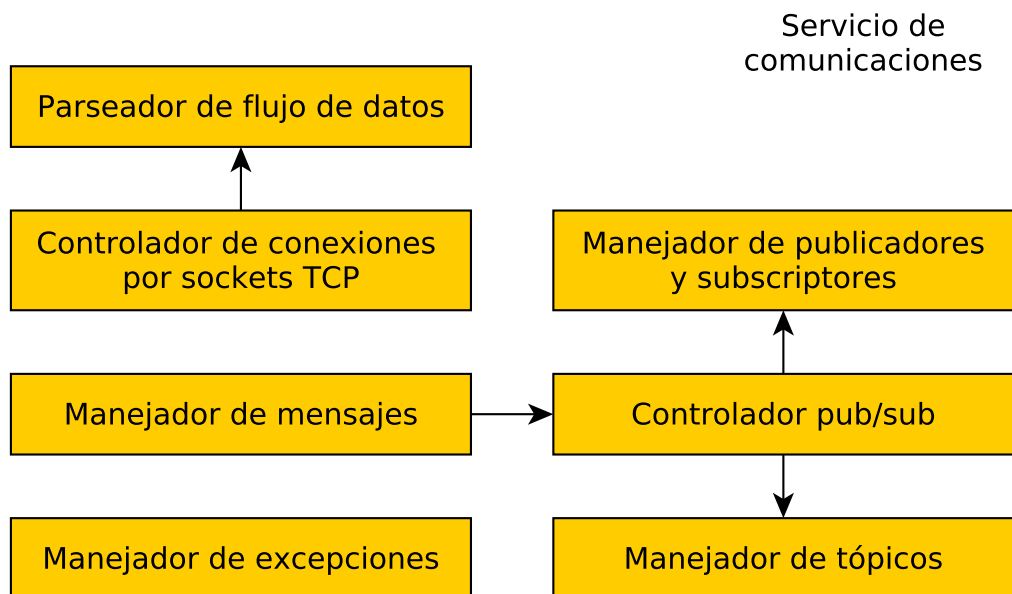


Figura 3.8: Servicio de comunicaciones

El manejo de las comunicaciones, el envío y la recepción de los mensajes, así como el manejo de excepciones en la comunicación tales como fallo de envío, conexión, desconexión y reconexión de dispositivos, manejo de latencias, etc. son controladas por este servicio.

La Figura 3.8 esboza los componentes del servicio de comunicaciones. Se tiene un controlador de conexiones por sockets TCP, el cual se encuentra en espera de clientes que deseen iniciar comunicación con el servicio de comunicaciones. Para llegar a este paso, los clientes debieron primero haber adquirido los permisos correspondientes con el servicio de permisos de comunicación.

Tras la petición de un usuario, el controlador de conexiones por sockets TCP abre una conexión bidireccional hacia el cliente, de modo que un hilo queda en espera de una petición del cliente para la transmisión de un flujo de bytes. Al mismo tiempo, el cliente queda en espera de un flujo de bytes por parte del servidor en un hilo por separado. De esta manera, la comunicación puede iniciarse en cualquier sentido a diferencia del protocolo HTTP, en el que la comunicación siempre funciona por el mecanismo petición-respuesta en el que el cliente inicia la comunicación y el servidor se limita a responder a las peticiones del cliente.

Cada vez que se detecta un flujo de bytes por parte del cliente, el parseador de flujo de datos lee la información y la encapsula en un mensaje, el cuál es transmitido al manejador de mensajes. De la misma manera, cuando el servicio de

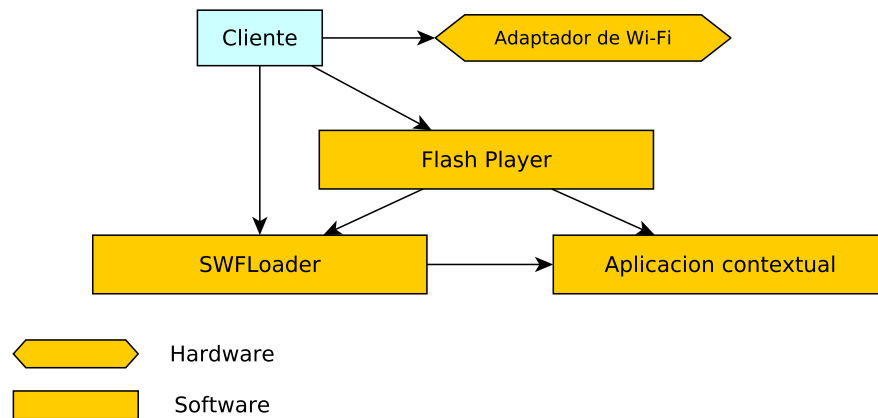


Figura 3.9: Componentes del cliente

comunicaciones desea transmitir un mensaje a uno o más clientes, el manejador de mensajes contacta al parseador de flujo de datos, el cual convierte el mensaje en un flujo de bytes para ser transmitido por el socket TCP hacia el cliente.

El manejador de mensajes utiliza el esquema *publish/subscribe* para la recepción, envío, manejo y administración de los mensajes. Este esquema es administrado por un controlador que se encarga de manejar las listas tanto de publicadores y suscriptores como de tópicos. El esquema *publish/subscribe* se describe más detalladamente en la sección 4.1.6.

Por otro lado, tenemos también dentro del servicio de comunicaciones al manejador de excepciones, el cual se encarga de escuchar errores en la transmisión de datos, desconexiones imprevistas, latencias demasiado altas, etc. y de notificar a los componentes interesados cuando alguna excepción ocurra, para que los componentes tomen las acciones correspondientes.

## Cliente

El dispositivo móvil del cliente dispone de los componentes que se muestran en la Figura 3.9 para poder utilizar la aplicación contextual e interactuar con el despliegue público.

El dispositivo del cliente necesita un adaptador Wi-Fi para poder conectarse a la red privada, donde se encuentra el servidor de la aplicación contextual y el despliegue público. Aunque el adaptador de Wi-Fi no es un componente de software como tal, consideramos importante mencionarlo porque es un requisito fundamental de la plataforma.

Por otro lado, el cliente requiere tener instalado Flash Player versión 10 o superior en su dispositivo. Normalmente, los dispositivos que son capaces de ejecutar Flash

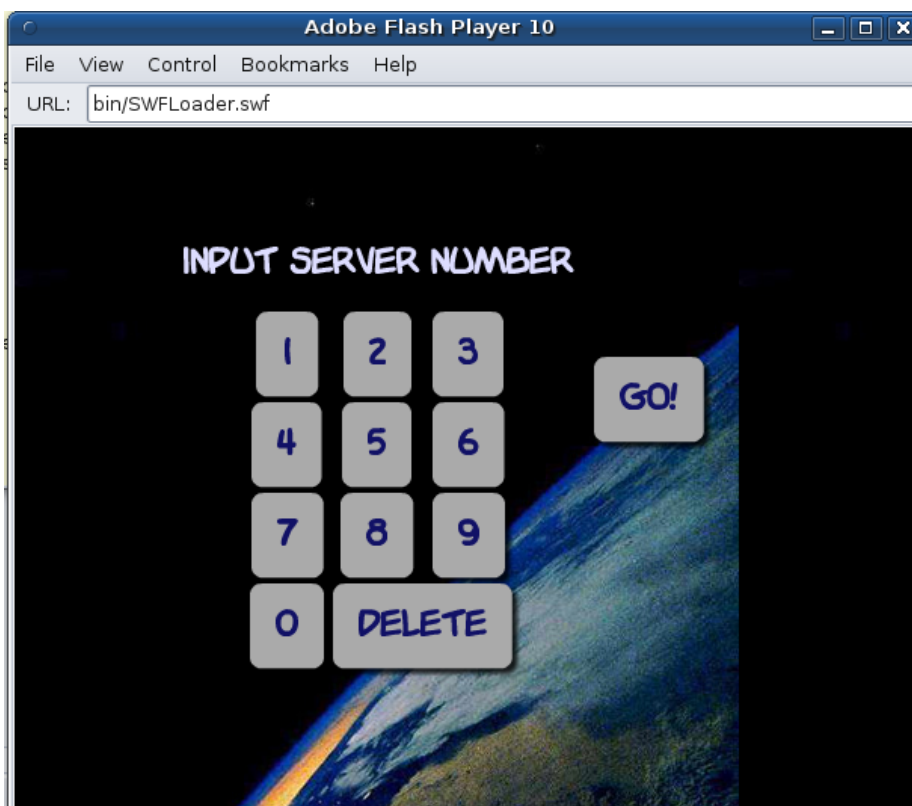


Figura 3.10: La aplicación SWFLoader

Player ya lo traen instalado por defecto, así que este requerimiento no debería ser un problema.

Otro de los componentes que deben encontrarse en el dispositivo del cliente es la aplicación SWFLoader, la cual es ejecutada por Flash Player y es el único programa que forma parte de la plataforma y que se encuentra instalado en el dispositivo del cliente. La aplicación SWFLoader funciona de manera análoga a un navegador Web, en el sentido que permite al usuario contactar diferentes servidores en busca de contenido descargable e interpretable. En este caso, el contenido que se descarga es la aplicación contextual, la cuál también es ejecutada por Flash Player.

### SWFLoader

Una vez que el cliente se encuentra conectado a la misma red privada que el servidor y el despliegue público, el usuario ejecuta la aplicación SWFLoader desarrollada como parte de la plataforma PACMEN (cf. Figura 3.10). SWFLoader solicita al usuario el número del servidor de PACMEN, que no es más que una versión corta de la dirección IP. Se recomienda que el número del servidor se encuentre visible a todos los usuarios, quizá cerca del despliegue público para que sea de fácil acceso.

Aplicación contextual



Figura 3.11: Componentes de la aplicación contextual

Una vez que el usuario introduce el número del servidor, SWFLoader contacta al servicio de la aplicación contextual y, por medio del protocolo HTTP, solicita la descarga de la aplicación contextual. Una vez que la descarga ha finalizado, la aplicación contextual es ejecutada directamente en el cliente por Flash Player.

### Aplicación contextual

La aplicación contextual es el corazón de la plataforma, ya que es el mecanismo bajo el cual la plataforma adquiere toda su funcionalidad y versatilidad. La aplicación contextual se divide en dos partes, como se muestra en la Figura 3.11. Por un lado, se tiene el componente a ser descargado y ejecutado por el cliente en su dispositivo móvil. Una vez descargado se ejecuta, solicita permisos para comunicación via sockets TCP al servicio correspondiente, establece una conexión y comienza a recibir entrada del usuario que, dependiendo de la aplicación, puede ser una gama variada de eventos. Según la aplicación en cuestión, algunos de estos eventos son comunicados al servicio de comunicaciones, el cual a su vez los envía a todos los componentes que estén interesados en dichos eventos. La forma en que un componente expresa su interés en recibir eventos es mediante la subscripción al tópico correspondiente. El mecanismo *publish/subscribe* se detalla mejor en la sección 4.1.6.

Por otro lado, se tiene el componente de la aplicación contextual que se ejecuta en el despliegue público. Esta aplicación muestra información a todos los usuarios y puede o no reflejar algo de la información que está recibiendo por parte de los clientes. Por ejemplo, en un videojuego multi-jugador que trate de una batalla entre tanques de guerra, donde cada usuario controle un tanque por medio de su dispositivo móvil, el componente de la aplicación contextual que se encuentra en el despliegue público podría mostrar un mapa con la localización de todos los jugadores. Mientras que el componente de la aplicación contextual que se ejecuta en el dispositivo móvil del cliente podría mostrar la visión del tanque del jugador, la mira, los controles o cualquier otra información que sea relevante únicamente para ese jugador.

### Interacción entre los componentes de la aplicación contextual

Los componentes de la aplicación contextual interactúan por medio de mensajes, los cuales no se envían directamente de un componente a otro, sino que se envían al

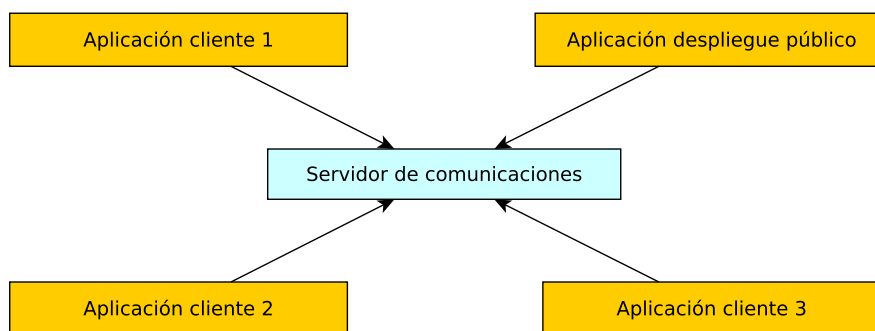


Figura 3.12: Comunicación entre los componentes de la aplicación contextual

servicio de comunicaciones, el cual utiliza un esquema de manejo de mensajes *publish/subscribe*. Esto significa que los componentes en realidad no conocen la existencia de los demás componentes por lo tanto no crean conexiones directas entre ellos. Un componente simplemente se suscribe a los tópicos de los que le interesa recibir información y publica sus propios eventos y mensajes en un tópico que tiene para él mismo.

Para ejemplificar esta interacción, supongamos el caso de una aplicación contextual en un cine en la que el despliegue público podría mostrar la cartelera disponible. Tras conectarse al servidor de PACMEN, los usuarios descargan y ejecutan el componente para el cliente de la aplicación contextual, que en este caso podría ser utilizado para leer sinopsis de películas, para ver los cortos de las películas o incluso para evaluar alguna película que el usuario haya visto.

### 3.3.2 Interfaz gráfica de usuario

La interfaz gráfica de usuario se encuentra dividida en dos partes:

- *Interfaz gráfica en el dispositivo móvil:* en su dispositivo móvil el usuario utiliza Flash Player para ejecutar la aplicación SWFLoader (Figura 3.10) cuya interfaz gráfica consiste de un teclado numérico. El usuario utiliza este teclado para introducir el número del servidor de PACMEN, el cuál se espera que esté visible para todos los usuarios de la aplicación contextual. Una vez introducido el número del servidor, la aplicación SWFLoader contacta al servicio de la aplicación contextual, y comienza la descarga del componente del cliente de dicha aplicación. Este componente es igualmente una aplicación que es ejecutada por Flash Player y tiene su propia interfaz gráfica, la cual depende del proveedor de la aplicación.
- *Interfaz gráfica en el despliegue público:* por otro lado, en el despliegue público también se muestra información. Esta interfaz gráfica, al igual que la interfaz del

componente para el cliente de la aplicación contextual, depende del proveedor de la aplicación y de su propósito.





# Capítulo 4

## La plataforma PACMEN

Teniendo en mente los aspectos descritos en el capítulo 3.1 de una plataforma que da soporte a la interacción entre dispositivos móviles y despliegues públicos, se procedió a implementar PACMEN (Plataforma para Aplicaciones Contextuales Multi-usuario en Entornos Nómadas).

### 4.1 Implementación de la plataforma PACMEN

La implementación de una plataforma involucra una serie de decisiones conceptuales y tecnológicas. Si una tecnología fuese superior a todas las demás en una tarea en particular y bajo un conjunto de restricciones específico, la decisión sería trivial, sin embargo éste no siempre es el caso. Cuando se elige una tecnología por encima de otra suele haber una compensación, se gana por un lado, pero se pierde por otro. Incluso cuando existe la posibilidad de usar varias tecnologías para el mismo objetivo de manera simultánea, intentando así ganar las ventajas de todas ellas, se suele pagar el costo en complejidad, estabilidad, tiempo de desarrollo y mantenimiento, entre otros.

En esta sección discutimos algunas de las razones detrás de las decisiones tecnológicas y conceptuales que se tomaron durante la implementación de PACMEN.

#### 4.1.1 Arquitectura básica

La arquitectura básica de PACMEN consiste de: 1) un despliegue público colocado estratégicamente en el ambiente donde los usuarios podrán observarlo desde una posición confortable sin necesidad de acercarse mucho, 2) uno o más usuarios utilizando dispositivos móviles como clientes para interactuar con el despliegue público y finalmente, 3) un servidor para mediar las comunicaciones y servir las aplicaciones. Los clientes no se conectan directamente al despliegue público, en vez de eso se comunican con el servicio de las comunicaciones. Esta arquitectura no solo provee de

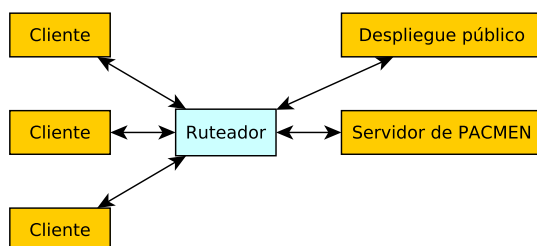


Figura 4.1: La arquitectura de red de PACMEN

desacoplamiento y robustez al permitir que diferentes entidades se encarguen de manejar tareas específicas, sino que también permite escalabilidad y flexibilidad pues una arquitectura de este estilo permitiría mantener varios servicios con un mismo servidor, en caso de pocos usuarios, o un sólo servicio en múltiples servidores, en caso de muchos usuarios.

Para establecer la comunicación inalámbrica entre los dispositivos móviles y el servicio de comunicaciones se tomaron en cuenta principalmente dos tecnologías: Bluetooth y Wi-Fi. Algunas ventajas de Bluetooth que resultaron sobresalientes fueron su bajo costo de energía y su disponibilidad relativamente común en dispositivos móviles. Sin embargo las ventajas de la alta velocidad, mayor rango de cobertura y escalabilidad para una gran cantidad de usuarios fueron decisivas para optar por Wi-Fi. Cuando se usa Wi-Fi se tiene el requerimiento adicional de un ruteador, pues a diferencia de Bluetooth los dispositivos no pueden conectarse unos con otros de manera directa. Sin embargo, dada la naturaleza del proyecto, este requerimiento adicional resulta despreciable pues podrían colocarse fácilmente ruteadores cerca de los despliegues públicos.

Es así que la arquitectura básica se implementa sobre una red privada que ahora incluye un ruteador, como se ve en la figura 4.1.

### 4.1.2 Contenido multimedia

Tomando en cuenta que deseamos ofrecer al cliente una experiencia enriquecida y atractiva, se decidió optar por una herramienta especializada en la presentación de contenido multimedia. Las principales herramientas que se tomaron en cuenta fueron, HTML5 y Flash Player 10. Se eligió Flash Player 10 debido a que es conocido por su utilización en el desarrollo de aplicaciones interactivas, multimedia. HTML5 resulta una interesante alternativa con mucho potencial, pero que a la fecha aún se encuentra en proceso de maduración.

Con estas tecnologías hemos definido los dispositivos móviles que serán soportados por la plataforma PACMEN, i.e., cualquier dispositivo capaz de ejecutar Flash Player 10 y que disponga de tecnología Wi-Fi.

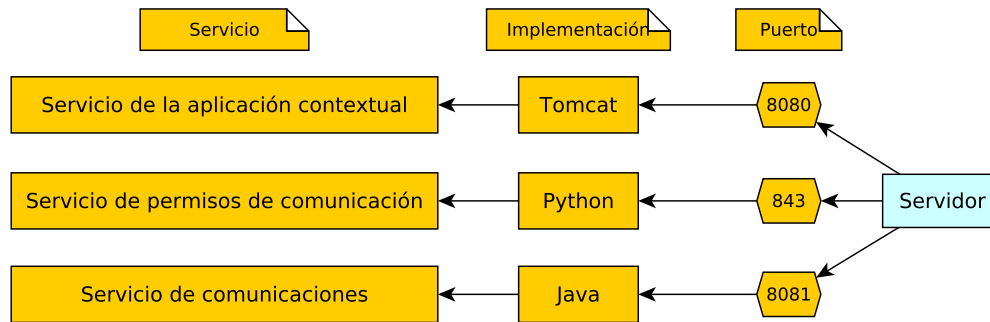


Figura 4.2: Los tres servicios principales del servidor, el tipo de implementación y el puerto correspondiente

### 4.1.3 Servicios proporcionados por el servidor

El servidor actúa como mediador de la comunicación entre los dispositivos móviles y el despliegue público. Provee tres servicios independientes mostrados en la figura 4.2. Cada servicio es ofrecido en un puerto distinto.

Para la implementación del servicio de la aplicación contextual se utilizó un servidor Web Tomcat, debido a que es una implementación ampliamente utilizada y contiene toda la funcionalidad necesaria para ofrecer la aplicación contextual por el protocolo HTTP. La elección del puerto 8080 es arbitraria, y es posible que en un ambiente de producción se quiera cambiar al puerto 80, el cual es utilizado normalmente para el protocolo HTTP.

El servicio de permisos de comunicación se encarga de proveer el archivo `crossdomain.xml`, el cual otorga los permisos para la utilización de sockets TCP. La aplicación utilizada es una ofrecida por Adobe como ejemplo de implementación. Fue desarrollada en Python para su fácil comprensión.

El servicio de comunicaciones se fue implementado en Java, esto debido a la estabilidad y robustez que presenta. Asimismo Java ofrece una variedad de paquetes que facilitan el desarrollo de servicios para conexiones por medio de sockets TCP.

### 4.1.4 Descarga y ejecución de aplicaciones contextuales

Para que PACMEN se considere una plataforma independiente y genérica, no se debería pedir a los usuarios que instalen algo nuevo cada vez que deseen interactuar con un despliegue público, aunque sea la primera vez que deseen interactuar con el despliegue en cuestión. PACMEN se instala una sola vez en el dispositivo del usuario, y cuenta con una herramienta llamada SWFLoader.

La herramienta SWFLoader facilita la conexión con el servicio de la aplicación contextual, el cual provee al cliente la aplicación contextual que contiene toda la

**Require:** “x.y” corresponde a los dos últimos bytes de la dirección IPv4 del servidor, e.g., “50.137”

**Ensure:** el número del servidor (SN)

- 1:  $x' \leftarrow \text{hexadecimal}(x)$
- 2:  $y' \leftarrow \text{hexadecimal}(y)$
- 3:  $SNX \leftarrow \text{concat}(x', y')$
- 4:  $SN \leftarrow \text{decimal}(SNX)$
- 5: **return**  $SN$

Figura 4.3: Cálculo del número del servidor

funcionalidad requerida para interactuar con el despliegue público en ese contexto en particular.

Para que SWFLoader contacte al servicio de la aplicación contextual, se requiere que el cliente introduzca el número del servidor, el cual es una representación corta de la dirección IP del servidor en la red privada.

### El número del servidor

Dado que la plataforma fue diseñada para trabajar dentro de una red privada, se decidió utilizar únicamente los dos últimos bytes de la dirección IPv4 del servidor, dando soporte a las redes privadas que comiencen con 192.168.x.x. Estas redes privadas permiten direccionar hasta 65,536 dispositivos, una cantidad que consideramos suficiente dada la naturaleza de las aplicaciones esperadas.

El número del servidor se espera que se encuentre visible a todos los usuarios de la aplicación contextual, probablemente cerca del despliegue público, y se forma por la unión de los últimos dos bytes de la dirección IPv4 en un solo número. La figura 4.3 muestra el método utilizado para calcular el número del servidor. El proceso consiste en tomar los últimos dos bytes de la dirección IPv4, convertirlos a su valor en hexadecimal, unirlos en un sólo número hexadecimal el cual finalmente es convertido de regreso a decimal. Por ejemplo, de la dirección IPv4 192.168.50.137, se toman los últimos dos bytes, 50.137, que en hexadecimal corresponden a 32.89, se juntan en un sólo número hexadecimal, en este caso 3289 el cual se regresa a decimal, 12937. Así pues, a un servidor con dirección IPv4 192.168.50.137 le corresponde el número de servidor 12937.

Se optó por utilizar esta representación, en vez de una dirección IPv4 completa, por las siguientes razones:

- El número del servidor tiene una longitud máxima de cinco dígitos, mientras que una dirección IPv4 tiene una longitud máxima de 12 dígitos y 3 caracteres de punto, dando una longitud total máxima de 15 caracteres.

- Aunque la dirección IP 192.168.1.3 es igual a 192.168.001.003, un usuario puede confundirse al tratar de saber el número de ceros que debe introducir.
- El caracter de punto puede ser otro factor de confusión o agobio para los usuarios.

Por supuesto que el cálculo del número del servidor es transparente para el usuario, quién sólo debe introducirlo en la aplicación SWFLoader tras haberlo visto en el despliegue público o en algún otro lugar accesible para todos los usuarios de la aplicación contextual.

### Servicio de administración de permisos

Una vez que se descarga la aplicación contextual, ésta es ejecutada en el dispositivo del cliente por Flash Player. Independientemente del tipo de aplicación que se trate, lo primero que hace es solicitar permisos al servidor para abrir conexiones basadas en sockets TCP. Para realizar esta tarea, el servidor dispone de un servicio de administración de permisos en un puerto independiente.

Para obtener los permisos, el cliente de la aplicación contextual intenta adquirir el archivo `crossdomain.xml` de alguna de las siguientes formas, siguiendo el orden en el que se enlistan:

- **A través del puerto 843:** el cliente de Flash Player ejecuta una petición del archivo al servidor por ese puerto.
- **Mediante la instrucción `Security.loadPolicyFile()`:** el cliente de Flash Player hace una petición explícita del archivo a través de esta instrucción.
- **Directamente sobre el socket destino:** el cliente de Flash Player hace una petición del archivo directamente sobre el socket que se desea utilizar para interactuar con el servidor de comunicaciones.

El proceso se detiene en el momento en que se logra adquirir el archivo `crossdomain.xml`, el cual autoriza o niega el uso de sockets, i.e., si se obtuvo el archivo por el puerto 843 pero en éste se niega la autorización para usar sockets, ya no se prueban las otras dos opciones.

Adobe solicitó ante la Agencia de Asignación de Números de Internet (IANA, por sus siglas en inglés), que se reserve el puerto 843 para servir archivos de políticas para sockets. La opción más segura es entonces adquirirlo a través de este puerto, ya que se puede confiar que el superusuario del sistema al que se desea conectar está de acuerdo con la comunicación. De hecho, en Linux y otros sistemas Unix se necesitan privilegios de super-usuario para poder servir en los puertos cuyo número es menor a 1024. Por esta razón, se optó por usar esta opción. La plataforma PACMEN hace uso de un programa en Python que se encarga de servir el archivo `crossdomain.xml` en el puerto 843.

### 4.1.5 Manejo de comunicaciones

En el lado del servidor, un robusto sistema de comunicaciones que de soporte a una topología dinámica es requerido, con clientes uniéndose y desconectándose en cualquier momento. Un sistema que releve a las aplicaciones de la responsabilidad del manejo de conexiones y desconexiones, y la correcta entrega de mensajes. Para ese propósito se sugirió utilizar un mecanismo de mensajería *Publish/Subscribe*. En PACMEN, este módulo está implementado en Java pues existen una serie de clases especializadas en el manejo de conexiones e hilos, lo que nos permite fácilmente y con pocas líneas de código realizar una implementación robusta y flexible del esquema de mensajería *Publish/Subscribe*.

### 4.1.6 Esquema de mensajería *Publish/Subscribe*

*Publish/Subscribe* es un esquema de mensajería donde los emisores de mensajes, llamados publicadores, no programan los mensajes para ser enviados directamente a los destinatarios, llamados subscriptores. En lugar de eso, los mensajes publicados son caracterizados en clases, sin que se tenga conocimiento de cuáles subscriptores, si acaso, existen para esa clase de mensajes. Los subscriptores expresan su interés en una o más clases de mensajes y sólo reciben mensajes de su interés sin conocimiento explícito de cuáles publicadores, si acaso, existen para esa clase de mensajes.

Este esquema de mensajería provee varias ventajas incluyendo, entre las más importantes para esta aplicación, una mayor escalabilidad y una topología de red más dinámica.

#### Filtrado

En el esquema *Publish/Subscribe*, los subscriptores típicamente reciben sólo un subconjunto del total de mensajes publicados. El proceso de elegir los mensajes para su recepción y procesamiento es llamado filtrado. Existen dos formas comunes de filtrado: basado en tópicos y basado en contenido.

- *Sistema basado en tópicos*: los mensajes son clasificados y publicados por tópicos o canales lógicos con un nombre en particular. Los subscriptores en un sistema basado en tópicos recibirán todos los mensajes que sean publicados en los tópicos a los que estén suscritos y todos los subscriptores suscritos a un mismo tópico recibirán los mismos mensajes. El publicador es el responsable de definir los tópicos o clases de mensajes a los cuales los subscriptores pueden suscribirse.
- *Sistema basado en contenido*: los mensajes sólo llegan al subscriptor si los atributos o el contenido de esos mensajes coinciden con el criterio definido por el subscriptor. El subscriptor es responsable de clasificar los mensajes.

El filtrado utilizado en la plataforma PACMEN es un sistema basado en tópicos. Se define un tópico para cada uno de los clientes que se conectan al servicio de comunicaciones, un tópico para el despliegue público y un tópico para el servicio de comunicaciones. Normalmente cada cliente al conectarse se suscribe al tópico del despliegue público, al tópico del servicio de comunicaciones y a su propio tópico, pero no a los tópicos correspondientes a otros clientes.

La aplicación del despliegue público, por otro lado, se suscribe a todos los tópicos, de esta manera cada vez que un cliente desea comunicar un mensaje, lo publica directamente en su tópico y el despliegue público puede recibirlo para procesarlo, sabiendo claramente de quién proviene para poder generar una respuesta adecuada en caso de ser necesario, e.g., para transmitir al cliente una respuesta o para reflejar algún cambio en la aplicación del despliegue público.

El tópico reservado para el servicio de comunicaciones, al cuál están suscritos normalmente todos los componentes puede ser utilizado para notificar eventos importantes, e.g., que una nueva versión de la aplicación está disponible, que la aplicación será apagada por mantenimiento o cualquier otra información importante que se quiera transmitir.

## Ventajas

- *Bajo acoplamiento*: los publicadores están débilmente acoplados a los suscriptores y no necesitan saber siquiera de su existencia. Mientras el tópico sea el punto de interés, los publicadores y los suscriptores tienen permitido permanecer ignorantes de la topología del sistema y cada uno de ellos puede seguir funcionando normalmente, sin importar el correcto funcionamiento de los demás.

En el tradicional esquema cliente-servidor, el cliente no puede publicar mensajes al servidor mientras que éste no esté funcionando, ni tampoco el servidor puede recibir mensajes a menos que el cliente esté funcionando. Algunos sistemas *Publish/Subscribe* no sólo desacoplan las localidades de los publicadores y suscriptores, sino también los desacoplan en el tiempo, permitiendo así que un mensaje sea publicado en el momento en que el publicador esté funcionando, y recibido en el momento que el suscriptor esté funcionando, sin importar que dichos momentos no coincidan.

- *Topología de red dinámica*: el tipo de aplicaciones para las cuales la plataforma PACMEN está diseñada sugiere que habrá usuarios entrando y saliendo de la red constantemente y en cualquier momento, lo cual requiere que el esquema de mensajería adoptado soporte una topología de red dinámica.

El esquema *Publish/Subscribe* ayuda a proveer este dinamismo, pues un cliente puede suscribirse en cualquier momento a los tópicos que le interesen y, en cuando un suscriptor se desconecta, simplemente se borra de la lista y automáticamente se eliminan sus suscripciones sin afectar a otros suscriptores

o publicadores. De la misma forma, en cuanto se desconecta un publicador, simplemente sus mensajes dejan de ser publicados, sin repercutir en el funcionamiento de otros publicadores o subscriptores.

## Desventajas

Los problemas más serios de los sistemas *Publish/Subscribe* son un efecto colateral de su principal ventaja: el desacoplamiento de los subscriptores y publicadores. El problema reside en que puede llegar a ser difícil especificar propiedades más fuertes que la aplicación pueda necesitar con respecto a los cabos del envío de un mensaje:

- Algunos sistemas *Publish/Subscribe* intentarán entregar el mensaje a los subscriptores por un rato y después se rendirán. Si una aplicación necesita de una garantía más fuerte (i.e. que los mensajes siempre se entreguen o que si la entrega no pudo ser confirmada, que el publicador sea notificado), entonces es probable que el esquema *Publish/Subscribe* no tenga alguna forma de proveer dicha propiedad.
- Otro problema surge cuando un publicador asume que existe algún suscriptor escuchándolo, i.e., si se utiliza el esquema *Publish/Subscribe* para registrar problemas de la aplicación de modo que cualquier aplicación que detecte algún error publica el mensaje apropiado y después, para que los errores sean desplegados en una consola, otra aplicación está suscrita al tópico correspondiente. Si la aplicación que despliega los errores en la consola se bloquea por alguna razón, los publicadores no tendrán forma de saber esto y seguirán tratando de publicar los errores que detecten, sin embargo no habrá quien los escuche y los mensajes se perderán.

Se recomienda que las aplicaciones contextuales se desarrollen con esta característica en mente, de modo que no sea crucial la entrega adecuada de cada uno de los mensajes o que se desarrolle un sistema de acuse de recepción por encima de la plataforma.

Otras desventajas del esquema *Publish/Subscribe* se encuentran por el lado de la seguridad, debido a que en sistemas que utilizan *brokers* (servidores), como es el caso de la plataforma PACMEN, el acuerdo de que un *broker* envíe un mensaje a un suscriptor se encuentra *en banda* con el mensaje, i.e., tanto la información del tópico como el mensaje mismo van en el mismo conjunto de datos, lo que puede causar varios problemas de seguridad. Los *brokers* podrían ser engañados para enviar notificaciones a los clientes equivocados o pueden ser objeto de ataques de negación de servicio, pues podrían ser sobrecargados mientras están obteniendo recursos para llevar el control de las suscripciones creadas.



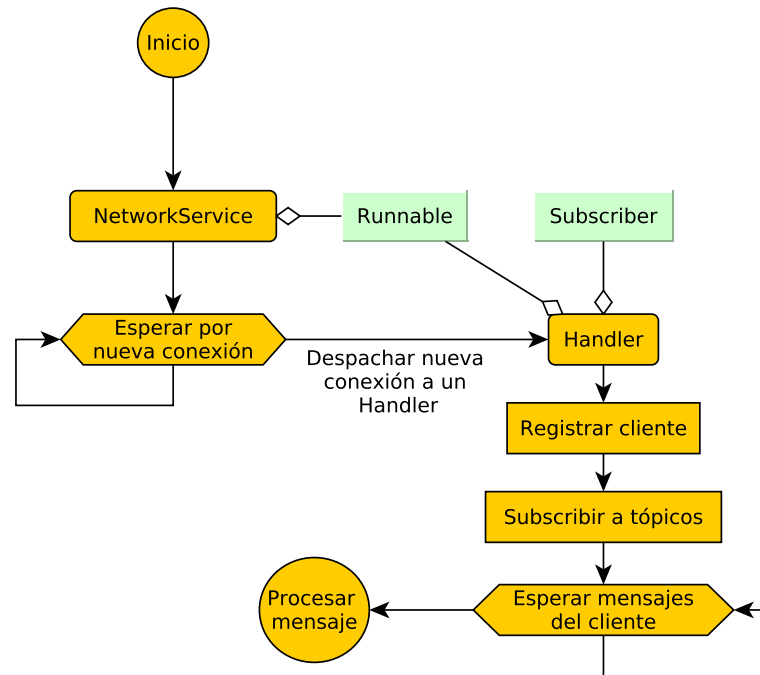


Figura 4.4: Creación de nuevas conexiones con clientes

Un publicador podrían también intentar introducir mensajes incorrectos o dañinos al sistema. El cifrado de datos podría ser una fuerte defensa en contra de estas vulnerabilidades, por lo que se recomienda a los desarrolladores de aplicaciones revisar la necesidad de dichas medidas.

#### 4.1.7 Implementación del esquema *publish/subscribe* en la plataforma PACMEN

En la plataforma PACMEN la implementación del esquema de mensajería *publish/subscribe* es multilenguaje: el servidor de comunicaciones está implementado en Java, mientras que los clientes están implementados en ActionScript 3 para Flash Player, esto debido a que del lado del servidor la prioridad era la estabilidad y robustez, mientras que del lado del cliente, la prioridad era el contenido enriquecido y la experiencia multimedia. El funcionamiento es relativamente sencillo y a continuación se detalla.

Todo comienza en el servicio de comunicaciones, donde la clase `NetworkService` implementa la interfaz `Runnable` para poder ser ejecutada en un hilo de ejecución independiente. Este componente se encuentra en espera de conexiones por parte de los clientes.

En cuanto se detecta una conexión, `NetworkService` crea un objeto de la clase

**Handler** y le asigna el cliente recién llegado para ser despachado. La clase **Handler** implementa las interfaces **Runnable** y **Subscriber**. Al implementar **Runnable** se convierte en un objeto capaz de ser ejecutado en su propio hilo de ejecución lo que, en efecto, permite estar en espera de mensajes por parte de todos los clientes conectados al mismo tiempo. De otra manera, si no se hiciera en un hilo de ejecución independiente, se tendría que estar iterando de manera secuencial sobre la lista de clientes para ver si alguno de ellos tiene algo que transmitir, o para transmitir algo hacia ellos. Al implementar la interfaz **Subscriber** se dota a la clase **Handler** de la capacidad de suscribirse a tópicos en el sistema *publish/subscribe* y, por lo tanto, de recibir los mensajes que se publiquen en ese tópico.

La figura 4.4 muestra la parte inicial del mecanismo de mensajería, que consta de la creación de nuevas conexiones hacia los clientes. Cada cliente se asocia con un objeto **Handler**, al cual le envía los mensajes que desea transmitir y del cual recibe los mensajes que le son enviados según los tópicos a los que está suscrito. Una vez que se crea un objeto **Handler** y se le asocia un cliente, éste abre dos flujos hacia el cliente, uno de entrada y uno de salida, después registra al cliente con el manejador de la lista de clientes. Este manejador asigna al cliente un identificador y se lo comunica al objeto **Handler**, el cual a su vez se lo comunica al cliente, como se muestra en la figura 4.5. Este identificador es muy importante y tiene muchos usos dentro del sistema *publish/subscribe* como se verá más adelante en esta misma sección.

Dentro de la implementación del esquema *publish/subscribe* se tiene también al manejador de tópicos o **TopicBoard**. La clase **TopicBoard** mantiene una lista de los tópicos existentes y una lista de los subscriptores suscritos a cada uno de los tópicos. Éstas listas se almacenan en un mapa que guarda la relación entre el identificador de un tópico y el vector de subscriptores. Existen tres tópicos permanentes que se crean desde el inicio de la ejecución del programa en el servicio de comunicaciones, a los cuáles todos tienen permiso de suscribirse, pero sólo el servidor tiene permiso para publicar en ellos. Estos tres tópicos reservados son:

- *Tópico de conexiones*: cuando se abre una nueva conexión con un cliente, el objeto **Handler** asociado a éste, lo registra con el manejador de la lista de clientes, el cual le asigna un identificador y publica, en el tópico de conexiones, que un nuevo cliente ha entrado al sistema e informa, en el mismo mensaje, el identificador de este nuevo cliente. Normalmente, la aplicación contextual del lado del despliegue público se encuentra suscrita a este tópico, de modo que es siempre notificada cada que un nuevo cliente entra al sistema y actualiza así lo que se muestra en pantalla o bien se suscribe al tópico correspondiente al nuevo cliente para recibir sus mensajes.
- *Tópico de desconexiones*: de la misma manera, cuando el objeto **Handler** asociado a un cliente detecta que este se ha desconectado, ya sea voluntariamente o por un error, se lo comunica al manejador de la lista de clientes, el cual lo remueve de la lista y notifica en el tópico de desconexiones que ese cliente ha

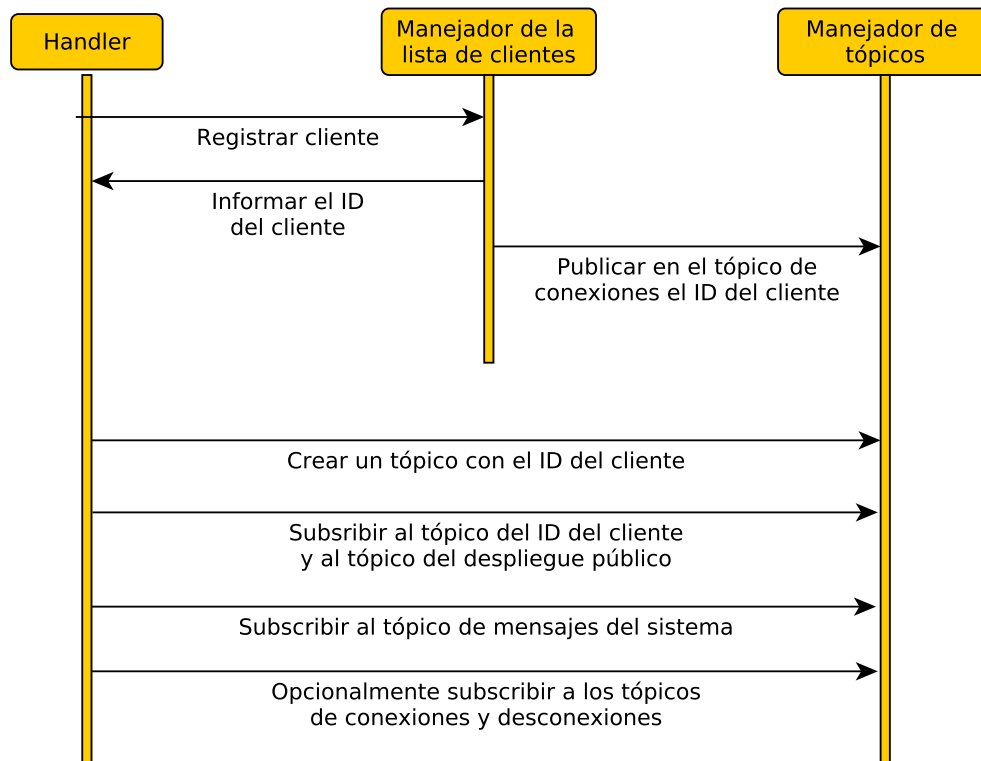


Figura 4.5: Registro de nuevos clientes y subscripción a tópicos

salido del sistema. Así la aplicación contextual en el despliegue público puede actualizar la información que muestra en pantalla y eliminar su suscripción del tópico de ese cliente.

- *Tópico de mensajes del sistema*: los mensajes del sistema están reservados al servidor para informar de cualquier eventualidad importante que concierna a la aplicación en general, e.g., que se encuentra disponible una nueva versión o que el sistema se apagará para mantenimiento, etc.

## 4.2 Aplicacion de prueba

La idea de tener una plataforma genérica es que se puedan desarrollar muchas y diferentes aplicaciones con distintas características, propósitos y capacidades. Se presentó la disyuntiva entre desarrollar varias aplicaciones pequeñas que ejemplificaran una a una las características y capacidades de la plataforma PACMEN o desarrollar una aplicación más completa y elaborada que demostrara, en conjunto, las diferentes capacidades y conceptos ofrecidos por la plataforma PACMEN. Se optó por la segunda opción, i.e., por desarrollar una aplicación con altos requerimientos de desempeño, responsividad, manejo dinámico de conexiones y desconexiones y entrada constante y simultánea de varios usuarios. Qué mejor aplicación con estas características que un videojuego multi-jugador.

### 4.2.1 Pacmen: The Game

Se desarrolló el videojuego Pacmen: The Game, el cual es una versión multi-jugador del clásico videojuego Pacman. Hasta 4 jugadores simultáneos pueden participar en el juego y hasta 8 pueden conectarse al sistema, en cuyo caso los últimos 4 esperarán su turno en fila para jugar. En cuanto un jugador pierde o se desconecta el juego rota a los jugadores y el siguiente participante toma su lugar.

La figura 4.6 muestra una captura de pantalla de la aplicación contextual Pacmen: The Game. Es una captura de lo que se muestra en el despliegue público durante la ejecución de la aplicación. Mientras que la figura 4.7 es una captura de la aplicación que el usuario descarga y utiliza en su dispositivo para interactuar con el despliegue público. En este caso, la aplicación que se ejecuta en el dispositivo móvil es un *gamepad*, el cual captura los eventos del usuario al presionar los botones direccionales y los encapsula en mensajes para enviarlos al servidor.

Al comenzar la ejecución de la aplicación contextual Pacmen: The Game, la aplicación del despliegue público inmediatamente se suscribe a los tópicos de conexiones y desconexiones en el servicio de comunicaciones. En cuanto un cliente contacta este servicio, descarga la aplicación contextual y comienza la interacción con el servicio de comunicaciones: se registra al cliente, se le asigna un identificador y se publica



Figura 4.6: Captura de pantalla de Pacmen: The Game en el despliegue público

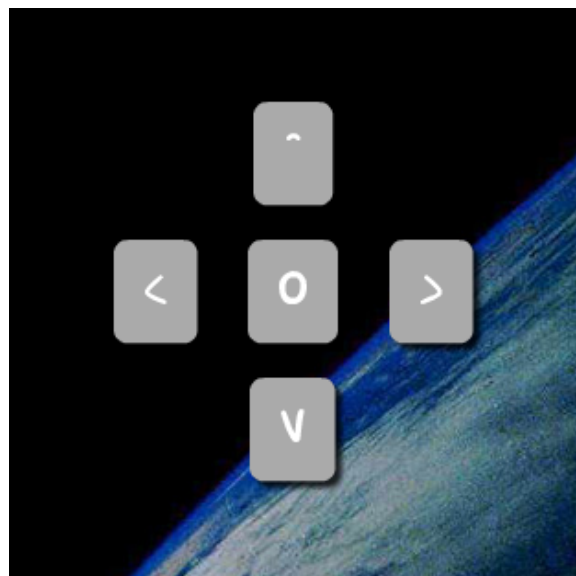


Figura 4.7: Captura de pantalla de Pacmen: The Game en el dispositivo del cliente

en el tópico de conexiones que un nuevo cliente se ha conectado. La aplicación en el despliegue público, al recibir la notificación de que un nuevo cliente se ha conectado, se suscribe al tópico del cliente y crea un pacman en la pantalla para que el usuario pueda controlarlo.

Cuando el usuario presiona los botones direccionales en la aplicación de su dispositivo móvil, la información del evento se encapsula en un mensaje y se publica en el tópico del cliente. Debido a que la aplicación en el despliegue público se encuentra suscrita al tópico de cada uno de los clientes, recibe estos eventos y puede entonces actualizar el contenido en el despliegue público como corresponda.

El servidor se encuentra a la espera de conexiones todo el tiempo. En cualquier momento un usuario puede conectar su dispositivo al servidor y comenzar la interacción. No es necesario esperar un momento en particular, de modo que los usuarios se conectan y desconectan constantemente, incluso en medio de juegos inconclusos. La aplicación reacciona a estas conexiones y desconexiones, creando y destruyendo pacmans, actualizando la lista de puntuaciones y rotando la lista de jugadores en espera, sin interrumpir el juego de otros usuarios.

Cuando un pacman es alcanzado por un fantasma, es destruido y el jugador es rotado en la lista de espera, de modo que si existen jugadores en cola para jugar, el siguiente jugador toma su lugar. Lo mismo sucede si un jugador se desconecta, ya sea intencionalmente o porque ocurrió algún evento inesperado, e.g., se acabó la batería de su dispositivo, apagó su adaptador de red o cualquier error en la comunicación.

## 4.2.2 Resultados

Los resultados de la aplicación de prueba desarrollada fueron satisfactorios. La plataforma PACMEN se muestra estable, responsiva y completamente funcional. La latencia entre la entrada de un usuario y la respuesta de la aplicación en el despliegue público es imperceptible, e.g., al presionar un botón de dirección en la aplicación contextual del dispositivo móvil causa inmediatamente que el pacman del jugador cambie de dirección en la pantalla del despliegue público.

La plataforma también muestra un manejo congruente y consistente de conexiones y desconexiones. No importa el orden en que los usuarios se conecten, desconecten o reconecten, la aplicación acepta y maneja las conexiones de manera correcta. Del mismo modo, no importa si la aplicación contextual se encuentra a la mitad de un juego, al principio o al final, esta acepta y maneja conexiones y desconexiones en todo momento sin interrumpir las actividades de los demás usuarios.

La aplicación contextual en el despliegue público muestra contenido enriquecido y una animación fluida, tal y como se podría esperar de una aplicación convencional para Flash Player. La versatilidad del lenguaje ActionScript 3 otorga a la plataforma PACMEN un gran potencial para el desarrollo de aplicaciones atractivas e interactivas.

La aplicación instalada en el dispositivo del cliente mostró gran estabilidad y un funcionamiento intuitivo y sencillo. La conexión con cada uno de los componentes del servidor se efectúa sin problemas, desde la descarga de la aplicación contextual a través del servidor Web, hasta la obtención de permisos para conexiones con base en sockets TCP y la transmisión de mensajes, tanto de entrada como de salida, con el servicio de comunicaciones.

La plataforma PACMEN cumple con las expectativas para las que fue desarrollada, incluyendo las cinco características clave que se definieron durante la etapa de diseño de la plataforma:

- **Genérica**

Las aplicaciones que se ejecutan sobre la plataforma son descargadas directamente en el sitio de uso e interpretadas por Flash Player, lo que permite que la aplicación se desarrolle tal y como el proveedor del servicio en el despliegue público lo requiera. La versatilidad de Flash Player permite, en realidad, el desarrollo de una gran variedad de aplicaciones atractivas y con diferentes propósitos. Su popularidad en Internet es prueba de ello.

- **Responsiva**

La velocidad de descarga de la aplicación contextual hacia el dispositivo móvil es la misma que se puede encontrar al descargar una aplicación para Flash Player en Internet con el mismo dispositivo móvil, pues el protocolo de conexión utilizado es el mismo, i.e., Wi-Fi, y el protocolo de comunicación también, i.e. HTTP. Incluso la velocidad de descarga puede ser superior, pues el servidor de la aplicación se encuentra en la misma red privada. Por otro lado, la latencia entre la entrada de un usuario y la respuesta del servidor resultó ser imperceptible, al grado de hacer un videojuego multi-usuario jugable sin problemas de latencia. El uso de conexiones basadas en sockets TCP permite que la conexión se mantenga abierta, por lo que no requiere que se abra una nueva conexión cada vez que el usuario genera una entrada hacia el servidor, ahorrando así recursos y tiempo. De la misma manera, los sockets TCP permiten el flujo de información en ambos sentidos, sin necesidad de esperar una solicitud por parte del cliente para otorgar una respuesta, como sucede en el protocolo HTTP.

- **Multimedia**

Flash Player es popularmente conocido como una herramienta para la distribución de contenido multimedia enriquecido y atractivo, con animaciones, sonido, imágenes y texto combinados bajo la utilización de efectos visuales atractivos. Todo esto permite llevar al usuario una experiencia completa y atractiva, haciendo a la plataforma más llamativa, amigable y en general agradable de utilizar, dándole así una mayor probabilidad de éxito.

- **Multi-usuario**

No sólo muchos usuarios pueden interactuar con el despliegue público de manera

simultánea, sino también puedan hacerlo tranquila y fluidamente, sin preocuparse por interrumpir o ser interrumpidos por otros usuarios. Los usuarios son capaces de comenzar la comunicación con el despliegue y terminarla en cualquier momento, sin que estas acciones afecten las actividades de otros usuarios del sistema.

- **Independiente**

La plataforma PACMEN no requiere software adicional ni hardware adicional. Únicamente se requiere Flash Player 10 o superior, instalado en el dispositivo móvil y la capacidad de conectarse a una red privada por medio de Wi-Fi. Estas dos son características cada vez más comunes en los dispositivos móviles. Estas características permiten a la plataforma PACMEN obtener mejores probabilidades de ser utilizada por usuarios casuales, quienes son el objetivo primario del proyecto, pues al ver que no requieren instalar nuevos paquetes o sistemas y que tampoco requieren comprar hardware adicional, tendrán menos razones para evitar intentarlo por primera vez. Y al igual que muchas otras tecnologías diferentes e innovadoras, la plataforma PACMEN requiere de ese primer intento por parte de los usuarios para cautivarlos y tener una posibilidad de éxito.



# Capítulo 5

## Conclusiones y trabajo futuro

Los despliegues públicos son utilizados para mostrar información a varios usuarios de manera simultánea. Estos despliegues públicos proveen a los usuarios aplicaciones contextuales que dependen del lugar donde se encuentren, e.g., en un aeropuerto la aplicación contextual mostraría información sobre llegadas, salidas, vuelos disponibles, etc. Más aún los usuarios pueden utilizar sus dispositivos móviles para acceder a un conjunto más grande de servicios provistos por la aplicación contextual, e.g., en el caso del aeropuerto, los usuarios podrían reservar boletos de avión, consultar el clima, etc.

Sin embargo, los despliegues públicos aún carecen de la funcionalidad para recibir entrada de varios usuarios al mismo tiempo. Algunos despliegues soportan interacción por medio pantallas táctiles, pero la limitación de esta solución reside en que sólo los usuarios que estén frente al despliegue pueden interactuar con él normalmente de manera secuencial.

Por otro lado, la proliferación de dispositivos móviles ha causado que comiencen a ser vistos como herramientas en la vida diaria y puedan usarse para obtener funcionalidades antes inalcanzables. En esta tesis se explora la idea de utilizar dispositivos móviles para interactuar con despliegues públicos. Esta idea ha comenzado a ser estudiada en algunos proyectos que se revisan en el capítulo del estado del arte del presente documento. El presente trabajo parte de las experiencias obtenidas en esos proyectos y trata de extraer algunas características fundamentales, para así emprender la tarea de desarrollar una plataforma genérica que dé soporte a la interacción entre dispositivos móviles y despliegues públicos.

La plataforma PACMEN es el resultado de ésta tesis, se trata de una plataforma que da soporte al desarrollo de aplicaciones contextuales, i.e., que dependen del contexto de utilización en general y del proveedor de la aplicación para ofrecer un servicio en particular, según el objetivo del despliegue público.

Las siguientes características fueron fundamentales para el desarrollo de la plataforma y se tomaron en cuenta al momento de enfrentar decisiones, tanto conceptuales como tecnológicas, a lo largo de la implementación de la plataforma PACMEN:

- **Genérica:** permite el desarrollo y ejecución de distintas aplicaciones contextuales sobre la misma plataforma.
- **Responsiva:** procura mantener la latencia de las comunicaciones al mínimo.
- **Multimedia:** tiene soporte para contenido multimedia enriquecido.
- **Multi-usuario:** soporta múltiples usuarios de manera simultánea, manejando conexiones, desconexiones y comunicaciones en todo momento, sin interrumpir las actividades de otros usuarios.
- **Independiente:** no requiere hardware adicional.

El resultado de que la plataforma PACMEN posea estas características es que los usuarios pueden utilizar el dispositivo móvil que tengan a su disposición, siempre y cuando posea un adaptador de red Wi-Fi y tenga Flash Player instalado, para utilizar las aplicaciones contextuales ofrecidas por diferentes proveedores e interactuar con los despliegues públicos disponibles, abriendo paso así a una gran variedad de posibilidades.

## 5.1 Conclusiones

Debido a que la cantidad y tipo de aplicaciones contextuales puede ser muy variado, resulta necesario el desarrollo de una plataforma que permita la implementación y ejecución de diferentes aplicaciones, con diferentes características y requerimientos, y que provea facilidades tanto a los desarrolladores como a los usuarios. De este modo, la misma plataforma puede ser usada en diferentes aplicaciones contextuales.

La construcción de una plataforma genérica para aplicaciones contextuales multi-usuario, como la que se desarrolló en esta tesis, otorga diferentes ventajas. A continuación, enumeramos algunas de las ventajas ofrecidas tanto a desarrolladores como a usuarios.

Las ventajas adquiridas al utilizar la plataforma PACMEN se pueden dividir en dos: 1) ventajas adquiridas por los desarrolladores de las aplicaciones contextuales y 2) ventajas adquiridas por los usuarios de las aplicaciones contextuales, quienes utilizan sus dispositivos móviles para interactuar con los despliegues públicos.

### 5.1.1 Ventajas adquiridas por los desarrolladores

La plataforma PACMEN contiene un conjunto de librerías que dan soporte a las comunicaciones entre dispositivos móviles y servidores, tanto para la descarga de aplicaciones contextuales, como para la comunicación entre usuarios y el manejo de permisos para realizar dicha comunicación.

De esta manera, los desarrolladores pueden hacer uso de estas librerías para desarrollar aplicaciones contextuales de manera más ágil, concentrándose únicamente en la funcionalidad y presentación, en vez de preocuparse por el manejo de las comunicaciones, los permisos o la descarga de aplicaciones contextuales.

El hecho de utilizar una misma plataforma para el desarrollo de varias aplicaciones contextuales permite a los desarrolladores librarse del problema de la compatibilidad, pues si un dispositivo funciona con una aplicación, lo más probable es que funcione con otras aplicaciones.

Libera también a los desarrolladores de la creación de instaladores, del manejo de permisos dentro del dispositivo móvil y del control de las dependencias, debido a que lo único que requiere el usuario para descargar y utilizar las aplicaciones contextuales es tener el cliente de la plataforma instalado.

Debido a que la aplicación contextual se descarga y ejecuta en el momento de su uso, tal y como lo haría un navegador Web al acceder a una página, la actualización de las aplicaciones resulta muy sencilla, pues sólo es necesario actualizarla dentro del servidor y la próxima vez que los usuarios la descarguen, obtendrán la versión actualizada y la podrán utilizar inmediatamente.

Flash Player 10 es ampliamente conocido entre los desarrolladores por su gran flexibilidad, eficiencia y capacidades, ya que permite a los desarrolladores crear una gran variedad de aplicaciones multimedia con un alto nivel de interactividad. El lenguaje de programación utilizado por Flash Player 10, Action Script 3 es sumamente poderoso y flexible porque permite un ágil desarrollo de aplicaciones y provee una gran cantidad de herramientas y utilerías, que ayuda a los desarrolladores a implementar prácticamente cualquier efecto audiovisual que se imaginen. Las aplicaciones obtenidas resultan ser muy llamativas, fluidas y eficientes. Realmente, consideramos que Flash Player 10 y Action Scrip 3 son herramientas muy adecuadas para este tipo de plataforma. Además de que los dispositivos que son capaces de ejecutar Flash Player 10 normalmente ya lo traen instalado, por lo que por ese lado también reduce los contratiempos de instalación de dependencias y la revisión de compatibilidad.

### **5.1.2 Ventajas adquiridas por los usuarios**

La plataforma PACMEN sólo requiere de Flash Player 10 y Wi-Fi para funcionar, por lo que es fácil para un usuario distinguir si su dispositivo móvil es compatible con la plataforma y lo mejor es que, una vez que determinó que su dispositivo es compatible, no tendrá que preocuparse más por ello. Aun cuando nunca haya utilizado una aplicación contextual en particular que utilice la plataforma, la aplicación podrá ser descargada y ejecutada en su dispositivo.

La instalación de aplicaciones puede llegar a ser un proceso tedioso, sobre todo en dispositivos móviles, sin mencionar los riesgos de seguridad que involucra la instalación constante de aplicaciones. Dado que las aplicaciones contextuales en la

plataforma PACMEN son descargadas en el sitio e interpretadas en un ambiente seguro, provisto por Flash Player 10, los usuarios no tienen que preocuparse por instalar las aplicaciones ni por los riesgos que eso involucraría. De nueva cuenta, se puede ver esta ventaja haciendo analogía al funcionamiento de un navegador Web, el cuál sólo se instala una vez y nos permite visitar páginas que son descargadas en ese momento e interpretadas por el navegador, sin que tengamos que instalar cada página.

De igual manera que sucede con los desarrolladores, el proceso de actualizar una aplicación contextual en la plataforma PACMEN resulta muy sencillo y transparente para los usuarios, quienes ni siquiera necesitan estar al tanto de qué versión es la que está siendo ejecutada en su dispositivo. Simplemente, cuando una aplicación es actualizada, la próxima vez que los usuarios interactúen con ella descargarán la última versión y comenzarán a utilizarla.

Flash Player 10 es ampliamente conocido por los usuarios como una herramienta capaz de ejecutar contenido multimedia interactivo, seguro y atractivo. Su utilización es muy popular en Internet para animaciones, juegos, interfaces gráficas, reproductores y otros tipos de aplicaciones. Además es multiplataforma y altamente compatible y disponible. Todo esto se traduce en ventajas para los usuarios quienes podrán utilizar un producto familiar, confiable y probado al emplear la plataforma PACMEN, teniendo así acceso a contenido multimedia, interactivo, flexible y funcional en sus dispositivos móviles para la interacción con despliegues públicos.

## 5.2 Trabajo futuro

En esta tesis se presentó una plataforma genérica para propiciar la interacción entre dispositivos móviles y despliegues públicos. Sin embargo, como podría suceder con cualquier otro tipo de plataforma, el proyecto no esboza su vitalidad hasta que existe una serie de aplicaciones aprovechando sus capacidades, i.e., siendo utilizada para lo que fue creada o, mejor aun, propiciando la creación de más y mejores plataformas genéricas en la misma dirección, pues no hay logro más grande que ser objeto de inspiración.

Es por ello que, como trabajo a futuro, observamos la implementación de diferentes aplicaciones contextuales que se apoyen en la plataforma PACMEN, ya sea alguna de las sugeridas a lo largo de este documento, de las implementadas en otros proyectos en el estado del arte, o alguna otra idea que se presente en el futuro. Conforme se desarrollen aplicaciones sobre la plataforma PACMEN, seguramente se irán descubriendo limitaciones de las que padece la plataforma, i.e., habrá mucho campo para mejorar la estabilidad, capacidad y versatilidad de la plataforma en futuras iteraciones.

Las capacidades de la plataforma PACMEN también pueden ser aumentadas, dando soporte simultáneo a otras tecnologías de comunicación, e.g., Bluetooth, o

a otras tecnologías de presentación de la información multimedia, e.g. HTML5 o Silverlight.

El objetivo más importante de esta tesis después de todo es propiciar la interacción entre dispositivos móviles y despliegues públicos, de modo que un paso importante para alcanzar el objetivo es la difusión del concepto, de la idea, de la plataforma y de las aplicaciones. De modo que, como trabajo a futuro, se prevee no sólo la implementación de nuevas aplicaciones, sino el despliegue de éstas en situaciones de uso reales, promocionando no sólo la plataforma, sino mucho más importante el concepto de la interacción entre dispositivos móviles y despliegues públicos y las oportunidades que dicha interacción presenta en el desarrollo de nuevas tecnologías.



# Publicaciones del autor

[Anzures and Mendoza, 2011] Anzures, H. and Mendoza, S. (2011) Multi-user Interaction with Public Screens Using Mobile Devices. 8th International Conference on Electrical Engineering, Computing Science and Automatic Control (*CCE*), pages 921–925, Mérida, Yucatán. México October 26-28, 2011. IEEE.





# Bibliografía

- [1] Jiten Rama and Judith Bishop. A survey and comparison of cscw groupware applications. In *Proceedings of the 2006 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*, SAICSIT '06, pages 198–205, , Republic of South Africa, 2006. South African Institute for Computer Scientists and Information Technologists.
- [2] Tim Paek, Maneesh Agrawala, Sumit Basu, Steven M. Drucker, Trausti T. Kristjansson, Ron Logan, Kentaro Toyama, and Andy Wilson. Toward universal mobile interaction for shared displays. In Herbsleb and Olson [17], pages 266–269.
- [3] Keith Cheverst, Alan J. Dix, Dan Fitton, Christian Kray, Mark Rouncefield, George Saslis-Lagoudakis, and Jennifer G. Sheridan. Exploring mobile phone interaction with situated displays. In Enrico Rukzio, Jonna Häkkinä, Mirjana Spasojevic, Jani Mäntyjärvi, and Nishkam Ravi, editors, *PERMID*, pages 43–47. LMU Munich, 2005.
- [4] Florian Echtler, Simon Nestler, Andreas Dippon, and Gudrun Klinker. Supporting casual interactions between board games on public tabletop displays and mobile devices. *Personal and Ubiquitous Computing*, 13(8):609–617, 2009.
- [5] Giuseppe Ghiani, Fabio Paternò, Carmen Santoro, and Lucio Davide Spano. Ubicicero: A location-aware, multi-device museum guide. *Interacting with Computers*, 21(4):288–303, 2009.
- [6] Himanshu Raj, Rich Gossweiler, and Dejan S. Milojicic. Contentcascade incremental content exchange between public displays and personal devices. In *MobiQuitous*, pages 374–381. IEEE Computer Society, 2004.
- [7] Eleanor F. Toye, Richard Sharp, Anil Madhavapeddy, David Scott, Eben Upton, and Alan F. Blackwell. Interacting with mobile services: an evaluation of camera-phones and visual tags. *Personal and Ubiquitous Computing*, 11(2):97–106, 2007.
- [8] Carl Gutwin. Book review: Public and situated displays: Social and interactional aspects of shared display technologies, kenton o'hara, mark perry, elizabeth churchill and daniel russell (eds.), the kluwer international series on computer

- supported cooperative work, 2003, 456 pp, isbn 1-4020-1677-8. *Computer Supported Cooperative Work*, 14(3):287–291, 2005.
- [9] Antonietta Grasso, Martin Mühlenbrock, Frederic Roulland, and Dave Snowdon. Supporting Communities of Practice with Large Screen Displays. In K. O’Hara, M. Perry, E. Churchill, and D. Russell, editors, *Public and Situated Displays: Social and Interactional Aspects of Shared Display Technologies*, CSCW Series. Kluwer International, 2003.
- [10] Daniel M. Russell and Rich Gossweiler. On the design of personal & communal large information scale appliances. In Gregory D. Abowd, Barry Brumitt, and Steven A. Shafer, editors, *Ubicomp*, volume 2201 of *Lecture Notes in Computer Science*, pages 354–361. Springer, 2001.
- [11] Harry Brignull, Shahram Izadi, Geraldine Fitzpatrick, Yvonne Rogers, and Tom Rodden. The introduction of a shared interactive surface into a communal space. In Herbsleb and Olson [17], pages 49–58.
- [12] Norbert Streitz, Thorsten Prante, Carsten Rucker, Daniel Van Alphen, Carsten Magerkurth, Richard Stenzel, and Daniela Plewe. *Ambient Displays and Mobile Devices for the Creation of Social Architectural Spaces*, pages 387–409. Kluwer Academic Publisher, 2003.
- [13] Seokhee Jeon, Jane Hwang, Gerard Jounghyun Kim, and Mark Billinghurst. Interaction with large ubiquitous displays using camera-equipped mobile phones. *Personal and Ubiquitous Computing*, 14(2):83–94, 2010.
- [14] Matthias Finke, Anthony Tang, Rock Leung, and Michael Blackstock. Lessons learned: game design for large public displays. In Sofia Tsekeridou, Adrian David Cheok, Konstantinos Giannakis, and John Karigiannis, editors, *DIMEA*, volume 349 of *ACM International Conference Proceeding Series*, pages 26–33. ACM, 2008.
- [15] Jürgen Scheible and Timo Ojala. Mobilenin combining a multi-track music video, personal mobile phones and a public display into multi-user interactive entertainment. In HongJiang Zhang, Tat-Seng Chua, Ralf Steinmetz, Mohan S. Kankanhalli, and Lynn Wilcox, editors, *ACM Multimedia*, pages 199–208. ACM, 2005.
- [16] Pedro Ventura, Hugo Sousa, and Joaquim Jorge. Mobile phone interaction with outdoor advertisements. In *CHI ’08 extended abstracts on Human factors in computing systems*, CHI EA ’08, New York, NY, USA, 2008. ACM.
- [17] James D. Herbsleb and Gary M. Olson, editors. *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work, CSCW 2004, Chicago, Illinois, USA, November 6-10, 2004*. ACM, 2004.