

Universidad Nacional de La Plata

Facultad de Informática

Licenciatura en Informática

Tesis de Grado

Aplicaciones Web para celulares

Director: Francisco Javier Díaz

Alumno: Diego Germán de la Riva
Legajo: 1828/6

Introducción

Las tecnologías inalámbricas -y en particular la telefonía celular- han tenido mucho auge y desarrollo en estos últimos años.

El teléfono celular hoy día no es solo una herramienta indispensable para la gente de negocios, sino también un elemento primordial para la comunicación entre las personas. El rápido desarrollo tecnológico, la reducción de costos y el incremento de las expectativas de la gente, han provocado una invasión de teléfonos celulares en toda la sociedad, a nivel de llegar a revolucionar las actividades que las personas realizan diariamente.

A pesar de que fue concebida estrictamente para la voz, la tecnología celular de hoy es capaz de brindar otro tipo de servicios, como transferencia de datos, audio y video, gracias a la cada vez más rápida expansión de las nuevas tecnologías.

Los dispositivos móviles cada vez son más pequeños, más poderosos y a medida que aumenta la demanda del consumidor, más baratos, incorporando nuevas funcionalidades diseñadas para permitir a los usuarios trabajar y acceder a la información en cualquier situación.

La nueva tendencia de la tecnología es dar a los usuarios la posibilidad de tener todo lo que pueden llegar a necesitar en un dispositivo de bolsillo. En particular la tecnología celular tiene como objetivo brindar "movilidad", definida como la capacidad de acceder a la información y a los servicios en cualquier momento, de cualquier manera y en cualquier parte.

El auge de los servicios que se proveen hoy en día en Internet, el avance tecnológico, la tendencia hacia dispositivos más pequeños y más rápidos, junto con la necesidad de acceso a la información en cualquier momento, son los factores determinantes del surgimiento de nuevas tecnologías de acceso a Internet desde cualquier tipo de dispositivos incluyendo a los teléfonos celulares.

El concepto de Internet en celulares tiene ya algunos años y si bien no ha logrado aún instalarse en el común de la gente, se espera que en los próximos años el acceso a Internet desde dispositivos móviles crezca exponencialmente y llegue a superar al uso desde dispositivos convencionales como PCs de escritorio o Laptops.

Índice de contenidos

Introducción.....	2
Índice de contenidos.....	3
Prefacio	6
Organización de la tesis	6
Agradecimientos.....	7
Capítulo 1. El celular	8
1.1 Historia de la telefonía celular	8
1.1.1 El teléfono	8
1.1.2 La comunicación inalámbrica.....	9
1.1.3 La telefonía celular	11
1.2 Generaciones de la telefonía celular	13
1.2.1 Primera generación (1G).....	13
1.2.2 Segunda generación (2G)	13
1.2.3 Generación 2.5 G	13
1.2.4 Tercera generación (3G).....	13
1.2.5 Cuarta generación (4G)	14
1.3 El celular en la actualidad	14
1.3.1 Servicios tecnológicos.....	14
1.3.2 Aspectos adicionales	16
1.3.3 El negocio.....	17
1.4 Datos de la telefonía celular.....	17
1.4.1 En Argentina.....	18
1.5 El futuro en Argentina	19
2.1 Operador de red	20
2.2 Redes inalámbricas	20
2.3 Protocolos móviles	21
2.3.1 TDMA (Time Division Multiple Access)	21
2.3.2 AMPS (Advanced Mobile Phone System)	21
2.3.3 GSM (Global System for Mobile Communications).....	21
2.3.4 HSCSD (High Speed Circuit Switched Data).....	22
2.3.5 GPRS (General Packet Radio Service)	22
2.3.6 EDGE (Enhanced Data rate for GSM Evolution)	22
2.3.7 CDMA (Code Division Multiple Access)	22
2.3.8 CDMA-2000 (Code Division Multiple Access 2000)	22
2.3.9 3G (Third Generation)	23
2.3.10 WCDMA (Wideband Code Division Multiple Access).....	23
2.3.11 UMTS (Universal Mobile Telephone System).....	23
2.4 Sistemas operativos	24
2.4.1 Symbian	25
2.4.2 Windows Mobile.....	25
2.4.3 RIM.....	26
2.4.4 Palm OS	26
2.5 Servicios.....	27
2.5.1 SMS (Short Message Service).....	27
2.5.2 MMS (Multimedia Messaging System)	27
2.5.3 Acceso a Internet.....	27
2.5.4 Navegación WAP	27
2.5.5 Java	28
2.5.6 Conexión inalámbrica con otros dispositivos.....	28
2.5.7 Otros servicios	29
Capítulo 3. Tecnologías inalámbricas.....	30
3.1 WAP.....	30
3.1.1 Surgimiento de WAP	30
3.1.2 Arquitectura.....	31
3.1.3 Dispositivos	32

3.1.4 El Cliente WAP	32
3.1.5 Gateway o proxy WAP	33
3.1.6 Funcionamiento	34
3.1.7 Configuración.....	35
3.1.8 Estructura interna de WAP (Pila WAP)	35
3.1.9 WAP 2.0	39
3.2 Tecnología Push.....	42
3.2.1 Estructura.....	43
3.2.2 Mensajes	44
3.2.3 Push y WAP	44
3.3 Java ME	44
3.3.1 Aplicaciones.....	45
3.3.2 Arquitectura.....	45
3.3.3 Mobile Information Profile (MIDP)	47
3.3.4 Java ME y WAP	48
3.5 BREW.....	48
3.5.1 Aplicaciones.....	49
3.5.2 Inconvenientes	49
3.5.2 BREW y Java ME	49
Capítulo 4. Aplicaciones para celulares	50
4.1 Aplicaciones WAP	50
4.1.1 Tipo de aplicaciones WAP	50
4.1.2 Servidor de aplicaciones WAP	50
4.1.3 Lenguajes de programación WAP	51
4.1.4 Entornos de desarrollo de aplicaciones WAP	53
4.1.5 Emuladores WAP.....	53
4.1.6 Seguridad de aplicaciones WAP.....	53
4.1.7 Inconvenientes con aplicaciones WAP.....	53
4.2 Aplicaciones Java ME.....	54
4.2.1 Tipo de aplicaciones Java ME.....	54
4.2.2 El Lenguaje Java	55
4.2.3 Entorno de desarrollo Java	57
4.2.4 Componentes de una aplicación Java ME	58
4.2.5 Acceso a Internet desde Java ME	58
4.2.6 Emuladores Java ME	58
4.2.7 Seguridad en Java ME	59
4.2.8 Inconvenientes con aplicaciones Java ME	59
4.3 Aplicaciones SMS	59
4.3.1 Tipos de aplicaciones SMS	60
4.3.2 El envío, recepción y respuesta de mensajes.....	60
4.3.3 Características de las aplicaciones SMS.....	61
4.3.4 Beneficios de las aplicaciones SMS	61
Capítulo 5. Desarrollo de una aplicación WAP	62
5.1 Objetivo	62
5.2 Motivación	62
5.3 Sistema SIU-Guaraní	62
5.3.1 Descripción técnica	63
5.3.2 Características principales.....	63
5.3.3 Guaraní3w	64
5.4 Aplicación WAP para Guaraní	64
5.4.1 Descripción técnica del prototipo.....	64
5.4.2 Decisiones en cuanto a implementación.....	65
5.4.3 Relevamiento de requerimientos técnicos	66
5.4.4 Relevamiento de requerimientos funcionales.....	66
5.4.5 Análisis de requerimientos técnicos	68
5.4.6 Análisis funcional	71
5.4.7 Diseño de la aplicación	79

5.4.8 Herramientas de desarrollo y prueba	84
5.4.9 Desarrollo del prototipo	84
5.4.10 Configuración del servidor y la aplicación	89
5.4.11 Puesta en marcha del prototipo.....	90
5.5 Resultados – Guaraní WAP	100
Capítulo 6. Conclusiones	106
Anexo A – Celulares con soporte WAP 2.0	108
Anexo B – Configuración de Guaraní WAP	118
Anexo C – Código fuente de Guaraní WAP.....	122
Bibliografía y sitios en Internet	147

Prefacio

La elección del tema surgió a partir de la investigación del autor en relación a las tecnologías disponibles para teléfonos celulares. A medida que comenzaba la exploración llegó a su conocimiento el surgimiento de WAP como protocolo estándar de acceso a Internet desde celulares.

Dado el auge de WAP para el desarrollo de aplicaciones móviles, el tema fue elegido como base para la construcción de un prototipo para el desarrollo de aplicaciones para celulares en la Facultad de Informática de la UNLP.

El surgimiento de nuevas tecnologías durante el proceso de desarrollo de la tesis, como WAP 2.0, Java ME y Brew, llevó al autor a realizar cambios en el plan de actividades y estructura de la tesis, incluyendo la evolución de la telefonía celular y la aparición de estas nuevas tecnologías.

El contacto con el SIU, entidad dependiente del Ministerio de Educación, Ciencia y Tecnología de la Nación, aportó el marco ideal para el desarrollo de un prototipo que pueda ser utilizado no solo en UNLP, sino en el resto de las universidades del país.

Así nace la presente tesis que describe la evolución, tecnologías, infraestructura y servicios de la tecnología celular y especifica un prototipo Guaraní WAP, como resultado del presente trabajo.

Organización de la tesis

La tesis está organizada de la siguiente forma:

- ✓ Introducción, describe brevemente los temas que serán abordados en la tesis.
- ✓ Prefacio, contiene información relevante a los orígenes y desarrollo del trabajo.
- ✓ Historia y evolución de la telefonía celular.
- ✓ Infraestructura y servicios actuales que posibilitan Internet en el celular.
- ✓ Tecnologías inalámbricas para el desarrollo de aplicaciones.
- ✓ Características de las aplicaciones para celulares.
- ✓ Descripción y desarrollo del prototipo.
- ✓ Conclusiones.
- ✓ Anexos.
- ✓ Referencias bibliográficas.

Agradecimientos

Quisiera agradecer a las siguientes personas que han contribuido de una u otra forma durante la elaboración de la tesis. A mi tutor, Francisco Javier Díaz, a las personas de SIU, María del Luján Gurmendi, Alejandro Delú, Víctor Darío Martínez, Darío Spagnuolo y Emilio Luque, al resto de las personas que colaboraron conmigo, Alejandra Osorio, Francisco Vives, Ana Smail y Javier Charne.

Asimismo hago extensivo el agradecimiento a toda mi familia que me ayudó incondicionalmente a que pudiera completar la tesis y demás personas que olvido en este momento mencionar.

Capítulo 1. El celular

El teléfono celular ha sido el dispositivo electrónico que más evolucionado en las últimas dos décadas. El avance de la tecnología ha sido aplicado en gran medida al desarrollo de aparatos más sofisticados con mayores funcionalidades. En esta sección se describe la historia de la telefonía celular, las diferentes generaciones, presente y futuro de esta tecnología.

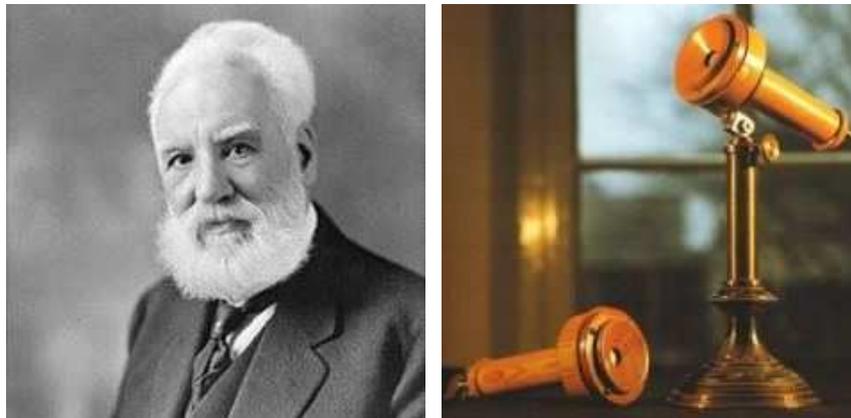
1.1 Historia de la telefonía celular

1.1.1 El teléfono

Hasta el 11 de junio de 2002 los libros de historia nombraban a Alexander Graham Bell (1847-1922) como inventor del teléfono. Bell nació en Edimburgo (Escocia), en el contexto de una familia marcada por los problemas de comunicaciones, su madre era sorda y su padre profesor de lenguaje de signos.

Instalado en Boston, Bell trabajaba en el telégrafo múltiple cuando se le cruzó por la cabeza un nuevo invento, el teléfono. En 1876 concluyó su primer aparato y pronunció la primera frase a través del mismo: "Mr. Watson, come here. I want you".

El 10 de marzo de 1876, presentó en la Exposición del Centenario, celebrada en Filadelfia, el teléfono, una máquina eléctrica que transportaba la voz.



Alexander Graham Bell y su invento

Gracias al patentamiento de su invento, Alexander Graham Bell embolsó una fortuna que reclamó durante años otro inventor, el italiano Antonio Meucci (1808-1889).

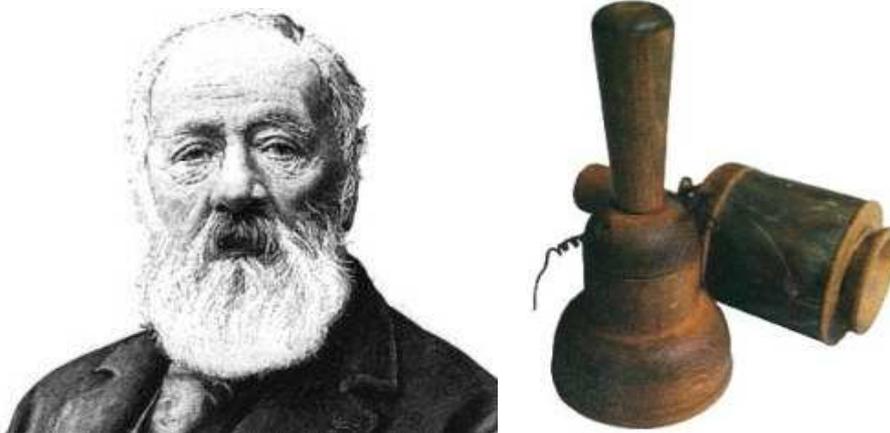
Meucci nació y cursó estudios de Ingeniería Mecánica en Florencia. En la década de 1830 emigró a Cuba a trabajar con enfermos reumáticos, a los que aplicaba pequeñas descargas eléctricas para paliar el dolor. Durante ese trabajo descubrió que la transformación de las vibraciones sonoras en impulsos eléctricos permitía transmitir la voz a distancia, a través de un cable.

En los años cincuenta, ya radicado en Nueva York comenzó a diseñar prototipos telefónicos. Construyó un primer modelo en 1855 y en 1871 presentó un aparato perfeccionado.

Ese mismo año, solicitó en Nueva York la demanda de patente del aparato y lo bautizó con el nombre de "teletrófono". Para renovar el documento de la demanda debía pagar diez dólares cada año. El científico italiano cumplió esta obligación los dos primeros, pero en 1874 no pudo conseguir el dinero y perdió los derechos legales sobre el invento.

Meucci buscó entonces apoyo económico y presentó su invento a la Western Union, pero la famosa compañía de telégrafos rechazó la oferta. En 1876 cuando Graham Bell patentó un aparato de transmisión de voz al que llamó teléfono, el científico italiano reclamó sus derechos en los tribunales pero murió sin que nadie reconociera su aporte a uno de los principales inventos del siglo XX.

Durante más de un siglo se mantuvo el error de atribuir a Graham Bell la paternidad del teléfono hasta que finalmente, el 11 de junio de 2002, el Congreso de los Estados Unidos reconoció oficialmente a Meucci como su verdadero inventor.



Antonio Meucci y su invento

1.1.2 La comunicación inalámbrica

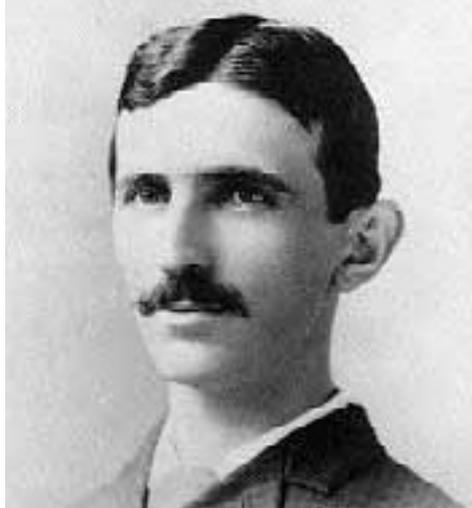
La comunicación inalámbrica tiene sus raíces en la invención del radio por Nikolai Tesla en la década de 1880 (formalmente presentado en 1894 por el italiano Guglielmo Marconi).

Tesla era un físico eléctrico Servio, nacido en Croacia el 1856 que estudió en Austria y Praga. La reputación científica de Tesla se basa en una cadena de invenciones trascendentales tales como corriente alterna (AC), el motor de inducción AC, el velocímetro, la turbina sin hélices, la bobina de Tesla y la radio frecuencia. También bosquejó planos para el radar y la iluminación fluorescente. Nikolai Tesla patentó más de 700 inventos.

La fama de Tesla comenzó a desperdigarse cuando emigró a EEUU donde, con el aval de George Westinghouse, plantó las bases del mundo moderno con su invención y defensa del sistema eléctrico AC.

En 1891 Tesla desarrolló un generador de alta frecuencia y alta tensión con el cual proyectaba transmitir la energía eléctrica sin necesidad de conductores. La Bobina de Tesla es un generador electromagnético que produce altas tensiones de elevadas frecuencias (radiofrecuencias) con efectos observables por el ojo humano como chispas, coronas y arcos eléctricos.

Luego de la creación de la bobina, el inventor pronto descubrió que él podría transmitir y recibir señales de radio de gran alcance cuando eran moduladas en la misma frecuencia. Cuando una bobina se modula a una señal en una frecuencia particular, magnifica literalmente la energía eléctrica entrante con la acción resonante. A principios de 1895, Tesla estaba listo para transmitir una señal 50 millas, pero un incendio destruyó su laboratorio.



Nikolai Tesla

Al mismo momento en Inglaterra, el inventor italiano Guglielmo Marconi estaba trabajando en la construcción de un telégrafo inalámbrico. Con este dispositivo, Marconi fue el primero en realizar una patente sobre tecnología inalámbrica, en Inglaterra en 1896. Su dispositivo tenía solamente un sistema del dos circuitos, el cual solo podía transmitir a pocos metros de distancia. Mas tarde Marconi realizó demostraciones de larga distancia, pero usando un oscilador de Tesla para transmitir las señales a través del canal de la mancha.

Tesla patentó sus propias aplicaciones de radio en 1897. Además la primera patente de Marconi en América, fechada el 10 de noviembre de 1900, fue revocada, al igual que las subsiguientes aplicaciones debido, entre otras cosas, a la prioridad de Tesla y de otros inventores.

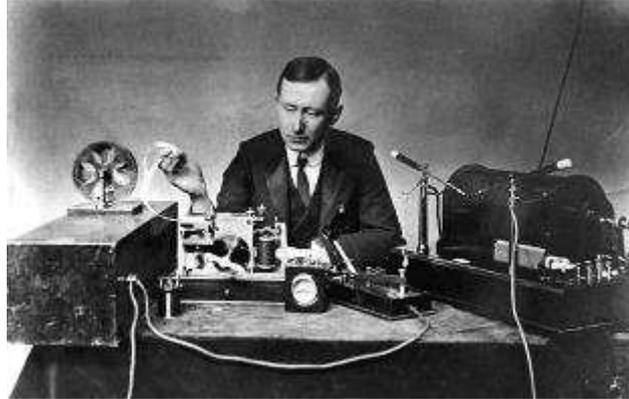
Marconi, cuya familia tenía contactos con la aristocracia inglesa, fundó la Marconi Wireless Telegraph Company, Ltd. que comenzó a crecer en la bolsa de EEUU y consiguió el apoyo de Tomas Edison, quien a esa altura, era el mayor proveedor de electricidad del país.

El 12 de diciembre de 1901, Marconi fue el primero en transmitir y recibir señales a través del Océano Atlántico.

En 1904 la oficina de patentes de EEUU repentina y sorpresivamente revirtió sus previas afirmaciones y le concedió a Marconi la patente por la invención de la radio. Las razones de esto nunca se han explicado completamente, pero el poder financiero de Marconi en los Estados Unidos sugiere una explicación posible.

Cuando Marconi ganó el premio Nobel en 1911, Tesla demandó a Marconi Company por plagio en 1915, pero no estaba en condiciones financieras para litigar en el caso contra una corporación semejante.

Recién en 1943, a pocos meses después de la muerte de Tesla, la Corte Suprema de EEUU dispuso que la patente de radio correspondiera a Tesla. La medida no fue en reconocimiento a Tesla, sino meramente a razones políticas. Marconi Company había demandado al gobierno de Estados Unidos por el uso de sus patentes en la Primera Guerra Mundial. Entonces la corte restauró la patente a Tesla simplemente para evitar el juicio.



Guglielmo Marconi y su invento

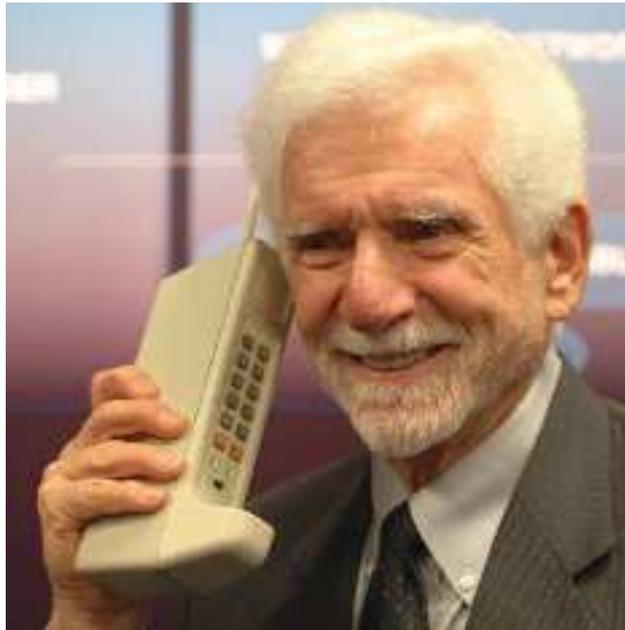
1.1.3 La telefonía celular

El concepto básico de teléfonos portátiles comenzó en 1947, cuando los investigadores buscaban un teléfono para usar en automóviles y descubrieron que usando rangos de áreas de servicio (células) y reutilizando frecuencia, podrían aumentar la capacidad de alcance de los teléfonos móviles substancialmente.

En ese año, AT&T solicitó que la FCC (Federal Communications Commission) asigne frecuencias de radio-espectro de modo que el servicio telefónico móvil pueda ser factible y AT&T tuviera un incentivo para investigar la nueva tecnología. Sin embargo la FCC limitó a la cantidad de frecuencias entregadas, disponiendo de solo veintitrés conversaciones posibles simultáneamente en la misma área de servicio.

En 1968, AT&T y Bell propusieron a la FCC un sistema celular compuesto de muchas y pequeñas torres de difusión de bajo alcance, donde cada una cubriera una 'célula' consistente en unas pocas millas en radio y colectivamente cubrieran un área más grande. Cada torre utilizaría solamente algunas de las frecuencias totales asignadas al sistema. Mientras los teléfonos viajaban a través de las áreas, las llamadas serían transferidas de torre a torre.

La primera llamada desde un teléfono celular la realizó Martin Cooper, ex director general de sistemas de Motorola, quien es considerado el inventor del teléfono celular. Cooper hizo la llamada justamente a su rival, Joel Engel, jefe de investigaciones de Bell Labs.



Martin Cooper con el primer teléfono celular

En 1977, AT&T y Bell comenzaron ensayos públicos con un prototipo de sistema celular en Chicago con unos 2000 clientes. Pero en 1979 la empresa japonesa NTT inició el primer sistema comercial en Tokio. En 1981, Motorola y American Radio comenzaron una segunda prueba de sistema celular en Washington. Durante el mismo año, los países nórdicos introdujeron un sistema celular similar a AMPS (Advanced Mobile Phone System). Finalmente en 1982, la FCC autorizó el servicio celular comercial para EEUU. Un año más tarde, se puso en operación el primer sistema comercial AMPS en la ciudad de Chicago.

Con ese punto de partida, en varios países se diseminó la telefonía celular como una alternativa a la telefonía convencional inalámbrica. La tecnología tuvo gran aceptación, por lo que a los pocos años de implantarse se empezó a saturar el servicio. En ese sentido, hubo la necesidad de desarrollar e implantar otras formas de acceso múltiple al canal y transformar los sistemas analógicos a digitales, con el objeto de dar cabida a más usuarios.

1.1.3.1 Historia celular Argentina

En Argentina comenzaron a verse los primeros teléfonos celulares de la mano de la empresa CRM, Compañía de Radiocomunicaciones Móviles S.A., denominada comercialmente Movicom, que operó la primera banda para el Área Múltiple Buenos Aires (AMBA) y su extensión. Luego de un año ya había 12.000 usuarios y a mediados de 1993 la cifra alcanzaba los 77.500 clientes. A partir de 1993 Movistar S.A., denominada comercialmente Miniphone, integrada en partes iguales por Telecom y Telefónica, prestó servicios de la segunda banda del servicio de radiocomunicaciones móvil celular en la misma zona.

En septiembre de 1994, la CTI, Compañía de Teléfonos del Interior, inició, con exclusividad hasta marzo de 1996, el servicio de telefonía celular móvil en el interior del país que cubre los principales centros urbanos y corredores viales. Luego de 2 años, en marzo de 1996, Telefónica Comunicaciones Personales, de Telefónica de Argentina (UNIFON) se presentó en sociedad

ofreciendo el servicio en la región sur y en mayo le toco el turno a Compañía de Comunicaciones Personales, de Telecom Argentina (PERSONAL) para la región norte.

En 2005 la compra de Bellsouth Latinoamérica por parte de Telefónica móviles, produce la unificación de Movicom y Unifon en Movistar, consolidada como la primera compañía de celulares en Argentina.

1.2 Generaciones de la telefonía celular

Para separar una etapa de la otra, la telefonía celular se ha caracterizado por contar con diferentes generaciones.

1.2.1 Primera generación (1G)

La 1G de la telefonía móvil hizo su aparición en 1979 y se caracterizó por ser analógica y estrictamente para voz. La calidad de los enlaces era muy baja al igual que su velocidad (2400 bauds). En cuanto a la transferencia entre celdas, era muy imprecisa ya que contaban con una baja capacidad (Basadas en FDMA, Frequency Division Multiple Access) y, además, la seguridad no existía. La tecnología predominante de esta generación es AMPS (Advanced Mobile Phone System).

1.2.2 Segunda generación (2G)

La 2G arribó en 1990 y a diferencia de la primera se caracterizó por ser digital.

EL sistema 2G utiliza protocolos de codificación más sofisticados que se emplean en los sistemas de telefonía celular actuales. Las tecnologías predominantes son: GSM (Global System Mobile Communications), IS-136, CDMA (Code Division Multiple Access) y PDC (Personal Digital Communications), éste último utilizado en Japón.

Los protocolos empleados en los sistemas 2G soportan velocidades de información más altas por voz, pero limitados en comunicación de datos. Se pueden ofrecer servicios auxiliares, como datos, fax y SMS (Short Message Service). La mayoría de los protocolos de 2G ofrecen diferentes niveles de encriptación.

1.2.3 Generación 2.5 G

La generación 2.5G ofrece características extendidas, ya que cuenta con más capacidades adicionales que los sistemas 2G, como: GPRS (General Packet Radio System), HSCSD (High Speed Circuit Switched), EDGE (Enhanced Data Rates for Global Evolution), IS-136B e IS-95Bm entre otros.

Esta tecnología es la que conocemos en Argentina en la actualidad.

1.2.4 Tercera generación (3G)

La 3G se caracteriza por contener a la convergencia de voz y datos con acceso inalámbrico a Internet; es apta para aplicaciones multimedia y altas transmisiones de datos.

Los protocolos empleados en los sistemas 3G soportan altas velocidades de información y están enfocados para aplicaciones más allá de la voz como audio (mp3), video en movimiento,

videoconferencia y acceso rápido a Internet, entre otras.

Asimismo, los sistemas 3G pueden alcanzar velocidades de hasta 384 Kbps, permitiendo viajar a usuarios a 120 kilómetros por hora en ambientes exteriores. En ambientes estacionarios de corto alcance o en interiores, la velocidad máxima de transferencia llega a los 2 Mbps.

Si bien se esperaba que la 3G tenga su auge entre 2003 y 2005 esto aún no ha ocurrido en la actualidad.

3G incluye las siguientes características:

- Multimedia mejorada (voz, datos, video, etc.)
- Diferentes formas de uso (telefonía celular, e-mail, pager, radio, fax, videoconferencia y explorador Web)
- Gran ancho de banda y alta velocidad (hasta 2 Mbps)
- Flexibilidad de ruteo (repetidores, satélites, LAN)
- Operación en frecuencia de transmisión y recepción de 2 GHz.
- Roaming entre continentes.

La 3G se utiliza con éxito en Japón y Corea del Sur desde el año 2001, sin embargo en Europa y Estados Unidos su masificación recién ha comenzado en 2005.

1.2.5 Cuarta generación (4G)

4G es la cuarta generación en tecnología de acceso inalámbrico mediante ondas de radio. Esta ha sido testada en Japón con transferencias de hasta 1Gbps a 20km/h.

4G no sólo define un estándar, sino que proporcionará un ambiente donde las conexiones podrán operar entre sí, para proporcionar la sensación de interacción en tiempo real con los distintos servicios multimedia como vídeo de alta calidad y videoconferencia, entre otros. Esperada para 2010, 4G cambiará radicalmente la forma a la cual el mundo ha estado acostumbrado a conectarse aumentando 10 veces las conexiones 3G.

Actualmente la compañía japonesa NTT DoCoMo realizó pruebas de este servicio, con buenos resultados.

De la tecnología 4G se espera obtener versiones mejoradas de las características prometidas en 3G, como por ejemplo Roaming en todo el planeta.

1.3 El celular en la actualidad

Más allá del uso como dispositivo de comunicación entre personas e inclusive como dispositivo de transmisión de datos, el teléfono celular ha evolucionado hacia nuevos usos. La aparición de servicios multimedia como cámaras digitales y walkman, son ejemplos de ello. El teléfono celular hasta se ha convertido en un símbolo de status social, como el reloj de pulsera, ya que han aparecido celulares con carcasa de oro, e inclusive con cobertura de diamantes.

1.3.1 Servicios tecnológicos

Movilidad, personalización y ubicuidad: estos paradigmas de la sociedad de la información se cumplen puntualmente en la evolución del celular. Mientras el teléfono fijo comunica a lugares, el celular comunica personas con métodos en permanente evolución.

La masificación de la tecnología celular, guió a los fabricantes de dispositivos y proveedores del servicio a pensar en nuevos horizontes comerciales. Entonces pensaron en la posibilidad de tener "Internet en la palma de la mano". La aparición de la tecnología WAP permitió esta funcionalidad, incluso con las limitaciones tecnológicas de los dispositivos disponibles a fines del siglo XX y las redes AMPS. La evolución hacia las redes y celulares 2G mejoró en buena medida el acceso a Internet, sin embargo, el negocio celular vino de la mano de otras ideas y soluciones tecnológicas.

La posibilidad de envío de mensajes de texto simple (SMS), una tendencia que se impuso en Europa y Asia hace algunos años, desembarcó en Argentina con fuerza luego de la crisis del 2001. En sólo un año, entre el 2003 y 2004, el porcentaje de usuarios de celulares que utilizan SMS pasó del 7% al 44%. El SMS permitió a las personas comunicarse de forma instantánea a un costo considerablemente menor al de una llamada.

La aparición de los celulares con pantalla color en el año 2001, impactó a las personas y muchas no dudaron en adquirir uno sin importar cuanto costara. El hecho de que los celulares fueran a color abrió un mundo de posibilidades para adaptarles nuevas funciones como por ejemplo una cámara fotográfica, o adosarle un fondo de pantalla. Mas adelante se agregó una característica muy importante que fue la de grabar videos y poderlos enviar como mensajes multimedia. Este último concepto, orientó a las empresas a crear teléfonos destinados al uso multimedia, llegando al extremo de poder ver videos, sacar fotos, escuchar música e incluso jugar con gráficos en 3D, gracias a la inclusión de memorias flash y RAM en los dispositivos.

Los servicios asociados como cámaras digitales, descargas de ringtones, canciones, fondos de pantalla, juegos han invadido el mercado y prácticamente no se venden celulares que no ofrezcan alguno de estos servicios.



Teléfono celular de la marca Sony Ericsson

1.3.2 Aspectos adicionales

Además de los servicios tecnológicos que pueda brindar un teléfono celular, hay ciertos aspectos que han producido que el celular tienda a convertirse en un elemento complementario a la persona logrando una relación muy estrecha entre ambos.

La movilidad, definida como la capacidad de acceder a la información y a los servicios en cualquier momento, de cualquier manera y en cualquier parte, ha hecho del celular un elemento muy útil para algunas personas e indispensable para otras.

La seguridad personal hoy en día está también ligada al celular. La posibilidad de recurrir a servicios de policía, bomberos y/o emergencias médicas, el hecho de permitir a un padre ubicar a sus hijos adolescentes, o de recibir una notificación por algún evento (como puede ser el disparo de una alarma domiciliaria), permiten a una persona disponer del celular como una medida de seguridad personal. Por ejemplo, Walt Disney, ha lanzado un servicio de telefonía celular dirigido a los niños en los Estados Unidos con un servicio por el cual los padres podrán controlar la utilización de los celulares por parte de los niños y conocer su localización en cualquier momento. De esta manera, los padres podrán controlar quien envía mensajes de texto al celular de sus hijos o con quien mantienen conversaciones telefónicas.

No puede obviarse tampoco el uso del celular como diferenciador de condición social, el hecho de que existan celulares realizados en oro 24Kilates con precios de mas de €2000, o inclusive un excéntrico celular de 120 Kilates de diamantes que cuesta mas de u\$s1.000.000 evidencia que la excentricidad ligada a las joyas y relojes, ha llegado a los celulares.



Celular de oro



Goldvish – Celular de oro y diamantes

1.3.3 El negocio

Una vez popularizado el servicio, a la comunicación interpersonal se le suma una gama de posibilidades que van desde negocios de contenidos especiales para estos sistemas hasta posibilidades de expresiones culturales.

La mayor parte de los expertos aseguran que el teléfono móvil será el dispositivo dominante a la hora de conectarse a la Internet del futuro. Además, estos dispositivos a diferencia de la PC, conseguirán que los usuarios paguen por los contenidos que se descargan.

Sin embargo, el negocio de Internet en celulares, en la actualidad ha sido postergado por nichos comerciales más atractivos. En Argentina, el 20% de la facturación de los celulares es por servicios asociados como ringtones, mensajes de textos, cambio de aparato y juegos.

El envío de mensajes de texto simple (SMS) se utiliza alrededor del planeta no solo para comunicarse con otras personas sino también para participar en concursos, encuestas o juegos, donde el usuario participa solamente enviando uno o múltiples mensajes de texto. Es también una herramienta funcional para acceder a servicios de Home Banking u otras aplicaciones de servicios como pago telefónico, compras electrónicas, alquiler de películas, reservación de hoteles y hasta conciertos.

Los usuarios de teléfonos celulares se familiarizan velozmente con las nuevas funciones de sus dispositivos. En Argentina, el 14% de los usuarios de celular bajó a su equipo alguna aplicación desde Internet como ringtones, juegos, salvapantallas, fondos de pantallas.

Si bien la descarga de ringtones, mp3 o wallpapers necesitan una conexión a Internet, no puede englobarse a estos servicios como uso de Internet a través de un celular en reemplazo de la PC.

1.4 Datos de la telefonía celular

Según la Asociación Internacional de GSM, que reúne a alrededor de 700 operadores de telecomunicaciones de todo el mundo, la cantidad de usuarios de teléfonos celulares con tecnología GSM ya llegó a los 2 mil millones en todo el mundo. Incluso, el organismo internacional aseguró que son más de mil los nuevos usuarios que se incorporan en todo el mundo por minuto a esta tecnología.

En Latinoamérica, las cifras dan cuenta de la importancia de la industria móvil en la región. Las empresas de telefonía celular han invertido unos 16 mil millones de dólares entre 2002 y 2004, lo que representa un 28% de sus ingresos comparado con los 13 a 18% en otras regiones como EE.UU. o Unión Europea. Eso genera alrededor de 10 mil millones de dólares anualmente en ingresos para los gobiernos (IVA, impuestos corporativos, aranceles, pagos de seguro social). Las inversiones móviles representan el 2,5% del total de inversiones de todas las industrias, y los servicios móviles contribuyen por sobre el 1,14% al PBI. Por último la tecnología celular aporta 2,3 millones de empleos, directa o indirectamente.

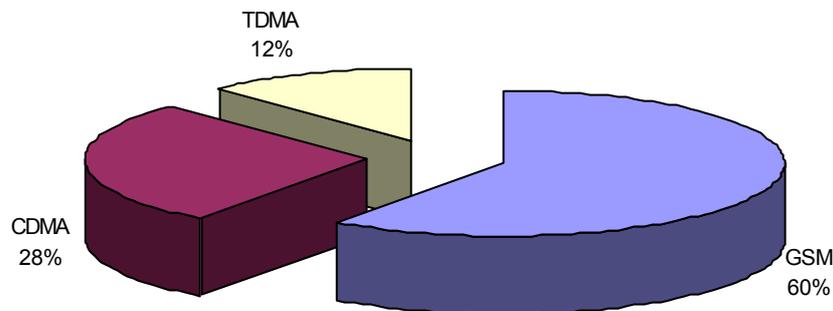
El proyecto Emerging Market Handset, creado con el objetivo de desarrollar un teléfono de bajo costo para promover la inclusión digital está siendo evaluado en nuestra región. Esta iniciativa

ha sido sumamente exitosas en Asia y África, donde se ha entendido con claridad el tremendo aporte que tiene para los países que una mayor parte de la población utilice servicios móviles. Además, se trata de países donde los niveles de impuestos son notoriamente más bajos que los que se observan en nuestra región, cuestión que ayuda notablemente a cumplir el objetivo de propagar lo más posible, y en el menor tiempo que se pueda, los servicios móviles en la población.

1.4.1 En Argentina

El Instituto Nacional de Estadísticas y Censos (INDEC) informó que para fin del año 2006 se espera que haya más de 30 millones de teléfonos celulares en funcionamiento en nuestro país.

Las ventas de celulares se multiplicaron en los últimos tiempos y permitieron acentuar los fuertes indicadores de consumo, con una penetración superior al 60% de la población. Desde 2004 en Argentina la cantidad de líneas móviles superaron a las fijas y actualmente hay alrededor de 21 millones de celulares. La tecnología predominante es GSM con un 60%, luego CDMA con 28% y TDMA con 12%.



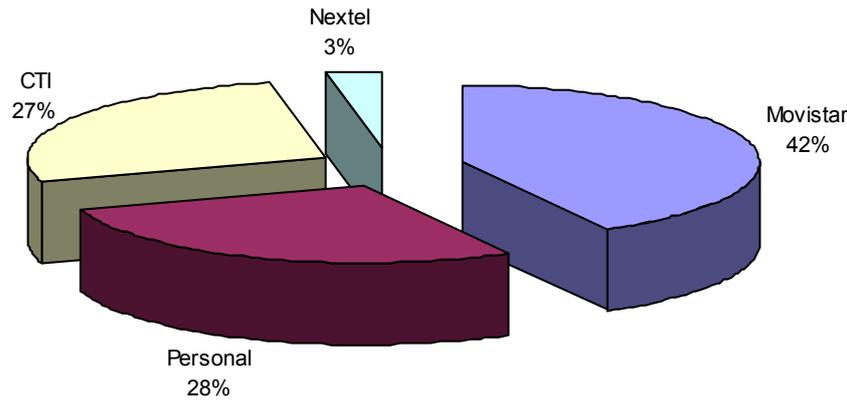
Tecnologías celulares en Argentina

Otro dato significativo es que la mayoría de los teléfonos en servicio son nuevos, ya que según las encuestas, el 80% tiene menos de dos años, y que el 27% de los usuarios planea la compra de un nuevo equipo en los próximos 6 meses.

Movistar, surgida a partir de la fusión entre Unifon y Movicom, es la compañía que lidera el mercado en nuestro país con un 42% de los usuarios de celulares. La empresa contabiliza 700 millones de mensajes de texto (SMS) por mes y 750.000 mensajes multimedia. Entre sus estadísticas también figuran los 1,5 millones de ringtones por mes, entre polifónicos y monofónicos.

Personal, la compañía celular de Telecom, que posee el 28% del mercado, registra unos 600 millones de SMS por mes, es decir un promedio de 92 mensajes por usuario. Esto significa, un 40% por encima de lo que suelen utilizar los usuarios europeos. Además el 20% de sus clientes cambia de ringtone al menos una vez por mes.

El mercado se completa con CTI que posee el 27% del mercado y Nextel con un 3%.



Operadoras de telefonía celular en Argentina

Con respecto a dispositivos, ya operan 20 marcas de teléfonos celulares en nuestro país. Palm y BalckBerry dominan el segmento ejecutivo. Motorola, Nokia, Samsung, Sony Ericsson, Siemens y Kyocera, son las marcas preferidas para el uso personal. La importación de dispositivos fue superior a los nueve millones en los últimos dos años. Gran parte se volcó a los nuevos clientes, pero otro tanto fue para abastecer el recambio de equipos.

1.5 El futuro en Argentina

Según los especialistas, al momento de migrar a una tecnología de tercera generación (como UTMS o WCDMA) es mejor contar previamente con tecnología CDMA en vez de GSM. A partir de la fusión de Unifón y Movicon en Movistar, se abrió en el país un debate sobre el tipo de tecnología que debe adoptarse. La migración de CDMA a GSM, que para algunos significa un retroceso tecnológico, es una decisión de naturaleza estratégica, vinculada a la actividad industrial.

A partir de la fusión, Movistar deberá optar por una de ambas tecnologías y ceder 42.5 MHz para ser utilizados por otra empresa. Desde el comienzo de las negociaciones, las cooperativas eran las principales beneficiadas ya que pasarían a conformar una nueva compañía celular pero hasta el momento, no existen fechas concretas para que Movistar ceda la frecuencia. Aparentemente, las cooperativas no tienen capital para tal emprendimiento de telefonía celular y es bastante improbable que puedan flanquear la capacidad ya instalada de Movistar, Personal y CTI. Igualmente, existe un nicho interesante donde las grandes empresas no llegan, ya sea porque no les interesa comercialmente o porque no tienen la cantidad suficiente de antenas para instalar.

A través de la resolución 268/04, la Secretaría de Comunicaciones de la Nación recomendó la unificación de estándares, lo que significó un voto a favor de GSM, que tiende a su consolidación en nuestro país. La venta de dispositivos actuales, las promociones y los servicios ofrecidos por las empresas evidencian esta tendencia. Las tecnologías 3G deberán seguir esperando evolucionar en Europa y EEUU para arribar a la región.

La llegada de la nueva generación, implicará un recambio de dispositivos, pero como ha ocurrido con el auge de GSM, se espera que el costo sea subsidiado por las operadoras de telefonía móvil, que no tienen interés en perder ningún cliente.

Capítulo 2. Infraestructura y Servicios

Como parte de la infraestructura celular, las redes inalámbricas o portadores proveen el medio para el funcionamiento de las comunicaciones inalámbricas. El avance tecnológico de los dispositivos, la capacidad de las nuevas redes y la sofisticación de los sistemas operativos para dispositivos inalámbricos, ha permitido a los fabricantes de teléfonos móviles y desarrolladores incorporar nuevos servicios.

2.1 Operador de red

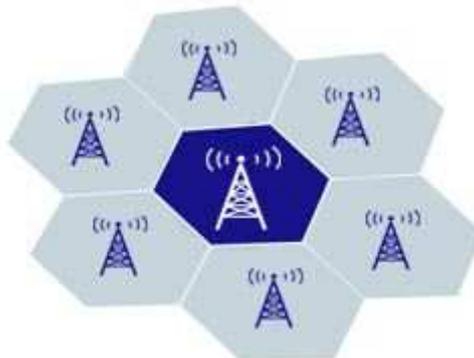
El operador de red es la empresa de telefonía o comunicaciones que provee el servicio de transporte a sus abonados. Un operador de red hace posible la comunicación entre teléfonos celulares y de línea, y además suministra otros servicios como, por ejemplo, buzón de voz, desvío de llamada, identificación de llamada, etc. También debe proveer el resto de los servicios de intercambio de datos como acceso a Internet, SMS o MMS.

2.2 Redes inalámbricas

Para cubrir un área con una red inalámbrica, se divide una región en secciones llamadas células, razón por la cual se las denomina redes celulares. Cada antena contiene una estación base o antena que comunica los dispositivos móviles.

Las estaciones base están agrupadas y controladas por un controlador de estaciones base que está junto a un centro de conmutación móvil. El controlador tiene acceso a una red fija y a la red inalámbrica, de esta manera los clientes pueden comunicarse tanto a teléfonos móviles como fijos.

Cuando un teléfono móvil está conectado, está comunicándose con la estación base. Si durante la conexión el dispositivo se está moviendo, cada vez que sale del área cubierta por una estación base se inicia un procedimiento para pasar la comunicación a la base que cubre el área en la que se encuentra.



Esquema de celdas

2.3 Protocolos móviles

A lo largo de la historia de la telefonía celular, han surgido diversos protocolos para intercambio de voz y datos entre los dispositivos.

2.3.1 TDMA (Time Division Multiple Access)

Es una tecnología inalámbrica que brinda servicios de alta calidad de voz y datos. TDMA distribuye las unidades de información en alternantes espacios de tiempo proveyendo acceso múltiple a un reducido número de frecuencias. Divide un único canal de frecuencia de radio en seis ranuras de tiempo. A cada persona que hace una llamada se le asigna una ranura de tiempo específica para la transmisión, lo que hace posible que varios usuarios utilicen un mismo canal simultáneamente sin interferir entre sí.

Se la relaciona con la segunda generación de telefonía celular ya que existen varios estándares digitales basados en TDMA como TDMA D-AMPS (Digital-Advanced Mobile Phone System), TDMA D-AMPS-1900, PCS-1900 (Personal Communication Services), GSM (Global System for Mobile Communication), DCS-1800 (Digital Communications System) o PDC (Personal Digital Cellular). También están basados en TDMA los estándares analógicos como AMPS, NMT (Nordic Mobile Telephone) y TACS (Total Access System).

2.3.2 AMPS (Advanced Mobile Phone System)

Es un sistema de telefonía móvil analógica de primera generación desarrollado por los laboratorios Bell. Se implementó por primera vez en 1982 en Estados Unidos. Se llegó a implantar también en Europa y Asia.

AMPS y los sistemas telefónicos móviles del mismo tipo dividen el espacio geográfico en celdas, de tal forma que las celdas adyacentes nunca usan las mismas frecuencias, para evitar interferencias. La estación base de cada celda emite con una potencia pequeña que favorece la reutilización de frecuencias y aumenta la capacidad del sistema. Sin embargo, también requiere un mayor número de estaciones base y por tanto una mayor inversión.

AMPS está siendo reemplazado por sistemas digitales tales como GSM y D-AMPS, pero ha sido un sistema de importancia histórica capital para el desarrollo de las comunicaciones móviles por el éxito obtenido y por las ideas novedosas que aportaba. Muchas operadoras todavía la usan como tecnología de respaldo.

2.3.3 GSM (Global System for Mobile Communications)

 Creado en 1991 en Europa, es el protocolo móvil que opera en la banda de frecuencia entre 900 y 1800 MHz en Europa, Asia y Australia, y en la frecuencia de 1900 MHz en Norte América, Latinoamérica y África. Es el estándar móvil que más se utiliza en el mundo en más de 100 países. Utiliza una banda estrecha TDMA y la misma radiofrecuencia puede ser ocupada por ocho llamadas simultáneas. Proporciona correo de voz integrado, alta

velocidad de datos, fax, pagin y servicio de mensajes cortos, así como comunicaciones seguras. Ofrece la mejor calidad de voz que cualquier estándar inalámbrico digital actual.

2.3.4 HSCSD (High Speed Circuit Switched Data)

Protocolo móvil modificador de circuitos basado en GSM. Es uno de los sistemas que pueden dar la opción de integración de datos en dispositivos móviles. Basados en circuitos de alta velocidad, mejora la codificación y posibilita un flujo de datos que va de 28.8 a 56Kbps, usando simplemente cuatro canales de radio simultáneamente.

Ofrece la posibilidad de acceder a varios servicios al mismo tiempo gracias a que funciona mediante múltiples canales independientes integrados en uno. El principal problema es que actualmente sólo Nokia suministra tarjetas de módem PCMCIA (CardPhone 2.0) para clientes HSCSD. El terminal típico para HSCSD es un PC móvil más que un teléfono.

2.3.5 GPRS (General Packet Radio Service)



Protocolo inalámbrico de circuitos de paquetes conmutados que ofrece acceso instantáneo a protocolos IP y a redes X.25. Permite Internet inalámbrico de alta velocidad con transmisiones de hasta 115 Kbits/s. La ventaja objetiva de GPRS es que ofrece una conexión permanente (es decir conectividad IP instantánea) entre el terminal móvil y la red a una velocidad tres o cuatro veces más rápida que GSM. Para impulsar el desarrollo de las aplicaciones GPRS se fundó en octubre de 1999 por Ericsson, Palm, Lotus, Oracle y Symbian la Alianza para Aplicaciones GPRS.

2.3.6 EDGE (Enhanced Data rate for GSM Evolution)

Versión de GPRS de banda más amplia que permite velocidades de transmisión de hasta 384Kbits/s. Es una evolución del estándar GSM para manejar servicios de la tercera generación de telefonía móvil. EDGE permite a los operadores móviles ofrecer alta velocidad y aplicaciones móviles multimedia. Usa la misma estructura de frames TDMA, canal lógico, y ancho de banda que las redes GSM actuales lo que permite que no haya que modificar las células existentes.

2.3.7 CDMA (Code Division Multiple Access)

También conocido como CDMAOne o IS-95 es una tecnología digital para brindar servicios de telefonía móvil.

CDMA es una tecnología de ancho espectro, permitiendo que muchos usuarios posean la misma asignación de tiempo y frecuencia en un ancho de banda dado. CDMA asigna códigos únicos a cada comunicación para distinguirla de otras en el mismo espectro. Las redes CDMA operan en bandas de frecuencia de 800 y 1900 MHz principalmente en América y Asia. La tecnología CDMA IS-95 provee voz y datos hasta velocidades de 64 kbits/sec, así como voice mail integrado y SMS.

2.3.8 CDMA-2000 (Code Division Multiple Access 2000)

También conocido como IS-2000, es una evolución de CDMAOne/IS-95 para brindar servicios 3G. Entrega mayor capacidad de red para resolver la demanda creciente de los servicios inalámbricos y los servicios de datos de alta velocidad. El CDMA2000 1X fue la primera tecnología 3G del mundo utilizado comercialmente.

El componente dominante de los sistemas CDMA2000 es una nueva Packet Core Network (PCN) que permite la entrega de servicios de datos empaquetados con mayor velocidad y seguridad. La PCN de CDMA2000 es uno de los primeros pasos en la evolución de los sistemas CDMA2000 hacia una arquitectura 100% IP y multimedia.

2.3.9 3G (Third Generation)

3G es una abreviatura para tercera-generación de telefonía móvil. Los servicios asociados con la tercera generación proporcionan la posibilidad para transferir tanto voz y datos (una llamada telefónica) y datos como la descarga de programas, intercambio de correo electrónico, y mensajería instantánea.

3G combina el acceso móvil de alta velocidad con servicios basados en Internet Protocol (IP). El estándar formal para 3G es el IMT-2000 (International Mobile Telecommunications 2000). Este estándar ha recibido diversos impulsos de diferentes comunidades de desarrolladores como CDMA-2000, o WCDMA con capacidades 8 veces mayor que las CDMA actuales y soportada por Ericsson, Nokia y los fabricantes japoneses de dispositivos de mano.

2.3.10 WCDMA (Wideband Code Division Multiple Access)

Es una tecnología para comunicaciones digitales radio de banda ancha de Internet, multimedia, vídeo y otras aplicaciones de gran demanda de capacidad. WCDMA se ha seleccionado como la tercera generación para los sistemas de telefonía móvil en Europa, Japón y Estados Unidos.

2.3.11 UMTS (Universal Mobile Telephone System)

Es el sistema de telecomunicaciones móviles de tercera generación que utiliza tecnología WCDMA. UMTS es el resultado de la evolución GSM y GPRS que posibilita velocidades de transmisión de hasta 2Mbps, alta seguridad y confidencialidad, alta resistencia a interferencias, roaming y cobertura a nivel mundial.

2.4 Sistemas operativos

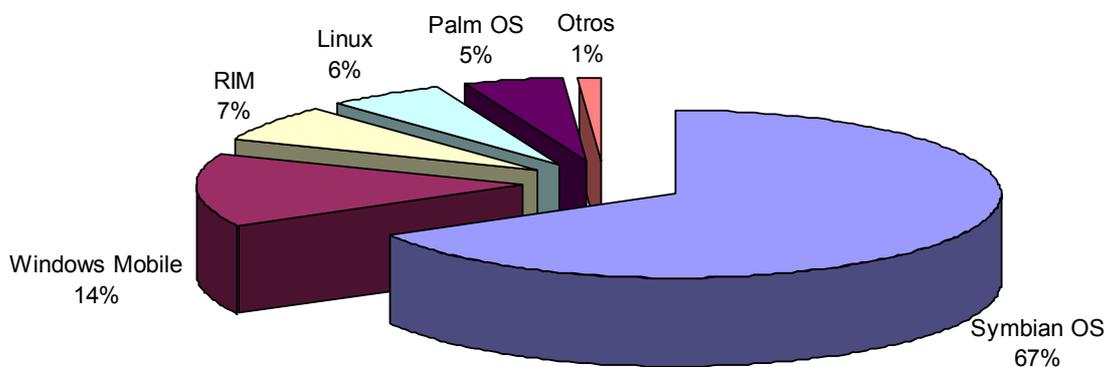
Para que el dispositivo móvil pueda ejecutar aplicaciones o acceder a un portal de Internet, debe contener un sistema operativo que lo soporte. A continuación se describen los principales sistemas operativos utilizados por los teléfonos celulares. Los primeros dispositivos inalámbricos contenían sistemas operativos nativos de funcionalidad básica. En la actualidad los sistemas operativos para celulares soportan la ejecución de una infinidad de servicios y aplicaciones.

La existencia de decenas de sistemas operativos en la telefonía móvil se ha convertido en un grave problema para la explosión de los nuevos contenidos y aplicaciones para los terminales celulares. Este obstáculo afecta por igual a los operadores y a los desarrolladores de software y contenidos, que se ven abocados a dispersar sus esfuerzos.

El problema es que hoy conviven en el mundo móvil los sistemas operativos Symbian, Windows Mobile, Palm, RIM (Research In Motion) y Linux, que son los más utilizados en los smartphones (equipos de alta gama que permiten conexión a Internet, envío de e-mail y uso de aplicaciones de gestión), con otros muchos sistemas operativos creados por los propios fabricantes de teléfonos móviles, lo que obliga a hacer múltiples adaptaciones tanto del contenido gráfico como de audio para entregar el mismo contenido a los diferentes dispositivos.

Algunas operadoras de telefonía móvil han concentrado sus esfuerzos en un subconjunto de sistemas operativos. Vodafone, por ejemplo, solo aceptará teléfonos que utilicen Windows Mobile de Microsoft, Symbian Serie 60 y Linux. También la japonesa NTT DoCoMo ha decidido impulsar sus proyectos en Symbian y en el sistema operativo de código abierto Linux. Telefónica Móviles, por su parte, asegura que trabaja con Symbian y Windows Mobile, pero que ha comenzado a hacerlo con Linux.

En el segmento de los smartphones, un 67% de los dispositivos de todo el mundo llevan el sistema operativo Symbian. Windows Mobile tendría el 14% frente al 7% de RIM, el 6% de Linux y el 5% de PalmOS. El sistema de Microsoft es el más fuerte en EEUU, mientras en Europa el dominio corresponde a Symbian. Este modelo también es el líder en China, donde está tomando mucha fuerza Linux, que ya tendría un 30% de este mercado asiático.



Sistemas operativos para smartphones

2.4.1 Symbian

symbian Es un sistema operativo diseñado para teléfonos celulares que incluye librerías, frameworks de interfaces de usuario, y especificaciones para implementación de aplicaciones y herramientas. Fue desarrollado por la empresa Symbian Ltd. empresa que actualmente pertenece a Ericsson, Panasonic, Siemens AG, Nokia y Sony Ericsson.

Symbian OS es estructurado y multitarea, como la mayoría de los sistemas operativos de escritorio actuales. La principal ventaja es el hecho de haber sido diseñado para dispositivos móviles, donde los recursos como memoria, disco y batería son limitados.



Celular con Symbian

2.4.2 Windows Mobile



El sistema operativo para teléfonos celulares de Microsoft incluye versiones móviles de las principales herramientas de conectividad que contiene cualquier Windows de escritorio. Además de realizar llamadas y enviar y recibir SMS, permite utilizar Outlook para enviar y recibir correo electrónico, navegar por Internet e incluso utilizar versiones móviles del paquete Office, incluyendo Word y Excel.



Pantalla de Windows Mobile 5.1

2.4.3 RIM



Research In Motion (RIM) es un sistema operativo propietario desarrollado por la empresa homónima para sus dispositivos BlackBerry. RIM OS posee aplicaciones para sincronizar información con una PC, enviar y recibir mails, mantener una libreta de direcciones, ingresar a Internet, ejecutar aplicaciones Java, tomar fotos y videos, entre otras.



BlackBerry Pearl con RIM

2.4.4 Palm OS



Palm OS es un sistema operativo desarrollado para PDA's, desarrollado por PalmSource Inc. Ha sido diseñado para ser fácil de usar y similar en varios aspectos a Microsoft Windows. Palm OS contiene un conjunto de aplicaciones básicas como reloj, calendario, alarma, libreta de direcciones y block de notas, así como la posibilidad de sincronizar información con una PC. Las versiones mas recientes de Palm OS incluyen avanzados editores de texto y planillas de cálculo.

En Septiembre de 2005 la empresa PalmSource fue adquirida por la compañía Japonesa ACCESS. A partir de tal adquisición, comenzó la diversificación de los sistemas operativos en los dispositivos Palm, permitiendo la aparición de Windows Mobile e inclusive Linux en reemplazo de Palm OS.

2.5 Servicios

2.5.1 SMS (Short Message Service)

Es un servicio disponible en los teléfonos móviles que permite el envío de mensajes cortos (también conocidos como mensajes de texto) entre teléfonos móviles, teléfonos fijos y otros dispositivos de mano. Además del envío de mensajes, SMS puede utilizarse para ordenar servicios como descargas de tonos, fondos de pantalla e incluso para participar de encuestas o competencias. La posibilidad de que el usuario pague por cada mensaje enviado o recibido, permitió nuevos servicios como respuesta a un simple mensaje de texto. Hoy día es muy común ver publicidades donde con solo enviar un mensaje de texto, uno puede recibir su horóscopo, consejos, chistes o incluso participar en sorteos y juegos. Una de las funciones que facilita el envío de SMS es la escritura predictiva, donde a medida que se escribe una letra, en la pantalla aparecen palabras posibles. También hay mensajes prearmados y pueden crearse y grabar propios.

Si bien la idea del envío de mensajes entre celulares viene de larga data, SMS fue diseñado como parte del estándar de telefonía móvil digital GSM y en la actualidad está disponible en una amplia variedad de redes, incluyendo las redes 3G.

2.5.2 MMS (Multimedia Messaging System)

Es un estándar de mensajería que le permite a los teléfonos móviles enviar y recibir contenido multimedia como sonido, video, fotos, etc. Es una evolución de los SMS y es necesario contar con un celular que soporte multimedia para poder enviarlos y recibirlos. Además, los mensajes MMS pueden ser enviados no solo a teléfonos celulares, sino también a cuentas de correo electrónico.

MMS es una norma abierta, diseñada para trabajar con redes inalámbricas de conmutación de paquetes como GPRS, EDGE y 3G.

2.5.3 Acceso a Internet

Para que un celular pueda acceder a Internet, es necesario que pueda iniciar una transferencia de datos y que posea un navegador WAP. A través de los protocolos GPRS o EDGE, que realizan transferencia de datos basada en conmutación por paquetes, los celulares pueden realizar una conexión a Internet.

El acceso a Internet brinda a los dispositivos móviles el acceso a otros servicios como WAP, permitiendo entre otras cosas, acceso a sitios Web, descargas y acceso a servicios multimedia.

2.5.4 Navegación WAP

Es un estándar que especifica un conjunto de protocolos de comunicaciones y un conjunto de aplicaciones que nos permiten acceder a Internet desde dispositivos móviles. Es un protocolo de comunicación, y un entorno de aplicaciones para el desarrollo de recursos de información,

servicios de telefonía y acceso a Internet desde dispositivos inalámbricos. La reutilización de las tecnologías Web facilita el desarrollo de aplicaciones WAP y las asimila al desarrollo Web basado en HTML, ya que es un protocolo basado en navegación.

La tecnología WAP se describe en detalle en la sección 3.1.

2.5.5 Java



Cuando hablamos de un celular que posee Java, estamos refiriéndonos a que el dispositivo móvil cuenta con una plataforma para ejecución de programas desarrollados en tecnología Java, en particular, la plataforma Java 2 Micro Edition, o Java ME.

Java ME es una colección de APIs (conjunto de clases de objetos) escritas en Java orientadas a productos de consumo como PDAs, teléfonos móviles o electrodomésticos. La plataforma Java ME permite desarrollar aplicaciones para en teléfonos móviles, permitiendo emular en un PC el resultado esperado en los dispositivos, para finalmente subir el producto terminado al teléfono.

La tecnología Java ME describe en detalle en la sección 3.3.

2.5.6 Conexión inalámbrica con otros dispositivos

La conexión con otros dispositivos puede realizarse a través de un cable (generalmente por USB) o de forma inalámbrica. Los tipos de conexión inalámbrica más comunes son los infrarrojos (IrDA) y Bluetooth.

Cada tecnología tiene sus ventajas y desventajas, y ninguna puede completar todas las necesidades de los usuarios. De hecho Bluetooth e IrDA juntas, crean un avance importante en la historia de la conexión sin cable de bajo alcance.

2.5.6.1 Infrarrojo IrDA (Infrared Data Association)



Es un tipo de transmisión inalámbrica de datos que utiliza radiación de infrarrojos, un tipo de radiación electromagnética cuya longitud de onda está por debajo de la de la luz roja.

Es un tipo de conexión universal sin cables con comunicación punto a punto, en un ángulo estrecho (30°) diseñado para operar en una distancia de 0 a 1 metro y a una velocidad de 9600 bps a 16 Mbps que soporta un amplio conjunto de plataformas de hardware y software.

2.5.6.2 Bluetooth



Bluetooth es la norma que define un estándar global de comunicación inalámbrica, que posibilita la transmisión de voz y datos entre diferentes equipos mediante un enlace por radiofrecuencia. Los principales objetivos que se pretende conseguir con esta norma comprenden la simplificación de las comunicaciones entre equipos móviles y fijos, la eliminación de cables y conectores y la posibilidad de crear pequeñas redes inalámbricas facilitando la sincronización de datos entre equipos.

Bluetooth es una propuesta de especificación de radiofrecuencia por transmisión de corto alcance, punto a multipunto que puede transmitir a través de objetos sólidos no metálicos. Su alcance es de 10 cm a 10 m, pero se puede extender a 100 m mediante el incremento de transmisión de energía.

2.5.7 Otros servicios

La incorporación de servicios como reloj, alarma, calendario, agenda, calculadora y otros a desplazado en gran medida el uso de estos dispositivos adicionales, consolidando al teléfono celular como elemento concentrador de servicios.

El avance tecnológico de la fotografía y la reproducción de música desde los dispositivos no es tan sorprendente como su uso y difusión. Diferentes estudios realizados por los principales fabricantes de dispositivos, indican que casi la mitad de las personas han reemplazado su cámara digital por el celular y más de la mitad ha hecho lo mismo con sus reproductores de MP3.

Las cámaras empezaron a ser incorporadas en los teléfonos celulares en 2000, pero durante años la calidad de las fotos fue tan deficiente que los aparatos eran considerados poco más que un juguete. Eso está cambiando, a medida que los fabricantes de celulares agregan cámaras que toman fotos de la misma calidad que las de las cámaras digitales promedio. Algunos de los teléfonos en el mercado tienen cámaras con lentes de dos megapíxeles, similares a las cámaras digitales más baratas. En Asia y Europa, ya se encuentran aparatos con cámaras de cinco a ocho megapíxeles para fotos de mayor calidad.

Capítulo 3. Tecnologías inalámbricas

Existen diversas tecnologías inalámbricas que permiten el acceso a datos desde un teléfono móvil. Este intercambio de información puede darse entre celulares, como una llamada o un mensaje de texto, o descargando contenido desde un servidor en Internet accediendo desde un navegador u otra aplicación instalada en el dispositivo. En esta sección se describen diferentes tecnologías inalámbricas que permiten realizar este intercambio y la relación entre las mismas.

3.1 WAP

WAP (Wireless Application Protocol, protocolo de aplicaciones inalámbricas) es una especificación abierta que provee a las redes inalámbricas, varias de las características de transmisión de datos usadas por los protocolos Web y agrega otras específicas para satisfacer los requerimientos especiales de los dispositivos inalámbricos.

Es un estándar que especifica un conjunto de protocolos de comunicaciones y un conjunto de aplicaciones para el desarrollo de recursos de información, servicios de telefonía y acceso a Internet desde dispositivos inalámbricos. La reutilización de las tecnologías Web facilita el desarrollo de aplicaciones WAP y las asimila al desarrollo Web basado en HTML, ya que es un protocolo basado en navegación.

WAP fue desarrollado por Phone.com con colaboración de Ericsson, Motorola y Nokia. Estas empresas formaron el Foro WAP con el objetivo de generar un estándar de acceso a Internet desde teléfonos celulares. En junio del año 2002 el foro WAP se convirtió en Open Mobile Alliance Ltd (OMA).

OMA fue formado por casi 200 compañías que representan a los operadores de telefonía móvil más importantes del mundo, a los desarrolladores de dispositivos y redes inalámbricas, a las compañías de tecnología de información y contenidos.

3.1.1 Surgimiento de WAP

El enorme incremento de las telecomunicaciones inalámbricas, la tendencia hacia dispositivos más pequeños y más rápidos, junto con la necesidad de acceso a la información en cualquier momento, son los factores determinantes del surgimiento de WAP.

Además, WAP cumple con las expectativas de movilidad deseadas en cualquier ambiente de aplicación lo que permite acceder a la información y a los servicios de Internet en cualquier momento, en cualquier lugar y de cualquier manera, mediante dispositivos del tamaño de la palma de la mano.

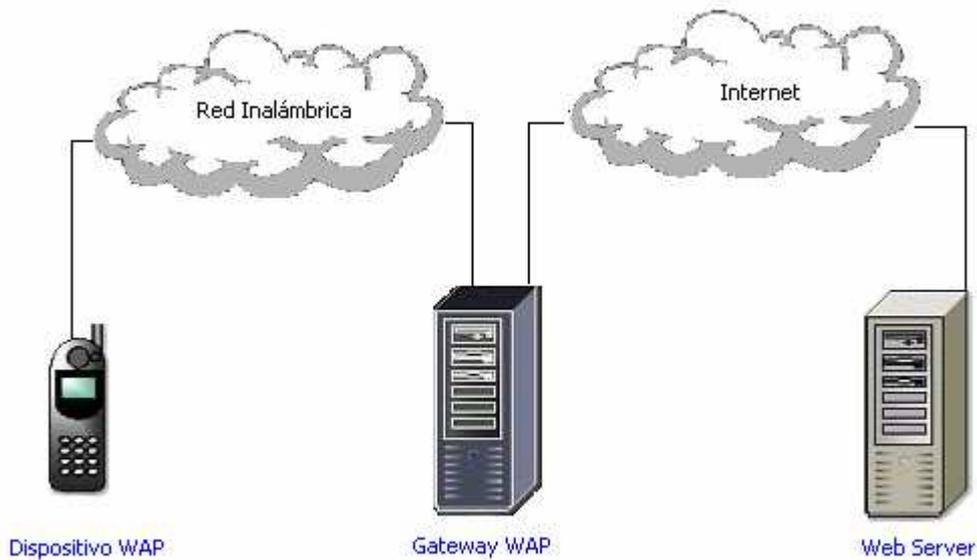
WAP es una solución unificada para brindar servicios de valor agregado existentes y futuros a la telefonía móvil. El protocolo incluye especificaciones para las capas de sesión y de transporte del modelo OSI, así como funcionalidades de seguridad y define un entorno de aplicaciones.

Es un protocolo escalable, que permite a las aplicaciones disponer de las capacidades de pantalla y recursos de red según su necesidad y en una gran variedad de tipos de terminales.

Los servicios pueden ser aplicables a pantallas de una sola línea o a terminales mucho más complejos.

3.1.2 Arquitectura

Desde sus inicios, el objetivo de WAP fue utilizar la estructura Web existente para producir la comunicación entre servidores de contenido y dispositivos móviles pero de una manera más eficiente y con menos consumo de recursos que los protocolos Web normales. La principal diferencia es la presencia del gateway o pasarela WAP para el intercambio de información entre HTTP y WAP. Este es el elemento que se coloca entre el dispositivo WAP y el servidor de origen. Actúa como un intérprete entre los dos, permitiendo la comunicación.



Arquitectura de WAP

La arquitectura WAP es muy similar a Internet: para acceder a una aplicación el cliente inicia una conexión con el gateway WAP y envía una petición de contenido; el gateway convierte la petición que recibe del cliente en formato HTTP y la pasa al servidor de origen. En el camino de retorno, el contenido se envía desde el servidor al gateway, la que se encarga de traducir el contenido a formato WAP y lo envía entonces al dispositivo móvil. El gateway entonces es el dispositivo que permite la comunicación de Internet y la red inalámbrica.

Los pasos para obtener contenido desde un teléfono celular son entonces:

1. El usuario solicita la página WAP que quiera ver escribiendo con el teclado su dirección en su móvil compatible con WAP.
2. El micronavegador del móvil envía la petición con la dirección (URL) de la página solicitada y la información sobre el abonado al gateway WAP.
3. El gateway examina la petición y la envía al servidor donde se encuentra la información solicitada.
4. El servidor añade la información HTTP o HTTPS pertinente y envía la información de vuelta al gateway.

5. En el gateway se examina la respuesta del servidor, se valida el código inalámbrico en busca de errores y se genera la respuesta que se envía al móvil.
6. El micronavegador examina la información recibida y si el código es correcto, lo muestra en pantalla.

3.1.3 Dispositivos

Es el dispositivo físico o equipo celular que se utiliza para acceder a contenido WAP. Existe una enorme y creciente variedad de dispositivos WAP con diferentes tipos de micronavegadores. Los primeros sólo permitían visualizar algunas líneas de texto, luego evolucionaron hasta poder mostrar imágenes sencillas en formato WBMP permitidas por WAP 1.x. En la actualidad con la especificación 2.0 los nuevos dispositivos pueden visualizar no solo imágenes en colores de alta definición, sino también sonidos y video.

Un navegador o micronavegador WAP es el software que se ejecuta en un dispositivo WAP que interpreta el contenido que llega desde Internet y decide cómo visualizarlo en la pantalla. También existen emuladores de navegadores que pueden utilizarse en una PC común y permiten ver contenido WAP.

3.1.4 El Cliente WAP

La especificación de WAP permite que los fabricantes tengan suficiente libertad para el desarrollo de dispositivos ya que no existe ninguna especificación WAP que indique el aspecto de un dispositivo WAP o cómo se debería presentar y visualizar el contenido que se recibe desde Internet.

El único requerimiento para que un dispositivo sea considerado WAP-enabled es que debe implementar un agente de usuario WAE (Wireless Application Environment), un agente usuario WTA (Wireless Telephony Application) y una Pila WAP. El agente usuario WTA forma parte de WAE el cual es uno de los componentes de la pila WAP.

El agente usuario WAE es el micronavegador que presenta el contenido para su visualización. Recibe archivos (WML, WMLScript, XHTMLMP, CSS, imágenes, etc.) del gateway WAP y los visualiza en la pantalla. Los detalles de implementación se dejan a criterio del fabricante y el navegador debe implementar toda la funcionalidad provista por los lenguajes WML, WMLScript o XHTMLMP. También debe gestionar la interacción con el usuario, por ejemplo la entrada de texto y los mensajes de error o de advertencia.

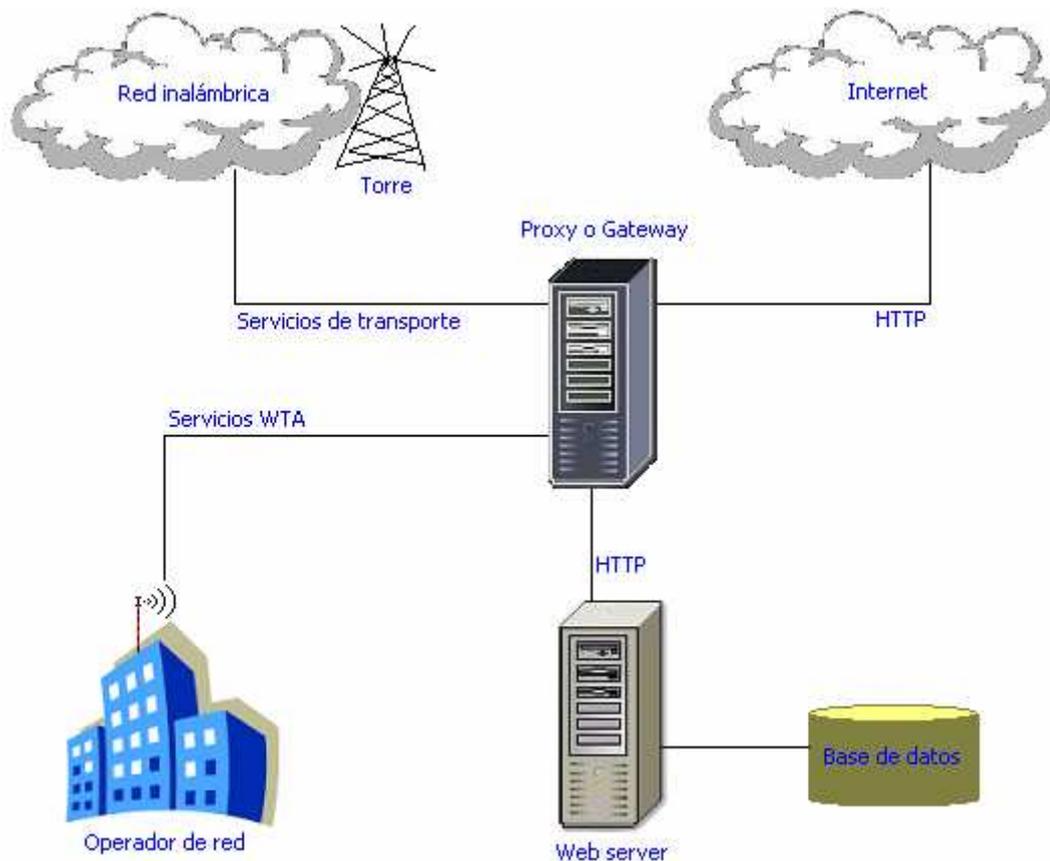
El agente usuario WTA es el encargado de ejecutar los archivos WTA recibidos del servidor. Estos archivos incluyen el acceso a la interfaz del teléfono y a las funcionalidades de red como el marcado telefónico, las respuestas de llamadas, la organización de la agenda, la gestión de mensajes y los servicios de ubicación entre otros. WAP 1.x solamente define la especificación del agente usuario WTA pero no lo implementa. En WAP 2.0 se provee soporte para WTA permitiendo por ejemplo el acceso a la funcionalidad telefónica del dispositivo.

La Pila WAP que permite al teléfono conectarse con el gateway WAP a través de los protocolos WAP.

3.1.5 Gateway o proxy WAP

Es un intermediario utilizado para conectar dos tipos de redes diferentes, inalámbrica e Internet. Un gateway WAP es un software que actúa como la unión entre dispositivos WAP y un servidor Web. Recibe peticiones de los clientes como si fuese realmente un servidor de origen con el cual se desea intercambiar información.

Interactúa con varios de los componentes de la telefonía celular e Internet para permitir el acceso a datos desde un dispositivo inalámbrico.



Interacciones del Gateway o Proxy WAP

El gateway WAP además contiene la funcionalidad de codificador y decodificador ya que convierte el contenido WML que se envía al cliente y que se recibe del cliente en un formato que está optimizado para las redes de bajo ancho de banda.

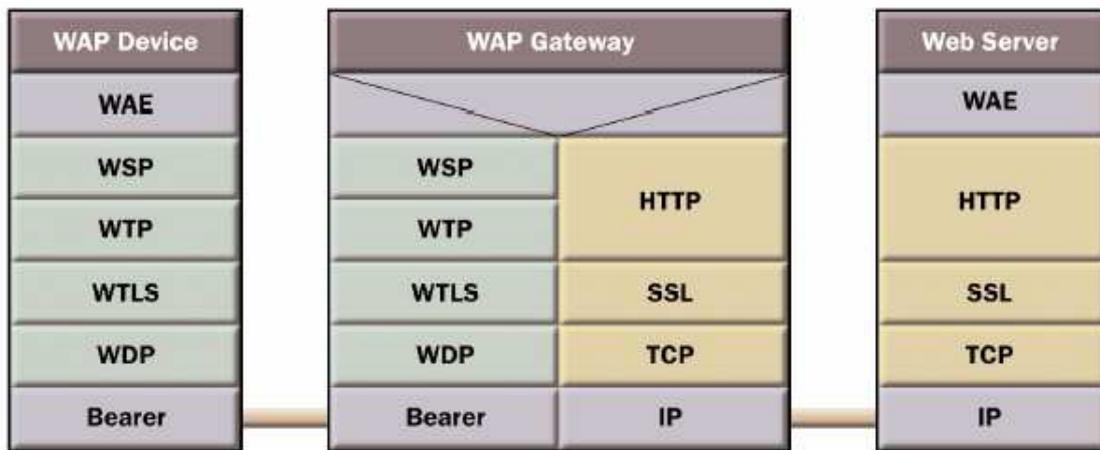
Convierte WML de un formato textual a uno legible por un dispositivo. Traduce los requerimientos del dispositivo WAP a requerimientos HTTP hacia el servidor WEB. Convierte la encriptación SSL usada en WEB hacia la encriptación WTLS usada en WAP. Convierte el protocolo TCP de WEB a WDP de WAP.

Con la introducción de los protocolos de Internet en el ambiente de WAP de la especificación 2.0 se proporciona soporte IP al dispositivo inalámbrico. WAP 2.0 define Wireless Profiled HTTP es un perfil de HTTP para ambientes inalámbricos compatible con HTTP/1.1. La interacción

entre el dispositivo de WAP y el Proxy WAP es ahora una transacción HTTP request/response. El nuevo TCP inalámbrico (Wireless Profiled TCP) es un TCP optimizado para ambientes inalámbricos y es completamente compatible con implementaciones estándares de TCP de Internet.

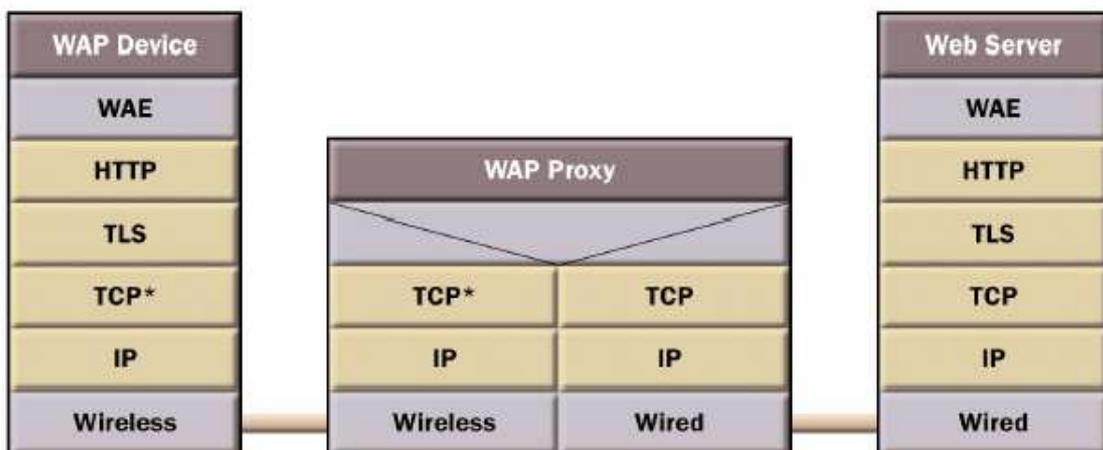
3.1.6 Funcionamiento

El gateway WAP 1.x se ocupa de la traducción de todas las peticiones que se envían y se reciben del cliente usando WSP al protocolo que utiliza el servidor de aplicaciones (por ejemplo HTTP). El servidor envía su información hacia el gateway utilizando HTTP, que envía el contenido recibido a los dispositivos WAP utilizando protocolos WAP.



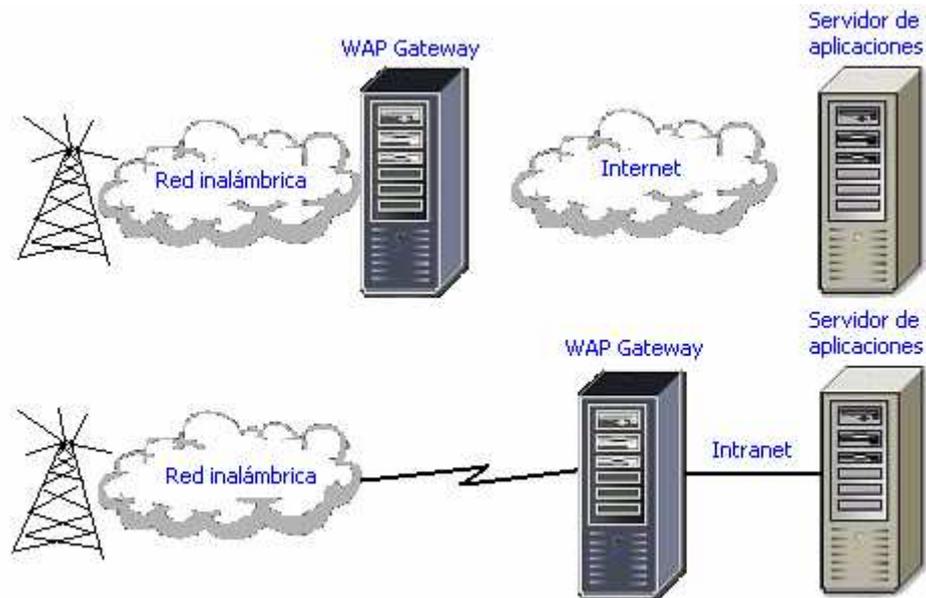
Gateway WAP 1.x

En WAP 2.0 simplemente crea un túnel entre el dispositivo WAP y el servidor WEB a través de un Proxy WAP permitiendo transacciones punto a punto HTTP.



Proxy WAP 2.x TCP* significa WP-TCP

3.1.7 Configuración



Diferentes configuraciones

Existen dos configuraciones básicas para proveer contenido WAP a un teléfono celular. La primera configuración permite acceder a Internet desde un dispositivo WAP utilizando el gateway público que proveen las empresas de telefonía celular. Con esta configuración no es necesario instalar un gateway y solo se necesita adaptar el servidor de aplicaciones para que provea contenido WAP. La segunda opción de configuración permite acceder a una Intranet desde un teléfono celular. Esta configuración es necesaria cuando se está ante una aplicación inalámbrica que requiere mayor grado de seguridad, cuyos datos sean privados y de alta confidencialidad. En este caso, es necesario instalar un gateway dentro de la Intranet para que traduzca el contenido WAP hacia los dispositivos ya que la transmisión de datos debe ser completamente segura y no se desea enviar los mismos hacia un operador de red.

3.1.8 Estructura interna de WAP (Pila WAP)

La pila WAP es un conjunto de protocolos en capas que forman la estructura interna de WAP. Es un derivado del modelo de referencia OSI ISO. Fue desarrollada para minimizar los requerimientos de ancho de banda y garantizar que las aplicaciones WAP puedan correr en diversas variedades de redes inalámbricas.

Una parte clave del lanzamiento de WAP 2.0 es la introducción de soporte para protocolos de Internet en los casos que el dispositivo móvil tenga conectividad por IP. Además se continúa con la pila WAP, que se utiliza sobre las redes no IP así como los portadores IP de bajo ancho de banda.



Capas de la pila WAP y WEB

Nótese que la pila WAP está diseñada para funcionar sobre una diversa variedad de redes inalámbricas con diferentes servicios de soporte como los vistos anteriormente.

La Pila WAP 1.x contiene cinco capas diferentes:

1. Capa de Aplicación: WAE (Wireless Application Environment) suministra un entorno de aplicación para el desarrollo y la ejecución de aplicaciones y servicios portátiles.
2. Capa de sesión: WSP (Wireless Session Protocol) suministra métodos para el intercambio organizado de contenidos entre un cliente y el servidor.
3. Capa de transacción: WTP (Wireless Transaction Protocol) suministra diferentes métodos para la ejecución de transacciones, para un grado variable de fiabilidad.
4. Capa de seguridad: WTLS (Wireless Transport Layer Security) es una capa de seguridad que suministra cuando está presente, conexiones para autenticación, privacidad y seguridad entre las aplicaciones.
5. Capa de transporte: WDP (Wireless Datagram Protocol) es la capa inferior de la pila WAP que provee a las capas superiores los servicios portadores ofrecidos por el operador de red.

3.1.8.1 Capa de aplicación (WAE)

La capa WAE brinda un entorno que incluye todos los elementos relacionados al desarrollo y ejecución de aplicaciones. WAE permite presentar contenido e interactuar con el teléfono celular. WAE está compuesto por dos elementos, el agente usuario WAE que incluye al micronavegador junto con el editor de texto simplificado y el agente usuario WTA que provee servicios de telefonía avanzada. Esta capa permite al desarrollador utilizar formatos y servicios específicos, creados y optimizados para presentar contenido e interactuar con dispositivos WAP.

Las especificaciones de la capa WAE no indican nada acerca de la implementación de la misma. Los navegadores, editores de mensajes y libretas telefónicas contenidas en el dispositivo WAP pueden ser muy diferentes entre ellos mientras sigan cumpliendo con las especificaciones. WAE solo especifica los formatos de imágenes y textos que los dispositivos deben compatibilizar.

Básicamente la especificación de WAE para WAP 1.x se compone de un lenguaje de marcas (WML, XHTMLMP), un lenguaje de script simplificado (WMLscript) y una interfaz para servicios locales y para servicios de telefonía avanzada WTA (Wireless Telephony Applications). Además WAE provee otros tipos de contenidos, tales como imágenes WBMP, vCard y vCalendar.

3.1.8.2 Capa de sesión inalámbrica (WSP)

La capa WSP (Wireless Session Protocol) hace posible el intercambio de datos entre aplicaciones de un modo organizado proveyendo servicios de sesión.

Un servicio de sesión es la funcionalidad que ayuda al establecimiento de la conexión entre un cliente y un servidor. El servicio se entrega por medio del uso de primitivas que lo suministran. Las primitivas son mensajes definidos que envía el cliente al servidor para pedir un servicio.

Para iniciar una nueva sesión, el cliente realiza una petición WSP que suministra algunos parámetros como la dirección del servidor, la dirección del cliente y los encabezados del cliente. Estos pueden estar vinculados a encabezados de cliente HTTP y pueden por ejemplo ser utilizados por el servidor para recuperar el tipo de agente usuario dentro del cliente WAP. Esto es útil cuando queremos formatear la salida de manera diferente dependiendo del tipo de dispositivo del cliente.

En WAP 2.0 WSP proporciona funcionalidad HTTP/1.1 e incorpora nuevas características, tales como sesiones de larga vida y la posibilidad de suspender y retomar una sesión.

3.1.8.3 Capa de transacción inalámbrica (WTP)

Esta capa suministra servicios para realizar transacciones fiables y no fiables entre el servidor y el cliente. WTP opera sobre la capa WDP y sobre la capa de seguridad opcional, WTLS. WTP está optimizada para adaptarse a un ancho de banda pequeño, tratando de reducir la cantidad de transacciones entre el cliente y el servidor.

En WAP 2.0 WTP se ha definido como protocolo orientado a transacciones ligeras que puede ser implementado en clientes livianos como estaciones móviles y funciona de manera eficientemente sobre redes inalámbricas. Las mejoras en WTP de WAP 2.0 incluyen:

1. Confiabilidad mejorada en servicios de datagrama. WTP alivia a la capa superior de las retransmisiones y de los reconocimientos necesarios cuando se utilizan servicios de datagrama.
2. Eficacia mejorada usada sobre servicios orientados a conexión. WTP no requiere el establecimiento de ninguna conexión explícita.
3. Las ventajas de usar un protocolo orientado a mensajes, diseñado para servicios orientados a transacciones, tales como la navegación.

3.1.8.4 Capa de seguridad inalámbrica (WTS o WTLS)

La capa WTLS está diseñada para proporcionar privacidad, integridad de datos y autenticación entre aplicaciones. Provee a la capa superior de WAP una interfaz de transporte segura que preserva el servicio de transporte provisto por las capas inferiores. Además, WTLS proporciona

un interfaz para administrar conexiones seguras.

WTLS es una capa opcional basada en SSL que opera sobre la capa de transporte. La traducción entre SSL y WTS se lleva a cabo en la memoria del gateway WAP sin dejar registro de almacenamiento para impedir que los datos sean vistos por personas no autorizadas.

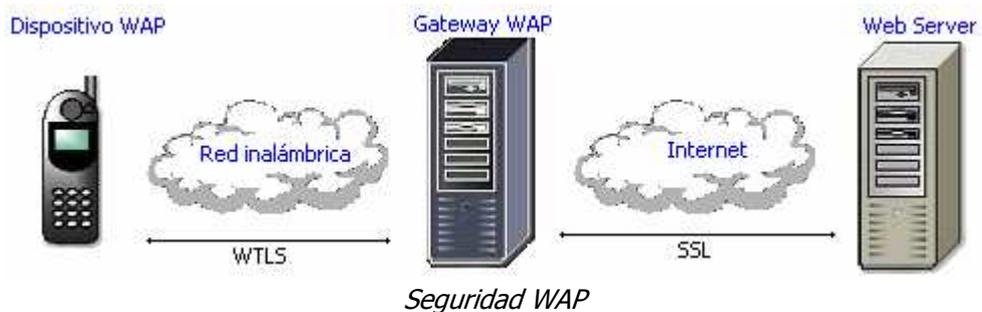
WTS brinda servicios que aseguran:

Privacidad: garantiza que los datos enviados entre el servidor y el cliente no son accedidos por ninguna otra persona. Nadie puede leer los mensajes cifrados aunque puedan ver físicamente el mensaje.

Autenticación del servidor: asegura que el servidor es realmente quien proclama ser y que no es un impostor.

Autenticación del cliente: suministra un modo para que el servidor origen limite el acceso al contenido que tiene a ciertos usuarios.

Integridad de datos: se ocupa de que nadie pueda alterar el contenido de un mensaje que se está transmitiendo entre el servidor y el cliente sin que ninguno de ellos lo detecte.



En WAP 2.0 WTLS proporciona una funcionalidad similar a TLS 1.0 e incorpora características adicionales como soporte de datagramas, handshake optimizado y refresco de claves dinámico.

El handshake es el intercambio de información entre dos módems que se realiza antes de una comunicación telefónica para acordar que protocolo se utilizará.

3.1.8.5 Capa de transporte (WDP)

WDP es la capa inferior de la pila WAP y es un protocolo sumamente portátil que opera en diferentes redes inalámbricas. WDP protege a las capas superiores de los servicios portadores suministrados por la red, por lo tanto permite que las aplicaciones funcionen de manera transparente sobre diferentes portadores.

WDP es un servicio de datagrama general, que ofrece un servicio constante a las capas superiores, comunicándose transparentemente sobre los servicios que provee el operador de red. WDP proporciona consistencia gracias a un conjunto de adaptaciones a las características específicas de los portadores. Esto proporciona así un interfaz común a las capas superiores que pueden funcionar independientemente de los servicios provistos por las diferentes redes

inalámbricas. Ejemplos de servicios brindados por el operador son SMS, CSD, y CDMA entre otros.

3.1.9 WAP 2.0

WAP 2.0 es un conjunto de especificaciones que adopta los protocolos y estándares de Internet más recientes. Además mantiene compatibilidades con el contenido y aplicaciones WAP existentes.

3.1.9.1 Objetivos

La especificación 2.0 de WAP salió a la luz en Marzo de 2002 y cumple varios objetivos:

Agrega soporte para los protocolos estándares de comunicación de Internet. WAP 2.0 soporta IP, TCP y HTTP. Con el agregado de estos protocolos junto con la optimización de los ambientes de telecomunicación inalámbrica, esta nueva especificación de WAP provee un ambiente que permite a los dispositivos inalámbricos utilizar las tecnologías de Internet ya existentes.

Continuando el trabajo de las especificaciones anteriores, WAP 2.0 permite que las aplicaciones y servicios operen sobre todas las interfaces aéreas y sus proveedores. Esto incluye las nuevas tecnologías conocidas como General Packet Radio Service (GPRS) y 3ra Generación (3G) de celulares que proveen mayor velocidad.

Provee un ambiente de aplicación que permite la entrega de información y servicios interactivos a diversos dispositivos inalámbricos como teléfonos celulares digitales, pagers, personal digital assistants (PDAs) u otros.

Agrega nuevas características para mejorar la experiencia del usuario con respecto al uso de los dispositivos celulares. Los dispositivos inalámbricos poseen factores que afectan el diseño tradicional de aplicaciones de Internet como limitaciones de hardware (pantallas pequeñas, ancho de banda limitado y RAM y ROM limitado) y consideraciones de interfaz de usuario como navegación one-finger. Las nuevas características tienden a eliminar estos problemas con los que se encuentra el usuario y lograr una mejor utilización de los dispositivos.

Minimiza el uso de procesamiento en el dispositivo y optimiza los recursos de red para minimizar costos y maximizar performance. WAP 2.0 también optimiza el uso del ancho de banda y las conexiones basadas en paquetes de las redes inalámbricas.

3.1.9.2 Componentes

Los principales componentes de la arquitectura de WAP 2.0 son:

Soporte de protocolos de pilas: Además de la Pila WAP 1.x, WAP 2.0 agrega soporte y servicio sobre protocolos basados en la pila de Internet incluyendo soporte para TCP, TLS y HTTP. De esta manera WAP 2.0 provee un modelo de conectividad sobre un amplio rango de redes y proveedores inalámbricos.

Ambiente de aplicación WAP: Llamado también 'WAP Browser', el ambiente de aplicación WAP ha evolucionado para adoptar los lenguajes estándares para navegadores de Internet. Esto ha llevado a la definición de XHTML Mobile Profile (XHTMLMP). XHTMLMP está basado en XHTML (eXtensible HyperText Markup Language). XHTML es un lenguaje modular desarrollado por W3C para reemplazar a HTML.

Servicios y capacidades adicionales: WAP 2.0 provee incrementos considerables en el número de características disponibles para los desarrolladores, operadores y usuarios. La motivación para desarrollar WAP siempre fue ampliar las tecnologías existentes en Internet a las redes, portadores y dispositivos inalámbricos. Versiones anteriores de WAP anticiparon progresos con respecto a imágenes, seguridad PKI end-to-end, mensajería, tecnología push, e implementación de estándares de W3C y de IETF. WAP 2.0 capitaliza una amplia gama de estas nuevas tecnologías y capacidades avanzadas, entre las que se destacan:

Redes y portadores: Los portadores por todo el mundo están aumentando sus redes existentes con protocolos de alta velocidad tales como GPRS, HSCSD y están ampliando el ancho de banda e incrementando la velocidad de las redes inalámbricas 3G tales como W-CDMA y CDMA2000. Estos portadores de mayor capacidad permiten interactuar con nuevos tipos de contenido y proporcionan mayor disponibilidad.

TCP/IP como protocolo de transporte: La mayoría de las nuevas tecnologías de redes inalámbricas proporcionan soporte de paquetes IP como protocolo de transporte de datos. WAP 2.0 desarrolló un perfil móvil de TCP (TCP Mobile Profile) para redes inalámbricas. Este perfil es compatible con el TCP de Internet actual.

Procesadores: Los fabricantes continúan introduciendo dispositivos más pequeños con procesadores más rápidos y con mayor definición de color. Además, han desarrollado tecnologías de empaquetado más eficiente que permite circuitos integrados más pequeños y con mayor sofisticación en el dispositivo. El efecto neto es una capacidad superior en los nuevos dispositivos inalámbricos orientada a realzar los servicios que WAP le brinda al usuario móvil.

Tecnologías amigables: Con el crecimiento del uso de dispositivos móviles, hay un conocimiento creciente de las necesidades específicas del usuario móvil. El foro WAP ha trabajado con W3C y IETF para ayudar a caracterizar las cuestiones claves que afectan el uso inalámbrico de la web. Con esa implicación, W3C ha presidido últimamente avances en tecnologías móviles "amistosas", incluyendo:

- El lanzamiento a fines de 2000 de la recomendación de un perfil móvil de XHTML conocido como XHTMLMP. Este perfil incorpora los elementos base de XHTML, y proporciona un marco para la expansibilidad y extensibilidad.
- Modificaciones recientes a Composite Capabilities/Preference Profile (CC/PP) para describir preferencias del usuario y capacidades del dispositivo. CC/PP proporciona la base técnica para describir el perfil del dispositivo a través de User Agent Profile (UAPROF). UAPROF es un protocolo para transmitir las capacidades disponibles en un dispositivo incluyendo características de red (ancho de banda, portador, etc.), capacidades de WAP (lenguaje, colores, WTA, etc.) y capacidades push.
- El lanzamiento de hojas de estilo. CSS Mobile Profile proporciona un subconjunto de la versión 2 de CSS dirigida a dispositivos tales como teléfonos celulares, personal digital assistants (PDAs), etc.

3.1.9.3 Características y funcionalidades mejoradas

Modelo de programación de WAP: El modelo de programación de WAP, es similar al modelo de programación web, ya que ambos utilizan el modelo Pull, (donde el cliente solicita el contenido al servidor). Sin embargo, WAP 2.0 amplía la arquitectura web agregando soporte para servicios de telefonía con WTA y permitiendo el modelo Push, donde un servidor puede enviar contenido al cliente sin que este lo solicite. Versiones anteriores de WAP requerían un gateway o proxy WAP para manejar el protocolo que operaba entre el cliente y el servidor. El gateway WAP se comunicaba con el dispositivo inalámbrico utilizando protocolos propios de WAP basados mayormente en los protocolos de comunicación de Internet, y se comunicaba con el servidor origen usando protocolos estándares de Internet. WAP 2.0 no requiere un gateway WAP, puesto que la comunicación entre el cliente y el servidor del origen se puede realizarse usando HTTP/1.1. Sin embargo, el uso de un proxy WAP puede optimizar el proceso de comunicación y ofrecer mejoras en el servicio, tales como localización, privacidad, y demás servicios. Además, el gateway WAP es necesario ofrecer funcionalidad Push.

Soporte IP: Una característica dominante de WAP 2.0 es la introducción de los protocolos de Internet en el ambiente WAP. Este soporte ha sido motivado por la aparición de las redes inalámbricas de la alta velocidad (CDMA y 3G) que proporcionan soporte IP al dispositivo inalámbrico.

HTTP inalámbrico (WP-HTTP): La especificación WP-HTTP es un perfil de HTTP para ambientes inalámbricos completamente compatible con HTTP/1.1. El modelo básico de la interacción entre un dispositivo WAP y un servidor WAP es una transacción HTTP request/response. WP-HTTP soporta compresión de mensajes y establecimiento de sockets seguros.

Seguridad en la Capa de Transporte (TLS): Un perfil inalámbrico del protocolo TLS permite transacciones seguras. Este perfil incluye cifrado, certificados, algoritmos de firma, etc. Además define métodos para que TLS establezca túneles a fin de soportar seguridad punto a punto a nivel de transporte.

TCP Inalámbrico (WP-TCP): WP-TCP proporciona servicios orientado a conexión. Está optimizado para ambientes inalámbricos y es completamente compatible con implementaciones estándares de TCP de Internet.

3.1.9.4 Mejoras en la experiencia del usuario

Además del ambiente de aplicación y de los aumentos en la capacidad del micro navegador, WAP 2.0 también soporta otras características para mejorar la experiencia del usuario. Estas características están orientadas a ampliar las capacidades de los dispositivos inalámbricos y mejoran la capacidad de entregar aplicaciones y servicios útiles. Algunas de estas características adicionales de WAP 2.0 son:

WAP Push: Este servicio permite que el contenido sea enviado o "empujado" desde las aplicaciones hacia los dispositivos inalámbricos a través de un proxy Push. Esta funcionalidad se ha enriquecido a partir del lanzamiento de WAP 2.0. Perfil de Agente Usuario (UAPProf): Este servicio que proporciona al servidor de aplicación, un mecanismo para describir las capacidades de los dispositivos clientes y de las preferencias de los usuarios. UAPProf soporta el modelo de

transacción cliente/servidor enviando información del dispositivo y del usuario al servidor junto con la petición. Esta información permite que los servidores adapten su contenido para variar la respuesta de acuerdo a la información recibida. Este modelo también permite que los proxys que intervienen proporcionen servicios de valor agregado.

Aplicaciones de telefonía inalámbrica (WTA): Este servicio proporciona herramientas para utilizar una amplia gama de aplicaciones de telefonía dentro del ambiente de una aplicación WAP. Estas aplicaciones incluyen realizar llamadas, contestarlas, derivarlas, etc., y pueden ser combinadas con los servicios de datos. Esto permite que los teléfonos WAP sean vistos como una plataforma de servicios de Internet completamente integrada.

Interfaz externa de funcionalidad (EFI): Este servicio especifica una interfaz entre WAE y aplicaciones que ejecutan fuera de las capacidades definidas en WAE. Esto es análogo a proporcionar un plugin, que amplía o realiza las capacidades de los navegadores web o de otras aplicaciones. EFI prevé el crecimiento y la extensibilidad futura de los dispositivos soportados por WAP. Este marco se puede utilizar para definir interfaces específicas necesarias para tener acceso a dispositivos externos (e.g. tarjetas, dispositivos de GPS, cámaras fotográficas digitales, etc.)

Interfaz persistente de almacenamiento: Esta capacidad especifica un sistema estándar de servicios del almacenamiento definido para organizar, tener acceso, almacenar y recuperar a datos en el dispositivo inalámbrico u otro dispositivo de memoria.

Sincronización de datos: Como solución para la sincronización de datos WAP 2.0 adopta el lenguaje SyncML.

Servicio de Mensajería Multimedia (MMS): MMS proporciona servicio de mensajería con características y funcionalidad que permite la entrega de tipos variados de contenido. Dependiendo del tipo de servicio provisto por el operador, MMS permite un sistema de entrega rápido (como SMS) o store-and-forward (como el correo electrónico) o ambos. A través de otros servicios de WAP como Push y UAProf, MMS proporciona una solución eficiente de mensajería capaz de proporcionar notificaciones a los dispositivos de forma eficiente.

Aprovisionamiento: Este servicio proporciona a los clientes WAP la información necesaria para funcionar en una red inalámbrica.

Pictograma: Este servicio permite el uso de imágenes minúsculas, tales como el sol sonriente, para transportar rápidamente conceptos que permiten comunicación eficiente en una cantidad de espacio pequeña.

3.2 Tecnología Push

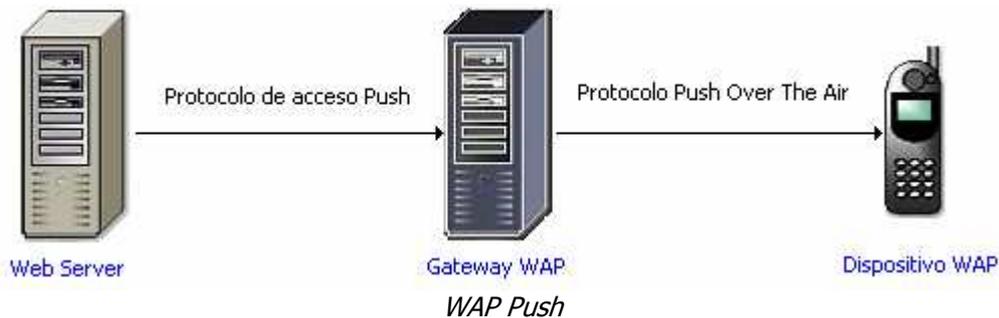
Existe mucha información útil que el usuario no puede visualizar porque no sabe cuando está disponible y sería conveniente que esta información sea enviada al usuario sin que este la solicite.

La tecnología Push permite justamente esto, enviar contenido a un dispositivo inalámbrico sin que este la solicite. La funcionalidad Push es especialmente relevante para las aplicaciones de

tiempo real que envían notificaciones a sus usuarios, tales como mensajería, precios, stock, alarmas de tráfico entre otros. Sin funcionalidad Push, estos tipos de aplicaciones requerirían que los dispositivos consulten a los servidores por nueva información o estado. En ambientes inalámbricos este tipo de actividad de interrogación lleva al uso ineficaz y derrochador de recursos de red. La funcionalidad Push de WAP controla el ciclo de vida de los mensajes enviados, el almacenamiento y al portador elegido para la entrega.

La tecnología Push define una estructura completa para la distribución de contenido de forma asincrónica sin necesidad de que el usuario haya requerido dicha información.

3.2.1 Estructura



El servidor de origen conocido como Push Initiator (PI) inicia el envío de contenido, el Push Proxy Gateway (PPG) es un intermediario entre el PI y el dispositivo WAP y funciona como punto de entrada al mundo inalámbrico para el PI. El Protocolo de Acceso Push (PAP) está diseñado para funcionar sobre diversos protocolos de aplicación de Internet como pueden ser HTTP o SMTP, generando un túnel a través de TCP (por ejemplo un POST de HTTP).

Por otro lado el protocolo Over The Air (OTA) trabaja sobre la capa de sesión (WSP) de la pila WAP y es el encargado de enviar la información al dispositivo.

PPG tiene que implementar toda la pila de protocolo WAP además de PAP y OTA. Por esto la funcionalidad Push forma parte del software dentro del Gateway.

La funcionalidad de PPG comprende la identificación y autenticación del PI, el análisis y detección de errores de la información enviada, la resolución y transformación de direcciones de dispositivos WAP, la codificación y compilación de código, la implementación del protocolo OTA y la selección del portador para la entrega del mensaje.

PAP se utiliza para distribuir contenido entre el PI y el PPG. PAP es una aplicación de XML por lo que su sintaxis y semántica es fácil de entender e implementar.

El protocolo OTA se encarga de la entrega del contenido desde el PPG hacia el dispositivo, y utiliza las primitivas de la capa WSP de la pila WAP.

3.2.2 Mensajes

El Push Initiator (PI) necesita saber donde se encuentra el dispositivo para saber a que gateway (PPG) encaminar la información. PI identifica a los clientes usando direcciones textuales que el PPG transforma a una dirección válida en la red inalámbrica que utiliza para distribuir el contenido Push.

Hay dos formas de enviar un mensaje hacia un cliente, una de ellas a partir del direccionamiento definido por el usuario donde la dirección del dispositivo no tiene significado dentro de la red inalámbrica, sino que PPG tiene una tabla de asignación de correspondencia de identificadores con las direcciones de dispositivos. La otra forma es utilizar las direcciones de los dispositivos, que son valores que tienen sentido en una red inalámbrica (la red las utiliza para encaminar información) y se pueden utilizar para identificar unívocamente cada dispositivo dentro de la red inalámbrica. Se utiliza por ejemplo ipv4, ipv6 o MSISDN (basado en el número de teléfono).

El mensaje Push se envía como mensaje MIME de multipartes relacionadas que siempre contienen una entidad de control en formato XML y una entidad de contenido en cualquier otro formato.

Luego del mensaje Push, se utiliza la respuesta Push para devolver al PI el resultado de aceptación inicial o la negación del mensaje Push y alguna información adicional relacionada con el procesamiento inicial por parte del gateway PPG.

3.2.3 Push y WAP

El modelo Push se encuentra definido en las especificaciones de WAP cosa que no sucede en las especificaciones de Internet, por ende la tecnología Push forma parte de WAP. Además el contenido Push está orientado hacia dispositivos que el usuario lleva normalmente consigo, por lo tanto tiene mucho sentido que exista un modelo Push estandarizado en WAP. Otra ventaja importante es que las redes inalámbricas poseen la capacidad de producir contenido Push basado en la ubicación. Como existe la posibilidad de saber la ubicación del usuario, podemos proveer contenido e información localizada, por ejemplo clima, información turística, rutas, etc.

Los servicios Push ya existen en las redes de telefonía móvil actuales con MMS y actualmente muchas personas lo utilizan para recibir noticias importantes, resultados deportivos, o notificaciones.

3.3 Java ME

Java ME (Java™ 2 Platform, Micro Edition) es la plataforma Java para consumidores como teléfonos celulares, PDA's y una amplia rama de dispositivos embebidos. Java ME incluye una máquina virtual Java y un conjunto de interfaces de programación de aplicaciones (API), que permite el desarrollo, descarga y posterior ejecución de programas Java en los dispositivos.

La plataforma Java ME brinda el poder y los beneficios de la tecnología Java incluyendo una interfaz de usuario flexible, un modelo de seguridad robusto, un amplio rango de protocolos de red y soporte para aplicaciones inalámbricas.

3.3.1 Aplicaciones

La plataforma Java ME permite crear aplicaciones inalámbricas 100% basadas en tecnología Java. Dentro del estándar J2EE las páginas JSP y los servlets Java se utilizan comúnmente entregar contenido dinámico a clientes navegadores web. En este modelo, un navegador obtiene HTML dinámico de un servidor de aplicaciones y lo interpreta para mostrar el contenido. Este mismo modelo puede utilizarse también para clientes inalámbricos, creando páginas JSP y servlets que generen WML o XHTMLMP para los micronavegadores de estos dispositivos.

Java ME posee un conjunto de interfaces de programación de aplicaciones (APIs) que proveen un ambiente de ejecución Java, llamado Mobile Information Device Profile (MIDP). MIDP mejora muchas de las limitaciones de las aplicaciones inalámbricas tales como interfaz de usuario, red, ancho de banda, almacenamiento persistente y modelado de aplicaciones.

MIDP proporciona las siguientes ventajas:

Disminuye el uso de la red y la carga del servidor: En una aplicación inalámbrica, el servidor es responsable de generar el contenido a mostrar. Esto requiere un ida y vuelta cada vez que la interfaz cambia. En cambio, con MIDP la interfaz del cliente está contenida dentro del dispositivo, así que este puede funcionar incluso estando desconectado. En las ocasiones que el dispositivo debe interactuar con un servidor, requiere menos tráfico de la red, porque descarga solamente datos de la aplicación, y no datos e interfaz como en el caso de WML o XHTMLMP.

Mejora la experiencia del usuario: La interacción que los lenguajes de marcas como WML y XHTMLMP pueden brindar es limitada. Las APIs de GUI de MIDP permiten interfaces con componentes visuales y eventos que posibilitan una mejor experiencia del usuario.

Java ME brinda acceso al mundo J2EE permitiendo que las aplicaciones empresariales puedan ser accedidas desde dispositivos móviles.

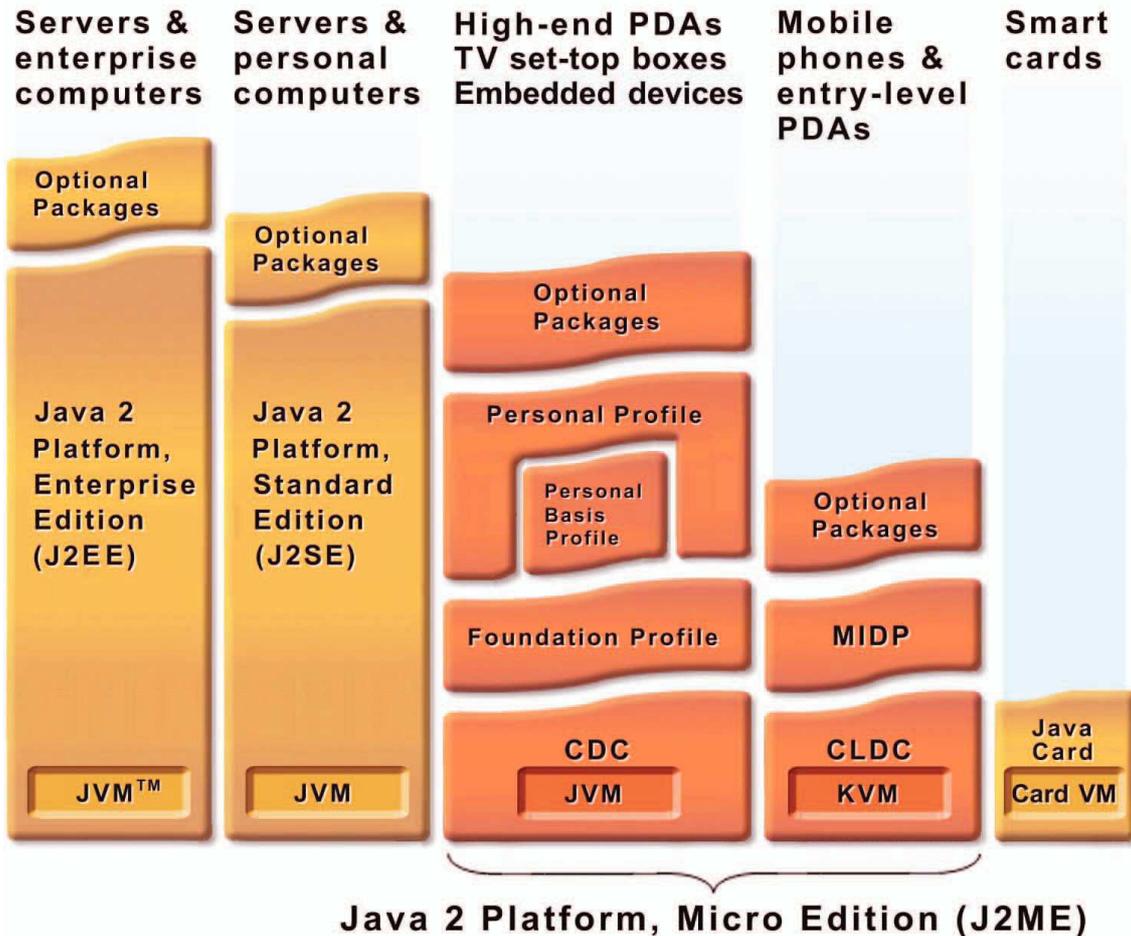
J2EE permite desarrollar aplicaciones del lado del servidor que pueden ejecutarse en cualquier servidor compatible con J2EE. Java ME a través de MIDP permite desarrollar aplicaciones cliente de servicios J2EE y ponerlas a disposición de cualquier dispositivo compatible con MIDP como celulares, o PDA's.

La tecnología Java ME parte del hecho de reconocer que los dispositivos no son todos iguales. Mientras muchas clases de dispositivos pueden utilizar la misma interfaz de usuario, otros tienen requerimientos propios que no coinciden con una solución general.

3.3.2 Arquitectura

La arquitectura Java ME comprende una variedad de clases, configuraciones, perfiles y paquetes adicionales que los desarrolladores pueden elegir y combinar para construir un ambiente de ejecución Java, que encuadre los requerimientos de un conjunto particular de

dispositivos y un mercado destino. En cada combinación puede optarse por optimizar el uso de memoria, el poder de procesamiento y capacidades de entrada/salida de una categoría particular de dispositivos, a fin de obtener las mayores ventajas de cada categoría.



Arquitectura Java ME (Sun Microsystems)

3.3.2.1 Configuraciones

Las configuraciones están compuestas de una máquina virtual (JVM) y un conjunto mínimo de clases Java. Estas proveen la funcionalidad básica para un amplio rango dispositivos que comparten características similares, tales como conectividad de redes y utilización de memoria. Concurrentemente, hay dos configuraciones Java ME: la configuración de dispositivo de conexión limitada (CLDC), y la configuración de dispositivo conectado (CDC). CLDC es más pequeña y diseñada para dispositivos con conexiones de red intermitentes, procesadores más pequeños y memoria limitada, como los teléfonos celulares, que pueden soportar una JVM limitada. CDC en cambio está diseñada para dispositivos con procesadores más importantes, más memoria y mayor ancho de banda, que permite ejecutar una JVM mas completa.

3.3.2.2 Perfiles

Para generar un entorno completo de ejecución de programas orientado a diferentes tipos de dispositivos inalámbricos, las configuraciones pueden ser combinadas con un conjunto de API's

de alto nivel denominadas perfiles que definen el ciclo de vida de la aplicación, la interfaz de usuario y acceso a propiedades específicas del dispositivo.

Un ejemplo de perfil consiste en combinar Mobile Information Device Profile (MIDP) con CLDC para generar de un perfil específico para teléfonos celulares u otros dispositivos de características comunes.

3.3.2.3 Paquetes opcionales

Java ME puede extenderse utilizando varios paquetes opcionales a una determinada configuración y a un perfil asociado. Los paquetes opcionales están diseñados para ofrecer tecnologías existentes como conectividad a bases de datos, Bluetooth, Web Services y mensajería. Como los paquetes adicionales son modulares, los desarrolladores pueden evitar la sobrecarga de funcionalidad innecesaria, incluyendo solo aquellos que la aplicación necesita.

3.3.3 Mobile Information Profile (MIDP)

MIDP es un elemento fundamental de Java ME. Al combinarse con CLDC, provee un entorno estándar de ejecución Java para los dispositivos móviles mas populares de la actualidad, como teléfonos celulares y PDA's.

CLDC y MIDP brindan una completa funcionalidad de aplicación requerida por las aplicaciones móviles, en la forma de un entorno estándar de ejecución de aplicaciones y un conjunto de API's Java. Los desarrolladores que utilizan MIDP pueden escribir las aplicaciones una vez, probarlas en diferentes emuladores y finalmente descargarlas en una amplia variedad de dispositivos móviles. MIDP ha sido adoptado como una plataforma estándar para el desarrollo de aplicaciones celulares, soportada por una enorme cantidad de dispositivos e incluso integrada dentro de los entornos de desarrollo (IDE) Java.

3.3.3.1 Beneficios

Interfaz de usuario enriquecida: Las aplicaciones MIDP proveen una base para desarrollar aplicaciones gráficas e intuitivas. La interfaz gráfica está optimizada para el tamaño pequeño de las pantallas, diferentes métodos de entrada de datos y otras características propias de los dispositivos móviles actuales. MIDP provee navegación intuitiva y entrada de datos aprovechando todas las ventajas del teclado telefónico, botones adicionales como las teclas de flecha, touch screen, y pequeños teclados QWERTY. Las aplicaciones MIDP son instaladas y ejecutadas localmente, pueden operar tanto en red como desconectadas, y pueden almacenar y administrar información local al dispositivo.

Conectividad extendida: MIDP permite a los desarrolladores explotar la funcionalidad propia de la red y las capacidades de mensajería de los dispositivos móviles. Soporta estándares como HTTP, HTTPS, sockets y puertos series, además de SMS a través de paquetes opcionales.

Multimedia y juegos: MIDP permite el desarrollo de juegos y aplicaciones multimedia portátiles. Una API de bajo nivel complementa a la API de alto nivel de la interfaz de usuario, permitiendo tener control de gráficos y entradas. Existe una API específica para juegos que agrega

funcionalidad específica, provee audio nativo que soporta tonos e incluso archivos WAV. Además la Mobile Media API (MMAPI), agrega la posibilidad de utilizar video u otro contenido multimedia a las aplicaciones MIDP.

Actualizaciones Over-the-Air: Una gran ventaja de MIDP es la posibilidad de publicar o actualizar aplicaciones de forma dinámica, de forma remota, a través del protocolo Over-the-Air.

Seguridad: MIDP incluye un modelo de seguridad estándar que permite proteger la red, las aplicaciones y los dispositivos móviles. El soporte HTTPS habilita a las aplicaciones el envío y recepción de datos a través de estándares existentes como SSL y WTLS.

3.3.4 Java ME y WAP

WAP y Java ME no son tecnologías competentes, sino que son tecnologías complementarias. El modelo de programación de ambas tiene similitud con modelos existentes como el modelo Web y Java. Programar WAP es similar a programar Web donde J2ME enriquece a WAP proveyendo funcionalidad extensible similar a la que un plugin provee a Web.

3.5 BREW

BREW (Binary Runtime Environment for Wireless) es una iniciativa de Qualcomm similar en varios aspectos a Java ME, que permite el desarrollo de aplicaciones inalámbricas, que pueden configurarse, distribuirse y con la característica que permite facturar por los contenidos que se descargan.

Las aplicaciones Brew están orientadas a mejorar la experiencia del usuario, como escuchar música ver videos, enviar y recibir mails y mensajes de texto, juegos, etc.

BREW incluye una plataforma de ejecución de aplicaciones (BREW VM), un conjunto de herramientas de desarrollo (BREW SDK) dispositivo, la BREW SDK™ (kit del desarrollo del software) y un sistema de distribución (BREW DS) que es controlado y manejado por los operadores de telefonía celular permitiéndoles administrar los procesos de facturación y pago.



Arquitectura BREW

3.5.1 Aplicaciones

BREW está pensado para que los desarrolladores puedan desarrollar en cualquier lenguaje. Las aplicaciones BREW naturales se escriben en c/C++, pero también permiten desarrollo de aplicaciones en otros lenguajes como Java y XML.

El BREW SDK incluye un emulador BREW que permite probar las aplicaciones en etapa de desarrollo como en Java ME, sin embargo para la publicación de aplicaciones, existen algunas diferencias. Las aplicaciones BREW deben estar firmadas digitalmente y como BREW permite un control absoluto sobre el dispositivo, solamente proveedores de contenido o desarrolladores autenticados, tienen las herramientas necesarias para firmar aplicaciones. Además una vez desarrollada y probada la aplicación, debe ser enviada a Qualcomm para un testeo oficial. Luego del testeo oficial, la aplicación puede ser ofrecida a los proveedores de contenido u operadores de red, para poder ser descargada e instalada. Por último, la aplicación es firmada digitalmente por el proveedor de contenido para permitir su ejecución en los dispositivos móviles.

3.5.2 Inconvenientes

Aunque Qualcomm ha demostrado que las aplicaciones BREW pueden ejecutar en teléfonos GSM, estas solo están disponibles en dispositivos CDMA debido a las limitaciones técnicas.

Si bien las aplicaciones pueden ser subidas a un dispositivo con soporte BREW para su testeo, si estas no están firmadas digitalmente, son eliminadas al reiniciar el dispositivo. Esto limita en gran medida el desarrollo de aplicaciones ad-hoc, sumando el hecho que no hay garantía que los proveedores de contenido dispongan de las aplicaciones desarrolladas a los usuarios.

Otro inconveniente es el costo de desarrollo de aplicaciones BREW. Además del costo de registro que impone Qualcomm para adquirir el compilador BREW, debe adicionarse el costo impuesto para el testeo oficial del código fuente y el costo de la firma digital de la aplicación.

3.5.2 BREW y Java ME

Actualmente Java ME es utilizado ampliamente en todo el mundo, en cambio BREW se utiliza masivamente solo en Estados Unidos y Japón. El hecho del costo de desarrollo, testeo y descarga que tienen las aplicaciones BREW, implica que los desarrolladores se inclinen hacia Java ME.

Verizon es la principal empresa que ofrece descarga de aplicaciones BREW pero con un costo. En cambio las aplicaciones Java ME pueden ser descargadas de diferentes portales de contenido, incluso de forma gratuita.

Capítulo 4. Aplicaciones para celulares

Las aplicaciones que pueden ejecutarse desde un dispositivo celular se dividen en tres grandes géneros, las que acceden a un sitio de Internet a través de un navegador WAP disponible en el dispositivo, las que ejecutan en el celular, como juegos y aplicaciones en general y las que se acceden a través del envío de mensajes de texto. A continuación se describen las aplicaciones WAP, Java ME y SMS como principales referentes de estos grandes grupos.

4.1 Aplicaciones WAP

Una aplicación WAP se compone de un sitio WAP en Internet y de un dispositivo inalámbrico que soporte WAP. Las páginas del sitio WAP están escritas en uno o varios lenguajes especiales, y publicadas en un servidor de aplicaciones WAP. Para que el dispositivo pueda interpretar el contenido del sitio WAP, debe contener un navegador WAP que interprete los lenguajes WAP.

4.1.1 Tipo de aplicaciones WAP

Una aplicación WAP puede brindar casi todos los servicios que se proveen hoy en día en Internet como servicios de comunicación (e-mail, mensajes cortos, postales electrónicas, mensajes multimedia, videotelefonía, pizarra electrónica), acceso a Internet e Intranets, acceso a bases de datos corporativas, páginas amarillas, información del tráfico, información turística, horarios de trenes, aviones, mapas, servicios de localización gestión y consultas bancarias, compras on-line, acceso a juegos y apuestas, carga de tarjetas de crédito, entre otros.

Debido a las limitaciones de red, teclado y pantalla de los teléfonos celulares, una aplicación Wap es aplicable a cualquier aplicación Web donde el intercambio de información entre el usuario y la aplicación sea mínimo y donde el diseño y los aspectos visuales no sean esenciales.

4.1.2 Servidor de aplicaciones WAP

Es el servidor donde residen la información y las aplicaciones.

Un servidor WAP es simplemente un servidor Web normal junto con un gateway WAP. Generalmente se piensa que debe necesariamente instalarse un servidor WAP para proveer Internet a los teléfonos celulares, pero en realidad cualquier servidor Web puede enviar contenido WAP y utilizar el gateway provisto por el operador de red.

El servidor de contenido o de aplicación WAP tiene exactamente la misma función que un servidor Web y ofrece las mismas funcionalidades a los clientes. La distinción entre ellos es de simplemente semántica, ya que ambos pueden coexistir en un mismo dispositivo físico y pueden suministrar ambas funciones usando el mismo fragmento de código. La única diferencia radica en el tipo de contenido que envían a los clientes. El servidor Web soporta archivos HTML, Javascript, multimedia y varios tipos de imágenes, por su parte, el servidor WAP solo soporta el envío de archivos WML, WMLScript, WBMP o XHTMLMP.

Es importante aclarar que el servidor de aplicaciones WAP puede también almacenar todas las tecnologías utilizadas para suministrar contenido dinámico, puede ejecutar Java, PHP, ASP o cualquier otra tecnología soportada por un servidor Web. Esto se debe a que un servidor WAP es simplemente un servidor Web con la funcionalidad WAP añadida. Suministrará todos los servicios normales para un servidor Web pero también actuará como servidor WAP.

Para permitir que un servidor Web normal envíe contenido WAP, simplemente necesitamos agregar los tipos MIME (Multipurpose Internet Mail Extensions) para los archivos WAP en la configuración del servidor. MIME es un método utilizado para convertir y transmitir archivos por Internet. Cuando se transmiten los archivos, el servidor añade un encabezado al archivo, que define el tipo de datos contenido. El cliente que recibe estos archivos puede saber que tipo de archivo es y utilizarlo adecuadamente. La mayoría de los navegadores WAP aceptan solamente tipos WAP MIME.

4.1.3 Lenguajes de programación WAP

Las versiones WAP 1.0 y WAP 2.0 utilizan diferentes lenguajes de programación.

4.1.3.1 WML (WAP 1.0)

El foro WAP creó un nuevo lenguaje el WML (Wireless Markup Language) destinado a minimizar los problemas de la transferencia de datos entre entornos inalámbricos sobre protocolos de Internet. Las sesiones inalámbricas están diseñadas para ser fácilmente suspendidas y retomadas, sin necesidad de la carga de conexiones que provocan los protocolos de Internet. De esta manera los protocolos son adecuados al bajo ancho de banda asociado con las comunicaciones inalámbricas.

WML está basado en XML, y es leído e interpretado por el micronavegador instalado en el dispositivo WAP. Las prestaciones de los navegadores están en relación directa con las capacidades del dispositivo. Cada navegador es distinto y puede interpretar el WML de forma distinta. El foro WAP solo proporciona una Definición de Tipo de Documento (DTD) para WML pero no indica como debe ser mostrado por los navegadores.

WML fue diseñado para visualizar principalmente páginas basadas en texto. Es un lenguaje basado en etiquetas y comparte algunos elementos de HTML. WML define tarjetas (cards) que incluyen otros elementos y atributos permitiendo la visualización del contenido en un teléfono WAP. De la misma manera que un navegador de red puede navegar de una página a otra, el navegador WAP puede navegar de una tarjeta a otra. Una tarjeta puede incluir texto, imágenes, links, campos de entrada de texto, menús, etc.

Cada documento WML está compuesto por una o más tarjetas. Cuando el usuario accede a un sitio WAP, se le envía un lote o paquete (deck) y se le muestra la primera tarjeta. El usuario puede leer el contenido, posiblemente ingresar alguna información y luego moverse a la siguiente tarjeta o a cualquier otra del paquete.

Aunque WML tiene capacidades limitadas si lo comparamos con HTML, tiene también un amplio rango de funcionalidades incluyendo soporte para texto, soporte para imágenes, ingreso de datos, variables, etc.

4.1.3.2 WMLScript (WAP 1.0)

Es un lenguaje de script simplificado, basado en ECMAScript, la versión estandarizada de JavaScript. Provee inteligencia en el cliente, suministrando un conjunto de bibliotecas para realizar operaciones matemáticas, manipular strings, etc. Colabora con WML y ayuda a impedir conexiones innecesarias con el servidor. En particular WMLScript suministra al programador la capacidad de verificación y validación de los datos introducidos por el usuario antes de ser enviados al servidor, impidiendo el envío de datos inválidos; el acceso a utilidades del dispositivo, como la libreta de direcciones, el calendario y la lista de mensajes; así como también métodos de interacción con el usuario sin necesidad del servidor de contenido, incluyendo métodos para visualizar errores y mensajes de advertencia.

A diferencia de lo que sucede con HTML donde el script puede estar embebido en el texto, los archivos WMLScript están separados en archivos diferentes agrupados en lotes de WML. Aunque las tarjetas WML contengan vínculos a archivos WMLScript, estos no se envían al cliente con los archivos WML, tal como sucede con HTML. Actualmente estos archivos WMLScript se envían al cliente WAP sólo cuando el cliente intenta explícitamente el acceso a la información contenida en uno de ellos.

4.1.3.3 XHTMLMP (WAP 2.0)

Como parte de los cambios en la nueva especificación, WAP 2.0 proporciona a los nuevos micronavegadores el nuevo lenguaje XHTML Mobile Profile (XHTMLMP) para brindar nuevo contenido y WML como soporte de WAP 1.x. Ambos lenguajes proporcionan servicios de presentación apropiados para los diferentes dispositivos inalámbricos. Para mejorar la eficiencia de la transmisión WAE 2.0 soporta tokenización de WML1.x y compilación de WMLScript antes de que el gateway envíe el contenido al dispositivo.

El lenguaje de marcado para WAE en WAP 2.0, XHTMLMP, amplía la base de XHTML definida por W3C. Esta base fue diseñada para ser extensible y WAE utiliza esta capacidad agregando características adicionales para extender la funcionalidad hacia los dispositivos móviles. XHTMLMP es muy extensible permitiendo el agregado de elementos si fuese necesario. Además, los documentos escritos en XHTML son totalmente operables en un navegador XHTMLMP. WAE en WAP 2.0 realiza la presentación de contenido con hojas de estilo utilizando una extensión a CSS. Soporta hojas de estilo en línea y externas, como la mayoría de los navegadores de Internet actuales. WAP 2.0 es compatible con WAP 1.x en la capa WAE a través de un soporte nativo para WML y XHTMLMP o por operaciones de transformación de WML1 a WML2. WML2 es una extensión de XHTMLMP que agrega características específicas de WML1 para proveer compatibilidad hacia atrás. El proceso de transformación incluye conversión de nombres y atributos de WML1 a XHTMLMP y soporte para características específicas de WML1. La transformación utiliza XSLT, que permite convertir documentos definidos en WML1 a código WML2 para ser utilizado en los nuevos micronavegadores.

4.1.4 Entornos de desarrollo de aplicaciones WAP

Como WML y XHTML son derivados de XML, y por ende archivos de texto plano, no es estrictamente necesario la creación de página WAP con un entorno de desarrollo (IDE) particular. Solo es necesario un editor de archivos de texto plano y respetar la nomenclatura de etiquetas para crear contenido WAP. Al igual que lo que ocurre en el desarrollo de página Web, es recomendable la utilización de algún editor de archivos que reconozca etiquetas y brinde funcionalidades como determinación de inicio y cierre de etiquetas, atributos y detección de errores.

4.1.5 Emuladores WAP

Una prueba de contenido WAP, consiste en subir el sitio WAP a Internet y corroborar con un celular la visualización de las páginas. Si bien puede tenerse una idea general del resultado esperado, existen un problema adicional: no todos los celulares muestran el contenido WAP de la misma forma. Lo ideal entonces es poder probar la visualización de contenido en la mayor cantidad de dispositivos inalámbricos de los que se disponga.

Para subsanar este inconveniente existen diferentes emuladores de contenido WAP, que permiten simular la visualización de las páginas WAP en una vasta cantidad de teléfonos celulares. La gran mayoría de los emuladores WAP se encuentran disponibles a través de sitios de Internet, lo que permite el testeado de una aplicación WAP subida a Internet. Existen también, pero en menor medida, emuladores disponibles para descarga que permiten probar aplicaciones WAP ejecutando localmente en una PC.

4.1.6 Seguridad de aplicaciones WAP

La seguridad de las aplicaciones WAP es provista a través del cifrado de datos, al igual que las aplicaciones Web.

En WAP 1.x la traducción de datos cifrado se lleva a cabo en la memoria del gateway. Por razones de seguridad ningún dato descifrado se guarda en almacenamiento secundario. El contenido que pertenece a sesiones no seguras si se guarda en un medio de almacenamiento de datos del gateway, reduciendo el tiempo de proceso y los recursos requeridos cuando alguien pide el mismo contenido.

En WAP 2.0 el TLS inalámbrico permite transacciones seguras. Esta implementación de TLS incluye cifrado, certificados digitales, algoritmos de firma, etc. También define el método para establecer túneles TLS que permiten seguridad end-to-end en la capa de transporte.

4.1.7 Inconvenientes con aplicaciones WAP

Internet tal como es actualmente, no está perfectamente preparado para WAP. Como principal problema Internet requiere demasiado ancho de banda y los dispositivos móviles actuales no están completamente preparados para soportar dicho requerimiento. Además existe una segunda limitación relacionada con la estructura interna de las redes inalámbricas, la latencia.

Cuando la información viaja por Internet hacia un dispositivo móvil tiene que pasar por varios elementos de la red móvil, en cada uno se introduce una pequeña demora.

Aunque la tecnología de celulares de tercera generación (3G) ha reducido el problema del ancho de banda y las nuevas redes inalámbricas tienden a reducir la latencia, aún quedan otros problemas que deben ser considerados antes de que los dispositivos móviles puedan adaptarse en su totalidad a Internet. El tamaño de las páginas Web siempre será inadecuado para las tecnologías WAP, teniendo en cuenta que la pantalla de un dispositivo móvil es muy pequeña, no es posible visualizar demasiada cantidad de texto y representar gráficos complejos. Además un teléfono móvil generalmente no tiene un teclado normal o un mouse, por lo tanto es mucho más difícil para un usuario introducir la información en un teléfono que en una computadora.

Al diseñar una aplicación móvil deben tenerse en cuenta estas limitaciones del dispositivo del cliente. Los protocolos de Internet no son adecuados para el uso en las comunicaciones con teléfonos móviles ya que requieren muchos mensajes entre los clientes y el servidor simplemente, por ejemplo, para establecer una comunicación de conexión. Al tratar de minimizar la sobrecarga de mensajes se requiere un alto poder de procesamiento en el dispositivo del cliente. El dispositivo deberá resolver internamente ciertas cuestiones tratando de evitar mensajes hacia el servidor.

4.2 Aplicaciones Java ME

Una aplicación Java ME es un programa escrito en lenguaje Java que puede ejecutarse sobre una plataforma o máquina virtual Java (JVM). Los dispositivos móviles que cuentan con soporte Java, poseen entonces una JVM instalada que permite la ejecución de aplicaciones Java ME. Para que la aplicación pueda ejecutarse, debe copiarse al teléfono celular a través de un cable de datos, por infrarrojo, por bluetooth o ser descargada de algún sitio en Internet. Una vez descargada la aplicación y ejecutando sobre la JVM del dispositivo móvil, puede acceder a Internet a través de funcionalidades particulares provistas por Java ME.

4.2.1 Tipo de aplicaciones Java ME

Si bien las aplicaciones Java ME se relacionan directamente con el entretenimiento, los límites de esta tecnología, superan ampliamente los alcances de este rubro. Java ME brinda la posibilidad de administrar dos tipos de interfaces, una a alto nivel a través del manejo de Screen y otra a bajo nivel a través de Canvas. Los juegos aprovechan al máximo la funcionalidad de bajo nivel, trabajando directamente con gráfica. En cambio, las aplicaciones utilizan mayormente Screen, ya que posee componentes GUI como campos de texto, listas y formularios, y eventos que permiten controlar la interacción con el usuario.

El manejo de eventos, también puede hacerse a bajo nivel, controlando por ejemplo cuando un usuario presiona una tecla del teléfono, o a alto nivel por ejemplo eligiendo un ítem de una lista. Al producirse un evento, puede notificarse a la aplicación y ejecutar las acciones adecuadas en respuesta al usuario.

Otro tipo de aplicaciones que pueden desarrollarse con Java ME, son aquellas que utilizan la infraestructura y servicios provistos por las redes celulares, como la navegación Web por HTTP

o HTTPS, correo electrónico por IMAP o aplicaciones que utilizan servicios Web disponibles en Internet.

4.2.2 El Lenguaje Java

El lenguaje de programación Java fue desarrollado en 1995 por James Gosling de la empresa Sun Microsystems. Inicialmente el lenguaje fue desarrollado para dispositivos electrónicos como un intento de resolver simultáneamente todos los problemas que se le plantean a los desarrolladores de software por la proliferación de arquitecturas incompatibles, tanto entre las diferentes máquinas como entre los diversos sistemas operativos que funcionaban sobre una misma máquina, añadiendo la dificultad de crear aplicaciones distribuidas en una red como Internet. Hace algunos años, Sun Microsystems decidió intentar introducirse en el mercado de la electrónica de consumo y desarrollar programas para pequeños dispositivos electrónicos. El mercado inicialmente previsto eran los equipos domésticos: microondas, tostadoras y, fundamentalmente, televisión interactiva. Este mercado, dada la falta de pericia de los usuarios para el manejo de estos dispositivos, requería interfaces mucho más cómodas e intuitivas que los sistemas que proliferaban en el momento. Otros requisitos importantes a tener en cuenta eran la fiabilidad del código y la facilidad de desarrollo. Gosling decidió que las ventajas aportadas por la eficiencia de C++ no compensaban el gran costo de pruebas y depuración. Gosling había estado trabajando en su tiempo libre en un lenguaje de programación que él había llamado Oak, el cual, aún partiendo de la sintaxis de C++, intentaba remediar las deficiencias que iba observando.

Sin embargo, el proyecto inicial fracasó y pese a lo que parecía ya un olvido definitivo, Bill Joy, cofundador de Sun y uno de los desarrolladores principales del Unix de Berkeley, juzgó que Internet podría llegar a ser el campo de juego adecuado para disputar a Microsoft su primacía casi absoluta en el terreno del software, y vio en Oak el instrumento idóneo para llevar a cabo estos planes. Tras un cambio de nombre y modificaciones de diseño, el lenguaje Java fue presentado en sociedad en agosto de 1995.



James Gosling – creador de Java

4.2.2.1 Características

Simple: Java ofrece toda la funcionalidad de un lenguaje potente, pero sin las características menos usadas y más confusas de éstos. C y C++ son lenguajes muy difundidos, por ello Java se diseñó en base a C++ para facilitar un rápido y fácil aprendizaje. Java elimina muchas de las características de otros lenguajes para mantener reducidas las especificaciones del lenguaje y añadir características útiles como el garbage collector (recolector de memoria dinámica).

Distribuido: Java se ha construido con extensas capacidades de interconexión TCP/IP. Existen librerías de rutinas para acceder e interactuar con protocolos como http y ftp. Esto permite acceder a la información a través de la red con tanta facilidad como a los ficheros locales. Java proporciona las librerías y herramientas para que los programas puedan ser distribuidos, es decir, que se corran en varias máquinas, interactuando.

Robusto: Java realiza verificaciones en busca de problemas tanto en tiempo de compilación como en tiempo de ejecución. La comprobación de tipos en Java ayuda a detectar errores, lo antes posible, en el ciclo de desarrollo. Java obliga a la declaración explícita de métodos, reduciendo así las posibilidades de error. Maneja la memoria para eliminar las preocupaciones por parte del programador de la liberación o corrupción de memoria.

Arquitectura Neutral: Un archivo Java es compilado y genera un archivo .class denominado bytecode. Este archivo es posteriormente ejecutado por un intérprete que varía según la arquitectura y el sistema operativo. Ese intérprete se lo conoce como la Máquina Virtual de Java (JVM). Con esta funcionalidad, se escribe un programa, se lo compila, y ese archivo puede funcionar en cualquier sistema operativo (para el que haya una máquina virtual).

Seguro: La seguridad en Java tiene dos facetas. Cuando se usa Java para crear un navegador, se combinan las características del lenguaje con protecciones de sentido común aplicadas al propio navegador. El código Java pasa muchos tests antes de ejecutarse en una máquina. El código se pasa a través de un verificador de byte-codes que comprueba el formato de los fragmentos de código y aplica un probador de teoremas para detectar fragmentos de código ilegal -código que falsea punteros, viola derechos de acceso sobre objetos o intenta cambiar el tipo o clase de un objeto-. En resumen, las aplicaciones de Java resultan extremadamente seguras, ya que no acceden a zonas delicadas de memoria o de sistema. Java imposibilita, también, abrir ficheros de la máquina local (siempre que se realizan operaciones con archivos, éstas trabajan sobre el disco duro de la máquina de donde partió el programa), no permite ejecutar aplicaciones nativa e impide que se utilicen otros ordenadores como puente, es decir, nadie puede utilizar nuestra máquina para hacer peticiones o realizar operaciones con otra. Además, los intérpretes que incorporan los navegadores de la Web son aún más restrictivos.

Interpretado: El intérprete Java puede ejecutar directamente el código objeto. Enlazar un programa, normalmente, consume menos recursos que compilarlo, por lo que los desarrolladores con Java pasarán más tiempo desarrollando y menos esperando por el ordenador.

Multithreaded: Al ser multithreaded Java permite muchas actividades simultáneas en un programa a través de diferentes hilos de control. Los threads o procesos ligeros, son básicamente pequeños procesos o piezas independientes de un gran proceso. El beneficio de

ser multithreaded consiste en un mejor rendimiento interactivo y mejor comportamiento en tiempo real. Aunque el comportamiento en tiempo real está limitado a las capacidades del sistema operativo subyacente (Unix, Windows, etc.), aún supera a los entornos de flujo único de programa (single-threaded) tanto en facilidad de desarrollo como en rendimiento.

Dinámico: Java se beneficia todo lo posible de la tecnología orientada a objetos. Java no intenta conectar todos los módulos que comprenden una aplicación hasta el tiempo de ejecución. Las librerías nuevas o actualizadas no paralizarán las aplicaciones actuales (siempre que mantengan el API anterior). Java también simplifica el uso de protocolos nuevos o actualizados.

4.2.3 Entorno de desarrollo Java

Existen distintos entornos de desarrollo que permiten generar código Java ME. Sun Microsystems distribuye gratuitamente el Java Development Kit (JDK). Se trata de un conjunto de programas y librerías que permiten desarrollar, compilar y ejecutar programas en Java. Incorpora además la posibilidad de ejecutar parcialmente el programa, deteniendo la ejecución en el punto deseado y estudiando en cada momento el valor de cada una de las variables a través de un Debugger. Cualquier programador con un mínimo de experiencia sabe que una parte muy importante del tiempo destinado a la elaboración de un programa se destina a la detección y corrección de errores. Existe también una versión reducida del JDK, denominada JRE (Java Runtime Environment) destinada únicamente a ejecutar código Java (no permite compilar).

Para el desarrollo de aplicaciones inalámbricas, Java provee el Java ME Wireless Toolkit, un conjunto de herramientas para crear aplicaciones que corren en dispositivos compatibles con Java ME. Este paquete se compone de librerías, herramientas de creación, utilitarios y emuladores de dispositivos que permite la creación de aplicaciones para CLDC y MIDP.

Un entorno de desarrollo integrado o en inglés Integrated Development Environment (IDE) es un programa compuesto por un conjunto de herramientas para un programador. Puede dedicarse en exclusiva a un sólo lenguaje de programación o bien, poder utilizarse para varios.

Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica GUI. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. . Estos entornos integrados permiten desarrollar las aplicaciones de forma mucho más rápida, incorporando en muchos casos librerías con componentes ya desarrollados, los cuales se incorporan al proyecto o programa.

Para el caso particular de Java, existen diferentes IDEs que permiten generar aplicaciones Java. Eclipse y NetBeans son dos entornos de software libre para desarrollo de aplicaciones Java, ambos posibilitan también la creación de aplicaciones Java ME. A su vez, también existen entornos propietarios desarrollados por las principales empresas, como WebSphere de IBM (versión comercial de Eclipse), JDeveloper de Oracle o JBuilder de Borland, que también posibilitan la creación de aplicaciones Java ME.

4.2.4 Componentes de una aplicación Java ME

Una aplicación Java ME consiste en una aplicación Java que utiliza los beneficios de la arquitectura Java ME, que provee una plataforma de ejecución de programas Java, un conjunto de paquetes opcionales o librerías para proveer funcionalidad a la aplicación, y diferentes perfiles que permiten desarrollar aplicaciones para diversos dispositivos móviles.

Una aplicación Java consiste en un conjunto de clases de objetos, agrupadas en paquetes y almacenados en archivos fuente con extensión .java. El compilador java, a partir de los archivos fuente, genera archivos de bytecode con extensión .class. Estos últimos archivos, pueden ser empaquetados en diferentes archivos JAR (JAVa Archive). La aplicación Java ME, consiste además de archivos descriptores que determinan la configuración de la aplicación.

4.2.5 Acceso a Internet desde Java ME

Para el acceso a Internet, a través de MIDP, Java ME brinda la posibilidad de utilizar los protocolos HTTP y HTTPS, e introduce soporte opcional para sockets, datagramas, SSL y serial. Además, a través de la especificación WMA, brinda soporte para mensajería SMS y MMS. También provee especificaciones para conexión Bluetooth.

4.2.6 Emuladores Java ME

El ideal de repetir el comportamiento exacto de un dispositivo es muy difícil de conseguir. Un desarrollador puede utilizar el emulador provisto por el J2ME Wireless Toolkit u otro disponibles en Internet. El problema es la enorme variedad de dispositivos y las diferencias entre cada uno de ellos, lo que provoca que las aplicaciones se ejecuten correctamente, en apariencia y comportamiento, en un grupo de dispositivos y en otros no. Estas diferencias se resaltan si se utilizan librerías o API's de vendedores específicos.

Entonces el emulador debe ser considerado conceptual, es decir, no representa un dispositivo específico sino un servicio para demostrar las características generales de cierta clase de dispositivo, como el comportamiento de botones o controles del celular.

El emulador provisto por J2ME Wireless Toolkit brinda una importante serie de características. Permite crear diferentes apariencias, para brindar una aproximación de la experiencia del usuario con los métodos de entrada y con la pantalla de un dispositivo particular. En una nueva apariencia puede especificarse el tamaño de pantalla, la profundidad de color, las fuentes, y los métodos de entrada de un dispositivo. Proporciona un ambiente sólido y robusto para la prueba de aplicaciones que no utilizan APIs de un vendedor específico, corriendo bajo Windows, Linux, y Solaris. Además, incluye herramientas útiles para el desarrollador de aplicaciones, incluyendo simulación de performance prueba de métodos y monitoreo de red.

Lo ideal es entonces, conseguir la mayor cantidad de emuladores posibles para testear aplicaciones, y luego conseguir la mayor cantidad de diferentes dispositivos celulares para verificar que la aplicación se comporta como se espera.

4.2.7 Seguridad en Java ME

El éxito de las aplicaciones comerciales móviles, como por ejemplo el pago de servicios e insumos, requiere un de alto nivel de seguridad y de confianza. El dispositivo móvil esencialmente se convierte en la identidad digital de su dueño, por ende debe haber una forma de autenticar al usuario y asegurarse que el dispositivo sea confiable. La proporción de datos seguros sobre redes móviles y la reducción de fraude en sistemas móviles de pago son dos cuestiones críticas.

Para realizar operaciones confiables, JavaME confía en servicios de seguridad proporcionados en smart cards o en dispositivos similares para asegurarse que las claves criptográficas y los procesos criptográficos se realizan con seguridad. A fin de implementar estos requisitos de seguridad para aplicaciones móviles se ha desarrollado, a través de Java Community Process (JCP), Security and Trust Services APIs (SATSA) para Java ME.

La mejor manera de autenticar usuarios y autorizar transacciones es obteniendo una firma digital desde el dispositivo. SATSA proporciona a las aplicaciones Java ME, la posibilidad de generar firmas digitales, administración de certificados, requerimientos de certificación ante autoridades certificadoras y alta y baja en un almacén de certificados. SATSA también incluye una API que proporciona un subconjunto de operaciones criptográficas básicas, tales como verificación de firma, cifrado, descifrado y acceso a aplicaciones seguras.

4.2.8 Inconvenientes con aplicaciones Java ME

La industria de datos móviles está emergiendo y en esta etapa de su evolución, es lógico que se produzcan fragmentos. Las aplicaciones móviles son principalmente consumidoras y para ser comercialmente viable, estas aplicaciones necesitan un mercado importante. Esto se logra a través de un único estándar propietario o con estándares abiertos no controlados y con pocas barreras por parte de la industria. La fragmentación implica que muy pocos servicios móviles sean útiles hoy día, por eso existen miles de aplicaciones de contenido, que permiten descarga de ringtones, imágenes, etc. pero muy pocas aplicaciones utilitarias de uso masivo.

La portabilidad de aplicaciones no es simple y requiere que los desarrolladores adapten diferencias en resolución de la pantalla, velocidad del procesador, límites de memoria, y capacidades de sonido, que pueden variar violentamente entre dispositivos.

4.3 Aplicaciones SMS

Una aplicación SMS consiste en un intercambio de mensajes de texto entre un celular y un servidor Web que recibe dicho mensaje a través de una aplicación Web y puede enviar una respuesta al dispositivo. Las aplicaciones SMS requieren de la participación del operador y de un proveedor de servicios que contenga un Gateway para permitir la comunicación entre el servidor Web y el celular.

4.3.1 Tipos de aplicaciones SMS

SMS fue diseñado inicialmente para soportar mensajes de tamaño limitado, sobre todo notificaciones y páginas numéricas o alfanuméricas. Mientras que estas aplicaciones continuarán siendo ampliamente utilizadas, hay nichos más recientes que SMS todavía puede explotar.

La utilización de textos cortos es el corazón de muchas aplicaciones que estaban restringidas a terminales fijos conectados a una red. Sin embargo, muchas de estas aplicaciones son aprovechadas de mejor forma si a la capacidad de comunicación de datos y localización, puede agregársele la movilidad de la estación de trabajo. Este tipo de solución es muy conveniente para aplicaciones que utilizan activos móviles como vehículos, cargas o incluso personas.

Otro tipo de aplicaciones que puede utilizar SMS como mecanismo de transporte es el banking. Si bien las transacciones por Internet son más económicas que los mensajes de texto, habilitando el acceso desde el celular a los clientes para chequear sus saldos, realizar transferencia entre cuentas y/o pagar deudas y tarjetas de crédito, tiene un valor agregado de importancia no solo para los clientes sino también para las instituciones financieras.

Los entretenimientos son también aplicables a SMS. Ejemplos son el intercambio de mensajes entre participantes de una trivía, encuesta o competencia, la suscripción a servicios diarios, o incluso la descarga de fondos de pantalla y tonos.

4.3.2 El envío, recepción y respuesta de mensajes

El envío, recepción y respuesta necesita de tres actores tecnológicos, el teléfono celular que envía y recibe los mensajes, el operador de red que permite la comunicación, el servidor de aplicaciones que brinda la funcionalidad deseada y el proveedor de contenidos que actúa como intermediario entre el operador de red y el servidor de aplicaciones.

Los pasos que involucra esta operatoria son los siguientes:

1. El usuario envía desde su celular un mensaje de texto a un número predeterminado. El mensaje de texto generalmente incluye una palabra clave y un dato adicional opcional. La palabra clave indica la funcionalidad deseada y el dato adicional puede utilizarse como parámetro de la funcionalidad.
2. El mensaje de texto llega al operador de red celular.
3. El operador de red deriva el mensaje hacia el proveedor de contenidos asociado con el número al que se envió el mensaje.
4. El proveedor de contenidos recibe el mensaje de texto, a través de un gateway WAP y a través de Internet lo deriva hacia el servidor de aplicaciones, con un acuerdo previo acerca del protocolo, del formato del requerimiento y sus parámetros.
5. El intercambio de mensajes entre el proveedor de contenido y el servidor de aplicaciones, puede realizarse vía correo electrónico (SMTP), SOAP sobre HTTP, SMPP, SNPP o por un POST HTTP(S).

6. El servidor de aplicaciones recibe el mensaje diferenciando la palabra clave para proveer la funcionalidad y los parámetros de la misma. También recibe el número de teléfono celular origen del mensaje.
7. El servidor de aplicaciones realiza la funcionalidad asociada a la palabra clave y puede enviar una respuesta al teléfono origen a través del proveedor de contenido.
8. El proveedor de contenido, deriva la respuesta a través del gateway WAP al operador de red, indicando también el número de teléfono destino.
9. El operador de red envía como mensaje de texto, la respuesta al dispositivo celular.

4.3.3 Características de las aplicaciones SMS

Las aplicaciones SMS se diferencian de las anteriores, por su simplicidad de desarrollo. En sí, puede consistir de un conjunto de servicios Web (o respuestas HTTP) con una escalabilidad estable, es decir, agregar nueva funcionalidad implica el desarrollo de un nuevo servicio o respuesta con menor, igual o mayor complejidad que los desarrollos anteriores. Quizá la mayor dificultad a la hora de implementar una aplicación SMS radica en las negociaciones que deben realizarse con los proveedores de contenido y con los operadores de red, para acordar no solo las cuestiones comerciales, sino también las operativas (autenticación, número destino, protocolo y puerto para intercambio de datos, formato de datos, etc.).

Por último es importante destacar la principal diferencia de las aplicaciones SMS versus las aplicaciones WAP y Java ME: la posibilidad de cobrar al receptor, cada invocación a una funcionalidad de la aplicación. SMS, permite que el costo de cada mensaje enviado, sea cobrado al receptor, adicionando el costo por funcionalidad ejecutada. Es decir, el envío de información al celular desde la aplicación SMS, pueden ser valorizados por el desarrollador de la misma para ser cobrado a cada receptor que lo solicite.

4.3.4 Beneficios de las aplicaciones SMS

Las principales ventajas de SMS tienen que ver con la conveniencia, flexibilidad, y la integración de servicios de mensajería y datos, es decir la posibilidad de utilizar el teléfono celular como extensión de la computadora. SMS elimina la necesidad de dispositivos separados para mensajería, integrando los servicios en un solo dispositivo.

Las ventajas de SMS incluyen:

- √ Notificaciones de entrega y alertas
- √ Entrega garantizada de mensajes
- √ Mecanismo de comunicación de bajo costo
- √ Envío selectivo y múltiple de mensajes
- √ Integración con otros servicios y aplicaciones de Internet
- √ Posibilidad de cobro de mensajes al receptor

Capítulo 5. Desarrollo de una aplicación WAP

El desarrollo de una aplicación WAP implica la toma de decisiones estratégicas en cuanto a tecnología, herramientas de desarrollo y diseño de la aplicación. En este capítulo se describen objetivos y motivación para el desarrollo de una aplicación WAP, las especificaciones necesarias de tecnología y herramientas y el diseño de la aplicación. Por último se describe el prototipo implementado, su funcionalidad y los resultados del desarrollo.

5.1 Objetivo

El objetivo de la aplicación es proveer un módulo WAP al sistema SIU Guaraní.

5.2 Motivación

El sistema SIU-Guaraní es un sistema de gestión de alumnos implementado en más de 150 unidades académicas de 30 universidades de la Argentina. Si bien su arquitectura principal es cliente-servidor, posee una interfaz Web, denominada Guaraní 3W para acceso de alumnos, docentes y autoridades. La importancia de varias operaciones disponibles vía Web, sumado a las necesidades de movilidad y accesibilidad de las mismas, crean una oportunidad única para un desarrollo WAP.

El SIU, entidad dependiente de la Secretaría de Políticas Universitarias del Ministerio de Educación, Ciencia y Tecnología, desarrolla soluciones informáticas y brinda servicios para el Sistema Universitario Nacional. Su objetivo es contribuir a mejorar la gestión de las instituciones, permitiéndoles contar con información segura, íntegra y disponible, optimizar sus recursos y lograr que el software sea aprovechado en toda su potencialidad.

El desarrollo de una interfaz WAP para Guaraní implica notorios beneficios a los alumnos, docentes y a la institución que lo implemente. La movilidad, característica principal de las aplicaciones WAP, brinda a alumnos y docentes posibilidades inmejorables de acceder a información académica como resultado de exámenes, inscripciones, horarios, mensajes, solicitudes, etc.

5.3 Sistema SIU-Guaraní

SIU-Guaraní es un sistema de información para gestión de alumnos de las universidades, que registra y administra todas las actividades académicas de la institución, desde que el alumno se inscribe como aspirante hasta que obtiene su título. Fue concebido para administrar la gestión de alumnos en forma segura, con la finalidad de obtener información consistente para los niveles operativos y directivos.

El sistema brinda servicios para alumnos, docentes, usuarios administrativos y autoridades, ya que pueden explorar los datos y obtener información como soporte para toma de decisiones.

Las prestaciones del sistema incluyen administración de planes de estudio y carreras, planificación, gestión de matrícula, gestión de cursadas, gestión de aulas, exámenes equivalencia, egresados, administración y misión de certificados, mensajería, encuestas.

Además brinda diferentes prestaciones diferentes tipos de perfiles, autoridades, docentes, alumnos.

5.3.1 Descripción técnica

La arquitectura principal del sistema corresponde a un diseño cliente y servidor, desarrollado con Power Builder como lenguaje de programación e Informix como servidor de base de datos. El subsistema denominado Guaraní3w es una aplicación Web desarrollado en PHP. Gran parte de las reglas de negocios están escritas en procedimientos almacenados en la base de datos, lo que simplifica el desarrollo de diferentes interfaces.

SIU-Guaraní cuenta con controles dinámicos y parámetros generales que permiten definir el comportamiento del sistema. Los parámetros generales son variables que al momento de la implementación toman un valor en función de ciertas necesidades que tenga la Unidad Académica. Los distintos valores que pueden tomar estos parámetros implican una forma diferente de operar del SIU-Guaraní, esto es parte de lo que hace flexible al sistema ante distintas realidades.

Otra herramienta para adaptarse a diferentes necesidades son los controles dinámicos. Estos controles son validaciones que se asocian a una operación (ítem de menú) y que al momento de ejecutarse la misma son disparados. La forma de comportarse de estos controles para cada operación a la que están asociados se determina al implementar el sistema junto con los parámetros generales. El sistema provee un conjunto de controles que puede ser ampliado por la Unidad Académica sin necesidad de modificar el sistema.

5.3.2 Características principales

Privacidad: El menú de la aplicación está compuesto por operaciones que se agrupan en módulos y submódulos. La aplicación permite crear usuarios del sistema con perfiles particulares en donde cada usuario debe tener asignado un conjunto de operaciones que serán las únicas que puede realizar. El usuario debe tener un nombre que lo identifique y una clave para asegurar su identidad.

Auditoria: El sistema registra la operatoria realizada (datos modificados y su estado previo), el usuario que la realizó y el momento en que se llevó a cabo (fecha y hora). Esto permite seguir el comportamiento de un usuario determinado.

Confiablez: La información se encuentra almacenada en un motor de base de datos relacional que asegura la consistencia de los datos y brinda mecanismos para realizar tareas de respaldo (back up). Ante algún imprevisto que cause la caída del sistema, deben contemplarse con celeridad, las distintas opciones que brinda el sistema para reconstruir el contenido de la base de datos. La tarea de back up es responsabilidad de la Unidad Académica.

Flexibilidad y evolución: El sistema tiene una arquitectura que le permite adaptarse a las realidades que existen en las distintas Universidades.

5.3.3 Guaraní3w

El subsistema Guaraní3w corresponde al desarrollo de una interfaz Web que permite a los alumnos, docentes y autoridades realizar distintos trámites y o consultas en forma autónoma utilizando terminales de autogestión colocados dentro de la institución o accediendo desde cualquier computadora a través de Internet.

Una característica muy importante del Sistema SIU-Guaraní es que la lógica principal del sistema está desarrollada dentro de la base de datos Informix, en forma de procedimientos almacenados. Esta arquitectura simplifica la tarea de mantener el sistema y el desarrollo de nuevas prestaciones funcionales ya que la lógica de la aplicación está escrita en un único lugar, independientemente de la interfaz que se utilice para acceder a los datos.

La aplicación Guaraní3w esta implementada en el lenguaje de programación PHP y utiliza la librería de abstracción ADOdb para la conexión a la base de datos, la cual permite independizarse de la sintaxis específica de cada lenguaje de consulta de la base de datos. En particular permite trabajar tanto con conexiones ODBC como con librerías nativas de Informix.

5.4 Aplicación WAP para Guaraní

Guaraní WAP será el módulo WAP del Sistema de gestión de alumnos SIU-Guaraní y permitirá a alumnos y docentes acceder a funcionalidad del sistema SIU-Guaraní desde un teléfono celular. Los beneficios que persigue este desarrollo consisten en brindar movilidad, accediendo a la información en cualquier momento desde cualquier lugar con un costo muy bajo.

Guaraní WAP estará disponible para todas las universidades que lo requieran, como una mejora del sistema SIU-Guaraní e incluirá la posibilidad de que el alumno consulte sus inscripciones a cursadas y exámenes, pueda ver el estado de sus solicitudes y acceda a los resultados de sus parciales. Además, los docentes podrán consultar la cantidad de inscriptos a sus cursadas y exámenes, sus actas pendientes y sus agendas de exámenes y cursadas.

5.4.1 Descripción técnica del prototipo

El destino objeto de la aplicación serán dispositivos celulares GSM con soporte GPRS o EDGE para transmisión de datos.

El prototipo a implementar será una aplicación compatible con el estándar WAP2.0 correspondiente a un módulo embebido dentro de la interfaz Guaraní3w.

Se reutilizará la lógica del sistema SIU-Guaraní y programación general del subsistema Guaraní3w, desarrollando un subconjunto de funcionalidades específicas, adecuando el esquema de navegación y la interfaz gráfica al común de los celulares disponibles en Argentina.

La interfaz será intuitiva facilitando la navegación y el acceso a la información por parte de los usuarios. Se minimizará el ingreso de texto al sistema y la cantidad de caracteres incluidos en las páginas WAP para optimizar el acceso a los datos y evitar la sobrecarga de información en el dispositivo.

5.4.2 Decisiones en cuanto a implementación

- ✓ La funcionalidad a implementar corresponde a un subconjunto de la funcionalidad disponible en Guarani3w, con adecuación de interfaces y esquema de navegación.
- ✓ El desarrollo formará parte de la aplicación Guarani3w, como un módulo embebido en la misma. Para ello deberá definirse un esquema de estandarización de nombres de páginas y procedimientos de base de datos.
- ✓ Se desarrollará la aplicación para el estándar WAP2.0, disponible para un amplio y creciente porcentaje del mercado de celulares GSM en Argentina.
- ✓ Como entorno de desarrollo no se utilizará un IDE predeterminado sino que sólo se utilizará un editor de texto que remarque etiquetas HTML.
- ✓ El lenguaje de programación a utilizar será PHP para reutilizar la funcionalidad ya implementada en Guarani3w.
- ✓ Se utilizará el servidor Web Apache versión 2.x, como entorno de prueba.
- ✓ El formato del contenido será XHTML, minimizando el uso de etiquetas y atributos de estilo, para reducir el impacto de uso en diferentes celulares.
- ✓ Para reducir costo a los usuarios, se utilizará una interfaz orientada a texto, sin imágenes u otros elementos multimedia.
- ✓ Se minimizará el tamaño de páginas retornado, tratando de que las mismas no superen los 10KB.
- ✓ Será reutilizado el esquema de menús y validación de Guarani y los procesos de base de datos utilizados en las operaciones de Guarani3w.
- ✓ Se reutilizarán páginas PHP desarrolladas para Guarani3w cuando sea posible, se agregarán nuevas páginas cuando sea necesario, pero no se sobrescribirá ninguna página PHP escrita para Guarani3w.
- ✓ Se reutilizarán procedimientos almacenados de base de datos Informix, cuando sea posible, se agregarán nuevos procedimientos cuando sea necesario, pero no se sobrescribirá ningún procedimiento almacenado escrito para Guarani o Guarani3w.
- ✓ El acceso a la aplicación WAP, será a través de un path diferente al acceso a Guarani3w, pudiendo definir alias virtuales para facilitar el acceso.
- ✓ No se hará detección de gente usuario (USER-AGENT), para determinar cuando un requerimiento proviene desde un navegador Web o desde un dispositivo celular.
- ✓ Se brindarán todos los mecanismos y herramientas para permitir la personalización de las operaciones en cada institución que implementen esta aplicación.

5.4.3 Relevamiento de requerimientos técnicos

Los requerimientos técnicos de la aplicación incluyen el hardware y software de servidor y las características de los dispositivos clientes de la aplicación.

5.4.3.1 Hardware y software de servidor

La aplicación será un módulo embebido en Guaraní 3w, por ende los requerimientos de hardware son equivalentes a los necesarios para dicho módulo. Guaraní 3w es una aplicación Web, desarrollada en PHP 4.x, que puede embeberse en los principales servidores Web como Apache 1.3.x, 2.x e Internet Information Server.

Como hardware mínimo es necesario un servidor que soporte Apache 1.3.x o IIS, por ende debe cumplir los siguientes requerimientos:

- Arquitectura monoprocesador basada en registros de 32 bits o superior, totalmente compatible con arquitecturas Intel Pentium IV o AMD Opteron.
- Memoria RAM de al menos 1GB DDR333, con una frecuencia de reloj de 166Mhz.
- Controladora de discos rígidos EIDE ATA100, con al menos 1 dispositivo de 80GB de capacidad y velocidad de rotación de 7200rpm.
- Placa de red Ethernet/Fast Ethernet IEEE 802.3, 10/100 Mbps.
- La cantidad y velocidad de procesadores así como la capacidad de memoria RAM puede variar de acuerdo a la cantidad de conexiones simultáneas que requiera Guaraní 3w para un óptimo funcionamiento.
- Como software mínimo se requiere, Apache 1.3 y PHP 4.

5.4.3.2 Hardware y software cliente

Para que los dispositivos celulares puedan soportar esta aplicación, se requiere como mínimo:

- ✓ Sistema Operativo Symbian OS, Windows Mobile o Palm OS.
- ✓ Conexión a Internet de 64Kbps con tecnología GPRS o EDGE.
- ✓ Navegador WAP 2.0 con interfaz visual y soporte XHTML.
- ✓ Soporte de sesiones a través de cookies.

Algunos dispositivos en Argentina que soportan estos requerimientos se describen en Anexo A.

5.4.4 Relevamiento de requerimientos funcionales

El relevamiento de requerimientos funcionales corresponde a un trabajo realizado en conjunto con el equipo de desarrollo del sistema SIU-Guaraní para recavar información sobre la funcionalidad a implementar y los beneficios del desarrollo de la aplicación.

5.4.4.1 Funcionalidad a implementar para docentes y alumnos

Se desea implementar un total de 12 operaciones con prioridad principal a través de una interfaz WAP:

Funcionalidad general

- Iniciar sesión: Implica la posibilidad de ingreso de un nombre de usuario y una contraseña para acceder a la aplicación. En caso de validación exitosa del usuario, se despliega el menú de operaciones disponibles según el perfil de la persona que ingresa al sistema.
- Finalizar sesión: Corresponde al fin de trabajo del usuario en la aplicación. Esta operación siempre estará disponible a través de alguna opción de menú de la aplicación.

Funcionalidad específica para alumnos

- Listado de inscripciones a exámenes: A través de esta opción de menú, el alumno puede consultar las materias a las cuales se ha inscripto a examen.
- Listado de inscripciones a cursadas: Ídem anterior, para consultar las materias que debe cursar.
- Evaluaciones parciales: Esta operación mostrará al alumno las materias que está cursando actualmente y que tengan evaluaciones parciales. El alumno deberá seleccionar la materia para la cual quiere consultar la evaluación parcial y como resultado del envío, debe mostrarse en forma de link los nombres de las evaluaciones parciales (parcial, recuperatorio, etc.) que tienen notas cargadas.
- Notas de evaluaciones parciales: Es la continuación de la operación anterior. Al seleccionar alguna de las evaluaciones parciales, se muestra al alumno el resultado de la evaluación.
- Certificados solicitados: Cuando el alumno selecciona esta opción de menú, la aplicación muestra todas las solicitudes que ha realizado a oficina de alumnos y que aún están pendientes de resolución.

Funcionalidad específica para docentes

- Agenda de exámenes: Mostrará al docente la información de las mesas de exámenes en las que participa con número de llamado, fecha, hora y aula del examen.
- Agenda de cursadas: Similar al caso anterior, pero con información de cursadas del docente.
- Cantidad de alumnos inscriptos en una mesa de examen: Esta operación mostrará al docente la lista de materias en las cuales participa tomando examen y luego de seleccionar una de ellas, el sistema retorna un listado con las mesas de exámenes de la misma, incluyendo la cantidad de inscriptos en cada mesa.
- Actas de examen sin cerrar: Este reporte indica cuales actas de examen en las que participo el docente, están aun sin cerrar.
- Cantidad de inscriptos a una comisión de cursada: Esta operación mostrará al docente la lista de materias en las cuales participa y luego de seleccionar una de ellas, el sistema retorna un listado con las comisiones de la misma, incluyendo la cantidad de inscriptos en cada comisión.

5.4.4.2 Funcionalidad a implementar en etapas posteriores

A futuro se prevé la implementación de la siguiente funcionalidad.

Funcionalidad general

- Visualización de mensajes.

Funcionalidad específica para alumnos

- Cronograma de evaluaciones parciales.
- Solicitud de certificados.
- Próximas mesas de exámenes.
- Consulta de créditos.
- Porcentaje de asistencia a cursadas.
- Materias regulares.
- Notas de un examen.
- Inscripción a exámenes.
- Inscripción a cursadas.

Funcionalidad específica para docentes

- Actas de cursada sin cerrar.
- Cronograma de evaluaciones parciales.

5.4.4.3 Beneficios perseguidos con la implementación

Entre los beneficiarios de la implementación del módulo WAP del sistema Guaraní 3w encontramos 2 grandes grupos de usuarios.

Alumnos: Los alumnos se ven beneficiados por contar con acceso a información académica en el momento, sin necesidad de acercarse a ventanilla, ni contar con una PC con acceso a Internet. La funcionalidad a implementar permite la interacción del alumno con la universidad, pudiendo consultar su información académica (resultados, horarios, inscripciones), realizar solicitudes (certificados analíticos y regulares) y transacciones (inscripciones para cursadas y/o exámenes).

Docentes: El beneficio para docentes se ve reflejado en la posibilidad de consultar la información referente a cantidad de inscriptos a cursadas y/o finales pudiendo organizar material de clase y planificar actividades, minimizando los diálogos con oficinas de alumnos, o con alumnos.

5.4.5 Análisis de requerimientos técnicos

El análisis de requerimientos técnicos presenta las dificultades de implementación de la aplicación, las limitaciones en cuanto acceso a la tecnología y las diferencias y adecuaciones de interfaz que deben aplicarse para el desarrollo de la solución. Asimismo se describen las decisiones tomadas para minimizar el impacto de los problemas de implementación.

5.4.5.1 Dificultades o imposibilidades de implementación

Las dificultades o imposibilidades de implementación se basan en los problemas inherentes a la interfaz de los celulares, que dificultan la creación de páginas con amplio contenido textual o gráfico como la disposición de un número máximo de filas y columnas para la visualización del texto, las limitaciones de tamaño de letras o imágenes y las limitaciones en cuanto al tamaño máximo en bytes de páginas.

Como resultado de este análisis se determina la utilización de una interfaz de tipo textual, basada en un número máximo de filas y fijo de columnas, donde se muestre la información solicitada y el menú de operaciones. Las operaciones de navegación, como "atrás", "adelante", "ingresar" a una opción de menú y "salir" de la aplicación estarán disponibles a través de teclas del teléfono celular. Sólo se prevé la presencia de una imagen (el logo de la Universidad) en la ventana de acceso al sistema con un tamaño prefijado ya que no habrá otros componentes multimedia en la aplicación. El tamaño máximo de páginas tratará de ser menor a 10KB.

5.4.5.2 Limitaciones de acceso a la tecnología

Si bien el acceso a la tecnología no es un tema de menor importancia, la aplicación fue pensada para funcionar en dispositivos celulares que soporten WAP 2.0, lo que requiere teléfonos de tecnología avanzada. Sin embargo debe considerarse que el auge de la telefonía celular en Argentina, sobre todo a partir de la aparición de GSM, provocó una venta masiva de teléfonos que soportan la tecnología requerida para la aplicación, con una considerable baja en los costos de estos dispositivos. En la actualidad, la telefonía celular tiene una penetración superior al 60% de la población, con un 70% de celulares con menos de 2 años de antigüedad. Estos nuevos dispositivos permiten acceso a mejores servicios, entre ellos, WAP 2.0.

También es de consideración el costo de un teléfono celular que representa aproximadamente un 15% del costo de adquisición de una computadora personal (el costo de un dispositivo con WAP 2.0 comienza en menos de \$200). Además, si bien el costo de navegación con el teléfono celular se factura por tiempo y por Bytes transmitidos, éste es similar o incluso inferior al costo de una conexión dial up domiciliaria. Por último, si se considera el pequeño tamaño que tendrán las páginas de la aplicación, el costo para utilizar alguna funcionalidad de la aplicación puede asemejarse al costo asociado que tendría un alumno que debe concurrir a un ciber para realizar la misma tarea por Internet.

5.4.5.3 Diferencias y adecuaciones de interfaz

Algunas funcionalidades a implementar, deben obligatoriamente adecuarse a las interfaces de los celulares, al punto de ser necesario en algunos casos, el desarrollo de nuevas funciones para dar soporte a esta interfaz.

Ejemplos claros son las pantallas con grandes volúmenes de datos, como por ejemplo la historia académica del alumno, que en su versión por Internet significa un listado que refleja todas las materias que el alumno ha cursado y ha rendido examen final inclusive con sus aplazos. Esta funcionalidad indefectiblemente no puede ser visualizada forma correcta un teléfono celular ya

que la cantidad de filas y columnas del resultado no tiene cabida en una interfaz de un dispositivo móvil.

Para listados de este tipo se prevé una estrategia diferente de visualización, que consiste en realizar una consulta sobre un ítem específico. En el caso particular de la historia académica del alumno, el ítem específico será la materia, que se buscará por su código. Esto representa la implementación de una nueva funcionalidad y ya no será "Historia Académica", sino "Materias Regulares", donde al encontrar coincidencia se mostrará si el alumno ha cursado esa materia y los finales rendidos sobre la misma.

Este tipo de adecuaciones debe aplicarse también al listado de inscripción a exámenes y cursadas, notas de evaluaciones parciales y cantidad de alumnos inscriptos a cursadas y examen.

5.4.6 Análisis funcional

5.4.6.1 Funcionalidad general

Iniciar sesión: Corresponde a la primer pantalla del sistema, que permite el acceso de docentes y alumnos a través del ingreso de un nombre de usuario y una contraseña.

Guaraní WAP!

Usuario:

Clave:

OK Volver

Pantalla de inicio (w_inicial.php)

Si la validación de usuario y contraseña es correcta, se invoca a la funcionalidad que retorna la lista de operaciones disponibles para dicho usuario.

Operaciones disponibles: esta operación es transparente para el usuario, ya que no debe invocarla explícitamente. Si la validación del usuario es exitosa, se invoca a esta funcionalidad que muestra el menú de operaciones disponibles según el perfil de la persona que ingresó al sistema. La lista de operaciones, se agrupa por categorías permitiendo una visualización más clara al usuario.

Seleccione una opción

Cursadas

Exámenes

Parciales

Certificados

OK Volver

Seleccione una opción

Agenda exámenes

Agenda cursadas

Inscr examen

Inscr cursadas

Actas abiertas

OK Volver

Operaciones para alumnos

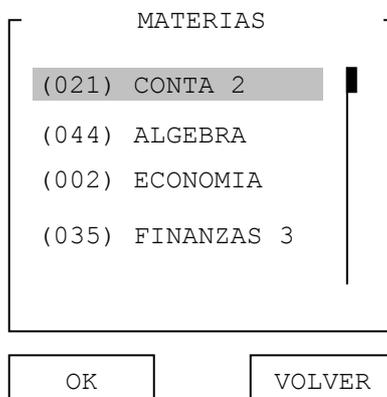
Operaciones para docentes

Pantalla de operaciones disponibles (w_operaciones.php)

Al ingresar en alguna de las opciones, se desplegará una lista con las operaciones disponibles de acuerdo al grupo seleccionado.

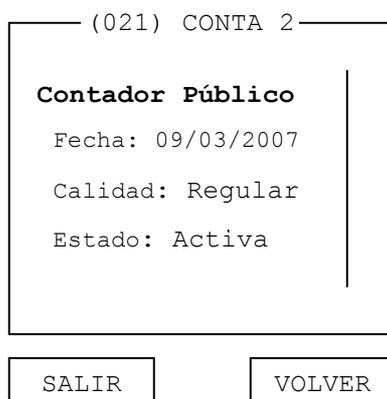
5.4.6.2 Funcionalidad para alumnos

Listado de inscripción a cursadas: esta operación se encuentra disponible al seleccionar la opción "Cursadas" en el menú principal del perfil alumno. El listado de inscripciones a cursadas debe mostrar al alumno todas las materias a las cuales se ha inscripto para cursar.



Pantalla de inscripciones a cursadas (w_inscripcion_cursadas.php)

Al seleccionar la materia aparece el resultado en pantalla. Los datos a mostrar serán: "Materia", "Carrera", "Fecha y hora", "Calidad" y "Estado" en versión reducida.



Pantalla de inscripción a cursada (w_inscripcion_cursada.php)

El botón "SALIR", vuelve a la página de operaciones del usuario.

Listado de inscripción a exámenes: esta operación se encuentra disponible al seleccionar la opción "Exámenes" en el menú principal del perfil alumno. El listado de inscripciones a examen debe mostrar al alumno todas las materias a las cuales se ha inscripto para rendir examen.

Seleccione una materia

(001) CONTA 1	█
(024) ANALISIS 2	
(015) DERECHO 1	

OK VOLVER

Pantalla de inscripciones a exámenes (w_inscripcion_examenes.php)

Al seleccionar la materia aparece el resultado en pantalla. Los datos a mostrar serán: "Materia", "Mesa", "Llamado", "Fecha y hora", "Tipo" y "Estado" en versión reducida.

(011) CONTA 1

Mesa 2	
Llamado 1	
29/03 14:30	
Libre	
Activa	

SALIR VOLVER

Pantalla de inscripción a examen (w_inscripcion_examen.php)

Se eliminarán las etiquetas y la fecha no incluirá año para reducir el tamaño de página.

Evaluaciones parciales: Está operación se encuentra disponible al seleccionar la opción "Parciales" del menú de alumnos. La funcionalidad de evaluaciones parciales requiere que el alumno seleccione la materia de un listado que contiene sus inscripciones a cursadas.

Luego de la selección de la materia se mostrará al alumno la lista de evaluaciones parciales disponibles para dicha materia. En caso que no existe conciencia, se retorna un mensaje indicando esta situación.

```
(058) SISTEMAS
PRIMER PARCIAL
RECUPERATORIO
SEGUNDO PARCIAL
OK          VOLVER
```

Pantalla de evaluaciones parciales (w_evaluaciones_parciales.php)

Nota de evaluaciones parciales: Una vez que selecciona una evaluación parcial, se mostrarán los siguientes datos: "Evaluación", "Fecha", "Nota", "Resultado", "Corregido por" y "Observaciones".

```
CONTA 2 PRIMER PARCIAL
15/02
DESAPROBADO (5)
Carlos Pérez:
Ej. 1 y 2 Mal
OK          VOLVER
```

Pantalla de nota parcial (w_nota_parcial.php)

Como en los casos precedentes, se eliminarán las etiquetas y la fecha no incluirá año para reducir el tamaño de página.

Certificados solicitados: Esta operación disponible en el menú "Certificados" permite visualizar todas las solicitudes que haya realizado a oficina de alumnos y que aún estén pendientes de resolución.

Certificados

Alumno Regular

29/03 Pendiente

OK VOLVER

Pantalla de certificados (w_certificados.php)

Mostrará cada certificado con una descripción de fecha de solicitud, estado y número de transacción.

5.4.6.3 Funcionalidad para docentes

Agenda de exámenes: Este reporte disponible a partir de la opción "Agenda exámenes" del perfil docente, mostrará al docente la información de las mesas de exámenes en las que participa con número de llamado, fecha, hora y aula del examen.

AGENDA

CONTA 1

Mesa/Llam 001-012

18/07/2007 10:00

Aula 23

OK VOLVER

Pantalla de agenda de exámenes (w_agenda_examenes.php)

Agenda de cursadas: Este reporte disponible a partir de la opción "Agenda cursadas" del perfil docente, mostrará al docente la información de las cursadas en las que participa con fecha, hora y aula y edificio de cursada.



AGENDA

CONTA 2

Lunes 10:00-12:00

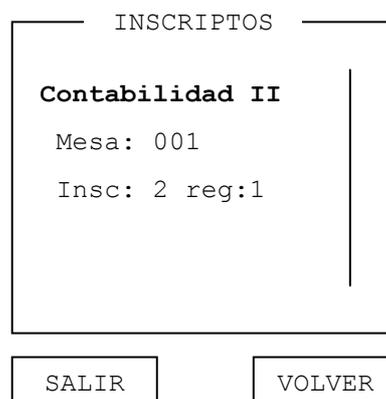
Aula 12

Edificio Anexo

OK VOLVER

Pantalla de agenda de cursadas (w_agenda_cursadas.php)

Cantidad de alumnos inscriptos en una mesa de examen: Esta operación disponible a partir de la opción "Inscriptos examen" del perfil docente, mostrará las mesas de examen disponibles y la información detallada de cantidad y condición.



INSCRIPTOS

Contabilidad II

Mesa: 001

Insc: 2 reg:1

SALIR VOLVER

Pantalla de inscriptos a examen (w_inscriptos_examen.php)

Como resultado, se visualiza un listado con cada mesa y la cantidad de inscriptos y regulares que se presentan a rendir el examen.

Cantidad de inscriptos a una comisión de cursada: Esta operación disponible a partir de la opción "CURSADAS" del perfil docente, muestra las comisiones de la materia y la cantidad de inscriptos.

INSCRIPTOS

Administración I

Com: C-487 (12)

Com: A-389 (42)

Pantalla de inscriptos a cursar (w_inscriptos_cursada.php)

Actas de examen sin cerrar: Este reporte disponible a partir de la opción "Actas abiertas" del perfil docente, indica cuales actas de examen en las que participo el docente, están aun sin cerrar. No requiere el ingreso de texto.

Actas abiertas

Contabilidad I:

5890 Abierta

Pantalla de actas de examen sin cerrar (w_actas_abiertas.php)

5.4.6.4 Consideraciones especiales

Búsqueda por materia: Las limitaciones de pantalla de WAP implican adecuaciones de interfaz para proveer la funcionalidad deseada. Para docentes y alumnos el principal escollo es el acceso a cualquier funcionalidad que requiere la búsqueda de materia. La cantidad de materias de un plan de estudios dificulta la selección en una interfaz WAP y el acceso por ingreso del nombre de materia es inviable debido a la longitud de las mismas y los problemas inherentes de sintaxis. Por último la búsqueda por código de materia, si bien es efectiva, implica conocimiento por parte de los usuarios de información adicional que no siempre se conoce o está disponible.

A partir de este análisis se determina la inclusión de un nombre abreviado o apodo de la materia, para realizar las búsquedas por tal criterio. El apodo de la materia debe corresponderse con el nombre por el cual el alumno y el docente reconocen a la materia. Ejemplos concretos serían los apodos CONTA 4, DERECHO 1, ANALISIS 2, que a veces son más conocidos que los verdaderos nombres de materias.

La pantalla de búsqueda deberá tener la siguiente estructura:

```

Búsqueda por materia
Código: [input]
Desconoce código?
Ingrese apodo: [input]
[BUSCAR] [VOLVER]

```

Pantalla de búsqueda de materia (w_busqueda_materia.php)

La búsqueda funcionará de la siguiente manera:

Si no ingresa código ni apodo, la búsqueda retornará un mensaje de error solicitando se ingrese alguna de las opciones. Si ingresa código de materia, se busca por dicho código, ignorando el apodo. Si, en cambio no ingresó código, pero ingresó apodo se realiza una búsqueda parcial con el apodo ingresado y se muestra en pantalla los resultados coincidentes. Por ejemplo la búsqueda por apodo CONTA debería retornar una lista con las siguientes probables posibilidades, incluyendo el código de materia en la respuesta para agilizar futuras búsquedas:

```

Seleccione materia
(011) CONTA 1
(021) CONTA 2
(031) CONTA 3
(041) CONTA 4
[OK] [VOLVER]

```

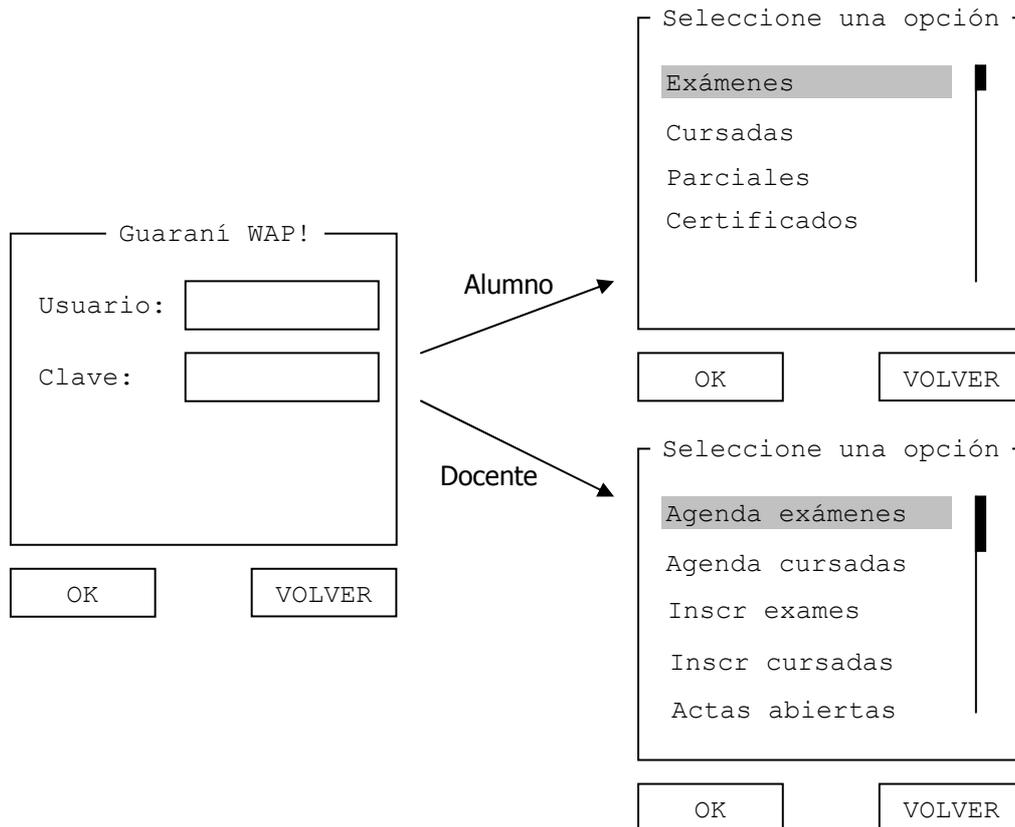
Pantalla de resultado de búsqueda de materia (w_seleccion_materia.php)

Al seleccionar la materia elegida, se continúa con la funcionalidad deseada.

5.4.7 Diseño de la aplicación

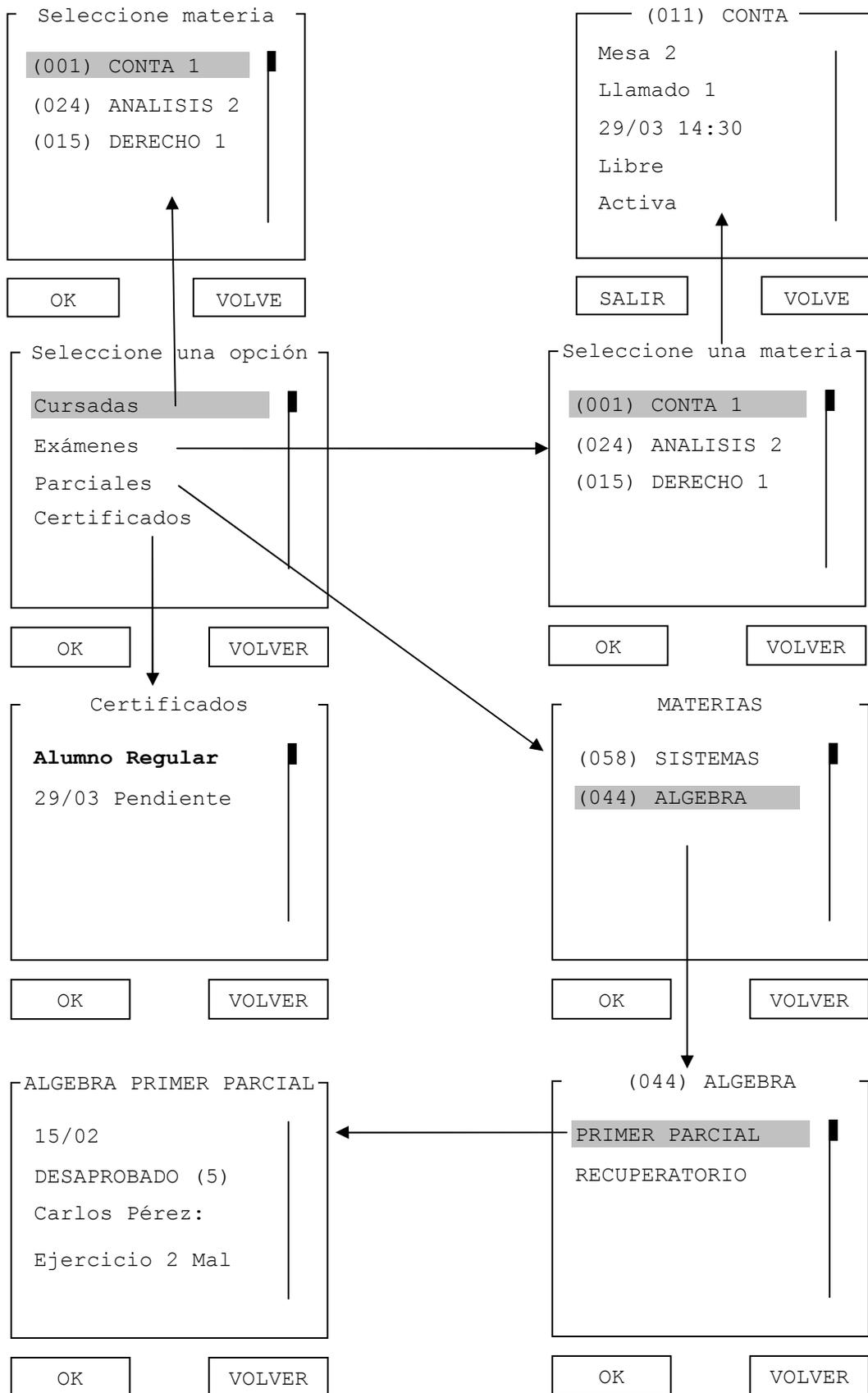
5.4.7.1 Estructura de navegación

La navegación por el sitio comienza a través de una ventana que solicita que se ingrese un nombre de usuario y una contraseña de acceso. De acuerdo al perfil de la persona que ingresa, se desplegará el menú específico de operaciones.



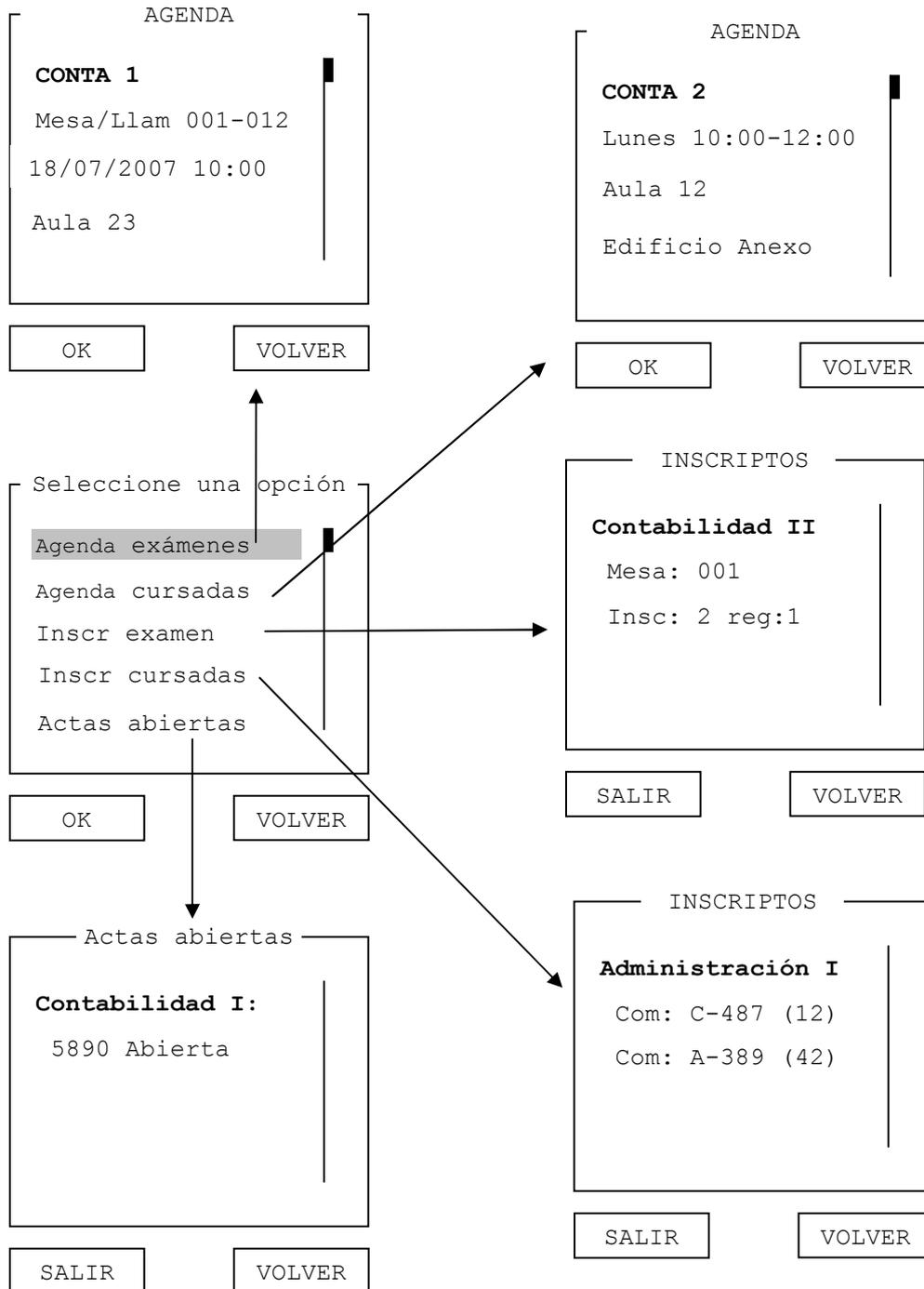
Inicio de sesión y listado de operaciones de acuerdo al perfil del usuario

Desde el perfil Alumno, puede accederse a exámenes, cursadas, parciales y certificados



Operaciones del perfil Alumno

Desde el perfil Docente, puede accederse a agendas, actas, exámenes y cursadas.



Operaciones del perfil Docente

5.4.7.2 Pantallas de la aplicación

Se definen las siguientes 16 pantallas a desarrollar en el prototipo:

1. Pantalla inicial (w_inicial.php): Página de inicio de la aplicación WAP. Muestra el formulario de ingreso al sitio.
2. Pantalla de operaciones disponibles (w_operaciones.php): Menú de operaciones disponibles según el perfil de la persona que ingresó al sistema.
3. Pantalla de búsqueda por materia (w_busqueda_materia.php): Página de búsqueda por código o apodo de materia.
4. Pantalla de resultado de búsqueda por materia (w_seleccion_materia.php): Página de selección del resultado de la búsqueda.
5. Pantalla de inscripciones a exámenes (w_inscripcion_examenes.php): Listado de materias a las cuales el alumno se ha inscripto a examen.
6. Pantalla de inscripción a examen (w_inscripcion_examen.php): Detalle de inscripción a examen.
7. Pantalla de inscripciones a cursadas (w_inscripcion_cursadas.php): Listado de materias a las cuales el alumno se ha inscripto a cursar.
8. Pantalla de inscripción a cursada (w_inscripcion_cursada.php): Detalle de inscripción a cursar.
9. Pantalla de evaluaciones parciales (w_evaluaciones_parciales.php): Listado de evaluaciones parciales de una materia.
10. Pantalla de nota parcial (w_nota_parcial.php): Detalle de evaluación parcial.
11. Pantalla de certificados (w_certificados.php): Listado de certificados pendientes de resolución.
12. Pantalla de agenda de exámenes del docente (w_agenda_examenes.php): Listado de exámenes del docente.
13. Pantalla de agenda de cursadas del docente (w_agenda_cursadas.php): Listado de cursadas del docente.
14. Pantalla de inscriptos a examen (w_inscriptos_examen.php): Listado de inscriptos por mesa de examen de una materia.
15. Pantalla de actas de examen sin cerrar (w_actas_abiertas.php): Listado de actas de examen abiertas.
16. Pantalla de inscriptos a cursar (w_inscriptos_cursada.php): Listado de comisiones con cantidad de inscriptos a cursar de una materia.

5.4.7.3 Tipo de páginas

Páginas de selección: Incluye a todas las páginas que solo contengan opciones para seleccionar. Estas páginas contienen listas de opciones, donde cada contiene un link hacia otra página. Se incluye en esta categoría a las siguientes pantallas:

- Pantalla de operaciones disponibles
- Pantalla de resultado de búsqueda por materia
- Pantalla de inscripciones a exámenes
- Pantalla de inscripciones a cursadas
- Pantalla de evaluaciones parciales

Listados: Incluye a todas las páginas que solo contengan resultados en forma de lista, pero sin posibilidad de selección para más detalle. Se incluye en esta categoría a las siguientes pantallas:

- Pantalla de certificados
- Pantalla de agenda de exámenes
- Pantalla de agenda de cursadas
- Pantalla de inscriptos a examen
- Pantalla de inscriptos a cursar
- Pantalla de actas de examen sin cerrar

Formularios: Incluye a todas las páginas que contengan campos para ingreso de texto, para ser enviados como parámetro a alguna funcionalidad de la aplicación. Se incluye en esta categoría a las siguientes pantallas:

- Pantalla inicial
- Pantalla de búsqueda por materia

Páginas de descripción: Incluye a todas las páginas que contengan información particular de algún ítem seleccionado o buscado previamente. Estas páginas no contienen links hacia ninguna otra. Se incluye en esta categoría a las siguientes pantallas:

- Pantalla de inscripción a examen
- Pantalla de inscripción a cursada
- Pantalla de nota parcial

5.4.7.4 Esquema de páginas

El esquema define la forma de visualización de cada tipo de página.

Páginas de selección

- Cada opción será especificada como un nuevo párrafo de texto.
- Cada opción contendrá un link hacia otra funcionalidad.
- El título de selección se incluirá como título de la página.

Listados

- Cada ítem será especificada como un nuevo párrafo de texto.
- Los ítems del listado serán diferenciados con resalto de texto en su descripción.
- El título del listado se incluirá como título de la página.

Formularios

- La pantalla contendrá solo los campos de ingreso de datos necesarios.
- Cada campo con su respectiva etiqueta serán especificados como un párrafo de texto.
- Las etiquetas de los campos estarán abreviadas.
- El título del formulario se incluirá como título de la página.

Páginas de descripción

- La página contendrá un único párrafo de texto.

- Cada valor a mostrar se separará por una nueva línea.
- Contendrá en lo posible, solo el valor a mostrar, sin etiquetas.
- El título se incluirá como título de la página.

Todas las pantallas contendrán la opción de volver al menú principal y de salir de la aplicación.

5.4.7.5 Etiquetas XHTML a utilizar

Se utilizarán las siguientes etiquetas:

- <HTML> </HTML>
- <HEAD> </HEAD>
- <TITLE> </TITLE>
- <BODY> </BODY>
- <P> </P>
-

-
- <A>
- <FORM> </FORM>
- <INPUT> </INPUT>

5.4.7.6 Estilo de páginas

El estilo de las páginas se definirá en un archivo de hoja de estilo separado (w_styles.css) y se definirán los estilos solo para las etiquetas definidas en el punto anterior.

5.4.8 Herramientas de desarrollo y prueba

Como herramienta de desarrollo, no se utilizará un IDE específico, sino solo editores de texto que remarquen sintaxis HTML, en particular EditPad, EditPlus y Dreamweaver.

Para prueba de la aplicación se utilizarán los emuladores Nokia Mobile Browser Simulator 4.0, Openwave V7 Simulator, el navegador Mozilla Firefox 2.0 y los dispositivos celulares Motorola (V235, U6 Pebl, V3 y V600), Nokia (6131 y 6020), Siemens CF76 y LG 200.

5.4.9 Desarrollo del prototipo

Se describe a continuación la configuración y el desarrollo de la funcionalidad requerida para el prototipo, la reutilización de código y procedimientos almacenados invocados para el desarrollo de la solución.

5.4.9.1 Configuración de la interfaz WAP, módulos, perfiles y operaciones

Como primer paso se definió una nueva interfaz al sistema Guaraní denominada Guaraní WAP, identificada con el código de interfaz nº 5. Las interfaces en el sistema Guaraní se definen en la tabla "par_interfaces".

Luego se definieron dos perfiles de usuario "WapAlu01" y "WapDoc01" para alumnos y docentes que accedan al sistema respectivamente y se relacionan con el perfil de usuario que se conecta a la base de datos.

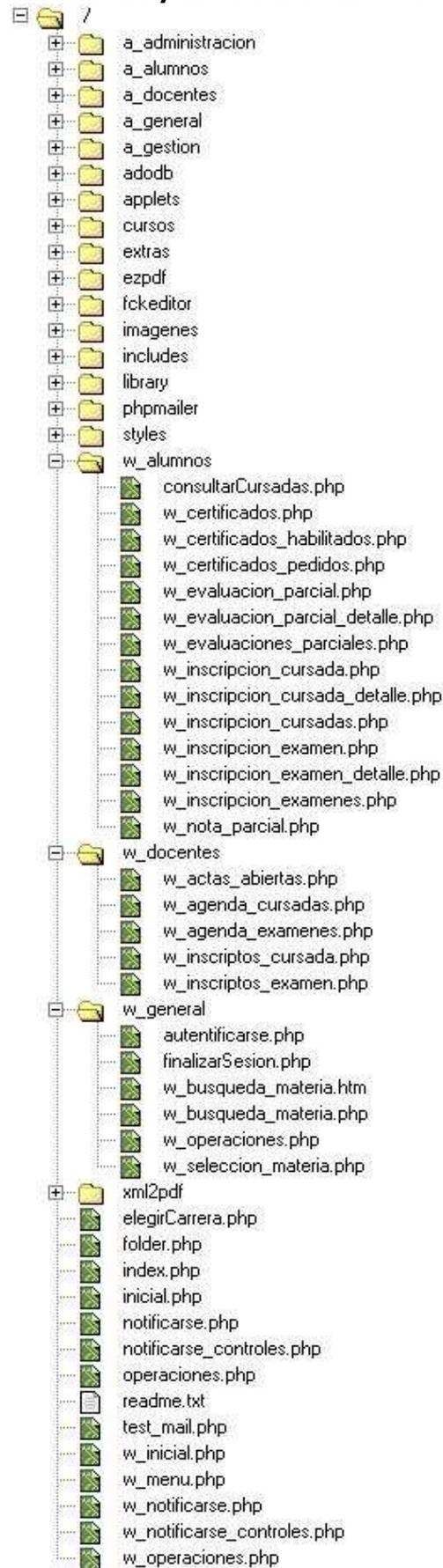
A continuación se ingresan los módulos incluidos en la interfaz WAP: "Exámenes", "Cursadas", "Certificados" y "Parciales" para alumnos y "Exámenes" y "Cursadas" para docentes.

Después se asociaron los módulos de la interfaz con los perfiles definidos previamente.

Por último, se crean las operaciones para la interfaz y se asocian con los módulos definidos anteriormente.

La configuración completa de Guaraní WAP se especifica en el Anexo B.

5.4.9.2 Estructura, nomenclatura y ubicación de directorios y archivos



Estructura de directorios y archivos

Nomenclatura de archivos y directorios: los archivos y directorios de Guaraní WAP comienzan con el prefijo "w_". Los archivos de Guaraní WAP se ubican dentro del directorio principal de Guaraní 3W y de los subdirectorios "w_alumnos", "w_docentes" y "w_general"

Dentro del directorio principal de Guaraní 3W (/) se ubican el archivo "w_inicial.php" para acceso al sistema por parte de alumnos y docentes, el archivo de operaciones por perfil de usuario "w_operaciones.php", el menú general que se incluye en todas las páginas "w_menu.php" y los archivos de notificaciones de errores "w_notificarse.php" y controles "w_notificarse_controles.php".

En el directorio "w_general" se ubicarán los archivos de funcionalidad general y compartida por todos los usuarios como "autenticarse.php" que chequea usuario y clave ingresado para verificar si es un usuario válido del sistema, "finalizarSession.php" para salir del sistema, y las páginas para búsquedas de materia.

En el directorio "w_alumnos" se encuentran los archivos para funcionalidad de alumnos y en el directorio "w_docentes" los archivos para funcionalidad de docentes en el sistema.

5.4.9.3 Implementación de la funcionalidad

Se describen a continuación el desarrollo de páginas PHP para la interfaz Guaraní WAP. El código fuente del prototipo se especifica en el Anexo III.

Ingreso a la aplicación (login): El ingreso a la aplicación se realiza desde la página inicial del sistema "/w_inicial.php". Esta página contiene un formulario para ingreso de usuario y clave que deriva hacia "/w_general/autenticarse.php". En caso de verificarse correctamente usuario y clave, se deriva el usuario hacia "/w_operaciones.php" para cargar las operaciones válidas para el perfil del usuario.

Egreso de la aplicación (logout): El egreso de la aplicación se realiza desde cualquiera de las páginas del sistema, ya que todas contienen un link hacia "/w_general/finalizarSesion.php". Esta página invalida la sesión actual del usuario y deriva a la página inicial del sistema.

Página de acceso a menú principal y salida: El menú general de acceso a la página de operaciones del usuario y a la salida del sistema, corresponde a una página PHP que se incluye en todas las demás páginas de la aplicación, excepto en la página inicial del sistema y la propia página de operaciones del usuario. El código contiene dos links XHTML que dirigen al usuario a la página inicial de opciones o hacia la salida del sistema respectivamente.

Menú de operaciones por perfil: El menú de operaciones por perfil se resuelve en la página "/w_operaciones.php" que se accede luego de ingresar al sistema o a través de la página de acceso al menú principal. Esta página invoca al procedimiento almacenado "sp_operaciones_net" que obtiene de la base de datos las operaciones habilitadas para el perfil del usuario.

Operación de consulta de inscripción a cursadas (para el perfil alumno): Esta operación se invoca desde el menú de operaciones del perfil alumno y corresponde a la página PHP `"/w_alumnos/w_inscripcion_cursadas.php"`. Esta página invoca al procedimiento `"sp_inscCursadas"` con los datos del alumno que ingresó al sistema y genera una lista con las materias en las que el alumno se ha inscripto para cursar. Cada materia del listado contiene un link hacia la página `"/w_alumnos/w_inscripcion_cursada_detalle.php"` para visualizar el detalle de los datos de inscripción en la cursada seleccionada.

Operación de consulta de inscripción a exámenes (para el perfil alumno): Esta operación se invoca desde el menú de operaciones del perfil alumno y corresponde a la página PHP `"/w_alumnos/w_inscripcion_examenes.php"`. Esta página invoca al procedimiento `"sp_inscExam"` con los datos del alumno que ingresó al sistema y genera una lista con las materias en las que el alumno se ha inscripto a examen. Cada materia del listado contiene un link hacia la página `"/w_alumnos/w_inscripcion_examen_detalle.php"` para visualizar el detalle de los datos de inscripción en el examen seleccionado.

Operación de consulta de evaluaciones parciales (para el perfil alumno): A esta operación se accede a través del menú de operaciones del perfil alumno y corresponde a la página PHP `"/w_alumnos/w_evaluaciones_parciales.php"`. Esta página invoca al procedimiento `"sp_inscCursadas"` con los datos del alumno que ingresó al sistema y genera una lista con las materias en las que el alumno se ha inscripto a cursar. Cada materia del listado contiene un link hacia la página `"/w_alumnos/w_evaluacion_parcial_detalle.php"` para visualizar las evaluaciones parciales de dicha materia. Una vez que se selecciona la materia, `"/w_alumnos/w_evaluacion_parcial_detalle.php"` invoca al procedimiento `"sp_evalparcalumno"` que retorna un listado con las evaluaciones parciales del alumno para la materia elegida. Cada evaluación del listado contiene un link hacia la página `"/w_alumnos/w_nota_parcial.php"` para visualizar el detalle de la evaluación.

Operación de consulta de certificados (para el perfil alumno): Esta operación se invoca desde el menú de operaciones del perfil alumno y corresponde a la página PHP `"/w_alumnos/w_certificados_pedidos.php"`. Esta página invoca al procedimiento `"sp_certif_pedidos"` con los datos del alumno que ingresó al sistema y genera una lista con los certificados solicitados por el alumno y el detalle de las solicitudes.

Operación de consulta de agenda de exámenes (para el perfil docente): A esta operación se accede a través del menú de operaciones del perfil docente y corresponde a la página PHP `"/w_docentes/w_agenda_examenes.php"`. Esta página invoca al procedimiento `"sp_agendaDocExa"` con los datos del docente que ingresó al sistema y genera una lista con las materias en las que el docente figura en mesa de examen con el detalle de los datos del llamado.

Operación de consulta de agenda de cursadas (para el perfil docente): Esta operación se invoca desde el menú de operaciones del perfil docente y corresponde a la página PHP `"/w_docentes/w_agenda_cursadas.php"`. Esta página invoca al procedimiento `"sp_agendaDocCur"` con los datos del docente que ingresó al sistema y genera una lista con los datos de las comisiones de materias en las que participa el docente.

Operación de consulta de inscriptos a exámenes (para el perfil docente): A esta operación se accede a través del menú de operaciones del perfil docente y corresponde a la página PHP "/w_docentes/w_inscriptos_examen.php". Esta página invoca al procedimiento "sp_mesasDoc" con los datos del docente que ingresó al sistema y genera una lista con las materias en las que el docente figura en mesa de examen junto con la cantidad de inscriptos regulares y libres.

Operación de consulta de inscriptos a cursadas (para el perfil docente): Esta operación se invoca desde el menú de operaciones del perfil docente y corresponde a la página PHP "/w_docentes/w_inscriptos_cursada.php". Esta página invoca al procedimiento "sp_ComisionesDoc" con los datos del docente que ingresó al sistema y genera una lista con los datos de las comisiones de materias en las que participa el docente y su cantidad de inscriptos.

Operación de consulta de inscriptos a cursadas (para el perfil docente): Esta operación se invoca desde el menú de operaciones del perfil docente y corresponde a la página PHP "/w_docentes/w_actas_abiertas.php". Esta página invoca al procedimiento "sp_busq_actaexa" con los datos del docente que ingresó al sistema y genera una lista con las actas de examen donde el docente es responsable y que aún figuran abiertas.

5.4.10 Configuración del servidor y la aplicación

5.4.10.1 Configuración de la aplicación

Para el correcto funcionamiento del sistema, debió modificarse el procedimiento almacenado "sp_paginas_net" que retorna los parámetros de una página determinada, incluyendo un nuevo parámetro al procedimiento, correspondiente a la interfaz. Además tuvo que modificarse la función "cargar_parametros()", disponible en la librería de funciones generales "/library/std_funcions.lib.php" para enviar el código de interfaz WAP al procedimiento.

5.4.10.2 Configuración del servidor

Alias de acceso a Guaraní 3W: En el archivo httpd.conf de Apache debe agregarse la siguiente línea para permitir el acceso a Guaraní3W y Guaraní WAP:

```
Alias guarani3w /<path-a-guarani3w>
```

Donde <path-a-guarani3w> es el directorio del servidor donde se encuentra la aplicación Guaraní3W.

Redirección de /wap hacia la página principal: Debe incluirse en la configuración de Apache un archivo con la siguiente configuración:

```
# server:/etc/apache/conf/wap.conf
<Location /wap>
    order allow,deny
    allow from all
    RewriteEngine on
    RewriteRule
        ^/(.*)$ http://<url-sitio>/<path-guarani3w>/w_inicial.php [R]
</Location>
```

Donde `<url-sitio>` es la URL del servidor donde se encuentra Guaraní 3w y `<path-guarani3w>` es el camino de acceso a la aplicación. De esta forma puede accederse al sitio WAP a través de la URL:

```
http://<url-sitio>/wap
```

Subdominio de wap para acceso directo a la página principal: También puede configurarse la aplicación para acceder como host virtual independizando la aplicación WAP de la aplicación Web. Para ello debe incluirse en la configuración de Apache un archivo con la siguiente configuración:

```
# server:/etc/apache/conf/wap.conf
NameVirtualHost *
<VirtualHost *>
    ServerName wap.<nombre-servidor>
    DocumentRoot <path-guarani3w>
    DirectoryIndex w_inicial.php
    <Location />
        order allow,deny
        allow from all
    </Location>
</VirtualHost>
```

Donde `<nombre-servidor>` es el nombre y dominio del servidor donde se encuentra Guaraní 3w (sin incluir "www") y `<path-guarani3w>` es el camino de acceso a la aplicación. De esta forma puede accederse al sitio WAP a través de la URL:

```
http://wap.<nombre-servidor>
```

Nota: estas especificaciones pueden cambiar de acuerdo a las configuraciones de servidores y servicios de cada universidad.

5.4.11 Puesta en marcha del prototipo

Una vez desarrollado la funcionalidad requerida, configurada la aplicación y el servidor, se procede a realizar las pruebas del mismo con diferentes dispositivos y emuladores, describiendo a continuación, resultado e inconvenientes de la aplicación.

5.4.11.1 Inconvenientes y soluciones

Redirección de páginas WAP: El primer inconveniente encontrado, fue el comportamiento diferente entre emuladores, navegadores y teléfonos celulares, en la redirección de páginas.

Guaraní 3W utilizaba para redireccionar páginas links de acceso relativo de la siguiente manera:

```
header("Location: ../operaciones.php");
```

Al utilizar la misma sentencia para redirigir las páginas WAP, funcionaba sin inconvenientes en los navegadores Web y en los emuladores, pero no así en algunos dispositivos celulares, retornando un error HTTP 404 (file not found), por no reconocer estos dispositivos el regreso al directorio anterior a través de "..".

La solución a este problema consistió en reemplazar la referencia al directorio anterior, por la especificación del camino completo, a través de variables de entorno que almacenan la URL y PATH de la aplicación:

```
$url = "http://" . $_SERVER["SERVER_NAME"] . $_SESSION["ses_sALIAS"];  
$url = $url . "w_operaciones.php";  
header("Location: $url");
```

Dificultad de ingreso de URL del sitio: Si no se ha configurado un subdominio para acceso directo a la aplicación, el ingreso de texto para acceder a la página inicial es una tarea tediosa y susceptible a errores.

Ingresar una URL del tipo:

```
http://www.universidadX.edu.ar/guarani3w/w_inicial.php
```

es una tarea compleja y molesta para el usuario final.

La solución consiste en configurar un subdominio de acceso directo a la aplicación como se explica en el punto 5.4.10.2, para acceder de la siguiente manera:

```
http://wap.universidadX.edu.ar
```

Una buena práctica será explicar a los usuarios la importancia de guardar el nombre del sitio entre sus accesos "Favoritos" en el teléfono celular, para no tener que escribir la URL del sitio cada vez que deseen ingresar al mismo.

Problemas de conectividad: Durante las pruebas de la aplicación, se sucedieron diversos problemas de conectividad por GPRS desde dispositivos celulares, por parte de dos diferentes prestadoras de servicios celulares.

Este problema escapa al dominio de la aplicación, pero debe considerarse que el acceso al sitio WAP puede estar limitado por un deficiente servicio de las operadoras celulares.

5.4.11.2 Ejecución del prototipo

A continuación se incluye la ejecución completa del prototipo desde el emulador Openwave V7 Emulator, que simula la ejecución desde un amplio conjunto de dispositivos celulares, con una base de datos de pruebas.

Ingreso a la aplicación por parte de un alumno



"w_inicial.php" > "w_operaciones.php"

Operación de consulta de inscripción a cursadas



"w_operaciones.php" > "w_inscripcion_cursadas.php" > "w_inscripcion_cursada_detalle.php"

Operación de consulta de inscripción a exámenes



"w_operaciones.php" > "w_inscripcion_examen.php" > "w_inscripcion_examen_detalle.php"

Operación de consulta de evaluaciones parciales



"w_operaciones.php" > "w_evaluaciones_parciales.php"



"w_evaluacion_parcial_detalle.php" > "w_nota_parcial.php"

Operación de consulta de certificados



"w_operaciones.php" > "w_certificados_pedidos.php"

Egreso de la aplicación



"w_operaciones.php" > "w_inicial.php"

Ingreso a la aplicación por parte de un docente



"w_inicial.php"

Acceso de un Docente

Menú de operaciones del perfil docente



"w_operaciones.php"

Operación de consulta de agenda de cursadas



"w_operaciones.php" > "w_agenda_cursadas.php"

Operación de consulta de agenda de exámenes



"w_operaciones.php" > "w_agenda_exámenes.php"

Operación de consulta de inscriptos a exámenes



"w_operaciones.php" > "w_inscriptos_examen.php"

Operación de consulta de inscriptos a cursadas



"w_operaciones.php" > "w_inscriptos_cursada.php"

Operación de consulta de actas abiertas



"w_operaciones.php" > "w_actas_abiertas.php"

Egreso de la aplicación



"w_operaciones.php" > "w_inicial.php"

5.5 Resultados – Guaraní WAP

Durante los últimos días del mes de junio de 2007, a través del trabajo conjunto del autor y del equipo de desarrollo del sistema SIU-Guaraní, el prototipo dio origen a una aplicación en producción: "Guaraní WAP", el módulo WAP del sistema SIU-Guaraní.

El módulo Guaraní WAP permite a alumnos y docentes acceder a funcionalidad del sistema SIU-Guaraní desde un teléfono celular. Los beneficios que persigue este desarrollo consisten en brindar movilidad, accediendo a la información en cualquier momento desde cualquier lugar con un costo muy bajo.

Desde el 29 de junio de 2007, Guaraní WAP está disponible para todas las universidades, en la versión 2.05.2 del sistema SIU-Guaraní, e incluye la posibilidad de que el alumno consulte sus inscripciones a cursadas y exámenes, vea el estado de sus solicitudes y acceda a los resultados de sus parciales. Además, los docentes pueden consultar la cantidad de inscriptos a sus cursadas y exámenes, sus actas pendientes y sus agendas de exámenes y cursadas.

Guaraní WAP fue presentado a las universidades el día 20 de abril de 2007 en el comité de usuarios del Sistema SIU-Guaraní realizado en el Palacio Pizzurno del Ministerio de Educación, Ciencia y Tecnología. Posteriormente, el 1º de Junio de 2007 durante la realización del taller de técnicos del sistema SIU-Guaraní desarrollado en la sede del Consejo Interuniversitario Nacional (CIN), se presentó la funcionalidad implementada en esta primera versión.

En ambas presentaciones, la aceptación del producto ha sido muy buena. Los participantes del comité de usuarios del sistema, comprendieron claramente los beneficios para alumnos, docentes y para ellos mismos, ya que éste módulo de autogestión ayuda a reducir el caudal de consultas y solicitudes a las oficinas o ventanillas de alumnos. Por su parte, los participantes del comité de técnicos entendieron que la implementación de Guaraní WAP es un servicio de valor agregado que puede brindar el área informática a la universidad.



Presentación de Guaraní WAP en el comité de usuarios, 20/04/2007 (gentileza SIU)

Al día de la finalización de esta tesis, viernes 3 de agosto de 2007, Guaraní WAP ha sido implementado en tres unidades académicas y está en condiciones de ser implementado en otras 14 universidades o institutos universitarios.

El SIU ha informado a través de su servicio de difusión la siguiente información el día 2 de julio de 2007:

"El SIU-Guaraní incorpora el uso de celulares"

El sistema de gestión de alumnos SIU-Guaraní ha habilitado el uso de celulares para consultas de trámites tanto para alumnos como para docentes.

De esta manera, una tecnología de uso masivo para la mayor parte de la población, como son los teléfonos celulares, se incorpora a la gestión académica mediante una serie de prestaciones que ya son habituales para los usuarios del SIU-Guaraní.

A partir de ahora, los alumnos de las universidades que trabajan con SIU-Guaraní pueden consultar desde su teléfono móvil sus notas de cursada, de exámenes finales, averiguar cuándo y dónde tienen clases, confirmar la inscripción a una materia, entre otras facilidades. También se incluyen prestaciones para docentes como por ejemplo conocer la agenda de clases y de cursadas, si tiene alumnos inscriptos para rendir exámenes, etc.

La incorporación de esta tecnología facilita el acceso a la información en cualquier momento desde cualquier lugar, sin necesidad de trasladarse y a un costo menor que el de una llamada de celular o similar al de una visita al cyber o conexión dial-up."

Por su parte, la Oficina Nacional de Tecnología de la Información (ONTI) se ha referido al proyecto en su boletín electrónico número 8, julio de 2007:

"Tramites Universitarios: Consulta por Celular.

El sistema de gestión de alumnos SIU-Guaraní ha habilitado el uso de celulares para consultas de trámites, tanto para alumnos como para docentes. De esta manera, una tecnología de uso masivo para la mayor parte de la población como los teléfonos celulares, se incorpora a la gestión académica mediante una serie de prestaciones que ya son habituales para los usuarios del SIU-Guaraní.

A partir de ahora, los alumnos podrán consultar desde su teléfono móvil sus notas de cursada, de exámenes finales, averiguar cuándo y dónde tienen clases, confirmar la inscripción a una materia, entre otros trámites. También se incluyen prestaciones para los docentes, como por ejemplo conocer la agenda de clases, la nómina de alumnos inscriptos para rendir exámenes, etc.

La incorporación de esta tecnología facilita el acceso a la información en cualquier momento y desde cualquier lugar, sin necesidad de traslado y a un costo menor que el de una llamada de celular o similar al de una visita a un cyber o conexión dial-up.

Cabe resaltar que el SIU-Guaraní forma parte de las soluciones informáticas que desarrolla el SIU como área integrante de la Secretaría de Políticas Universitarias del Ministerio de Educación, Ciencia y Tecnología de la Nación, para contribuir a la mejora de la gestión de las universidades del país."

También los medios periodísticos nacionales también han hecho referencia al proyecto. El diario La Nación en su sección Cultura, edición del jueves 26 de julio de 2007 ha suscrito:

"Universitarios podrán hacer trámites por celular

En 30 universidades del país los alumnos podrán consultar desde el teléfono móvil sus notas, cuándo y dónde cursan, confirmar la inscripción a materias, entre otras facilidades (Télam).-

Alumnos de 30 universidades del país pueden consultar desde su celular sus notas de cursada, averiguar cuándo y dónde tienen clases y confirmar la inscripción a una materia, entre otras facilidades que permite este novedoso sistema que suma una nueva tecnología masiva, como lo es el teléfono móvil, para la realización de trámites.

Los usuarios de este mecanismo de gestión, que se implementó recientemente, son los alumnos de facultades de al menos 30 casas de altos estudios que conforman parte del Sistema de Información Universitaria (SIU).

El SIU, dependiente de la Secretaría de Políticas Universitarias del Ministerio de Educación, es un sistema que permite la administración de todas las actividades académicas de los alumnos desde que ingresan como aspirantes hasta que obtienen el diploma.

El objetivo de este sistema es contribuir a mejorar la gestión de las instituciones, permitiéndoles contar con información segura, íntegra y disponible, optimizar sus recursos y lograr que el software sea aprovechado en toda su potencialidad.

Los sistemas desarrollados por el SIU toman los nombres de tribus indígenas de distintas regiones de la Argentina, como guaraní, tehuelche, puelche, araucano y wichi. "Estos nombres transmiten una noción de integridad entre comunidades diversas y a su vez remiten a una identidad común y la decisión de retomar estos nombres es una forma de tenerlas presentes en la conformación actual de nuestra sociedad", indicaron desde el SIU.

En tanto, los encargados del SIU-Guaraní decidieron incorporar a la gestión académica una tecnología de uso masivo para la mayor parte de la población, como son los teléfonos celulares.

El propósito fue que los estudiantes de algunas de las facultades o unidades académicas que integran esta red puedan consultar desde su teléfono móvil diversos tipos de información vinculados con su carrera.

De esta manera, los universitarios que, por diversas cuestiones no pueden acercarse hasta las sedes, entonces acceden por celular a información sobre sus notas de cursada, de exámenes finales, averiguar cuándo y dónde tienen clases, y confirmar la inscripción a una materia.

Este sistema también incluye prestaciones para los docentes como, por ejemplo, conocer la agenda de clases y de cursadas y si tiene alumnos inscriptos para rendir exámenes.

El acceso al sistema se realiza a través de un código de usuario y una clave con la misma identificación que se utiliza para consultar la información en internet desde una computadora.

La incorporación de esta tecnología facilita el acceso a la información en cualquier momento y lugar, sin necesidad de trasladarse y a un costo menor que el de una llamada de celular o similar al de una visita al cyber o conexión dial-up.

Entre las universidades nacionales que utilizan esta herramienta se encuentran la de Catamarca, Cuyo, Córdoba, Formosa, La Pampa, La Plata y Mar del Plata. También están la

Universidad Nacional de Misiones, Rosario, San Juan, San Luis, Tucumán, Patagonia San Juan Bosco, del Centro de la Provincia de Buenos Aires, Comahue, Litoral y la de Buenos Aires."

Página 12, sección Universidad, edición del viernes 27 de julio de 2007:

"Los alumnos de treinta universidades nacionales que tengan celular podrán acceder vía telefónica a información sobre su historial académico y realizar operaciones administrativas.

Por Julián Bruschtein

Cerca de 600 mil estudiantes de treinta universidades de todo el país podrán hacer un seguimiento de sus actividades académicas a través del teléfono celular (siempre y cuando tengan uno). La red de navegación de telefonía móvil fue integrada por la Secretaría de Políticas Universitarias al sistema informático de las universidad para que los alumnos puedan acceder a los datos sobre fecha y lugar de clases, confirmar inscripciones a materias o consultar las notas de sus exámenes y otros trámites. Entre las universidades nacionales que ya están en condiciones de implementar el servicio están las de Buenos Aires, del Comahue, Litoral, San Luis, Formosa, Misiones y Córdoba.

El SIU-Guaraní es un sistema informático que ya viene funcionando con el propósito de agilizar la gestión de las universidades. Por su intermedio, los estudiantes pueden obtener, a través de Internet, información sobre su historia académica. Ahora, según difundió ayer el Ministerio de Educación, también pueden hacerlo desde el teléfono celular. El sistema facilita el flujo de información entre las unidades académicas que lo utilizan y los alumnos, sin necesidad de que los universitarios se acerquen hasta las sedes donde cursan.

A distancia pueden consultar notas de exámenes finales, confirmar su inscripción a alguna materia o averiguar el cronograma de clases.

La coordinadora del proyecto, María Luján Gurmendi, explicó a Página/12 que su función era "brindar software de gestión a las entidades que lo requieran. El SIU-Guaraní registra toda la actividad académica que realicen los alumnos en las aulas, desde que ingresan al sistema universitario hasta que egresan". El objetivo primordial del sistema es impulsar "una mejora en la gestión institucional. Funciona con la misma seguridad con la que se pueden hacer movimientos bancarios electrónicamente", agregó Gurmendi, "por lo tanto, la información es privada y segura".

La masificación del uso de los teléfonos móviles entre los estudiantes decidió a los impulsores del proyecto a incorporarlo como instrumento de gestión. Sobre las 41 universidades existentes, por ahora son treinta las instituciones que formalmente utilizan la red, mientras que otras nueve se encuentran probando el software. Aunque depende de la organización de cada universidad –sea centralizada o descentralizada–, en total suman 208 las unidades académicas que están en condiciones de ofrecer el servicio WAP, la red de navegación para los celulares.

Para obtener el permiso de acceso al servicio "sólo hace falta pedir un usuario y generar la clave de acceso en la misma unidad académica en la que estudian, y si ya la tienen para la navegación en la web, no hace falta generar otra, porque es la misma", explicó la funcionaria. Los docentes también están incluidos para el uso de la aplicación.

Desde allí pueden verificar la agenda de las clases o la cantidad de alumnos que se inscribieron para rendir exámenes, con un sistema que "facilita en tiempo y forma el acceso a los datos", en cualquier momento, sin tener que acercarse al establecimiento, y a un costo "menor al de una llamada de celular o similar al de una visita al ciber o conexión dial-up", según consta en la página web del SIU (www.siu.edu.ar). A su vez el sistema guarda el recorrido administrativo-académico de cada uno de los alumnos, información que luego se cruza con otro programa utilizado para generar datos acerca del alumnado de cada institución, con el fin de elaborar informes estadísticos."

Por último, algunas de las universidades interesadas en el proyecto también han difundido los beneficios de Guaraní WAP.

Periódico Hoy La Universidad de la Universidad Nacional de Córdoba, viernes 27 de julio de 2007:

"Inscripciones por celular en la UNC

Pedir mesa para examen, anotarse en el cursado de materias y consultar notas, algunos de los trámites que se podrán hacer en un año.

Durante el año próximo, los alumnos que asistan a la Universidad Nacional de Córdoba (UNC) podrán inscribirse para rendir materias, consultar fechas de exámenes y conocer sus notas ingresando al campus virtual a través del browser instalado en los teléfonos celulares con tecnología digital. Así lo confirmó a Día a Día Miguel Montes, prosecretario de Informática de la Casa de Trejo.

El técnico de la UNC especificó que se trata de una "evolución" del sistema Guaraní, el soporte informático que administra el campus virtual para la autogestión de los alumnos universitarios. Ante este cambio, la Universidad deberá proveer en el sitio web <http://guarani3w.unc.edu.ar> de una visualización y herramientas que permitan el acceso desde los celulares que cuenten con navegador.

La aplicación de esta nueva herramienta se producirá tras la renovación del equipamiento informático. Montes estimó que el acceso móvil estaría habilitado en todas las unidades académicas para las inscripciones a exámenes de julio de 2008.

"Las facultades se irán incorporando a medida que vayamos mudando la tecnología, calculamos terminar ese proceso antes de los exámenes de julio de 2008", indicó Montes.

Sin embargo, Montes advirtió: "los trámites no se harán por mensajitos de textos".

En 30 universidades

Este desarrollo del Sistema Informático Universitario, realizado por la Secretaría de Políticas Universitarias de la Nación, permitirá a los alumnos de 30 universidades nacionales, entre ellas la de Córdoba, ingresar desde el teléfono celular al campus virtual que permite la administración de todas las actividades académicas de los alumnos activos.

Este sistema también incluye prestaciones para cada docente como, por ejemplo, conocer la agenda de clases y si tiene alumnos inscriptos para rendir exámenes.

Actualmente, el acceso al sistema se realiza únicamente por PC, utilizando un código de usuario y una clave, similares a los mecanismos de identificación utilizados en distintas páginas para acceso a información personal.

La aplicación browser facilitará el acceso a la información universitaria en cualquier momento a través de una tecnología masificada, como es el celular.

Entre las universidades nacionales que utilizan esta herramienta se encuentran, además de la UNC, la de Catamarca, Cuyo, Formosa, La Pampa, La Plata, Mar del Plata; Misiones, Rosario, San Juan, San Luis, Tucumán, Patagonia San Juan Bosco, del Centro de Buenos Aires, Comahue, Litoral y la de Buenos Aires.”

Por último, la Universidad Nacional del Noroeste de la Provincia de Buenos Aires en su boletín informativo del jueves 2 de agosto de 2007 ha suscrito:

"La UNNOBA cuenta con consultas por celular

Guaraní WAP, un módulo del Sistema de gestión de alumnos SIU-Guaraní, permite acceder a funcionalidad del sistema de gestión de alumnos, SIU-Guaraní, desde un teléfono celular. Guaraní WAP incluye la posibilidad de que el alumnos consulte sus inscripciones a cursadas y exámenes, pueda ver el estado de sus solicitudes y acceda a los resultados de sus parciales. Además, los docentes pueden consultar la cantidad de inscriptos a sus cursadas y exámenes, sus actas pendientes y sus agendas de exámenes y cursadas.

El SIU, entidad dependiente de la Secretaría de Políticas Universitarias del Ministerio de Educación, Ciencia y Tecnología, desarrolla soluciones informáticas y brinda servicios para el Sistema Universitario Nacional. Su objetivo es contribuir a mejorar la gestión de las instituciones, permitiéndoles contar con información segura, íntegra y disponible, optimizar sus recursos y lograr que el software sea aprovechado en toda su potencialidad.

El sistema SIU-Guaraní es un sistema de gestión de alumnos implementado en mas de 150 unidades académicas de 30 universidades de la Argentina y posee una interfaz Web, denominada Guaraní 3W que permite el acceso a través de Internet por parte de alumnos, docentes y autoridades.

Docentes y alumnos de la UNNOBA pueden acceder a información académica desde su celular ingresando al sitio wap.unnoba.edu.ar, utilizando los mismos datos de usuario y clave solicitados para Guaraní 3W.

Cabe recordar que Guaraní WAP es producto del trabajo conjunto del personal de la Dirección de Informática de la UNNOBA, encabezado por Diego de la Riva, quien estuvo a cargo del proyecto y del equipo de desarrollo de SIU.”

Estos testimonios muestran la importancia del desarrollo implementado y abren un camino para la continuidad del proyecto Guaraní WAP, en base a nuevos desarrollos propios de SIU y al aporte que pueden brindar todas las universidades interesadas en implementarlo.

Capítulo 6. Conclusiones

Como es sabido, tanto la telefonía como la transmisión inalámbrica tienen más de cien años de desarrollo, pero los mayores avances en cuanto a tecnología se han producido en las últimas tres décadas cuando se unieron estas tecnologías dando origen a la telefonía celular. Los avances en los últimos diez años de la telefonía celular han sido asombrosos, llegando al punto de dejar obsoletos dispositivos de menos de dos años de antigüedad en varias oportunidades.

Esto resalta aún más la brecha tecnológica existente entre los países del primer mundo encabezados por Japón, Estados Unidos y Europa y los países del tercer mundo como los latinoamericanos, africanos y gran parte de los asiáticos. Hoy día Latinoamérica y en particular la Argentina transita a pie de plomo la generación 2.5 de telefonía celular y los países del primer mundo disfrutan de los beneficios de la tercera generación y experimentan con la cuarta.

Los servicios de valor agregado como fotografía, video, mensajes, ringtones, juegos y aplicaciones han producido un boom en la venta de dispositivos en todo el mundo, que ha dirigido el negocio de desarrollo de aplicaciones para celulares en este sentido, relegando en parte la generación de productos de software para aplicaciones específicas.

GSM es el estándar predominante en Argentina y a nivel mundial, aunque existen otros tecnológicamente superiores pero que proveen menos servicios que este. Si bien WCDMA es el estándar más utilizado en la tercera generación, el dominio de GSM hace pensar que la cuarta generación estará centrada en UMTS.

Aún persiste dos problemas que limita en gran medida el desarrollo de aplicaciones específicas: la diversificación de sistemas operativos en los celulares y la presencia de diferentes tecnologías para el desarrollo. Estos dos factores deben tenerse en cuenta al momento de diseñar una aplicación para celulares.

Hoy día el desarrollo de aplicaciones para celulares consiste principalmente en adaptar aplicaciones existentes con cambios que permitan satisfacer los requerimientos de los dispositivos inalámbricos. En el caso de las aplicaciones residentes, Java ME es una plataforma reducida de Java SE para ejecución de aplicaciones adaptadas a dispositivos móviles. BREW es una plataforma específica para el desarrollo de aplicaciones para celulares. En ambos casos, las aplicaciones a ejecutar están acotadas a un número limitado de dispositivos compatibles, lo que hace poco viable el desarrollo de aplicaciones genéricas que ejecuten en variedades diversas de celulares. Además, el avance tecnológico de los dispositivos, como la mejora en procesadores y la inclusión de memoria RAM y disco rígido, está visualizando la posibilidad de que los celulares puedan ejecutar las mismas aplicaciones (y los mismos sistemas operativos) que las PC's. Llegado a este punto, las aplicaciones desarrolladas para Java ME o BREW pasarían a ser obsoletas, dejando lugar a la inclusión de una mayor variedad de aplicaciones en los teléfonos celulares.

Como ocurre en los desarrollos Web para PC's, las aplicaciones para celulares también tienden a implementarse en estructuras de varias capas, delegando al dispositivo solo la visualización de la capa de interfaz de usuario. Las características de los dispositivos actuales son beneficiosas

para aplicaciones de Internet, aunque en nuestro país siguen existiendo dificultades de conectividad en las redes celulares, acrecentadas a mayor distancia del conurbano bonaerense.

El surgimiento de WAP 2.0 ha simplificado de gran manera el desarrollo de aplicaciones Web o WAP para celulares, al punto de requerir una adaptación de interfaces de usuario al tipo de dispositivos que ejecutan la aplicación. Si bien los dispositivos cada día son más avanzados, las redes de comunicación mejoran, acercando la ejecución de programas y servicios pensados para computadoras, seguirá perdurando la limitante del tamaño del dispositivo, que implica una adaptación de interfaz de cualquier aplicación que quiera ejecutarse en un teléfono celular.

El prototipo implementado, justamente consiste en la creación de una nueva interfaz para un sistema existente, el cual funciona en la actualidad con una interfaz cliente/servidor y otra interfaz Web para ejecutar a través de Internet. Reutiliza la lógica de negocio y la programación de la aplicación madre e implementa la interfaz WAP para teléfonos celulares con soporte WAP 2.0.

Esta reutilización de funcionalidad, sumada a la simplicidad de configuración de aplicaciones y servidores para soporte WAP 2.0, vislumbra un camino a seguir en el futuro desarrollo de sistemas y servicios para teléfonos celulares.

Anexo A – Celulares con soporte WAP 2.0

Marca	Modelo	Marca	Modelo
Airam	TSM340	KDDI	A5509T
Alcatel	GlamPhone/ELLE-N1	KDDI	W21K
Alcatel	One Touch 535	KPT	SD-528
Alcatel	One Touch 556/557/565	Kyocera	KX1
Alcatel	One Touch 735i	Kyocera	KX2
Alcatel	One Touch 756/757	LG	A7110
Alcatel	One Touch 835	LG	A7150
Alcatel	One Touch C551	LG	B2000
Alcatel	One Touch C552	LG	C1100
Alcatel	One Touch C651	LG	C1200
Alcatel	One Touch S853	LG	C130
Amoi	A9B	LG	C1300
Amoi	CA6	LG	C1300i
Amoi	CS6	LG	C1500
Amoi	F8	LG	C2000
Amoi	V600	LG	C2200
Amoisonic	9201	LG	C3300
Amoisonic	F9	LG	C3310
Asus	J100	LG	C3320
Audiovox	CDM-8450	LG	CE500
Audiovox	CDM-8450SP	LG	CU8380
Audiovox	CDM-8900 (Telus network)	LG	CX5450
Audiovox	CDM-8910	LG	DG200
Audiovox	CDM-8912	LG	Easy Shot BX5450
Audiovox	CDM-8920	LG	F1200
Audiovox	CDM-8930	LG	F2100
Audiovox	CDM-8940	LG	F2300
Audiovox	CDM-9500	LG	F7200
Audiovox	CDM-9900	LG	F7250
Audiovox	PM-8920	LG	F9100
BenQ	A500	LG	G1100
BenQ	A5001	LG	G210
BenQ	Athena S830C	LG	G4011
BenQ	M300	LG	G4015
BenQ	M315	LG	G4020
BenQ	Morpheus	LG	G4050
BenQ	Nike1 S660C	LG	G5500
BenQ	S668C	LG	G5600
BenQ	S670C	LG	G6070
BenQ	S680C	LG	G7000
BenQ	S7001	LG	G7050
BenQ	S700C	LG	G7200
Capitel	C8188	LG	KG220
Capitel	F600	LG	KU730
Dai Telecom	G83	LG	L1100
Danger	hiptop	LG	L1150
Ericsson	T610	LG	L1200
Haier	P7	LG	L1400
HUAWEI	U626	LG	L1400i
KDDI	A1305SA	LG	L3100
KDDI	A1404S	LG	L5100

Marca	Modelo	Marca	Modelo
LG	LG4600	Motorola	A780
LG	LG500	Motorola	A820
LG	LG5400	Motorola	A830
LG	LG6100	Motorola	A835
LG	LX5350	Motorola	A845
LG	M4300	Motorola	A890
LG	MG300D	Motorola	A920
LG	MG610c	Motorola	A925
LG	MG800	Motorola	C155
LG	MM535	Motorola	C350
LG	OG200	Motorola	C350 US
LG	PM225	Motorola	C370
LG	PM325	Motorola	C375
LG	T5100	Motorola	C380
LG	U8110	Motorola	C381
LG	U8120	Motorola	C385
LG	U8130	Motorola	C390
LG	U8138	Motorola	C450
LG	U8150	Motorola	C510
LG	U8180	Motorola	C550
LG	U8290	Motorola	C650
LG	U8330	Motorola	C651
LG	U8360	Motorola	C975
LG	U8380	Motorola	C980
LG	U8500	Motorola	CN620
LG	U8550	Motorola	E1000
LG	U880	Motorola	E1070
LG	U890	Motorola	E365
LG	VI-125	Motorola	E375
LG	vi5225	Motorola	E380
LG	VX3100	Motorola	E390
LG	VX3200	Motorola	E398
LG	VX4500	Motorola	E550
LG	VX4600	Motorola	E610
LG	VX6100	Motorola	E680
LG	VX7000	Motorola	E680i
LG	VX8000	Motorola	E770-Vodafone
LG	VX8100	Motorola	E790
LG	W800	Motorola	E798
Lobster	485	Motorola	E815
Maxon	O2-X1i	Motorola	E815 (Verizon Wireless)
Microsoft	Mobile Explorer	Motorola	E895
Motorola	385	Motorola	i605
Motorola	A1000	Motorola	i740
Motorola	A630	Motorola	i850
Motorola	A668	Motorola	i860
Motorola	A680	Motorola	L6
Motorola	A728	Motorola	L7-Vodafone
Motorola	A732	Motorola	MPX220
Motorola	A760	Motorola	PEBL U6
Motorola	A768	Motorola	PEBL V6

Marca	Modelo	Marca	Modelo
Motorola	ROKR E1	Motorola	V551
Motorola	ROKR E2	Motorola	V555
Motorola	SLVR L7	Motorola	V557
Motorola	SLVR V8	Motorola	V560
Motorola	T720	Motorola	V600
Motorola	T720g	Motorola	V620
Motorola	T720i	Motorola	V635
Motorola	T725E	Motorola	v710
Motorola	V1050	Motorola	V710 (Verizon Wireless)
Motorola	V1075	Motorola	V80
Motorola	V150	Motorola	V810
Motorola	V170	Motorola	V860
Motorola	V171	Motorola	V870
Motorola	V172	Motorola	V880
Motorola	V173	Motorola	V975
Motorola	V180	Motorola	V980
Motorola	V185	Motorola	V980 (Vodafone)
Motorola	V186	Motorola	W220
Motorola	V188	NEC	530
Motorola	V190	NEC	A232
Motorola	V220	NEC	A525
Motorola	V235	NEC	c313
Motorola	V265	NEC	c338
Motorola	V280	NEC	c616
Motorola	V290	NEC	e101
Motorola	V3	NEC	e228
Motorola	V3 BLK	NEC	e232
Motorola	V3-Vodafone	NEC	e238
Motorola	V300	NEC	e242
Motorola	V303	NEC	e313
Motorola	V330	NEC	e338
Motorola	V360	NEC	e540
Motorola	V360-Vodafone	NEC	e606
Motorola	V360i	NEC	e616
Motorola	V3c	NEC	e808
Motorola	V3i	NEC	N110
Motorola	V3r	NEC	N411i
Motorola	V3x	NEC	N535
Motorola	V3x-Vodafone	NEC	N700
Motorola	V400	NEC	N710
Motorola	V500	NEC	N820
Motorola	V501	NEC	N840
Motorola	V505	NEC	N900
Motorola	V525	Newgen	C620
Motorola	V525M	Newgen	E1200
Motorola	V535	Nokia	2650
Motorola	V536	Nokia	2652
Motorola	V540	Nokia	3100
Motorola	V545	Nokia	3100 US
Motorola	V547	Nokia	3105
Motorola	V550	Nokia	3108

Marca	Modelo	Marca	Modelo
Nokia	3120	Nokia	6233
Nokia	3125	Nokia	6235
Nokia	3152	Nokia	6235i
Nokia	3155	Nokia	6255
Nokia	3155i	Nokia	6255i
Nokia	3200	Nokia	6260
Nokia	3205	Nokia	6265
Nokia	3205 SPR	Nokia	6265i
Nokia	3205i	Nokia	6270
Nokia	3220	Nokia	6280
Nokia	3230	Nokia	6585 (US)
Nokia	3250	Nokia	6590
Nokia	3300	Nokia	6600
Nokia	3300 US	Nokia	6610i
Nokia	3410 CDMA	Nokia	6620
Nokia	3590	Nokia	6630
Nokia	3595	Nokia	6638
Nokia	3600	Nokia	6650
Nokia	3620	Nokia	6651
Nokia	3630	Nokia	6670
Nokia	3650	Nokia	6680
Nokia	3650 UP.Browser	Nokia	6681
Nokia	3660	Nokia	6682
Nokia	5100 US	Nokia	6800 US
Nokia	5140	Nokia	6810
Nokia	5140i	Nokia	6820
Nokia	6010	Nokia	6822
Nokia	6020	Nokia	7200
Nokia	6021	Nokia	7250i
Nokia	6030	Nokia	7260
Nokia	6030 US	Nokia	7270
Nokia	6060	Nokia	7280
Nokia	6061	Nokia	7360
Nokia	6070	Nokia	7370
Nokia	6101	Nokia	7380
Nokia	6102	Nokia	7600
Nokia	6103	Nokia	7610
Nokia	6111	Nokia	7700
Nokia	6131	Nokia	7710
Nokia	6136	Nokia	8310
Nokia	6152	Nokia	8390
Nokia	6155	Nokia	8800
Nokia	6155i	Nokia	8801
Nokia	6170	Nokia	9110
Nokia	6200	Nokia	9210
Nokia	6220	Nokia	9210c
Nokia	6225	Nokia	9210i
Nokia	6225 (US)	Nokia	9290
Nokia	6225i	Nokia	9300
Nokia	6230	Nokia	9500
Nokia	6230i	Nokia	E60

Marca	Modelo	Marca	Modelo
Nokia	E61	Research In Motion Ltd.	BlackBerry 5820
Nokia	E70	Research In Motion Ltd.	BlackBerry 6210
Nokia	N-Gage	Research In Motion Ltd.	BlackBerry 6220
Nokia	N-Gage QD	Research In Motion Ltd.	BlackBerry 6230
Nokia	N70	Research In Motion Ltd.	BlackBerry 6280
Nokia	N71	Research In Motion Ltd.	BlackBerry 6510
Nokia	N80	Research In Motion Ltd.	BlackBerry 6710
Nokia	N90	Research In Motion Ltd.	BlackBerry 6720
Nokia	N91	Research In Motion Ltd.	BlackBerry 6750
Nokia	N92	Research In Motion Ltd.	BlackBerry 7100
Nokia	Opera for 6600	Research In Motion Ltd.	BlackBerry 7100 v4
Nokia	Opera for 6670	Research In Motion Ltd.	BlackBerry 7210
Nokia	Opera for 6680	Research In Motion Ltd.	BlackBerry 7220
Nokia	Opera for 7610	Research In Motion Ltd.	BlackBerry 7230
Nokia	Opera for N-Gage	Research In Motion Ltd.	BlackBerry 7230 v4
Nokia	THR880i	Research In Motion Ltd.	BlackBerry 7250
O2	X1b	Research In Motion Ltd.	BlackBerry 7280
O2	X2i	Research In Motion Ltd.	BlackBerry 7280 v4
O2	X4	Research In Motion Ltd.	BlackBerry 7290
Openwave	OPWV-SDK/60	Research In Motion Ltd.	BlackBerry 7290 v4
Openwave	OPWV-SDK/61	Research In Motion Ltd.	BlackBerry 7510
Openwave	OPWV-SDK/62	Research In Motion Ltd.	BlackBerry 7510 v4
Openwave	OPWV-SDK/7.0.2.3	Research In Motion Ltd.	BlackBerry 7520
Openwave	OPWV-SDK/70	Research In Motion Ltd.	BlackBerry 7730
Opera	Opera for Series 60	Research In Motion Ltd.	BlackBerry 7750
Opera	Symbian Client	Research In Motion Ltd.	BlackBerry 7780
Palm	M130	Research In Motion Ltd.	BlackBerry 8700
Palm	M130B	Research In Motion Ltd.	BlackBerry Charm
Panasonic	A200	Sagem	CO210
Panasonic	A210	Sagem	my300X
Panasonic	G50	Sagem	my301X
Panasonic	G60	Sagem	my400X
Panasonic	G70	Sagem	my401X
Panasonic	GD87	Sagem	my700X
Panasonic	SA6	Sagem	myC-3
Panasonic	SA7	Sagem	myC-3b
Panasonic	X200	Sagem	myC-3m
Panasonic	X200P	Sagem	myC-4
Panasonic	X300	Sagem	myC3-2
Panasonic	X400	Sagem	myC4-2
Panasonic	X400P	Sagem	myC5-2
Panasonic	x410	Sagem	myC5-2 Vodafone
Panasonic	X500	Sagem	myC5-2T
Panasonic	X60	Sagem	myMobileTV
Panasonic	X66	Sagem	myV-55
Panasonic	x68	Sagem	myV-56
Panasonic	X700	Sagem	myV-65
Panasonic	X701	Sagem	MYV-66
Philips	568	Sagem	myV-75
QCI	OPWV 6	Sagem	myV-76
Research In Motion Ltd.	BlackBerry 5810	Sagem	myX-2G

Marca	Modelo	Marca	Modelo
Sagem	myX-4	Samsung	SGH-E630C
Sagem	myX-4T	Samsung	SGH-E635
Sagem	myX-5-2T	Samsung	SGH-E638
Sagem	myX-5d	Samsung	SGH-E640
Sagem	myX-5e	Samsung	SGH-E700
Sagem	myX-5m	Samsung	SGH-E700A
Sagem	myX-6	Samsung	SGH-E708
Sagem	myX-6-2	Samsung	SGH-E710
Sagem	myX-7	Samsung	SGH-E710i
Sagem	myX-8	Samsung	SGH-E715
Sagem	myX5-2	Samsung	SGH-E720
Sagem	myX6-2	Samsung	SGH-E720C
Sagem	myZ-5	Samsung	SGH-E730
Sagem	myZ-55	Samsung	SGH-E738
Sagem	SG345i	Samsung	SGH-E750
Samsung	D820	Samsung	SGH-E760
Samsung	E320	Samsung	SGH-E770
Samsung	MM-A700	Samsung	SGH-E800
Samsung	SCH-A600	Samsung	SGH-E800C
Samsung	SCH-A650	Samsung	SGH-E808
Samsung	SCH-A670	Samsung	SGH-E820
Samsung	SCH-W109	Samsung	SGH-E860V
Samsung	SCH-X700	Samsung	SGH-E880
Samsung	SGH-C108	Samsung	SGH-P207
Samsung	SGH-C110	Samsung	SGH-P510
Samsung	SGH-D500	Samsung	SGH-P518
Samsung	SGH-D500C	Samsung	SGH-P716
Samsung	SGH-D500E	Samsung	SGH-P730
Samsung	SGH-D508	Samsung	SGH-P730C
Samsung	SGH-D600	Samsung	SGH-P735
Samsung	SGH-D608	Samsung	SGH-P738
Samsung	SGH-E100	Samsung	SGH-P777
Samsung	SGH-E108	Samsung	SGH-T309
Samsung	SGH-E310	Samsung	SGH-T809
Samsung	SGH-E315	Samsung	SGH-X100
Samsung	SGH-E316	Samsung	SGH-X108
Samsung	SGH-E317	Samsung	SGH-X120
Samsung	SGH-E318	Samsung	SGH-X426
Samsung	SGH-E330	Samsung	SGH-X427
Samsung	SGH-E330C	Samsung	SGH-X427m
Samsung	SGH-E330N	Samsung	SGH-X460
Samsung	SGH-E335	Samsung	SGH-X460C
Samsung	SGH-E338	Samsung	SGH-X468
Samsung	SGH-E340	Samsung	SGH-X480
Samsung	SGH-E350	Samsung	SGH-X480C
Samsung	SGH-E350E	Samsung	SGH-X495
Samsung	SGH-E350V	Samsung	SGH-X497
Samsung	SGH-E360	Samsung	SGH-X600
Samsung	SGH-E610	Samsung	SGH-X600A
Samsung	SGH-E618	Samsung	SGH-X608
Samsung	SGH-E630	Samsung	SGH-X610

Marca	Modelo	Marca	Modelo
Samsung	SGH-X620	Sendo	P600
Samsung	SGH-X640	Sendo	S600
Samsung	SGH-X660	Sendo	X
Samsung	SGH-X660-Vodafone	Sharp	550SH
Samsung	SGH-X680-Vodafone	Sharp	703SH
Samsung	SGH-X910	Sharp	770SH
Samsung	SGH-Z105U	Sharp	802SH
Samsung	SGH-Z140	Sharp	902SH-Vodafone
Samsung	SGH-Z300	Sharp	903SH
Samsung	SGH-Z308	Sharp	SX813
Samsung	SGH-Z500	Sharp	TM-100
Samsung	SGH-Z510	Sharp	TM-150
Samsung	SGH-Z520	Sharp	TM-200
Samsung	SGH-Z540	Sharp	TQ-GX-E30
Samsung	SGH-ZV10	Sharp	TQ-GX-i98
Samsung	SGH-ZV50	Sharp	TQ-GX1
Samsung	SPH-A660	Sharp	TQ-GX10
Samsung	SPH-A680	Sharp	TQ-GX10i
Samsung	SPH-A740	Sharp	TQ-GX10m
Samsung	SPH-A760	Sharp	TQ-GX10N
Samsung	SPH-A790	Sharp	TQ-GX13
Samsung	SPH-A800	Sharp	TQ-GX15
Samsung	SPH-A920	Sharp	TQ-GX17
Samsung	Z100	Sharp	TQ-GX20
Samsung	Z105	Sharp	TQ-GX20i
Samsung	Z107	Sharp	TQ-GX20N
Samsung	Z110	Sharp	TQ-GX21
Samsung	Z130	Sharp	TQ-GX22
Samsung	Z150	Sharp	TQ-GX22S
Samsung	Z310	Sharp	TQ-GX23
Sanyo	MM-5600	Sharp	TQ-GX25
Sanyo	MM-7400	Sharp	TQ-GX27
Sanyo	MM9000	Sharp	TQ-GX29
Sanyo	PLS4920	Sharp	TQ-GX30
Sanyo	PLS7400	Sharp	TQ-GX30i
Sanyo	PLS8100	Sharp	TQ-GX31
Sanyo	PLS8200	Sharp	TQ-GX32
Sanyo	PM-8200	Sharp	TQ-GZ100
Sanyo	S750	Sharp	TQ-GZ100S
Sanyo	SCP-2300	Sharp	TQ-GZ100T
Sanyo	SCP-4900	Sharp	TQ-GZ200
Sanyo	SCP-4920	Sharp	TQ-GZ200S
Sanyo	SCP-5300	Sharp	TQ-V750
Sanyo	SCP-5400	Sharp	TQ-V802SH
Sanyo	SCP-5500	Sharp	WX-T71
Sanyo	SCP-6400	Sharp	WX-T91
Sanyo	SCP-7200	Siemens	A60
Sanyo	SCP-7300	Siemens	A62
Sanyo	SCP-8100	Siemens	A65
Sanyo	SCP-8200	Siemens	A65-China
Sanyo	SCP-8300	Siemens	A75

Marca	Modelo	Marca	Modelo
Siemens	AX72	Siemens	S68
Siemens	AX75	Siemens	S6C
Siemens	C60	Siemens	S75
Siemens	C61	Siemens	S81
Siemens	C62	Siemens	S88
Siemens	C65	Siemens	SG75
Siemens	C65-China	Siemens	SK65
Siemens	C65-Vodafone	Siemens	SL55
Siemens	C66-NAFTA	Siemens	SL56
Siemens	C70	Siemens	SL5C
Siemens	C72	Siemens	SL5E
Siemens	C75	Siemens	SL65
Siemens	CF62	Siemens	SL65-China
Siemens	CF62T	Siemens	SL75
Siemens	CF6C	Siemens	SP65
Siemens	CF75	Siemens	ST55
Siemens	CFX65	Siemens	ST60
Siemens	CL75	Siemens	SX1
Siemens	CT65	Siemens	SX1 UP.Browser
Siemens	CT66	Siemens	SXG75
Siemens	CV65	Siemens	U15
Siemens	CX65	Siemens	Z60
Siemens	CX65-China	Sony	PEG-TH55
Siemens	CX65-Vodafone	Sony	PEG-TJ37
Siemens	CX70	Sony	PEG-UX50
Siemens	CX70-Vodafone	SonyEricsson	D750i
Siemens	CX75	SonyEricsson	F500i
Siemens	CXT65	SonyEricsson	J100
Siemens	CXT70	SonyEricsson	J100a
Siemens	EF51	SonyEricsson	J100c
Siemens	EF81	SonyEricsson	J100i
Siemens	EF91	SonyEricsson	J200
Siemens	EL71	SonyEricsson	J200c
Siemens	M55	SonyEricsson	J200i
Siemens	M55C	SonyEricsson	J210
Siemens	M56	SonyEricsson	J210c
Siemens	M65	SonyEricsson	J210i
Siemens	M65-Vodafone	SonyEricsson	J220
Siemens	M65C	SonyEricsson	J220a
Siemens	M65i	SonyEricsson	J220c
Siemens	M75	SonyEricsson	J220i
Siemens	MC60	SonyEricsson	J230
Siemens	MC6C	SonyEricsson	J230a
Siemens	MT65	SonyEricsson	J230c
Siemens	S55	SonyEricsson	J230i
Siemens	S55i	SonyEricsson	J300
Siemens	S56	SonyEricsson	J300a
Siemens	S57	SonyEricsson	J300c
Siemens	S65	SonyEricsson	J300i
Siemens	S65-Vodafone	SonyEricsson	K300a
Siemens	S66	SonyEricsson	K300c

Marca	Modelo	Marca	Modelo
SonyEricsson	K300i	SonyEricsson	S700c
SonyEricsson	K310	SonyEricsson	S700i
SonyEricsson	K310a	SonyEricsson	S710
SonyEricsson	K310c	SonyEricsson	S710a
SonyEricsson	K310i	SonyEricsson	T226
SonyEricsson	K500c	SonyEricsson	T230
SonyEricsson	K500i	SonyEricsson	T237
SonyEricsson	K506c	SonyEricsson	T290
SonyEricsson	K508	SonyEricsson	T290a
SonyEricsson	K508c	SonyEricsson	T290c
SonyEricsson	K508i	SonyEricsson	T290i
SonyEricsson	K510	SonyEricsson	T300
SonyEricsson	K510a	SonyEricsson	T302
SonyEricsson	K510c	SonyEricsson	T306
SonyEricsson	K510i	SonyEricsson	T310
SonyEricsson	K600	SonyEricsson	T312
SonyEricsson	K600i	SonyEricsson	T316
SonyEricsson	K608	SonyEricsson	T608
SonyEricsson	K608i	SonyEricsson	T610
SonyEricsson	K610	SonyEricsson	T612
SonyEricsson	K610c	SonyEricsson	T616
SonyEricsson	K610i	SonyEricsson	T618
SonyEricsson	K700	SonyEricsson	T620
SonyEricsson	K700c	SonyEricsson	T628
SonyEricsson	K700i	SonyEricsson	T630
SonyEricsson	K750	SonyEricsson	T637
SonyEricsson	K750c	SonyEricsson	T687i
SonyEricsson	K750i	SonyEricsson	T68i
SonyEricsson	K758c	SonyEricsson	V600i
SonyEricsson	K790	SonyEricsson	V800
SonyEricsson	K790a	SonyEricsson	V802
SonyEricsson	K790c	SonyEricsson	W300
SonyEricsson	K790i	SonyEricsson	W300c
SonyEricsson	K800	SonyEricsson	W300i
SonyEricsson	K800c	SonyEricsson	W550i
SonyEricsson	K800i	SonyEricsson	W600i
SonyEricsson	M600	SonyEricsson	W800
SonyEricsson	M600c	SonyEricsson	W800c
SonyEricsson	M600i	SonyEricsson	W800i
SonyEricsson	P800	SonyEricsson	W810
SonyEricsson	P802	SonyEricsson	W810c
SonyEricsson	P900	SonyEricsson	W810i
SonyEricsson	P908	SonyEricsson	W850
SonyEricsson	P910	SonyEricsson	W850i
SonyEricsson	P910a	SonyEricsson	W900i
SonyEricsson	P910c	SonyEricsson	W950
SonyEricsson	P910i	SonyEricsson	W950c
SonyEricsson	P990	SonyEricsson	W950i
SonyEricsson	P990c	SonyEricsson	Z1010
SonyEricsson	P990i	SonyEricsson	Z500
SonyEricsson	S700	SonyEricsson	Z500a

Marca	Modelo
SonyEricsson	Z520
SonyEricsson	Z520a
SonyEricsson	Z520c
SonyEricsson	Z520i
SonyEricsson	Z530
SonyEricsson	Z530c
SonyEricsson	Z530i
SonyEricsson	Z600
SonyEricsson	Z608
SonyEricsson	Z800
SonyEricsson	Z800c
SonyEricsson	Z800i
Telit	G82
Telit	G83
Toshiba	803
Toshiba	CDM-9950
Toshiba	TS 921
Vitelcom	TSM-5m
Vitelcom	TSM-5mt
Voxtel	BD40
Voxtel	RX100
Voxtel	RX200
Voxtel	V100
Voxtel	V300
Voxtel	V310
Voxtel	V50
Yospace Nokia	3650
Yospace Sharp	TQ-GX10
Yospace SonyEricsson	T610

Anexo B – Configuración de Guaraní WAP

Se incluye solo el texto relevante para WAP de los archivos de configuración.

Configuración de Interfaz WAP

Archivo: interfaz_5_Wap.dca

```
INSERT INTO par_interfaces (interfaz, nombre, descripcion) VALUES
(5, 'Wap', 'Interface wap - celulares');
```

Configuración de Perfiles de usuario

Archivo: Perfil_WapAlu01.dca

```
INSERT INTO acc_perfiles (perfil, nombre, bitmap) VALUES
('WapAlu01', 'Wap - Alumno', NULL);
```

Archivo: perfil_WapDoc01.dca

```
INSERT INTO acc_perfiles (perfil, nombre, bitmap) VALUES
('WapDoc01', 'Wap - Docente', NULL);
```

Archivo: Wap_Perfil_db_perfil_tipo_usuario.dca

```
-- Perfil del Alumno
INSERT INTO acc_perfilinterfaz (perfil_usuario_db, tipo_usuario,
interfaz, perfil) VALUES ('Alumno', 'ALU', 5, 'WapAlu01') ;
-- Perfil del Docente
INSERT INTO acc_perfilinterfaz (perfil_usuario_db, tipo_usuario,
interfaz, perfil) VALUES ('Alumno', 'DOC', 5, 'WapDoc01') ;
```

Configuración de módulos y submódulos

Archivo: WAP_modulos.dca

```
-- Módulos Genéricos
-- Modulo de inicio de sesión para Docentes y Alumnos
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES
('Wap00001', 'Mi sesión', NULL);
-- Modulos para el Alumno
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES ('WapAlu01',
'Cursadas', NULL);
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES ('WapAlu02',
'Exámenes', NULL);
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES ('WapAlu03',
'Consultas', NULL);
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES ('WapAlu04',
'Certificados', NULL);
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES ('WapAlu05',
'Parciales', NULL);
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES ('WapAlu06',
'Mensajes', NULL);
-- Modulos para el Docente
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES ('WapDoc01',
'Cursadas', NULL);
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES ('WapDoc02',
'Exámenes', NULL);
```

```
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES ('WapDoc03',
'Consultas', NULL);
INSERT INTO acc_modulos (modulo, nombre, bitmap) VALUES ('WapDoc04',
'Mensajes', NULL);
```

Archivo: Submodulo_WAP.dca

```
INSERT INTO acc_submodulos (submodulo, nombre, bitmap) VALUES
('Wap', 'Wap', NULL);
```

Asociación de módulos con perfiles

Archivo: WAP_modulos_x_perfil.dca

```
-- Modulos para el Alumno
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapAlu01',
'Wap00001');
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapAlu01',
'WapAlu01');
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapAlu01',
'WapAlu02');
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapAlu01',
'WapAlu03');
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapAlu01',
'WapAlu04');
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapAlu01',
'WapAlu05');
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapAlu01',
'WapAlu06');
-- Modulos para el Docente
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapDoc01',
'Wap00001');
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapDoc01',
'WapDoc01');
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapDoc01',
'WapDoc02');
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapDoc01',
'WapDoc03');
INSERT INTO acc_modulos_perfil (perfil, modulo) VALUES ('WapDoc01',
'WapDoc04');
```

Operaciones

Archivo: Operacion_netid003.dca

```
-- WAP
INSERT INTO dba.acc_oper_nom_x_int ( operacion, interfaz, nombre )
VALUES ( 'netid003', 5, 'Finalizar' );
INSERT INTO acc_parametros_wap(operacion, orden, pagina, titulo)
VALUES ('netid003', 1, 'w_general/finalizarSesion.php', 'Finalizar
sesión');
```

Archivo: Operacion_agcert01.dca

```
-- WAP
INSERT INTO dba.acc_oper_nom_x_int ( operacion, interfaz, nombre )
VALUES ( 'agcert01', 5, 'Certificados disponibles' );
INSERT INTO dba.acc_parametros_wap ( operacion, orden, pagina, titulo
) VALUES ( 'agcert01', 1, 'w_alumnos/w_certificados.php',
'Certificados' );
```

Archivo: Operacion_agcert02.dca

```
-- WAP
INSERT INTO dba.acc_oper_nom_x_int ( operacion, interfaz, nombre )
VALUES ( 'agcert02', 5, 'Certificados solicitados' );
INSERT INTO dba.acc_parametros_wap ( operacion, orden, pagina, titulo
) VALUES ( 'agcert02', 1, 'w_alumnos/w_certificados_pedidos.php',
'Certificados' );
```

Archivo: Operacion_agrep001.dca

```
-- WAP
INSERT INTO dba.acc_oper_nom_x_int ( operacion, interfaz, nombre )
VALUES ( 'agrep001', 5, 'Insc. a Exámenes' );
INSERT INTO dba.acc_parametros_wap ( operacion, orden, pagina, titulo
) VALUES ( 'agrep001', 1, 'w_alumnos/w_inscripcion_examenes.php',
'Exámenes' );
```

Archivo: Operacion_agrep002.dca

```
-- WAP
INSERT INTO dba.acc_oper_nom_x_int ( operacion, interfaz, nombre )
VALUES ( 'agrep002', 5, 'Insc. a Cursadas' );
INSERT INTO dba.acc_parametros_wap ( operacion, orden, pagina, titulo
) VALUES ( 'agrep002', 1, 'w_alumnos/w_inscripcion_cursadas.php',
'Cursadas' );
```

Archivo: Operacion_cur00037.dca

```
-- WAP
INSERT INTO dba.acc_oper_nom_x_int ( operacion, interfaz, nombre )
VALUES ( 'cur00037', 5, 'Parciales' );
INSERT INTO dba.acc_parametros_wap ( operacion, orden, pagina, titulo
) VALUES ( 'cur00037', 1, 'w_alumnos/w_evaluaciones_parciales.php',
'Parciales' );
```

Archivo: Operacion_doccur02.dca

```
-- WAP
INSERT INTO dba.acc_oper_nom_x_int ( operacion, interfaz, nombre )
VALUES ( 'doccur02', 5, 'Inscriptos cursada' );
INSERT INTO dba.acc_parametros_wap ( operacion, orden, pagina, titulo
) VALUES ( 'doccur02', 1, 'w_docentes/w_inscriptos_cursada.php',
'Cursadas' );
```

Archivo: Operacion_docexa02.dca

```
-- WAP
INSERT INTO dba.acc_oper_nom_x_int ( operacion, interfaz, nombre )
VALUES ( 'docexa02', 5, 'Inscriptos examen' );
INSERT INTO dba.acc_parametros_wap ( operacion, orden, pagina, titulo
) VALUES ( 'docexa02', 1, 'w_docentes/w_inscriptos_examen.php',
'Exámenes' );
```

Archivo: Operacion_exa00120.dca

```
-- WAP
INSERT INTO dba.acc_oper_nom_x_int ( operacion, interfaz, nombre )
VALUES ( 'exa00120', 5, 'Actas abiertas' );
INSERT INTO dba.acc_parametros_wap ( operacion, orden, pagina, titulo
) VALUES ( 'exa00120', 1, 'w_docentes/w_actas_abiertas.php',
'Exámenes' );
```

Archivo: OperacionesModulos_wap.sql

```
INSERT INTO acc_operacion_mod (operacion, modulo, submodulo, orden)
VALUES ('netid003','Wap00001','Wap',1000);
-- Configuración operaciones de ALUMNOS
INSERT INTO acc_operacion_mod (operacion, modulo, submodulo, orden)
VALUES ('agrep001','WapAlu02','Wap',10);
INSERT INTO acc_operacion_mod (operacion, modulo, submodulo, orden)
VALUES ('agrep002','WapAlu01','Wap',20);
INSERT INTO acc_operacion_mod (operacion, modulo, submodulo, orden)
VALUES ('agcert01','WapAlu04','Wap',30);
INSERT INTO acc_operacion_mod (operacion, modulo, submodulo, orden)
VALUES ('agcert02','WapAlu04','Wap',35);
INSERT INTO acc_operacion_mod (operacion, modulo, submodulo, orden)
VALUES ('cur00037','WapAlu05','Wap',40);
-- Configuración operaciones de Docentes
INSERT INTO acc_operacion_mod (operacion, modulo, submodulo, orden)
VALUES ('doccur02','WapDoc01','Wap',10);
INSERT INTO acc_operacion_mod (operacion, modulo, submodulo, orden)
VALUES ('docexa02','WapDoc02','Wap',20);
INSERT INTO acc_operacion_mod (operacion, modulo, submodulo, orden)
VALUES ('exa00120','WapDoc02','Wap',30);
```

Anexo C – Código fuente de Guaraní WAP

Se incluyen solo las páginas PHP utilizadas en el módulo Guaraní WAP.

Directorio principal (/)

Archivo: w_inicial.php

```
<?
    session_start();

    $_SESSION["ses_interfaz"] = "5";

    require_once("includes/parametros.inc.php");           // Carga los
parametros de la conexión
    require_once("adodb/adodb.inc.php");
    require_once("library/std_functions.lib.php");

    // Se conecta con la base y controla la unidad academica
    $objCon = conectar();

    $strSQL = "execute procedure sp_unidadAcademica('" .
$_SESSION["sUA"] . "')";
    $rsUAcad = $objCon->Execute($strSQL);

    if ( !$rsUAcad )
    {
        $_SESSION["ses_unidad_academica"] = "NADA";
        $_SESSION["ses_nombre_ua"]       = "NADA1";
    }
    elseif ( empty($rsUAcad->fields[0]) )
    {
        $_SESSION["ses_unidad_academica"] = "NADA";
        $_SESSION["ses_nombre_ua"]       = "NADA1";
    }
    else
    {
        $_SESSION["ses_unidad_academica"] = $rsUAcad->fields[0];
        $_SESSION["ses_nombre_ua"]       = $rsUAcad->fields[1];
    }

    registrarSALIAS();
?>
<html>
<head>
<title>GuaraniWAP!</title>
</head>
<body>
    <form action="<? echo "http://" . $_SERVER["SERVER_NAME"] .
$_SESSION["ses_sALIAS"] ?>w_general/autenticarse.php" method="POST"
>
        Usuario: <input type="text" name="fUsuario" size="10"
maxlength="20" value=""><br/>
        Clave: <input type="password" name="fClave" size="10"
maxlength="20" value=""><br/>
        <input type="submit" value="OK">
    </form>
</body>
</html>
```

Archivo: w_menu.php

```

<?
    include("includes/w_session.inc.php");
    $urloper = "http://" . $_SERVER["SERVER_NAME"] .
$_SESSION["ses_sALIAS"] . "w_operaciones.php";
    $urlexit = "http://" . $_SERVER["SERVER_NAME"] .
$_SESSION["ses_sALIAS"] . "w_general/finalizarSesion.php";
?>
<p>
<a href="<?echo $urloper?>">Menú Principal</a><br/>
<a href="<?echo $urlexit?>">Finalizar</a>
</p>

```

Archivo: w_notificarse.php

```

<?
require("adodb/adodb.inc.php");
require("adodb/crypt.inc.php");
include("includes/w_session.inc.php");
include("library/std_functions.lib.php");

if (isset($_SESSION["sCon"]))
{
    // Llama a la funcion que toma los parametros de la página.
    $parametros = Cargar_parametros();
}

// Si el llamado se hace desde la session porque
// la misma caducó entonces tengo que recuperar los parametros
directamente
// desde GET porque desde la session no tengo acceso a las funciones
de librerias.
if (! isset($parametros["fStatus"]) &&
    ! isset($parametros["fueraFrameSet"]) &&
    ! isset($parametros["fMsg"]))
    {
        $parametros["fStatus"] = $_GET["fStatus"];
        $parametros["fueraFrameSet"] = $_GET["fueraFrameSet"];
        $parametros["fMsg"] = $_GET["fMsg"];
    }

$strLeyendaStatus = "";
$strVoyAElegirOperacion = "N";
$strLeyendaStatus = "OK";

// Almaceno en fueraFrameSet si debo terminar la session y salir del
frameset.
$fueraFrameSet = (isset($parametros["fueraFrameSet"]) &&
    $parametros["fueraFrameSet"] == "1");

if ($parametros["fStatus"] <> 1)
    $strLeyendaStatus = "Problemas !!!";
?>
<html>
<body>
<?
    if ($strLeyendaStatus != "") echo $strLeyendaStatus .
'<br><br>';

    // realiza la conexión si la sesion no se encuentra vacia
    if (isset($parametros["fMsg"]))

```

```

    {
        $fMsg = urldecode($parametros["fMsg"]);
        // Si el mensaje tiene un mensaje
        // adjunto: Ejecuto el stored procedure.
        if (ereg("^[0-9]|-)", $fMsg) && isset($_SESSION["sCon"]))
        {
            $objCon = conectar();
            $strSQL = "EXECUTE PROCEDURE mme_muestraMensaje('" . $fMsg
. "' )";
            $rsMensajes = $objCon->Execute($strSQL);
            echo $rsMensajes->fields[0];
        } else {
            echo $fMsg;
        }
    }

    if ((($strVoyAElegirOperacion == "S") && ($parametros["fStatus"]
== "1")) || ($fueraFrameSet))
    {
        ?>
        Hace click en el boton para ingresar de nuevo al sistema.
        <a href=? echo ArmaLink("w_inicial.php"); ?>>Iniciar
sesi&oacute;n</a>
        <?
            unset($_SESSION["ses_sUsuario"]);
            unset($_SESSION["ses_nro_inscripcion"]);
            unset($_SESSION["ses_Apellido"]);
            unset($_SESSION["ses_Nombres"]);
        }
        ?>
    </body>
</html>

```

Archivo: w_notificarse_controles.php

```

<?
include("includes/w_session.inc.php");
require("adodb/adodb.inc.php");
require("adodb/crypt.inc.php");
include("library/std_functions.lib.php");
include("library/validador.lib.php");

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();

$operacion = "";
$paramsPagAnterior = "";
$paramsPagSiguiente = "";
$pagAnterior = "";
$pagSiguiente = "";

if (isset($parametros["operacion"]))
    $operacion = $parametros["operacion"];
// Por defecto si no hay pagina a la que ir
// se carga folder.php
if (!isset($parametros["PagSiguiente"]))
    $parametros["PagSiguiente"] = "./folder.php";
else
    $PagSiguiente = $parametros["PagSiguiente"];

if (isset($parametros["PagAnterior"]))
    $PagAnterior = $parametros["PagAnterior"];

```

```

if (isset($parametros["ParamsPagAnterior"]))
    $ParamsPagAnterior = $parametros["ParamsPagAnterior"];

if (isset($parametros["ParamsPagSiguiente"]))
    $ParamsPagSiguiente = $parametros["ParamsPagSiguiente"];

$strActuaComo = "M";
$strTitulo = "Mensaje !!!";
$strLeyenda = "Podes continuar";

// Si tengo un estado que mostrar
// tengo que cambiar el titulo.
if (isset($parametros["fStatus"]))
{
    if ($parametros["fStatus"] <= -1)
    {
        $strTitulo = "Problemas !!!";
        $strLeyenda = "NO podes continuar";
    }
    elseif ($parametros["fStatus"] == 0)
    {
        $strTitulo = "Advertencia !!!";
        $strLeyenda = "Podes continuar";
    }
}

// crea la conexión a la BD
$objCon = conectar();
?>
<html>
<head>

</head>
<body class="cuerpo">
<?
    for ($j= 0; $j <= count($_SESSION["sPilaMensajes"]) -1; $j++)
    {
        if (($strActuaComo == "M" &&
$_SESSION["sPilaMensajes"][$j][4] == "A") ||
            ($strActuaComo == "M" &&
$_SESSION["sPilaMensajes"][$j][4] == "E") ||
            ($strActuaComo == "A" &&
$_SESSION["sPilaMensajes"][$j][4] == "E")
        )
            $strActuaComo = $_SESSION["sPilaMensajes"][$j][4];
    }
?>

<?

    if (!empty($_SESSION["sPilaMensajes"]))
    {

        for ($j= 0; $j <= count($_SESSION["sPilaMensajes"]) -
1; $j++)
        {
            $strSQL = "EXECUTE PROCEDURE mme_muestraMensaje('"
.$_SESSION["sPilaMensajes"][$j][3] . "')";
            $rsMensajes = $objCon->Execute($strSQL);

            // evaluo el resultado de la operacion
            $strWarning = $rsMensajes->fields[0];

            echo $strWarning . '<br/>';

```

```

        $rsMensajes->Close();
    }
}
?>
<?
    if ($strActuaComo != "M")
    {
        $url = ArmaLink($PagAnterior . "?" .
str_replace("|", "&", $ParamsPagAnterior) . "&Notificarse=S&operacion="
. $operacion);
        ?>
    <?
    }

    if ($parametros["fStatus"] >= 0)
    {
        $url = ArmaLink($PagSiguiente . "?" .
str_replace("|", "&", $ParamsPagSiguiente) . "&operacion=" . $operacion
. "&Notificarse=S");
        ?>
    <?
    }
?>
</body>
</html>

```

Archivo: w_operaciones.php

```

<?
    require_once("includes/w_session.inc.php");
    require_once("adodb/adodb.inc.php");
    require_once("adodb/crypt.inc.php");
    require_once("library/std_functions.lib.php");

    // Llama a la funcion que toma los parametros de la página.
    $parametros = Cargar_parametros();

    // realiza la conexión
    $objCon = conectar();

    $strSQL = "EXECUTE PROCEDURE sp_operaciones_net('" .
    $parametros["unidad_academica"] . "', '" . $parametros["sUsuario"] .
    "', 5)";
    $rsOperac = $objCon->Execute($strSQL);
?>
<html>
    <head>
        <title>GuaraniWAP!</title>
    </head>
    <body>
        <?
            if ($rsOperac && !$rsOperac->EOF)
            {
                while (!$rsOperac->EOF) {
                    $strCat = $rsOperac->fields[4];
                ?>

                <?
                    // loopeo de categorias
                    while (($rsOperac->fields[4] == $strCat) &&
(! $rsOperac->EOF))
                    {

```

```

        $strItem = $rsOperac->fields[6];
        $sLink = ArmaLink("http://" .
$_SERVER["SERVER_NAME"] . $_SESSION["ses_sALIAS"] . trim($rsOperac-
>fields[7]) . "?operacion=" . trim($rsOperac->fields[2]) .
"&nombre_operacion=" . urlencode(trim($rsOperac->fields[6]));
        ?>
        <p>
        <a href="<? echo $sLink; ?>" ><? echo
trim($rsOperac->fields[6]); ?></a>
        </p>
        <?
        $rsOperac->MoveNext();
        }
        ?>
    <?
    }
    } else {
    ?>
    No hay operaciones para este perfil !!
    <?
    }
    ?>
</body>
</html>

```

Directorio de librerías (/library)

Archivo: std_functions.lib.php (función Cargar_parametros())

```

/**
 * Retorna en el arreglo $parametros
 * las variables de $_GET, $_POST y $_SESSION
 * ademas:
 * "nombre_operacion"
 * "titulo_pagina"
 * "sub_pagina_XX"
 * "PagAnterior"
 * "PagSiguiente"
 */
function Cargar_parametros()
{
    $objCon = conectar();

    if (isset($_GET["qs"]))
    {
        $crypt = new MD5Crypt();
        $qs_crypt = $crypt->Decrypt($_GET["qs"],qs_clave);

        parse_str($qs_crypt,$qs);
        $parametros = array_merge($qs,$_POST);
    } else
        $parametros = $_POST;

    if (array_key_exists("nombre_operacion",$parametros))
        $_SESSION["ses_nombre_operacion"] =
urlencode($parametros["nombre_operacion"]);

    foreach ($_SESSION as $key => $value)
    {
        if (substr($key,0,4) == "ses_")
            $parametros[substr($key,4)] = $_SESSION[$key];

        if (substr($key,0,9) == "par_oper_")

```

```

        $parametros[substr($key,9)] = $_SESSION[$key];
    }

    // Ahora saco la pagina siguiente y anterior si es que existen...
    $sPagina = ereg_replace('^' . $parametros["sALIAS"] , '',
$_SERVER["SCRIPT_NAME"]);

    if (isset($parametros["operacion"]))
    {
        $strSQL = "EXECUTE PROCEDURE sp_paginas_net('" .
        $parametros["operacion"] . "', '" . $sPagina . "', '" .
        $_SESSION["ses_interfaz"] . "')";
        $rsPaginas = $objCon->Execute($strSQL);

        $iOrden = $rsPaginas->fields[2];
        $_SESSION["ses_titulo_pagina"] = $rsPaginas->fields[3];
        $parametros["titulo_pagina"] = $rsPaginas->fields[3];

        while (!$rsPaginas->EOF)
        {
            if ($iOrden > $rsPaginas->fields[1] && ($rsPaginas->fields[1]
            != $rsPaginas->fields[2]))
                $parametros["PagAnterior"] = $parametros["sALIAS"] .
                $rsPaginas->fields[0];
            else
                $parametros["PagSiguiente"] = $parametros["sALIAS"] .
                $rsPaginas->fields[0];

            /* Tengo carpetas? */
            if ($rsPaginas->fields[4] != "")
            {
                $strSQL = "EXECUTE PROCEDURE sp_paginas('" . $rsPaginas-
                >fields[4] . "')";
                $rsSubPag = $objCon->Execute($strSQL);

                if (!$rsSubPag->EOF)
                {
                    $parametros["sub_paginas"] = $rsSubPag->RecordCount();
                    $i = 1;
                    while (!$rsSubPag->EOF)
                    {
                        $parametros["sub_pagina_" . $i]["titulo"] = $rsSubPag-
                        >fields[2];
                        $parametros["sub_pagina_" . $i]["pagina"] = $rsSubPag-
                        >fields[3];
                        $rsSubPag->MoveNext();
                        $i++;
                    }
                }
            }

            $rsPaginas->MoveNext();
        }
    }
    return $parametros;
}

```

Directorio de archivos comunes (/includes)

Archivo: w_session.inc.php

```
<?
ini_set('display_errors','0');
//Constantes de uso general
define("qs_clave","guar*387#3w");

define("SPATH", "http://" . $_SERVER["SERVER_NAME"] . "/" .
substr($_SERVER["PHP_SELF"],1,strpos($_SERVER["PHP_SELF"],"/",1)));

session_start();
?>
```

Directorio de uso general (/w_general)

Archivo: autenticarse.php

```
<?
require_once("../adodb/adodb.inc.php");
require_once("../adodb/encrypt.inc.php");
require_once("../includes/w_session.inc.php");
require_once("../library/std_functions.lib.php");

unset($_SESSION["ses_UsuarioValidado"]);

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();

// Si no esta qs, entonces le pego el GET
$parametros = array_merge($parametros,$_GET);

// controlo que el usuario haya sido ingresado
if (! array_key_exists("fUsuario",$parametros))
{
    $url = ArmaLink("../w_notificarse.php?" .
        "fStatus=". urlencode(-1) .
        "&fMsg=" . urlencode("-746, NO existe un usuario en la sesión,
identificate otra vez.")
        . "&operacion=".urlencode($parametros["operacion"]));

    header("Location: $url");
    exit;
}

// realiza la conexión
$objCon = conectar(false);

$strSQL = "execute procedure sp_autenticarse('" .
$parametros["unidad_academica"] . "', '" . $parametros["fUsuario"] .
"', '" . $parametros["fClave"] . "',5)";
$rsAutentif = $objCon->Execute($strSQL);

if (! $rsAutentif || ($rsAutentif->fields[0] < 0))
{
    // Limpia la session
    if (isset($_SESSION["ses_sUsuario"]))
unset($_SESSION["ses_sUsuario"]);
}
```

```

    if (isset($_SESSION["ses_nro_inscripcion"]))
unset($_SESSION["ses_nro_inscripcion"]);
    if (isset($_SESSION["sApellido"]))
unset($_SESSION["sApellido"]);
    if (isset($_SESSION["sNombres"]))
unset($_SESSION["sNombres"]);
    if (isset($_SESSION["sAlerta"]))
unset($_SESSION["sAlerta"]);

    if ($rsAutentif)
    {
        $url = ArmaLink("../w_notificarse.php?" .
            "fStatus=". $rsAutentif->fields[0] .
            "&fMsg=" . urlencode($rsAutentif->fields[1])
            . "&operacion=" . urlencode($parametros["operacion"]) );
        header("Location: $url");
        exit;
    }
    else
    {
        $url = ArmaLink("../w_notificarse.php?" .
            "fStatus=" . -1 .
            "&fMsg=" . urlencode($objCon->ErrorMsg())
            . "&operacion=" . urlencode($parametros["operacion"]));
        header("Location: $url");
        exit;
    }
}

$intStatus = $rsAutentif->fields[0];
$strMsg = $rsAutentif->fields[1];

// El resultado fue OK, entonces recupera los datos personales del
usuario
$strSQL = "execute procedure sp_datosPersUsrAG('" .
$parametros["unidad_academica"] . "', '" . $parametros["fUsuario"] .
"')";
$rsDatosPer = $objCon->Execute($strSQL);

if (!$rsDatosPer)
{
    $url = ArmaLink("../w_notificarse.php?" .
        "fStatus=" . -1 .
        "&fMsg=" . urlencode("800301," .
$parametros["fUsuario"]) . "&operacion=" . urlencode($parametros["operacio
n"]));
    header("Location: $url");
    exit;
}

if (empty($rsDatosPer->fields[4]))
{
    $url = ArmaLink("../w_notificarse.php?" .
        "fStatus=" . -1 .
        "&fMsg=" . urlencode("800301," . $parametros["fUsuario"])
        . "&operacion=" . urlencode($parametros["operacion"]));
    header("Location: $url");
    exit;
}

// incluyo al usuario en la session
$_SESSION["ses_sUsuario"] = $parametros["fUsuario"];
$_SESSION["ses_nro_inscripcion"] = $rsDatosPer->fields[1];
$_SESSION["ses_Apellido"] = $rsDatosPer->fields[4];

```

```

$_SESSION["ses_Nombres"] = $rsDatosPer->fields[5];

unset($_SESSION["sAlerta"]);

$_SESSION["ses_UsuarioValidado"] = "S";
$_SESSION["ses_nombre_operacion"] = "";
// voy a mostrar las operaciones disponibles

$url = "http://" . $_SERVER["SERVER_NAME"] . $_SESSION["ses_sALIAS"]
. "w_operaciones.php";
header("Location: $url");
?>

```

Archivo: finalizarSesion.php

```

<?
include("../includes/w_session.inc.php");

session_destroy();

header("Location: ../w_inicial.php");
?>

```

Directorio de archivos de alumnos (/w_alumnos)

Archivo: w_certificados_habilitados.php

```

<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/encrypt.inc.php");
include("../library/std_functions.lib.php");
include("../library/validador.lib.php");

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();

// crea la conexión a la BD
$objCon = conectar();

// recupero los certificados de la persona
$strSQL = "execute procedure sp_certif_habilit ('" .
$parametros["unidad_academica"] . "', '" .
$parametros["nro_inscripcion"] . "')";
$rsCertifHabilit = $objCon->Execute($strSQL);
?><html>
<head>
<title>CERTIFICADOS</title>
</head>
<body>

<?
$rsCertifHabilit->MoveFirst();
while (!$rsCertifHabilit->EOF)
{
    if ($rsCertifHabilit->fields[3] == "S")
    {
        echo $rsCertifHabilit->fields[1] . "<br/>";
    }
    $rsCertifHabilit->MoveNext();
}

```

```
?>
    <? include("../w_menu.php"); ?>
</body>
</html>
```

Archivo: w_certificados_habilitados.php

```
<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/crypt.inc.php");
include("../library/std_functions.lib.php");
include("../library/validador.lib.php");

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();

    // crea la conexión a la BD
    $objCon = conectar();

    // recupero los certificados de la persona
    $strSQL = "execute procedure sp_certif_pedidos ('" .
$parametros["unidad_academica"] . "', '" .
$parametros["nro_inscripcion"] . "')";
    $rsCertifHabilit = $objCon->Execute($strSQL);
?><html>
<head>
<title>CERTIFICADOS</title>
</head>
<body>

<?
    $rsCertifHabilit->MoveFirst();
    while (!$rsCertifHabilit->EOF)
    {
//        if ($rsCertifHabilit->fields[3] == "S")
//            {
                echo "<b>" . $rsCertifHabilit->fields[1] . "</b><br/>";
                echo "F. Solicitud: " . $rsCertifHabilit->fields[5] .
"<br/>";
                echo "F. Emisi&oacute;n: " . $rsCertifHabilit->fields[6]
. "<br/>";
                echo "Nro Trans: " . $rsCertifHabilit->fields[7] .
"<br/>";
                echo "<br/>";
//            }
            $rsCertifHabilit->MoveNext();
        }
?>
    <? include("../w_menu.php"); ?>
</body>
</html>
```

Archivo: w_evaluacion_parcial_detalle.php

```
<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/crypt.inc.php");
include("../library/std_functions.lib.php");

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();
```

```

// recupero los parámetros enviados por la pagina anterior
if (empty($parametros["fCarrera"])) {
    $url = ArmaLink("../w_notificarse.php?fStatus=" . urlencode("-1") . "&fMsg=" . urlencode("-746, NO existe la Carrera en la sesión, identificate otra vez") . "&operacion=" . urlencode($parametros["operacion"]));
    header("Location: $url");
    exit;
}

if (empty($parametros["fLegajo"])) {
    $url = ArmaLink("../w_notificarse.php?fStatus=" . urlencode("-1") . "&fMsg=" . urlencode("-746, NO existe el Legajo en la sesión, identificate otra vez") . "&operacion=" . urlencode($parametros["operacion"]));
    header("Location: $url");
    exit;
}

if (empty($parametros["fComision"])) {
    $url = ArmaLink("../w_notificarse.php?fStatus=" . urlencode("-1") . "&fMsg=" . urlencode("-746, NO existe la Comisión en la sesión, identificate otra vez") . "&operacion=" . urlencode($parametros["operacion"]));
    header("Location: $url");
    exit;
}

// crea la conexión a la BD
$objCon = conectar();
// recupera las comisiones en las que está inscripto el alumno
$strSQL = "execute procedure sp_evalparcalumno('" . $parametros["unidad_academica"] . "', '" . $parametros["fCarrera"] . "', '" . $parametros["fLegajo"] . "', '" . $parametros["fComision"] . "')";
$rsEvalParc = $objCon->Execute($strSQL);
?>
<html>
<head>
<title>EVALUACIONES</title>
</head>
<body>
<?
    if (!$rsEvalParc || $rsEvalParc->EOF)
    {
    ?>
        No hay información disponible.
        <? include("../w_menu.php"); ?>
    <?
    }
    else
    {
    ?>
        <?
        while ( !$rsEvalParc->EOF)
        {
        ?>
            <p><a href="<?
                $url = "http://" . $_SERVER["SERVER_NAME"] . $_SESSION["ses_sALIAS"] . "w_alumnos/w_nota_parcial.php".
                    "?materia=" . trim($rsEvalParc->fields[15]) .
                    "&comision=" . trim($rsEvalParc->fields[16]) .
                    "&evaluacion=" . trim($rsEvalParc->fields[13]) .

```

```

        "&fecha=" . $rsEvalParc->fields[8] .
        "&nota=" . $rsEvalParc->fields[9] .
        "&resultado=" . trim($rsEvalParc->fields[10]) .
        "&corregido=" . trim($rsEvalParc->fields[11]) .
        "&observaciones=" . trim($rsEvalParc->fields[12])
;
        echo ($url);
        ?>>
        <? echo substr($rsEvalParc->fields[13], 0, 40); ?>
(<? echo substr($rsEvalParc->fields[9], 0, 40); ?>)
        </a></p>
        <?
                $rsEvalParc->MoveNext();
        }
    }
    ?>
</body>
</html>

```

Archivo: w_evaluaciones_parciales.php

```

<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/crypt.inc.php");
include("../library/std_functions.lib.php");
include("../library/validador.lib.php");

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();

// Si viene del notificador de errores o mensajes de pntos de
control no ejecuto nuevamente el control
if (!array_key_exists("Notificarse", $parametros))
{
    // limpio la cola de mensajes del Pto. de Control y ejecuto los
controles
    resetMensajes();
    resetPuntosControl();

    // LLamo al validador
    $resultado =
ejecutar_puntos_pagina($parametros["unidad_academica"], $parametros["op
eracion"], $_SERVER["SCRIPT_NAME"]);

    // Verifico el resultado y resuelvo si sigue o muestra el error
...
    if ($resultado <= 0){
        $url = ArmaLink("../w_notificarse_controles.php?fStatus=" .
$resultado . "&PagAnterior=folder.php&PagSiguiente=" .
$_SERVER["SCRIPT_NAME"] . "&operacion=" . $parametros["operacion"]);
        header("Location: $url");
        exit();
    }
}

$strEstado = "A"; // Recupera solo las inscripciones Activas

// recupero los parámetros enviados por la pagina anterior
if (empty($parametros["operacion"]))
{

```

```

    $url = ArmaLink("../w_notificarse.php?fStatus=" . urlencode("-
1")."&fMsg=".urlencode("-746, NO existe la Operacion en la sesi3n,
identificate otra vez"));
    header("Location: $url");
    exit;
}

// crea la conexi3n a la BD
$objCon = conectar();
// recupera las comisiones en las que est1 inscripto el alumno
$strSQL = "execute procedure sp_inscursadas('' .
$params["unidad_academica"] . "', ' ' .
$params["nro_inscripcion"] . "', ' ' . $strEstado . ' ' )";
$rsInscCurs = $objCon->Execute($strSQL);
?>
<html>
<head>
<title>MATERIAS</title>
</head>
<body>
    <?
    if (!$rsInscCurs || $rsInscCurs->EOF)
    {
    ?>
        No hay cronogramas disponibles.
        <? include("../w_menu.php"); ?>
    <?
    }
    else
    {
    ?>
        <b><? echo "Materia/Comisi3n"; ?></b><br/>
    <?
        // es para hacer el corte por carrera
        $strCarrera = $rsInscCurs->fields[1];
        while (!$rsInscCurs->EOF)
        {
        ?>

            <a href=<?
                $url = "http://" . $_SERVER["SERVER_NAME"] .
$_SESSION["ses_sALIAS"] .
"w_alumnos/w_evaluacion_parcial_detalle.php".
                    "?operacion=" . $params["operacion"] .
                    "&fCarrera=" . $rsInscCurs->fields[1] .
                    "&fLegajo=" . $rsInscCurs->fields[2] .
                    "&fComision=" . $rsInscCurs->fields[3] ;
                    echo ArmaLink($url);
                    ?>>

                <? echo substr($rsInscCurs->fields[15], 0, 40) . '/' .
$rsInscCurs->fields[12]; ?><br/>
                </a>

            <?
                $rsInscCurs->MoveNext ();
            }
        }
    ?>
</body>
</html>

```

Archivo: w_inscripcion_cursada_detalle.php

```

<?
    include("../includes/w_session.inc.php");
    require("../adodb/adodb.inc.php");
    require("../adodb/crypt.inc.php");
    include("../library/std_functions.lib.php");
    include("../library/validador.lib.php");
?>

<html>
<head>
<title><? echo nvl(urldecode($_GET["materia"])); ?></title>
</head>
<body>
<p>
<b><? echo nvl(urldecode($_GET["carrera"])); ?></b><br/>
<? echo "Fecha: " . nvl(urldecode($_GET["fecha_inscripcion"]));
?><br/>
<? echo "Calidad: " . nvl(urldecode($_GET["calidad"])); ?><br/>
<? echo "Año: " . nvl(urldecode($_GET["anio_academico"])); ?><br/>
<? echo "P. Lectivo: " . nvl(urldecode($_GET["periodo_lectivo"]));
?><br/>
<? echo "Estado: " . nvl(urldecode($_GET["estado"])); ?><br/>
<? echo "Nro Trans: " . nvl(urldecode($_GET["nro_transaccion"])); ?>
</p>
    <? include("../w_menu.php"); ?>

</body>
</html>

```

Archivo: w_inscripcion_cursadas.php

```

<?
    include("../includes/w_session.inc.php");
    require("../adodb/adodb.inc.php");
    require("../adodb/crypt.inc.php");
    include("../library/std_functions.lib.php");

    // Llama a la funcion que toma los parametros de la página.
    $parametros = Cargar_parametros();

    // creo la conexión
    $objCon = conectar();

    // recupero todas las materias en las que está inscripto
    $strEstado = "T"; // Puede ser: A:Activa, B:Baja, D:Pendiente,
T:Todas

    // recupero las materias en las que el alumno está inscripto
    $strSQL = "execute procedure sp_inscCursadas('" .
    $parametros["unidad_academica"] . "', '" .
    $parametros["nro_inscripcion"] . "', '" . $strEstado . "')";
    $rsInscCurs = $objCon->Execute($strSQL);
?>
<html>
    <head>
        <title>MATERIAS</title>
    </head>
    <body >
        <?

            if ($rsInscCurs->EOF)
            {

```

```

?>
No existen inscripciones a cursadas.
<? include("../w_menu.php"); ?>
<?
}
else
{
?>
<?
while (!$rsInscCurs->EOF)
{
?>
    <? $sLink = "http://" . $_SERVER["SERVER_NAME"] .
$_SESSION["ses_sALIAS"] .
"w_alumnos/w_inscripcion_cursada_detalle.php".
"?materia=" . trim($rsInscCurs-
>fields[15]) .
"&carrera=" . trim($rsInscCurs->fields[14]) .
"&fecha_inscripcion=" .
ConvertirFecha($rsInscCurs->fields[7],1) .
"&estado=" . $rsInscCurs->fields[11] .
"&calidad=" . $rsInscCurs->fields[6] .
"&nro_transaccion=" . $rsInscCurs-
>fields[10] .
"&anio_academico=" . $rsInscCurs-
>fields[9] .
"&periodo_lectivo=" . $rsInscCurs-
>fields[8]; ?>
    <p><a href="<? echo $sLink; ?>" ><? echo trim($rsInscCurs-
>fields[15]); ?></a></p>
    <?
        $rsInscCurs->MoveNext ();
    }
?>
<?
}
?>
</body>
</html>

```

Archivo: w_inscripcion_examen_detalle.php

```

<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/crypt.inc.php");
include("../library/std_functions.lib.php");
include("../library/validador.lib.php");
?>

<html>
<head>
<title><? echo nvl(urldecode($_GET["materia"])); ?></title>
</head>
<body>
<p>
<b><? echo nvl(urldecode($_GET["carrera"])); ?></b><br/>
<? echo "Mesa/llam: " . nvl(urldecode($_GET["mesa"])) . "/" .
nvl(urldecode($_GET["llamado"])); ?><br/>
<? echo "F. Insc: " . nvl(urldecode($_GET["fecha_inscripcion"]));
?><br/>
<? echo "F. Examen: " . nvl(urldecode($_GET["fecha_examen"])); ?><br/>
<? echo "Tipo Insc.: " . nvl(urldecode($_GET["tipo"])); ?><br/>

```

```

<? echo "Estado: " . nvl(urldecode($_GET["estado"])); ?><br/>
<? echo "Nro Trans: " . nvl(urldecode($_GET["nro_transaccion"]));
?><br/>
</p>
<? include("../w_menu.php"); ?>
</body>
</html>

```

Archivo: w_inscripcion_examenes.php

```

<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/crypt.inc.php");
include("../library/std_functions.lib.php");
include("../library/validador.lib.php");

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();

// Si viene del notificador de errores o mensajes de pntos de
control no ejecuto nuevamente el control
// if (!array_key_exists("Notificarse",$parametros))
// {
//     // limpio la cola de mensajes del Pto. de Control y ejecuto los
controles
//     resetMensajes();
//     resetPuntosControl();

// LLamo al validador
// $resultado =
ejecutar_puntos_pagina($parametros["unidad_academica"],$parametros["op
eracion"],$_SERVER["SCRIPT_NAME"]);

// Verifico el resultado y resuelvo si sigue o muestra el error
...
// if ($resultado <= 0)
// {
//     $url = ArmaLink("../w_notificarse_controles.php?fStatus=" .
$resultado . "&PagAnterior=folder.php&PagSiguiente=" .
$_SERVER["SCRIPT_NAME"] . "&operacion=" . $parametros["operacion"]);
//     header("Location: $url");
//     exit();
// }
// }

// crea la conexión a la BD
$objCon = conectar();

// Consulto las Materias en las que esta inscripto
$strSQL = "execute procedure sp_inscExam('" .
$parametros["unidad_academica"] . "', '" .
$parametros["nro_inscripcion"] . "')";
$rsInscExam = $objCon->Execute($strSQL);
?>

<html>
<head>
<title>MATERIAS</title>
</head>
<body >
<?

```

```

        if ($rsInscExam->EOF)
        {
        ?>
        No existen inscripciones a exámenes.
        <? include("../w_menu.php"); ?>
        <?
        }
        else
        {
        ?>
        <?
        while (!$rsInscExam->EOF)
        {
        ?>
                <? $sLink = "http://" . $_SERVER["SERVER_NAME"] .
$_SESSION["ses_sALIAS"] .
"w_alumnos/w_inscripcion_examen_detalle.php".
                "?materia=" . trim($rsInscExam->fields[19]) .
                "&carrera=" . trim($rsInscExam->fields[22]) .
                "&mesa=" . trim($rsInscExam->fields[8]) .
                "&llamado=" . trim($rsInscExam->fields[7]) .
                "&fecha_inscripcion=" . $rsInscExam-
>fields[4] .
                "&fecha_examen=" .
ConvertirFecha($rsInscExam->fields[16],2) . " " . substr($rsInscExam-
>fields[17],0,5) .
                "&tipo=" . $rsInscExam->fields[20] .
                "&nro_transaccion=" . $rsInscExam-
>fields[24] .
                "&estado=" . $rsInscExam->fields[21]; ?>
                <p><a href="<? echo $sLink; ?>" ><? echo trim($rsInscExam-
>fields[19]); ?></a></p>
                <?
                $rsInscExam->MoveNext();
                }
        ?>
        <?
        }
        ?>
        </body>
</html>

```

Archivo: w_nota_parcial.php

```

<?
    include("../includes/w_session.inc.php");
    require("../adodb/adodb.inc.php");
    require("../adodb/crypt.inc.php");
    include("../library/std_functions.lib.php");
    include("../library/validador.lib.php");
?>

<html>
    <head>
        <title><? echo nvl(urldecode($_GET["materia"])); ?> - <? echo
nvl(urldecode($_GET["comision"])); ?></title>
    </head>
    <body>
        <b><? echo nvl(urldecode($_GET["evaluacion"])); ?></b><br/>
        <? echo "Fecha: " . nvl(urldecode($_GET["fecha"])); ?><br/>
        <? echo "Nota: " . nvl(urldecode($_GET["nota"])) . " (" .
nvl(urldecode($_GET["resultado"])) . ")"; ?><br/>

```

```
<? // echo nvl(urldecode($_GET["corregido"])) . ": " .  
nvl(urldecode($_GET["observaciones"])); ?>  
  <? include("../w_menu.php"); ?>  
</body>  
</html>
```

Directorio de archivos de docentes (/w_docentes)

Archivo: w_actas_abiertas.php

```

<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/crypt.inc.php");
include("../library/std_functions.lib.php");
include("../library/validador.lib.php");

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();

// crea la conexión a la BD
$objCon = conectar();

$strSQL = "EXECUTE PROCEDURE sp_turnos_examen()";
$rsTurnos = $objCon->Execute($strSQL);

$arrTurnos = ordenarRS($rsTurnos, array(0 => array(3,"date,ASC")));

$strSQL = "EXECUTE PROCEDURE sp_busq_actaexa('" .
$params["unidad_academica"] . "' ,'" .
$params["nro_inscripcion"] . "' ,'" .
(array_key_exists("anio_academico",$parametros) &&
($parametros["anio_academico"] != ''))
? "'" . $parametros["anio_academico"] . "'" : "NULL") . "' ,'" .
(array_key_exists("turno_examen",$parametros) &&
($parametros["turno_examen"] != ''))
? "'" . stripslashes($parametros["turno_examen"]) . "'" :
"NULL") . "' ,'" .
(array_key_exists("materia",$parametros) &&
($parametros["materia"] != ''))
? "'" . $parametros["materia"] . "'" : "NULL") .
";" ;
user_error($strSQL, E_USER_NOTICE);
$rsActas = $objCon->Execute($strSQL);

?>
<html>
<head>
<title>ACTAS</title>
</head>
<body>
  <?
    if (isset($rsActas) && !$rsActas->EOF)
    {
      $indiceSel = 0;

      $arrOrd = ordenarRS($rsActas, array(0 =>
array(1,"integer,DESC")));

      if (!array_key_exists("indice", $parametros))
      {
        $arr = scrollArray($arrOrd, -1, 1);
        $indiceSel = 0;
      } else {
        $arr = scrollArray($arrOrd, $parametros["indice"], 1);
        $indiceSel = $parametros["indice"];

```

```

    }
    for ($i=0; $i < count($arr[0]); $i++)
    {
    ?>
        <b><? echo $rsActas->fields[4]; ?></b><br/>
        &nbsp;<? echo "Nro Acta: " . $arr[0][$i][1]; ?><br/>
        &nbsp;<? echo "Año: " . $arr[0][$i][5]; ?><br/>
        &nbsp;<? echo "Turno: " . $arr[0][$i][6]; ?><br/>
        &nbsp;<? echo "Estado: "; ?>
        <? switch ($arr[0][$i][9])
            {
                case 'A': echo 'Abierta'; break;
                case 'C': echo 'Cerrada'; break;
                case 'B': echo 'Invalidada'; break;
                case 'I': echo 'Impresa sin notas'; break;
                case 'N': echo 'Impresa con notas'; break;
                default : echo 'Indeterminado';
            }
        ?><br/>
    <?
    }
    ?>
    <? include("../w_menu.php"); ?>
</body>
</html>

```

Archivo: w_agenda_cursadas.php

```

<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/crypt.inc.php");
include("../library/std_functions.lib.php");
include("../library/validador.lib.php");

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();

// crea la conexión a la BD
$objCon = conectar();

$strSQL = "EXECUTE PROCEDURE sp_AgendaDocCur('" .
$parametros["unidad_academica"] . "', '" .
$parametros["nro_inscripcion"] . "')";
$rsAgendaDocCur = $objCon->Execute($strSQL);

?>
<html>
<head>
<title>AGENDA</title>
</head>
<body>
    <?
        if (!$rsAgendaDocCur || $rsAgendaDocCur->EOF) {
    ?>
            No hay información sobre la agenda de clases.
    <?
        }
        else
        {
            $strTemp = $rsComisionesDoc->fields[4];
            while (!$rsAgendaDocCur->EOF)

```

```

    {
        if ($strTemp != $rsAgendaDocCur->fields[5])
        {
            $strTemp = $rsAgendaDocCur->fields[5];
        }
    }
    ?>
    <b><? echo $rsAgendaDocCur->fields[6] . " - " .
$rsAgendaDocCur->fields[4]; ?></b><br/>
    &nbsp;&nbsp;<? echo "Dia: " . $rsAgendaDocCur->fields[10];
?><br/>
    &nbsp;&nbsp;<? echo "Hs: " . $rsAgendaDocCur->fields[11] . " - "
. $rsAgendaDocCur->fields[16]; ?><br/>
    &nbsp;&nbsp;<? echo "Aula: " . $rsAgendaDocCur->fields[13] . " -
" . $rsAgendaDocCur->fields[15] ?><br/>
    <?
        $rsAgendaDocCur->MoveNext();
    }
}
?>
<? include("../w_menu.php"); ?>
</body>
</html>

```

Archivo: w_agenda_exámenes.php

```

<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/crypt.inc.php");
include("../library/std_functions.lib.php");
include("../library/validador.lib.php");

// Llama a la función que toma los parámetros de la página.
$parametros = Cargar_parametros();

// crea la conexión a la BD
$objCon = conectar();

$strSQL = "EXECUTE PROCEDURE sp_AgendaDocExa('" .
$parametros["unidad_academica"] . "', '" .
$parametros["nro_inscripcion"] . "')";
$rsAgendaDocExa = $objCon->Execute($strSQL);

?>
<html>
<head>
<title>AGENDA</title>
</head>
<body>
    <?
        if (!$rsAgendaDocExa || $rsAgendaDocExa->EOF) {
            ?>
                No hay información sobre la agenda de exámenes.
            <?
                }
            else
            {
                $strTemp = $rsAgendaDocExa->fields[4];
                while (!$rsAgendaDocExa->EOF)
                {
                    if ($strTemp != $rsAgendaDocExa->fields[1])
                    {
                        $strTemp = $rsAgendaDocExa->fields[1];
                    }
                }
            }
        }
    }

```

```

    }
    ?>
    <b><? echo $rsAgendaDocExa->fields[2]; ?></b><br/>
    &nbsp;&nbsp;&nbsp;<? echo "Mesa/Llam: " . $rsAgendaDocExa->fields[5] .
" - " . $rsAgendaDocExa->fields[6]; ?><br/>
    &nbsp;&nbsp;&nbsp;<? echo "Fecha: " . $rsAgendaDocExa->fields[11];
?><br/>
    &nbsp;&nbsp;&nbsp;<? echo "Hs: " . $rsAgendaDocExa->fields[12] . " - "
. $rsAgendaDocExa->fields[17] ; ?><br/>
    &nbsp;&nbsp;&nbsp;<? echo "Aula: " . $rsAgendaDocExa->fields[14] . " -
" . $rsAgendaDocExa->fields[16];?><br/>
    <?
        $rsAgendaDocExa->MoveNext();
    }
}
?>
<? include("../w_menu.php"); ?>
</body>
</html>

```

Archivo: w_inscriptos_cursada.php

```

<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/encrypt.inc.php");
include("../library/std_functions.lib.php");
include("../library/validador.lib.php");

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();

// crea la conexión a la BD
$objCon = conectar();

$strSQL = "EXECUTE PROCEDURE sp_ComisionesDoc('" .
$parametros["unidad_academica"] . "', '" .
$parametros["nro_inscripcion"] . "')";
$rsComisionesDoc = $objCon->Execute($strSQL);

?>
<html>
<head>
<title>INSCRIPTOS</title>
</head>
<body>
    <?
        if (!$rsComisionesDoc || $rsComisionesDoc->EOF) {
    ?>
        No figura en ninguna comisi&oacute;n vigente.
    <?
        }
    else
    {
        $strTemp = $rsComisionesDoc->fields[4];
        while (!$rsComisionesDoc->EOF)
        {
            if ($strTemp != $rsComisionesDoc->fields[4])
            {
                $strTemp = $rsComisionesDoc->fields[4];
            }
        }
    ?>
        <b><? echo $rsComisionesDoc->fields[5] ?>:</b><br/>

```

```

        &nbsp;   <? echo "Com: " . $rsComisionesDoc->fields[3];
?><br/>
        &nbsp;   <? echo "P Lect: " . $rsComisionesDoc->fields[6]
?><br/>
        &nbsp;   <? echo "Cnt Insc: " . $rsComisionesDoc->fields[9]
?><br/>
        <?
            $rsComisionesDoc->MoveNext();
        }
    }
?>
<? include("../w_menu.php"); ?>
</body>
</html>

```

Archivo: w_inscriptos_examen.php

```

<?
include("../includes/w_session.inc.php");
require("../adodb/adodb.inc.php");
require("../adodb/crypt.inc.php");
include("../library/std_functions.lib.php");
include("../library/validador.lib.php");

// Llama a la funcion que toma los parametros de la página.
$parametros = Cargar_parametros();

// crea la conexión a la BD
$objCon = conectar();

$strSQL = "EXECUTE PROCEDURE sp_MesasDoc('" .
$parametros["unidad_academica"] . "', '" .
$parametros["nro_inscripcion"] . "')";
$rsMesasDoc = $objCon->Execute($strSQL);

?>

<html>
<head>
<title>INSCRIPTOS</title>
</head>
<body>
    <?
        if (!$rsMesasDoc || $rsMesasDoc->EOF) {
    ?>
        No figura en ninguna mesa de examen.
    <?
        }
        else
        {
            $materia_anterior = "";
            $strTemp = $rsMesasDoc->fields[1];
            while (!$rsMesasDoc->EOF)
            {
                if ($strTemp != $rsMesasDoc->fields[1])
                {
                    $strTemp = $rsMesasDoc->fields[1];
                }
            }
    ?>
            <? if ($rsMesasDoc->fields[1] <> $materia_anterior)
                echo "<b>" . $rsMesasDoc->fields[2] . "</b><br/>";

            $materia_anterior = $rsMesasDoc->fields[1];

```

```

?>

<? echo "Turno/Mesa: " . $rsMesasDoc->fields[4] . " - " .
$rsMesasDoc->fields[5] . "<br/>" ?>
<? echo "Fecha: " . $rsMesasDoc->fields[7] . "<br/>" ?>
<?
    $strSQL = "EXECUTE PROCEDURE sp_AluInscExa('" .
$parametros["unidad_academica"] . "','" . $rsMesasDoc->fields[1] .
"', '" . $rsMesasDoc->fields[3] . "','" . $rsMesasDoc->fields[4] .
"', '" . $rsMesasDoc->fields[5] . "','" . $rsMesasDoc->fields[6] .
"');"
    $rstotal = $objCon->Execute($strSQL);
    if ($rstotal)
    {
        $libres = 0;
        $regulares = 0;
        echo "Cnt Insc: " . count($rstotal->GetArray()) .
"<br/>";
        while (!$rstotal->EOF)
        {
            if (substr(ucfirst($rstotal->fields[16]),0,1) ==
"R")
                $regulares = $regulares + 1;
            if (substr(ucfirst($rstotal->fields[16]),0,1) ==
"L")
                $libres = $libres + 1;
            $rstotal->MoveNext();
        }
        echo "Reg: " . $regulares . " Libres: " . $libres .
"<br/>";
    }

?> <br/>
<?
    $rsMesasDoc->MoveNext();
}
}
?>
<? include("../w_menu.php"); ?>
</body>
</html>

```

Bibliografía y sitios en Internet

Bibliografía

- 4th Generation Wireless Networks. Jaime Sanchez García. 2005.
- Antonio Meucci, l'inventore del telefono. Notiziario Técnico Telecom Italia. 2003.
- BlackBerry Mobile Data System, Technical Overview. Research In Motion Limited. 2006.
- Bluetooth Protocol Architecture. Riku Mettala. 1999.
- Cellular Communications Whitepaper. The International Engineering Consortium. 2001.
- Creating a BREW Application from Scratch. Qualcomm Incorporated. 2002.
- Datasheet Java 2 Platform, Micro Edition. Sun Microsystems Inc. 2002.
- Desarrollo informático colaborativo en el sistema universitario: La experiencia SIU-Guaraní. Luján Gurmendi, Ricardo Williams. 2006.
- Designing Wireless Enterprise Applications Using Java Technology. Sun Microsystems. 2002.
- Global System for Mobile Communications (GSM). The International Engineering Consortium. 2001.
- Integrating WAP and Java Technology: The End -to- End Story. Dana H. Myers, Sun's 2001 JavaOne Conference, San Fransisco, USA. 2001.
- Introducción a la Telefonía Celular. Renzo Mare. 2003.
- J2ME Java 2 Micro Edition. Agustín Froufe Quintas, Patricia Jorge Cárdenas. 2004.
- J2ME Tutorial. Vikram Goyal, Sun Microsystems. 2005.
- Martin Cooper Biography. ArrayComm Inc. 2003.
- Mobile Application Developement with NetBeans Mobility Pack. Kiran Patel, Madrid Sun Tech Days 2005-2006. 2006.
- Palm Developer Guide, Palm OS Platform. Palm OS. 2007.
- Profesional WAP. Arehart Charles, Chidambaram Nirmal, Guruprasad Shashikiran, Homer Alex, Howell Ric, Kasiollai Stephan. 2001.
- SIU-Guaraní - Curso Guaraní3W. SIU, Secretaría de Políticas Universitarias, Ministerio de Educación, Ciencia y Tecnología. 2006.
- UAProf An overview. Mikael Nilsson. 2003.
- UMTS Protocols and Protocol Testing. The International Engineering Consortium. 2001.
- WAP 2.x architecture - Features, services and functions. Peter Arnby, Johan Hjelm and Peter Stark. 2001.
- WAP Guía Práctica. Dornan Addy, 2001.
- Wireless Application Protocol WAP 2.0 Technical Specifications. Wireless Application Protocol Forum, Ltd. 2002.

- Wireless Application Protocol WAP 2.0 Technical White Paper. Wireless Application Protocol Forum, Ltd. 2002.
- Wireless Short Message Service (SMS). The International Engineering Consortium. 2001.

Sitios de referencia

- Alexander Graham Bell Institute: bell.uccb.ns.ca
- All Net Devices: www.allnetdevices.com
- AnywhereYouGo.com: www.anywhereyougo.com
- Benq Siemens: www.benq-siemens.com
- BlackBerry: www.blackberry.com
- BREW (Qualcomm): brew.qualcomm.com
- Canadian Biography Online: www.biographi.ca
- Canalys: www.canalys.com
- Comitato Guglielmo Marconi: www.radiomarconi.com
- CTI: www.cti.com.ar
- Ericsson: www.ericsson.com
- Garibaldi-Meucci Museum: www.garibaldimeuccimuseum.org
- Gartner: www.gartner.com
- IEEE Computer Society: www.computer.org
- Indec: www.indec.gov.ar
- Java ME (Java): java.sun.com
- Motorola: www.motorola.com
- Movistar Argentina: www.movistar.com.ar
- Museo dell'informatica e del calcolo scientifico: www.museoai.ca.it
- Nobel: www.nobelprize.org
- Nokia: www.nokia.com
- Telefónica Argentina: www.telefonica.com.ar
- Tesla Museum: www.tesla-museum.org
- Tesla Society: www.teslasociety.com
- The International Engineering Consortium (IEC): www.iec.org
- O'Reilly: www.oreilly.com
- Open Mobile Alliance (Wap Forum): www.wapforum.com
- OpenWave: www.openwave.com
- Onti: www.sgp.gov.ar/contenidos/onti

- Palm: www.palm.com (access)
- Personal: www.personal.com.ar
- Research In Motion: www.rim.com
- Secretaría de Comunicaciones: www.secom.gov.ar
- SIU: www.siu.edu.ar
- Symbian: www.symbian.com
- Windows Mobile (Microsoft): www.microsoft.com
- World Wide Web Consortium (W3C): www.w3c.org
- Yospace: www.yospace.com

Información de actualidad y noticias

- 3G Today: www.3gtoday.com
- BBC News: news.bbc.co.uk
- Celularis: www.celularis.com
- CincoDías.com: www.cincodias.com
- Clarín: www.clarin.com
- CNET News.com: news.com.com
- Datafull: www.datafull.com
- IBL News: www.iblnews.com
- InfoBae profesional: www.infobaeprofesional.com
- Internet.com: www.internet.com
- La Nación: www.lanacion.com
- Movilsur: www.movilsur.com
- Página 12: www.pagina12.com.ar
- SearchMobileComputingcom: searchmobilecomputing.techtarget.com
- Sun Mobility Developer Newsletter: developers.sun.com
- Syscon Media: www.syscon.com
- Wireless Business Technology: www.wbt2.com
- ZD Net: zdnet.co.uk