

**DESARROLLO DE UNA APLICACIÓN DE CONSULTA BIBLIOGRÁFICA PARA LA
BIBLIOTECA CENTRAL A TRAVÉS DE DISPOSITIVOS MÓVILES BASADA EN LA
ARQUITECTURA DE SERVICIOS WEB XML**

**ALEX RICARDO MÁRQUEZ CORONEL
OSCAR DARÍO MORALES CÁRDENAS**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICOMECAICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA**

2006

**DESARROLLO DE UNA APLICACIÓN DE CONSULTA BIBLIOGRÁFICA PARA LA
BIBLIOTECA CENTRAL A TRAVÉS DE DISPOSITIVOS MÓVILES BASADA EN LA
ARQUITECTURA DE SERVICIOS WEB XML**

**ALEX RICARDO MÁRQUEZ CORONEL
OSCAR DARÍO MORALES CÁRDENAS**

**Trabajo de grado para optar al título de
Ingeniero de Sistemas**

Director

**Sergio Castillo Castelblanco
Ingeniero de Sistemas, Profesor EISI**

Co-Director

**Benjamín Augusto Pico Merchán
Ingeniero de Sistemas, Profesional División de Servicios de Información**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICOMECAICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2006**

**“Si haces lo que siempre has hecho,
Nunca llegarás más allá de donde siempre has llegado”**

Anónimo

CONTENIDO

| | Página |
|--|--------|
| ETAPA 1: PRELIMINARES | |
| A. Lista de Anexos | VII |
| B. Lista de Esquemas | VIII |
| C. Lista de tablas | X |
| D. Lista de figuras | XI |
| E. Glosario | XIV |
| INTRODUCCIÓN | 1 |
| | |
| 1. FUNDAMENTOS TEÓRICOS | |
| | |
| 1.1 TECNOLOGÍA INALÁMBRICA | 3 |
| 1.1.1 Dispositivos móviles | 5 |
| 1.1.2 Dispositivos móviles existentes | |
| 1.1.2.1 Teléfono móvil | 6 |
| 1.1.2.2 Asistente Personal Digital (PDA) | 7 |
| | |
| 1.2 SERVICIOS WEB XML | |
| 1.2.1 Descripción de los Servicios Web XML | 9 |
| 1.2.2 Estructura de los Servicios Web XML | 11 |
| 1.2.3 Consumo De Los Servicios Web Por Parte De Los Clientes | 12 |
| | |
| 1.3 XML eXtensible Markup Language (<i>lenguaje de marcado ampliable o extensible</i>) | 13 |
| | |
| 1.4 SOAP Simple Object Access Protocol (<i>Protocolo de acceso a objetos sencillos</i>) | 14 |
| 1.4.1 Estructura del Mensaje SOAP | 15 |
| 1.4.2 Ejemplo de un mensaje SOAP | 15 |
| | |
| 1.5 RPC Remote Procedure Call (<i>Llamado a procedimiento Remoto</i>) | 16 |

| | | |
|------|--|----|
| 1.6 | WSDL <i>Web Service Description Language (Lenguaje de descripción de los servicios Web XML)</i> | 17 |
| 1.7 | UDDI <i>Universal Description, Discovery and Integration (Descubrimiento, descripción, e integración Universal)</i> | 19 |
| 1.8 | Métodos de comunicación distribuida entre dispositivos | |
| | 1.8.1 Formato Propietario Binario | 20 |
| | 1.8.2 Serialización de Objetos | 21 |
| | | 22 |
| 1.9 | Servicios Web XML y la computación distribuida | |
| | 1.9.1 RMI, DCOM y CORBA | 22 |
| | | 23 |
| 1.10 | Servicios Web XML y los Dispositivos Móviles | |
| | | 24 |

2. TECNOLOGÍA EMPLEADA

| | | |
|-----|--|----|
| 2.1 | SOFTWARE DE DESARROLLO | |
| | 2.1.1 Microsoft .NET Framework | 25 |
| | 2.1.2 NetBeans 4.1 | 26 |
| | 2.1.3 J2ME (Java 2 MicroEdition) | 27 |
| 2.2 | SISTEMAS OPERATIVOS | |
| | 2.2.1 Windows XP Service Pack 2 Professional Edition | 29 |
| | 2.2.2 Windows 2003 Server Standard Edition (W2K3) | 30 |
| | 2.2.3 Windows Mobile 2003 | 32 |
| | 2.2.4 Linux SUSE 10 | 33 |
| | 2.2.5 Symbian OS | 34 |
| 2.3 | HARDWARE | 35 |
| | 2.3.1 Equipos para el desarrollo de las aplicaciones | 35 |
| | 2.3.2 Equipos para la implantación | 36 |
| 2.4 | MANEJO Y ALMACENAMIENTO DE BASES DE DATOS | 38 |

| | | |
|--------------------------------|---|-----------|
| 2.4.1 | Informix | 38 |
| 2.4.2 | SqlServerCE | 39 |
| 2.5 | SERVIDOR WEB | |
| 2.5.1 | Conceptos de Servidor Web | 40 |
| 2.5.2 | Servidor IIS (Internet Information Server) | 41 |
| 3. | METODOLOGÍA PARA EL DESARROLLO DEL SOFTWARE | 42 |
| 3.1 | Escogiendo la metodología de desarrollo | 42 |
| 3.2 | Metodología de Programación Extrema | 43 |
| 3.3 | El Modelo de desarrollo de la aplicación | 46 |
| ETAPA 2: DESARROLLO | | |
| 4.1 | Definición del problema | 47 |
| 4.1.1 | Síntomas, causas y consecuencias del problema | 48 |
| 4.2 | Objetivos general y específicos | 49 |
| 4.3 | Justificación y alcance del plan propuesto | 50 |
| 4.4 | Viabilidad e Impacto | 51 |
| 4.5 | Especificación y análisis de requerimientos | |
| 4.5.1 | Requerimientos de la comunidad | 53 |
| 4.6 | Implementación | 54 |
| 4.6.1 | Vista Conceptual | 54 |
| 4.6.2 | Vista Física | 57 |
| 4.6.3 | Vista lógica | 60 |
| 4.7 | Diseño detallado de las aplicaciones | |
| 4.7.1 | Esquema de funcionamiento de los módulos de Consulta (Clientes) | 62 |
| 4.7.2 | Esquema de funcionamiento de los módulos de Configuración del Cliente | 66 |
| 4.7.3 | Esquema de funcionamiento de la Aplicación Servidor | 68 |
| 4.7.4 | Diagrama Entidad-Relación de la base de datos de biblioteca | 70 |

| | |
|--|----|
| 4.7.5 Diagrama Entidad-Relación de la base de datos del Cliente Windows Mobile | 72 |
| 4.7.6 Diagrama Entidad-Relación de la base de datos del cliente J2ME | 73 |
| 4.7.7 Vista a las Pantallas de la aplicación cliente | 74 |
| 4.8 Pruebas Técnicas | 75 |

ETAPA 3: CONCLUSIÓN

| | |
|---|----|
| 5. Conclusiones | 77 |
| 6. Recomendaciones | 79 |
| 7. Bibliografía | 80 |
| 7.1 Sitios Web disponibles para consulta de información | 83 |
| 8. Anexos | 84 |

A. LISTA DE ANEXOS

Anexo 1 - Instalación y configuración del ambiente de desarrollo

Anexo 2 - Instalación y configuración del Servidor IIS

Anexo 3 - Crear un servicio Web XML en .NET para servidores IIS

Anexo 4 - Manejo de los objetos session y application

Anexo 5 - Atributos aplicables a los servicios Web XML

Anexo 6 - Implementar Servicios Web XML en el Servidor

Anexo 7 - Probar un servicio Web XML desde el explorador

Anexo 8 - Acceder a la descripción del servicio Web XML

Anexo 9 - Controlar e iniciar excepciones en los servicios Web XML

Anexo 10 - Como dar visibilidad al servicio Web XML

Anexo 11 - Como los clientes buscan los servicios Web XML

Anexo 12 - Seguridad en los servicios Web XML

Anexo 13 - Crear el programa Cliente de Windows Mobile del servicio Web XML del servidor de Biblioteca

Anexo 14 - Instalar el programa Cliente de Windows Mobile en el dispositivo Móvil compatible con Pocket PC

B. LISTA DE ESQUEMAS

ESQUEMA No 1, Esquema de diseño de los Servicios Web XML

ESQUEMA No 2, Ejemplo de un documento XML, donde se muestra en este formato la descripción de las características de un libro

ESQUEMA No 3, Estructura de un mensaje SOAP

ESQUEMA No 4, Ejemplo de una petición SOAP a un servicio Web XML

ESQUEMA No 5, Ejemplo de la respuesta SOAP a la anterior petición del cliente

ESQUEMA No 6, Esquema donde se muestra la posición lógica del servidor UDDI

ESQUEMA No 7, Esquema donde se muestran las principales variables de desempeño de los métodos existentes para comunicar dispositivos

ESQUEMA No 8, Estructura de KSOAP

ESQUEMA No 9, Diagrama de conjuntos de la plataforma Java

ESQUEMA No 10, Esquema de seguridad basado en Corta-Fuegos de Windows 2003 Server

ESQUEMA No 11, Esquema que muestra la disposición lógica del Servidor Web en una red local LAN

ESQUEMA No 12, Disposición del Servidor IIS en un equipo personal con Windows como sistema operativo

ESQUEMA No 13, Modelo de desarrollo de la Metodología de Programación Extrema

ESQUEMA No 14, Vista Conceptual de la aplicación, en el se muestran los diferentes componentes que intervienen, en la parte superior se encuentran los servidores de

Biblioteca, Base de datos y Servidor Web, y en la parte inferior se encuentra el componente cliente de la aplicación, la flecha bidireccional central es el medio que usan para comunicarse, en este caso Internet y la red Celular

ESQUEMA No 15, Vista física de la aplicación, Esta vista muestra la distribución física de los equipos que conforman la aplicación. De arriba hacia abajo se cuenta desde el servidor donde reside el servicio Web XML hasta el cliente del servicio, representados por la PDA y el Celular

ESQUEMA No 16, Diagrama lógico de los métodos existentes en el servicio Web XML

ESQUEMA No 17, Diagrama de métodos del programa cliente para el SO Windows CE

ESQUEMA No 18, Desarrollo del Flujo de consulta bibliográfica de las aplicaciones en los diferentes clientes

ESQUEMA No 19, Diagrama de flujo del programa cliente J2ME en el diseñador de aplicaciones JavaMicroEdition de NetBeans

ESQUEMA No 20, Esquema de las tablas más relevantes de la base de datos del servidor de Biblioteca

ESQUEMA No 21, Esquema de la base de datos local del cliente Windows Mobile

ESQUEMA No 22, Diagrama Entidad-Relación de la base de datos que se creó para el cliente J2ME

C. LISTA DE TABLAS

TABLA No 1, Semejanzas entre Servicios Web XML y CORBA

TABLA No 2, La tabla recoge esquemáticamente las principales diferencias de las metodologías ágiles con respecto a las tradicionales (no ágiles). Estas diferencias que afectan no sólo al proceso en sí, sino también al contexto del equipo así como a su organización.

TABLA No 3, Tabla que muestra los síntomas causas y consecuencias del problema planteado

TABLA No 4, Atributos que se le aplican a la definición de los servicios Web XML para cambiar su desempeño

TABLA No 5, Excepciones que son lanzadas al momento de consumir los servicios Web XML por parte de los clientes

TABLA No 6, Opciones de autenticación en los servicios Web XML

D. LISTA DE FIGURAS

Figura 1, Ejemplos de dispositivos móviles

Figura 2, Modelos de celulares disponibles en el mercado

Figura 3, Modelos de PDA's disponibles en el mercado

Figura 4, Participación en el mercado de las PDA's por parte de los fabricantes

Figura 5, Esquema de funcionamiento básico de los Servicios Web XML

Figura 6, Guía básica de los Servicios Web: elementos de Conexión, descripción y descubrimiento

Figura 7, Proceso de serialización y deserialización que sufren tanto la petición del cliente como la respuesta por parte del usuario del servicio Web XML

Figura 8, Logo del entorno de desarrollo del Visual Studio .NET 2003

Figura 9, Logo del grupo de desarrollo de NetBeans

Figura 10, Logo del sistema operativo Windows XP

Figura 11, Entorno gráfico del SO Windows Mobile 2003

Figura 12, Logo del sistema operativo Linux SUSE, de la empresa Novell

Figura 13, Symbian UIQ 7.0, la interfaz de Symbian para los móviles de última generación

Figura 14, Modelo de desarrollo del Prototipado Clásico

Figuras 15-16, Imágenes correspondientes a la interfase del cliente J2ME, la primera imagen es el listado de resultados después de haber realizado la consulta, la segunda imagen muestra los detalles del libro que se escogió en el listado de libros

Figuras 17-18, Imágenes correspondientes a la interfase del cliente Windows Mobile

Figura 19, Pantalla de consulta del cliente ASP.NET del servicio Web XML

Figura 20, Pantallas de la aplicación cliente donde el usuario puede cambiar varios parámetros de la aplicación, tales como la dirección IP del Servidor de biblioteca, la pantalla de la izquierda corresponde al cliente J2ME, mientras que el de la derecha es la del cliente de Windows Mobile

Figura 21, Pantalla donde se muestran los detalles del libro con la opción de guardar resaltada (cliente Windows Mobile)

Figura 22, Pantalla principal de la aplicación cliente, en ella se muestran los diferentes menús de acceso a la aplicación

Figura 23, de Izquierda a derecha. Formularios que corresponden a la búsqueda general de la aplicación cliente Windows Mobile (Búsqueda General, Resultado, Detalles)

Figura 24, Disposición de las páginas que corresponde a la aplicación que reside en el servidor de desarrollo, en ella se ven los diferentes componentes de la aplicación servidor, el servicio Web XML tiene como nombre buscador.asmx, el archivo descriptor buscador.wsdl y el archivo de descubrimiento tiene el nombre de buscador.disco

Figura 25, Vista de la ejecución del servicio Web XML desde el explorador del servidor

Figura 26, Vista de la ejecución de un método del servicio Web XML en el explorador del servidor

Figura 27, Vista de la respuesta en formato XML que el servidor le enviaría al cliente cuando este lo invoque con los mismos parámetros con los cuales se le hizo la solicitud

Figura 28, Vista del archivo .WSDL (descripción del servicio Web XML) desde el explorador del Servidor

Figura 29, Directorio donde se muestran servicios Web XML disponibles en Internet

Figura 30, Imagen que muestra en un programa cliente la localización en el proyecto donde se agrega la referencia Web al servicio Web en el entorno de .NET

Figura 31, Pantalla que muestra los métodos Web que posee el servicio Web al cual se le esta agregando la referencia

Figura 32, Pantalla que muestra como crear un objeto a partir de la clase creada de la referencia al servicio Web XML

E. GLOSARIO¹

Ancho de Banda: Término que se usa para determinar la capacidad de un medio para transportar datos, sea de forma análoga o digital, el ancho de banda es la anchura, medida en Hercios, del rango de frecuencias en el que se concentra la mayor parte de la potencia de la señal.

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 son:

- Facilitar las comunicaciones entre equipos móviles y fijos.
- Eliminar cables y conectores entre éstos.
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre nuestros equipos personales.

Byte: Se describe como la unidad básica de almacenamiento de información, generalmente equivalente a ocho bits, pero el tamaño del byte depende del código de caracteres o código de información en el que se defina.

CDMA: Code division multiple access (*Acceso Multiplexado por División de código*)

La *multiplexación por división de código* o CDMA es un término genérico que define una interfaz de aire inalámbrica basada en la tecnología de espectro extendido (spread spectrum). Para telefonía celular, CDMA es una técnica de acceso múltiple especificada por la TIA como IS-95.

CORBA: *Common Object Request Broker Architecture*, es un protocolo de comunicaciones y mecanismos necesarios para permitir la interoperabilidad entre diferentes aplicaciones escritas en diferentes lenguajes y ejecutadas en diferentes plataformas.

Computación Distribuida: La computación distribuida, computación en grilla o informática en rejilla, es un nuevo modelo para resolver problemas de computación

¹ Tomado de la enciclopedia en línea <http://wikipedia.org>

masiva utilizando un gran número de ordenadores organizados en racimos incrustados en una infraestructura de telecomunicaciones distribuida.

Cliente: Es un ordenador que accede a recursos y servicios brindados por otro llamado Servidor, generalmente en forma remota.

Dispositivo: Es una entidad de red que es capaz de enviar y recibir paquetes de información y son identificados por una única dirección. Un dispositivo puede servir a un número de clientes actuando como servidor y ser al mismo tiempo un cliente para otro servidor.

EDGE: *Enhanced Data rates for GSM Evolution (Velocidades de datos ampliadas para la evolución de GSM)*. Es una tecnología de la telefonía móvil celular, que actúa como puente entre las redes 2G y 2.5G. EDGE se considera una evolución del GPRS (General Packet Radio Service). Esta tecnología funciona con redes TDMA y su mejora (GSM).

GPRS: *General Packet Radio Service (Servicio General de Paquetes de Radio)*

Es considerada la generación 2.5, entre la segunda generación (GSM) y la tercera (UMTS). Proporciona altas velocidades de transferencia de datos (especialmente útil para conectar a Internet) y se utiliza en las redes GSM.

GSM: *Global System for Mobile Communications (Sistema Global para las Comunicaciones Móviles)* Formalmente conocida como "Group Special Mobile" (GSM, Grupo Especial Móvil) es un estándar de comunicación mundial para teléfonos móviles digitales. El estándar fue creado por la CEPT² y posteriormente desarrollado por ETSI³ como un estándar para los teléfonos móviles europeos, con la intención de desarrollar una normativa que fuera adoptada mundialmente. El estándar es abierto, no propietario y evolutivo (aún en desarrollo). Es el estándar predominante en Europa, así como el mayoritario en el resto del mundo (alrededor del 70% de los usuarios de teléfonos móviles del mundo en 2001 usaban GSM).

² CEPT, *Conferencia Europea de Administraciones de Correos y telecomunicaciones*. organismo internacional que agrupa a las entidades responsables en la administración pública de cada país europeo de las políticas y la regulación de las comunicaciones

³ ETSI, *European Telecommunications Standard Institute*, Instituto de Estándares de Telecomunicaciones Europeos, Es una organización de estandarización de la industria de las telecomunicaciones de Europa

Historia de Usuario: Son la técnica utilizada para especificar los requisitos del software en el desarrollo de la metodología de programación Extrema.

IDE: Un IDE es un ambiente 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.

IRDA: Es la comunidad que define los estándares y especificaciones técnicas que rigen la comunicación inalámbrica vía rayos infrarrojos.

ROAMING: Es un concepto utilizado en comunicaciones inalámbricas que está relacionado con la capacidad de un dispositivo para moverse de una zona de cobertura a otra. Es una palabra de procedencia inglesa que significa *vagar* o *rondar*.

TDMA: *Time Division Multiple Access (Acceso multiplexado por división de tiempo)*

TDMA es una tecnología basada en ondas de radio que se usa como medio para el intercambio de información, TDMA distribuye las unidades de información en espacios alternantes de tiempo proveyendo acceso múltiple a un reducido número de frecuencias. TDMA es una tecnología inalámbrica de segunda generación que brinda servicios de alta calidad de voz y datos.

Tecnología inalámbrica: Tecnología inalámbrica hace referencia al desarrollo de tecnologías que permitan entablar comunicaciones en la ausencia de hilos que enlacen los equipos que establecen la conexión.

Teléfono móvil: El teléfono móvil también conocido como celular, es un dispositivo de comunicación electrónico de onda corta, puede ser digital o analógico, con las mismas capacidades básicas de un teléfono alámbrico convencional.

UDDI: *Universal Description, Discovery and Integration (Descubrimiento, descripción, e integración Universal)* UDDI es un registro público diseñado para alojar información acerca de los servicios Web en forma estructurada.

UMTS: *Universal Mobile Telecommunications System (Sistema de Telecomunicaciones Móviles Universales)*. La tecnología UMTS es el sistema de telecomunicaciones móviles de tercera generación, que evoluciona desde GSM pasando por GPRS.

SOAP: *Simple Object Access Protocol (Protocolo de acceso a objetos sencillos)* El protocolo SOAP se basa en XML y describe un formato de mensajería para la comunicación entre equipos.

SYMBIAN: Symbian OS es un sistema operativo de código abierto para celulares y dispositivos móviles.

WI-FI: WI-FI o red de área local inalámbrica (WLAN) es una red de TI (tamaño intermedio) que utiliza la frecuencia de radio 802.11a, 802.11b o 802.11g en lugar de cables y permite realizar diversas conexiones inalámbricas a Internet.

XML: Es un conjunto de reglas para definir etiquetas semánticas que organizan un documento en diferentes partes.

TÍTULO:

DESARROLLO DE UNA APLICACIÓN DE CONSULTA BIBLIOGRÁFICA PARA LA BIBLIOTECA CENTRAL A TRAVÉS DE DISPOSITIVOS MÓVILES BASADA EN LA ARQUITECTURA DE SERVICIOS WEB XML⁴

AUTORES:

MORALES CÁRDENAS, Oscar Darío

MÁRQUEZ CORONEL, Alex Ricardo

PALABRAS CLAVES:

XML, Servicios Web, SOAP, UDDI, RPC, Dispositivos Móviles, celulares, PDA, buscador, biblioteca

DESCRIPCIÓN:

Para continuar con el perfil de investigación y desarrollo de nuevas tecnologías con que cuenta la Universidad, se ha planteado este proyecto de consulta bibliográfica para la biblioteca central. Tesis que busca llevar a la plataforma móvil⁵ a través de los *Servicios Web XML* el servicio de prestación de libros con que cuenta la Universidad actualmente. Mejora que puede llegar a ser muy útil y beneficiosa para un excelente apoyo al desarrollo de la academia.

Adicionalmente, se busca la profundización el área de una de las formas de integración de plataformas heterogéneas más importantes que se han creado hasta la fecha y que representan una tecnología creciente como lo son los Servicios Web XML.

Con este proyecto se busca otorgarle a la comunidad universitaria⁶ la libertad de poder consultar el archivo de materiales bibliográficos que tiene disponible la biblioteca central en sus instalaciones, sin que sea necesario desplazarse físicamente hasta ellas. Se ha transformado en una necesidad, el hecho de poder acceder en tiempo real a través de un dispositivo móvil, la descripción de los libros que tiene disponible la biblioteca en temas relacionados con las diferentes carreras que se cursan en la universidad, esta información permite a los usuarios de la biblioteca ahorrar tiempo, al admitir que la persona que realiza la consulta tenga conocimiento de primera mano y desde virtualmente cualquier lugar⁷. Asimismo se puede saber si el libro, tesis o revista que esta buscando se encuentra en la biblioteca de la universidad y de su disponibilidad para ser sacado en calidad de préstamo.

⁴ Proyecto de grado Modalidad investigación

⁵ Plataforma que da soporte a dispositivos con acceso inalámbrico a redes de datos

⁶ La comunidad Universitaria esta compuesta además de estudiantes y profesores, de empleados, visitantes, estudiantes de otras universidades y colegios, y de la comunidad en general

⁷ Posición geográfica en la cual el usuario tenga acceso a la red Internet de forma inalámbrica

TITLE:

Development of an application of bibliographical consultation for the central library through movable devices based on the architecture of XML Web Services⁸

AUTHORS:

MORALES CÁRDENAS, Oscar Darío

MÁRQUEZ CORONEL, Alex Ricardo

KEY WORDS:

XML, Web Services, SOAP, UDDI, RPC, Mobile devices, cell phones, PDA, search, library

DESCRIPTION:

In order to continue with the development and investigation profile of new technologies which the university has, the consultation of bibliographical resources for the central library project has been proposed. This thesis points to take the mobile platform⁹ through the *XML Web Services* the book lending service which is provided by the university actually. This improvement might become really useful y beneficial for an excellent support to the academy.

Additionally, it's looked for the deepening in the area of one of the most important ways for heterogeneous platform integration that have been created so far and represents a growing technology such as the *XML Web Services*.

This project tries to give to the university community¹⁰ the freedom to find in the bibliographical resources database available in the central library facilities, without being necessary to get physically there. The power to access in real time by a mobile device has become essential, this information allows library users save time, because it permits the person search on the database so it can have first hand results and virtually from any place¹¹. Furthermore, the person might know if the book, thesis o review the person is looking for is placed in the university library and the availability in order to be loaned.

⁸ Graduation Project in the category of investigation

⁹ Platform that gives device support with wireless access to data networks

¹⁰ The University community is compose in addition by students and teachers, employees, visitors, other universities and schools students, and the community in general

¹¹ Any geographical location where the user have wireless access to Internet

INTRODUCCIÓN

Los conocimientos plasmados en este libro se centran en la investigación de los mecanismos necesarios para consultar la información bibliográfica de la Universidad Industrial de Santander a modo de servicio, de tal manera que esta se pueda acceder desde dispositivos móviles. Se considera este proyecto uno de los pioneros en el área, al emplear y documentar una de las tecnologías más revolucionarias creadas en el área de la informática y los negocios en el mundo que está empezando a tenerse en cuenta en Colombia. De la misma forma, el trabajo se proyecta hacia los dispositivos móviles de uso común, tratando de explotar características actualmente subvaluadas y subutilizadas como lo son la computación y el Internet Móvil.

Este trabajo de grado representa el primer paso adelante tanto de la Universidad Industrial de Santander como de la región hacia los negocios vía dispositivos móviles (m-Business) que simbolizan el futuro del comercio a nivel mundial. Horizonte ya visible en países industrializados que han desarrollado herramientas como m-Wallet de Motorola, permitiendo realizar transacciones bancarias, compras en línea, entre otras características. Aplicaciones como esta no están lejos de ser también hechas en Colombia.

En cuanto al desarrollo, Inicialmente se optó por usar el lenguaje Java como herramienta para la construcción de la aplicación para los dispositivos móviles y el lenguaje C# de la tecnología .NET para crear la aplicación residente en el servidor con sistema operativo Microsoft. ¿Pero de que forma comunicar estas dos plataformas?, Existen muchas posibilidades para realizar dicha interacción, cada una de ellas con sus respectivas ventajas, las más relevantes del caso son explicadas en capítulos posteriores en este libro; cabe aclarar que es posible crear además un protocolo propio para la comunicación entre las dos plataformas, pero se verían comprometidos la escalabilidad, portabilidad y el mantenimiento de la aplicación global.

Actualmente, Se ha producido una fuerte tendencia hacia los estándares para el intercambio de datos entre diferentes plataformas, esta tendencia deriva de una visión que está evolucionando constantemente hacia un novedoso paradigma informático denominado Servicios Web XML, el cual busca la perfecta integración entre ambientes de software heterogéneos y que está respaldado por las grandes empresas del

software como Microsoft, IBM, Sun Microsystems, Oracle, entre otras. Asimismo, ha llegado a imponerse en las aplicaciones B2C¹² y B2B¹³ reemplazando de esta forma estándares empresariales tradicionales como EDI¹⁴, ya que los Servicios Web están basados en un lenguaje universal y auto descriptivo como XML, y que se presentan como una alternativa más simple y menos costosa.

La universidad como toda institución de servicio público, debe permanecer necesariamente en contacto con otras instituciones. Para el caso en que la universidad requiera abrir las puertas de su información (sea bibliográfica o no) a otras entidades¹⁵, exponer los componentes internos a entidades externas para que la información se acceda directamente, implicaría problemas de seguridad de grandes proporciones, debido a que necesariamente se compartirían bases de datos, contraseñas, esquemas de seguridad, etc. Es en este momento cuando es recomendable revelar la información de manera indirecta, segura, e independiente del tipo de cliente que se trate, en lo concerniente a características como su lenguaje de programación, sistema operativo, arquitectura de red, entre otras. Por estas razones, explicadas con mayor detalle a lo largo de este documento, se hace indispensable el uso del avance tecnológico de la arquitectura orientada a servicios, conocido como los servicios Web XML.

¹² B2C Business to Costumer (Negocio al cliente).

¹³ B2B Business to Business (Negocio a Negocio).

¹⁴ EDI Electronic Data Exchange (Intercambio Electrónico de Datos).

¹⁵ Entendiendo por entidades a personas, alumnos, bibliotecas, empresas, etc.

ETAPA 1

PRELIMINARES

1. FUNDAMENTOS TEÓRICOS

El marco teórico establece los conceptos teóricos involucrados en el desarrollo de este proyecto, define los conceptos necesarios para un primer acercamiento a las tecnologías involucradas y permite que los lectores construyan las bases conceptuales necesarias para el entendimiento de los temas involucrados en el proyecto.

1.1 TECNOLOGÍA INALÁMBRICA

El termino "tecnología inalámbrica" es el concepto que hace referencia al desarrollo de tecnologías que permiten entablar comunicaciones en la ausencia de hilos que enlazan los equipos que establecen la conexión. Actualmente esta tecnología ha permitido desarrollar la comunicación a través de ondas de radio frecuencia de bajo poder y también a través de medios ópticos denominados infrarrojos.

Los medios de comunicación inalámbricos han sufrido cambios sustanciales en sus protocolos de transmisión a través del tiempo, estos cambios de tecnología se agruparon en lo que se conoce como generaciones de la comunicación inalámbrica.

Iniciando en 1979 con la primera generación (1G) la primitiva comunicación celular hizo su salto a esta plataforma, con velocidades que no superaban los 2400 baudios, totalmente análoga, limitada a transmitir voz, y con un esquema de seguridad prácticamente inexistente, fue el primer intento para la estandarización de los protocolos de la naciente telefonía celular.

Después de 11 años La comunicación móvil terrestre es el servicio de comunicación más difundido y con el mayor número de usuarios alrededor del mundo, en ese momento la telefonía celular estaba lista para hacer su salto a la plataforma digital, con protocolos de transmisión como GSM, CDMA, TDMA logro transmitir en un mismo canal voz y datos, o únicamente datos a mayores velocidades. El tamaño de los terminales se hace cada vez más pequeño, las coberturas se extienden, y se empiezan a transmitir datos, aunque a velocidades muy pequeñas.

Con la segunda generación se introduce al mercado el envío de mensajes SMS¹⁶, además de permitir la compatibilidad entre las distintas redes con lo cual se empieza a mejorar la calidad del servicio. Al existir varios estándares de transmisión de datos, cada sector del mundo sea por inconvenientes técnicos o estrategias comerciales escoge su propia tecnología de segunda generación, en este caso GSM se implantó en Europa y en otros países del resto del mundo. TDMA y CDMA en EEUU, mientras que PDC en Japón.

En la actualidad, se están empezando a desplegar sistemas de lo que se ha denominado generación 2,5 (HSCSD, GPRS, EDGE)¹⁷ que harán de puente entre los de segunda generación y la telefonía móvil de tercera generación (UMTS¹⁸). Esta última responde a un intento de estandarizar las comunicaciones móviles a nivel mundial, aunque ya están empezando a surgir pequeñas diferencias entre EEUU y el resto. Como tercera generación ofrecerá grandes velocidades de conexión, por lo que se espera que se convierta en la forma más habitual de acceso a Internet. Permitiendo la transmisión de todo tipo de comunicaciones inalámbricas que soporten: voz, datos, imágenes, vídeo, radio,...

Algunos sistemas 2,5 (GPRS, EDGE) introducen la conmutación de paquetes en la telefonía móvil, es decir, la comunicación se produce al "estilo" Internet. La información se divide en trozos o paquetes, que siguen caminos diferentes hasta alcanzar el destino. GPRS alcanzará los 115 Kbps, mientras que EDGE los 384 Kbps. Además, EDGE permitirá a los operadores de GSM y TDMA integrar en sus redes actuales este nuevo sistema.

En investigación se encuentra la cuarta generación, aunque aún no tiene nombre específico se espera que alcance velocidades de hasta 50 veces las actuales, su implementación esta pensada para el año 2010.

¹⁶ SMS, Short Message Service, Servicio de Mensajes Pequeños, o también conocidos como mensajes de texto

¹⁷ HSCSD, *High Speed Circuit-Switched Data*, Circuito Intercambiado de Datos de alta Velocidad
GPRS, *General Packet Radio Service*, Servicio General de Paquetes de Radio

EDGE, *Enhanced Data Rates for GSM Evolution*, Tasas de Datos Realzadas para la Evolución de GSM

¹⁸ UMTS, *Universal Mobile Telecommunications System*, es el sistema de telecomunicaciones móviles de tercera generación, que evoluciona desde GSM

1.1.1 Dispositivos móviles

El concepto de dispositivo móvil en el común de la gente aún no se ha estandarizado a una única definición, los usuarios conocen como dispositivo móvil toda esa amplia gama de aparatos inalámbricos que pueden comprar en las tiendas electrónicas, esta distorsión del concepto de dispositivo móvil esta alimentado en gran medida por los diferentes términos que existen en el idioma ingles para describir estos dispositivos. Existen términos tales como "information Device" (dispositivo de Información), "information appliance" (Aplicación de Información), "consumer electronic" (Cliente Electrónico), "embedded device" (Dispositivo Empotrado) o "small device" (Dispositivo Pequeño) para nombrar algunos ejemplos.

Otro concepto importante es el término "Wireless" (Inalámbrico). Un dispositivo inalámbrico es aquel que es capaz de intercambiar información con otro dispositivo en la ausencia de hilos dando la impresión de estar conectados a una red, por ejemplo los teléfonos celulares y los modelos más recientes de PDA's¹⁹.

Basándose en la explicación de estos importantes conceptos se puede crear una definición más aproximada de que es un dispositivo móvil.

Un dispositivo móvil es aquel elemento lo suficientemente pequeño para su transporte, conexión e intercambio de información con otros equipos en la ausencia de hilos, permitiendo hacer uso de el durante su transporte.

Las características que hacen que los dispositivos móviles sean diferentes a otros tipos de ordenadores como los de mesa, portátiles o servidores son:

- El poseer una funcionalidad realmente limitada.
- No son necesariamente extensibles o actualizables, en muchos casos la mejor opción es comprar un nuevo dispositivo.
- El precio de adquisición de un dispositivo móvil es menor al de equipo de mesa.
- Posee un número limitado de controles para interactuar con el usuario, debido a su reducido tamaño.

¹⁹ PDA, *Personal Digital Asistent*, Asistente Personal Digital, Es un computador de mano diseñado para ser transportado como agenda por el usuario, los modelos más populares son los fabricados por la empresa Palm

- Es fácil de usar, no se requieren usuarios expertos para acceder a las funciones del dispositivo.



FIGURA No 1, Ejemplos de dispositivos móviles²⁰

Algunos dispositivos móviles disponibles al público son:

- Los comunicadores de bolsillo o Beeper
- Los teléfonos móviles
- Sistemas de navegación de los automóviles
- Los organizadores y asistentes personales (PDA)

1.1.2 Dispositivos Móviles Existentes

1.1.2.1 Teléfono Móvil

El teléfono móvil también conocido como celular, es un dispositivo de comunicación electrónico de onda corta, puede ser digital o analógico, con las mismas capacidades básicas de un teléfono alámbrico convencional, posee la característica de ser portátil, al no requerir una línea alámbrica para estar conectado a la red telefónica.

La mayoría de los teléfonos actuales utilizan para conectarse a las redes de tecnología inalámbrica la vía de las ondas de radio.



²⁰ <http://www.cbc.ca/wireless/images/Palm.jpg>,
http://www.palm.com/us/images/wireless/photo_treo650.jpg
<http://www.iwm.com.mx/servicios/facil/biper.gif>

FIGURA No 2²¹, Modelos de celulares disponibles en el mercado

Actualmente el estándar de comunicación de los teléfonos convencionales es la voz. Por otra parte un teléfono móvil puede soportar adicionalmente a la voz, SMS (Short Message Service) para enviar mensajes de texto, intercambio de paquetes para acceder a Internet y MMS (Multimedia Message Service) para enviar y recibir fotos y videos.

Los celulares usan sistemas operativos que gestionan sus procesos básicos de comunicación y datos, entre estos se destacan Windows Mobile 5.0 (Microsoft) con el 28% y Symbian (Nokia) con el 60%, son los sistemas operativos que tiene más participación en el mercado.

1.1.2.2 PDA (Personal Digital Assistant)

Un PDA (Personal Digital Assistant o Ayudante personal digital) es un dispositivo de pequeño tamaño, móvil y versátil que le permite al usuario la posibilidad comunicarse, realizar cálculos y almacenar información.

A los PDAs también se les llama palmtops, handheld computers (ordenadores de mano) y pocket computers (ordenadores de bolsillo).

Un PDA típico puede funcionar como teléfono móvil, fax, explorador de Internet, organizador personal, GPS, etc.

La mayoría de PDAs empezaron a usarse con una especie de bolígrafo en lugar de teclado, por lo que incorporaban reconocimiento de escritura a mano. Hoy en día los PDAs pueden tener teclado y/o reconocimiento de escritura. Algunos PDAs pueden incluso reaccionar a la voz, mediante tecnologías de reconocimiento de voz.



Palm tungsten
Palm Source inc



Palm ONE
Palm Source inc



PocketPc n10
Acer

²¹ Modelos de la Empresa Nokia, líder en el mercado de Celulares

FIGURA No 3,²² Modelos de Pdas disponibles en el mercado

Apple fue una de las primeras compañías en ofrecer PDAs, pero al poco tiempo otras compañías empezaron a ofrecer productos similares, siendo actualmente la empresa Palm la dominadora del mercado.

Lo normal en el mercado tecnológico de los PDA's es diferenciarlos en función de su sistema operativo.

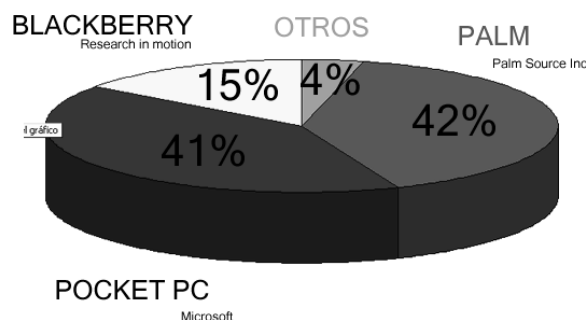


FIGURA No 4²³, Participación en el mercado de las PDA's por parte de los fabricantes

Aunque los Palm y Pocket PC son los más utilizados, los Pocket PC son de reciente creación. Los Palm, con su inicial fabricación como Palm Pilots por la compañía Palm Source Inc., llevan más tiempo en el mercado.

Los Blackberry, cuyo uso se está incrementando, son PDAs fabricados por la compañía canadiense Research In Motion (RIM) y se venden por compañías de telefonía móvil de todo el mundo.

²² Imagen tomada y modificada de <http://www.palm.com> y <http://www.acer.com>

²³ Participación en el mercado de junio de 2005

1.2 SERVICIOS WEB XML

1.2.1 Descripción de los Servicios Web XML

Un servicio Web XML es una entidad programable que proporciona un elemento de funcionalidad determinado, cuya finalidad es posibilitar intercambio de información entre diversas plataformas a través de un protocolo de comunicación estándar (SOAP) basado en XML como lenguaje de representación de datos y HTTP como protocolo de transporte, de tal manera que el intercambio sea independiente de la plataforma, sistema operativo o lenguaje de programación utilizada por el servicio y cliente (consumidor) adicionalmente que no se requiera de un agente intermediario en la comunicación.

Basándose en estos conceptos un servicio Web XML presenta las siguientes características:

Esta Basado en XML

Usando XML como la capa de representación y HTTP como transportador de datos, elimina cualquier atadura del servicio Web con respecto a la plataforma, sistema operativo o infraestructura de red que tiene el sistema.

Ligeramente Ligado

El consumidor de un servicio Web no esta directamente atado al servicio directamente, es decir, la interfase del servicio puede ser cambiada en cualquier momento sin comprometer la habilidad del cliente para consumirlo. Por otro lado, en los sistemas donde se presenta alta cohesión la lógica del servidor y del cliente esta muy ligada lo que implica que si una interfase cambia la otra también lo debe hacer.

No requiere de un agente intermedio

El intercambio de datos XML se hace del tipo servidor-cliente directamente, no interviene ningún tipo de objeto externo de tipo traductor como CORBA.

Esta disponible para ser usado por cualquier cliente en cualquier parte del mundo

Lejos de estar atado a la intranet corporativa, los servicios Web XML están disponibles para ser consumidos por cualquier cliente que este conectado a la red Internet en cualquier posición geográfica.

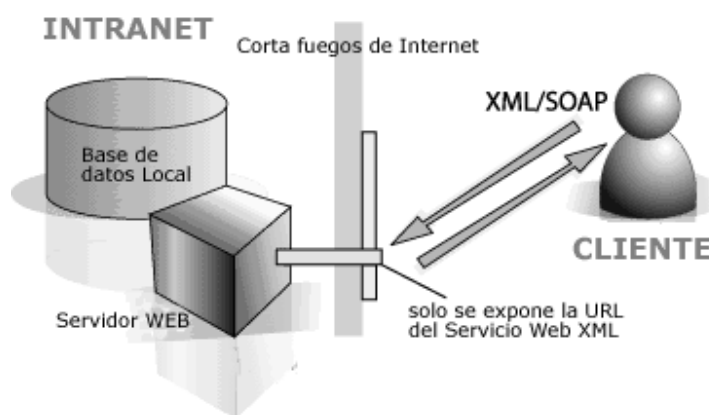


FIGURA No 5²⁴, Esquema de funcionamiento básico de los Servicios Web XML

En vez de centrarse en las posibilidades genéricas de portabilidad del código, los servicios Web XML proporcionan una solución viable para habilitar la interoperabilidad de datos y sistemas. *Los servicios Web XML utilizan representación de datos basada en XML como medio fundamental para contribuir a reducir las diferencias existentes entre entornos que utilizan distintos modelos de representación, sistemas operativos y lenguajes de programación.* Los programadores pueden crear aplicaciones que entrelacen servicios Web XML de una diversidad de orígenes, de modo similar a como utilizan tradicionalmente componentes en la creación de aplicaciones distribuidas.

Un servicio Web XML puede ser utilizado internamente por una aplicación o bien ser expuesto de forma externa en Internet por varias aplicaciones. Dado que a través de una interfaz estándar es posible el acceso a un servicio Web XML, éste permite el funcionamiento de una serie de sistemas heterogéneos como un conjunto integrado.

XML es un lenguaje soportado por la gran mayoría de las empresas del desarrollo de software, Al utilizar mensajería basada en XML como mecanismo de descripción y representación, el cliente del servicio Web XML y el proveedor de servicios Web XML sólo necesitan el conocimiento mutuo de las entradas, las salidas y la forma como deben ser escritos los mensajes en XML.

Los servicios Web XML posibilitan una nueva era para el desarrollo de aplicaciones distribuidas. Quedan así relegadas las competiciones entre modelos de objetos y entre lenguajes de programación. Cuando los sistemas se acoplan estrechamente mediante

²⁴ Imagen tomada y modificada del sitio Web <http://webservices-strategy.com/>

infraestructuras patentadas, esto se realiza a expensas de la interoperabilidad entre las aplicaciones. Los servicios Web XML ofrecen interoperabilidad en un nivel completamente nuevo que no deja lugar para esas rivalidades contraproducentes. Como siguiente avance revolucionario de Internet, los servicios Web XML se convierten en la estructura fundamental que vincula a todos los equipos y dispositivos.

1.2.2 Estructura de los Servicios Web XML

La infraestructura de servicios Web XML se fundamenta en la comunicación por medio de mensajes basados en XML que cumplen con una descripción de servicio publicada. La descripción del servicio es un documento escrito en una gramática XML denominada WSDL (Lenguaje de descripción de servicios Web) que define el formato de mensaje comprensible para el servicio Web XML. La descripción del servicio sirve como acuerdo que define el comportamiento de un servicio Web XML e indica a los clientes potenciales cómo interactuar con él. El comportamiento de un servicio Web XML está determinado por modelos de mensajería que el servicio define y admite. Estos modelos dictan conceptualmente lo que el consumidor del servicio puede esperar que ocurra cuando se envía un mensaje con formato correcto al servicio Web XML.

Por ejemplo, el modelo de solicitud y respuesta asociado a un servicio del estilo de RPC (Remote Procedure Call, llamada a procedimiento remoto) definiría qué esquema de mensajes SOAP debe utilizarse para invocar un método en concreto. Este modelo determinaría igualmente el formato que debería seguir el mensaje SOAP de respuesta. Otro ejemplo de modelo de mensajería representa interacciones unidireccionales. Este modelo se emplea cuando una comunicación se va a establecer en un sólo sentido. En esta situación, el emisor no recibirá ningún mensaje del servicio Web XML, ni siquiera los mensajes de error. Existe una limitación cuando se establece una comunicación unidireccional mediante un protocolo que es tradicionalmente de solicitud y respuesta, donde se puede devolver un mensaje de error.



FIGURA No 6, Guía básica de los Servicios Web: elementos de Conexión, descripción y descubrimiento

Los esquemas que definen los formatos de mensajes SOAP pueden definirse internamente para la descripción del servicio o bien externamente e importarse en la descripción.

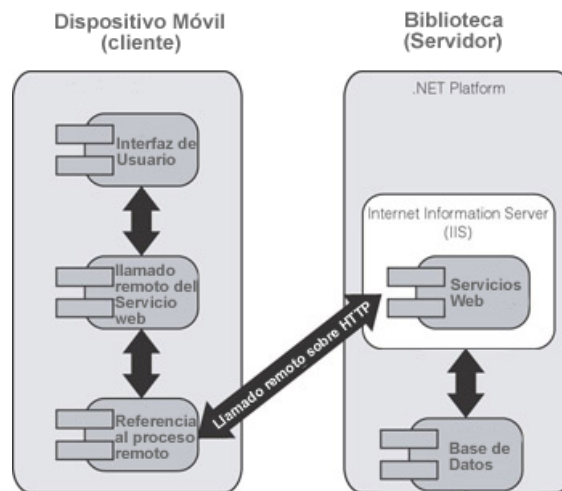
Además de las definiciones de formato de mensaje y los modelos de mensajería, la descripción del servicio puede contener la dirección asociada a cada punto de entrada del servicio Web XML. El formato de esta dirección se ajusta al protocolo utilizado para el acceso al servicio, como una dirección URL para HTTP o una dirección de correo electrónico para SMTP.

1.2.3 Consumo de los Servicios Web Por Parte De Los Clientes

Dada la dirección URL de un documento de descubrimiento que reside en un servidor Web, el programador de una aplicación cliente puede conocer la existencia de un servicio Web XML, sus capacidades y cómo interactuar correctamente con él. Este proceso se conoce como descubrimiento de servicios Web XML.

Mediante el proceso de descubrimiento de servicios Web XML, se descarga un conjunto de archivos al equipo local, que contienen información acerca de la existencia de servicios Web XML. Los archivos pueden ser descripciones de servicio, esquemas XSD o documentos de descubrimiento. Los documentos de descubrimiento que se

descargan contienen información acerca de la existencia de otros servicios Web XML que pueden residir en otros servidores Web.



ESQUEMA No 1, Esquema de diseño de los Servicios Web XML

1.3 XML *eXtensible Markup Language* (lenguaje de marcado ampliable o extensible)

XML Es un conjunto de reglas para definir etiquetas semánticas que organizan un documento en diferentes partes. Al igual que el HTML se basa en documentos de texto plano en los que se utilizan etiquetas para delimitar los elementos de un documento. Sin embargo, XML define estas etiquetas en función del tipo de datos que está describiendo y no de la apariencia final que tendrán en pantalla o en la copia impresa, además permite definir nuevas etiquetas y ampliar las existentes.

XML es un lenguaje de marcas no uno de programación aunque se definen igual como lenguajes, donde se utiliza para describir información, normalmente documentos, donde es notorio la marca de principio y final del elemento descrito.

```

<?xml version="1.0" encoding="utf-8" ?>
<MaterialBib>
  <nroClasificacion>515.15/P985c</nroClasificacion>
  <titulo>CALCULO CON GEOMETRIA ANALITICA / EDWIN J. PURCELL, DALE
  VARBERG.</titulo>
  <autor>PURCELL, EDWING J.</autor>
  <tipo>LIBRO</tipo>
  <Ejemplares>
    <string>GENERAL - PISO 2 & PRESTADO</string>
    <string>GENERAL - PISO 2 & PRESTADO</string>
    <string>RESERVA CENTRAL 2 & DISPONIBLE</string>
  </Ejemplares>
</MaterialBib>

```

ESQUEMA No 2, Ejemplo de un documento XML, donde se muestra en este formato la descripción de las características de un libro

1.4 SOAP Simple Object Access Protocol (Protocolo de acceso a objetos sencillos)

SOAP es el protocolo de acceso a objetos simple. La versión actual es la 1.1 y la especificación real se puede encontrar en www.w3.org/tr/soap (en inglés). El protocolo SOAP se basa en XML y describe un formato de mensajería para la comunicación entre equipos. Contiene también varias secciones opcionales que describen las llamadas a métodos (RPC) y proporcionan detalles sobre el envío de mensajes SOAP a través de HTTP.

Originalmente diseñado por Dave Winer, Bob Atkinson y Mohsen Al-Ghosein con el respaldo de Microsoft, SOAP fue cedida al grupo de trabajo del protocolo XML del consorcio de World Wide Web, entidad encargada de su difusión.

HTTP fue escogida como la capa de transporte de SOAP ya que esta trabaja bien sobre la infraestructura actual de la red mundial, su mayor ventaja sobre otros protocolos distribuidos es que SOAP es transparente a los cortafuegos, característica que no poseen protocolos como Corba, D-COM, GIOP/IIOP, los cuales son normalmente filtrados por los cortafuegos.

Por otro lado XML fue escogida como el lenguaje estándar para escribir los mensajes SOAP, su escogencia fue debido a su amplia aceptación por las grandes casas de software y esfuerzos de desarrolladores de código abierto.

1.4.1 Estructura del mensaje SOAP

Un mensaje SOAP esta contenido en un sobre, dentro de este sobre hay dos secciones adicionales: el encabezado y el cuerpo del mensaje, el encabezado contiene información relevante acerca del mensaje, por ejemplo se puede enviar en el encabezado la información de autenticación, aunque el encabezado del mensaje SOAP no es obligatorio, cuando esta presente siempre debe estar al principio del mensaje.

```
<soap:Envelope>
  <soap:Head>
    .....
  </soap:Head>

  <soap:Body>
    .....
  </soap:Body>
</soap:Envelope>
```

ESQUEMA No 3, Estructura de un mensaje SOAP

1.4.2 Ejemplo de un mensaje SOAP

Aquí se presenta un ejemplo de cómo un cliente usando un mensaje SOAP requiere información de un libro en un servicio Web XML. (El cliente necesita conocer el código del libro)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ObtenerDetalles xmlns="http://alcatraz.uis.edu.co/bibliotecasw/buscador.asmx">
      <IDLibro>827635</IDLibro>
    </ObtenerDetalles>
  </soap:Body>
</soap:Envelope>
```

ESQUEMA No 4, Ejemplo de una petición SOAP a un servicio Web XML

Y el Servicio Web le contestará con un mensaje SOAP con un esquema parecido al siguiente:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ObtenerDetallesResponse xmlns="http://alcatraz.uis.edu.co/bibliotecasw/buscador.asmx">
      <ObtenerDetallesResult>
        <Nombre>3D Studio Max 3, Guia de Aprendizaje</Nombre>
        <IDLibro >827635</ IDLibro >
        <descripcion>Libro de Aprendizaje para programación</descripcion>
        <precio>46000</precio>
        <disponible>>true</disponible>
      </ObtenerDetallesResult>
    </ObtenerDetallesResponse>
  </soap:Body>
</soap:Envelope>
```

ESQUEMA No 5, Ejemplo de la respuesta SOAP a la anterior petición del cliente

1. 5 **RPC-XML Remote Procedure Call** (*Llamado a procedimiento Remoto*)

Antes de mirar las diferencias con SOAP entraremos a determinar las características que estas dos especificaciones tienen en común:

Están basados en XML como lenguaje para la representación de datos y pueden utilizar HTTP como protocolo de transporte.

Tienen como objetivo primordial la integración de sistemas heterogéneos a través de paso de mensajes de texto en formato XML.

Implementación de funciones remotas de manera transparente para el usuario. Es decir que el cliente no tenga que saber en que lenguaje está hecho el servicio o en que sistema operativo está.

RPC-XML esta diseñado para ser lo mas simple y liviano posible sin perder las características de interoperabilidad, por el contrario SOAP es un protocolo que permite el manejo de estructuras de datos complejos personalizado por el *usuario*, donde el orden no es lo importante dado que cada elemento posee un nombre. Además SOAP posee características adicionales como la utilización de esquemas para la verificación de la validez de un mensaje. Estas características requieren un costo adicional en el ancho de banda utilizado para transmitir los mensajes.

SOAP es la tecnología con mayor impulso por parte de las grandes compañías de software como son Microsoft, IBM, Sun Microsystems, etc. y de gran aceptación por parte de los desarrolladores WEB.

En particular SOAP se podría tomar como una evolución del XML-RPC teniendo en cuenta que posee gran parte de las características y ventajas pero adicionales con costo extra agregado.

Las ventajas de SOAP se evidencian cuando las estructuras de los datos son complejas.

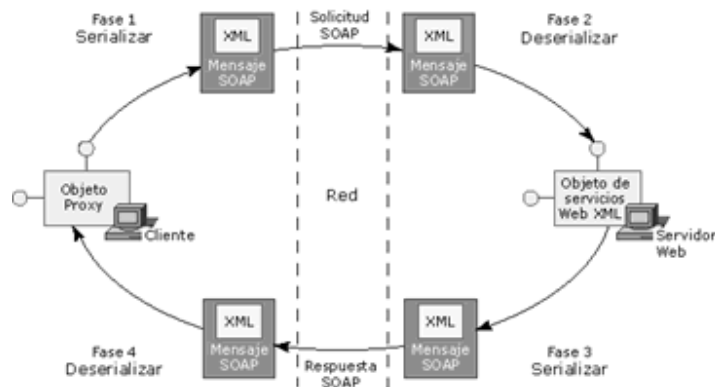


FIGURA No 7, Proceso de serialización y deserialización que sufren tanto la petición del cliente como la respuesta por parte del usuario del servicio Web XML

1.6 WSDL *Web Service Description Language* (Lenguaje de descripción de los servicios Web XML)

Dado que los protocolos de comunicaciones y los formatos de mensajes están estandarizados en la comunidad del Web, cada día aumenta la posibilidad e importancia de describir las comunicaciones de forma estructurada. WSDL afronta esta necesidad definiendo una gramática XML que describe los servicios de red como colecciones de puntos finales de comunicación capaces de intercambiar mensajes. Las definiciones de servicio de WSDL proporcionan documentación para sistemas distribuidos y sirven como fórmula para automatizar los detalles que toman parte en la comunicación entre aplicaciones.

Los documentos WSDL definen los servicios como colecciones de puntos finales de red o puertos. En WSDL, la definición abstracta de puntos finales y de mensajes se separa de la instalación concreta de red o de los enlaces del formato de datos. Esto permite la reutilización de definiciones abstractas: mensajes, que son descripciones abstractas de los datos que se están intercambiando y tipos de puertos, que son colecciones abstractas de operaciones. Las especificaciones concretas del protocolo y del formato de datos para un tipo de puerto determinado constituyen un enlace reutilizable.

Un puerto se define por la asociación de una dirección de red y un enlace reutilizable; una colección de puertos define un servicio. Por esta razón, un documento WSDL utiliza los siguientes elementos en la definición de servicios de red:

Types: contenedor de definiciones del tipo de datos que utiliza algún sistema de tipos (por ejemplo XSD).

Message: definición abstracta y escrita de los datos que se están comunicando.

Operation: descripción abstracta de una acción admitida por el servicio.

Port Type: conjunto abstracto de operaciones admitidas por uno o más puntos finales.

Binding: especificación del protocolo y del formato de datos para un tipo de puerto determinado.

Port: punto final único que se define como la combinación de un enlace y una dirección de red.

Service: colección de puntos finales relacionados.

Es importante observar que WSDL no introduce un nuevo lenguaje de definición de tipos. WSDL reconoce la necesidad de disponer de diferentes sistemas de tipos para describir los formatos de mensaje y admite como sistema de tipos canónico la especificación de los esquemas XML (XSD).

Sin embargo, puesto que no es razonable esperar una única gramática del sistema de tipos que se utilice para describir todos los formatos de mensajes presentes y futuros, WSDL permite el uso de otros lenguajes de definición de tipos mediante la extensibilidad.

Asimismo, WSDL define los mecanismos de enlace común. Éstos se utilizan para adjuntar un protocolo, un formato de datos o una estructura específica a un mensaje abstracto, una operación o un punto final de red. Permite la reutilización de definiciones abstractas.

Además del marco de definición de servicio principal, esta especificación introduce extensiones de enlace específicas para los siguientes protocolos y formatos de mensaje: *SOAP 1.1, HTTP GET / POST, MIME*

Aunque se han definido en este documento, las extensiones de lenguaje anteriores están en una capa superior respecto al marco de definición de servicio principal. Nada impide el uso de otras extensiones de enlace con WSDL.

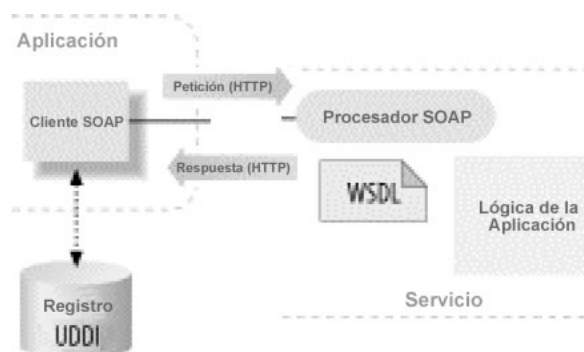
1.7 UDDI *Universal Description, Discovery and Integration* (Descubrimiento, descripción, e integración Universal)

UDDI es un registro público diseñado para alojar información acerca de los negocios en forma estructurada. A través de UDDI, se pueden publicar y descubrir los diferentes negocios y sus Servicios Web. Estos datos pueden ser clasificados usando taxonomías estándares para que puedan ser encontrados basados en su categoría.

UDDI contiene información técnica detallada acerca de los servicios que ofrece un negocio. A través de un conjunto de llamadas SOAP se puede interactuar con UDDI en tiempo de diseño y de ejecución para encontrar información técnica, para que los servicios puedan ser invocados y usados.

La información UDDI es alojada en nodos operadores, que corresponden a compañías comprometidas en mantener nodos que cumplan con las especificaciones del consorcio UDDI.org. Se requiere que los nodos operadores constantemente estén replicando los datos de un servidor a otro a través de un canal seguro, de tal forma que se mantenga una copia de todo el registro en todos los nodos. Una vez que se crea un registro en un nodo este debería poder ser encontrado en otro nodo.

UDDI es relativamente liviano ya que esta diseñado como un registro de aplicaciones y no como un repositorio de aplicaciones, la diferencia es sutil pero crucial, es decir sólo se encuentran las direcciones donde están ubicados los servicios más no los servicios en si.



ESQUEMA No 6, Esquema donde se muestra la posición lógica del servidor UDDI

WSDL emerge como una pieza importante en el protocolo de los servicios Web. Por esto es importante mencionar como se acoplan WSDL y UDDI y como el concepto de interfaces e implementaciones hace parte de cada protocolo. Ambos conceptos fueron diseñados para complementarse y delimitar claramente metadatos y la implementación en si de la aplicación.

WSDL hace una distinción clara entre los mensajes y los puertos, Los mensajes son la sintaxis y semántica requerida de un servicio que son abstractos, mientras que los puertos son las direcciones donde el servicio Web puede ser invocado que son siempre concretos.

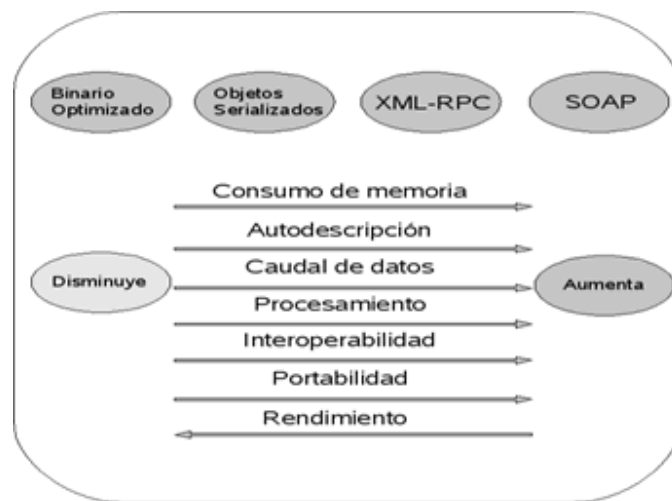
Una de las características más impresionantes es que puede haber múltiples implementaciones de una interfase WSDL. Este diseño permite escribir implementación de la misma interfase, garantizando que un sistema pueda hablar con un semejante.

UDDI describe una diferenciación muy similar entre abstracción e implementación con el concepto de tModels (Modelos de tecnología), que representan huellas digitales, interfaces y tipos abstractos de metadatos.

1.8 Métodos de Comunicación distribuida entre dispositivos

Existen diferentes formas de acceder a información que se encuentra disponible en Internet a través de dispositivos móviles, al hablar de Java 2 Micro Edition es indispensable saber que el perfil con menores características MIDP 1.0 establece que debe existir compatibilidad con el protocolo HTTP versión 1.1 por lo tanto para hacer una aplicación que cumpla con las especificaciones para la mayoría de los dispositivos móviles implica el uso de este protocolo.

A través de este protocolo existen diversos métodos de comunicación y cada uno de ellos con sus respectivas ventajas y desventajas pero tienen en común que sacrifican ciertas características como el tamaño de la aplicación que es muy importante para dispositivos con memoria limitada, auto descripción de la comunicación para la compatibilidad con otras aplicaciones, utilización del caudal de datos y cantidad de procesamiento que requiere la aplicación. Este sacrificio se hace con el fin de ganar otros beneficios como compatibilidad, interoperabilidad, portabilidad, entre otras.



ESQUEMA No 7, Esquema donde se muestran las principales variables de desempeño de los métodos existentes para comunicar dispositivos

1.8.1 Formato Propietario Binario

Esta es una forma bastante eficiente de comunicación por que los formatos se establecen de tal forma que optimicen el consumo de ancho de banda, son altamente flexibles y complejos. Estos formatos tienen esencialmente que ver con la forma de enviar peticiones HTTP GET y HTTP POST.

El objetivo es minimizar el tamaño de la carga útil y agilizar la comunicación. Las desventajas por otro lado son:

El cliente y el servidor deben conocer de antemano el formato establecido aun antes de comenzar el desarrollo ya que el mensaje no es autodescriptivo, es por esto que el cliente está altamente ligado al servidor.

Los cambios hechos en el formato requieren cambios en el tanto en cliente como en el servidor pero deben mantener compatibilidad con versiones lo que con el tiempo codificación se torna más compleja.

1.8.2 Serialización de Objetos

Es una metodología que se aproxima a la solución de los problemas anteriores por que consiste en serializar objetos de java aunque toca implementar esta función en cada clase que se desea serializar o deserializar en el lado del cliente J2ME por que no se encuentra implementada. Además sólo se podría utilizar clientes J2ME y servidor J2EE para la aplicación, cuestión que carece de sentido si estamos buscando interoperabilidad.

1.9 Servicios Web XML y la Computación distribuida

Existen precursores de la tecnología de los Web Services cuyo objetivo es el de buscar la interacción entre funcionalidades ubicadas en algún lugar remoto de la red, éstas continúan vigentes aunque su uso este disminuyendo por la utilización de nuevas tecnologías de integración como los Web Services que es capaz de llegar donde CORBA y RMI no pueden interactuar con una aplicación antigua como por ejemplo de COBOL:

RMI - *Remote Method Invocation*- Fue el primer framework para crear sistemas distribuidos de Java. El sistema de Invocación Remota de Métodos (RMI) de Java permite, a un objeto que se está ejecutando en una Máquina Virtual Java (VM), llamar a métodos de otro objeto que está en otra VM diferente. Esta tecnología está asociada al lenguaje de programación Java, es decir, que permite la comunicación entre objetos creados en este lenguaje.

DCOM - *Distributed Component Object Model*- El Modelo de Objeto Componente Distribuido, esta incluido en los sistemas operativos de Microsoft. Es un conjunto de conceptos e interfaces de programa, en el cual los objetos de programa del cliente, pueden solicitar servicios de objetos de programa servidores en otros ordenadores dentro de una red. Esta tecnología esta asociada a la plataforma de productos Microsoft.

CORBA - *Common Object Request Broker Architecture*- Tecnología introducida por el Object Management Group OMG, creada para establecer una plataforma para la invocación de métodos remotos bajo un paradigma orientado a objetos independiente

del lenguaje de programación. Esta propuesta es la que más se asemeja a servicios Web, no sólo por que tienen el objetivo de integrar diferentes sistemas operativos sino por su similitud en arquitectura.

En un sentido general CORBA "envuelve" el código escrito en otro lenguaje en un paquete que contiene información adicional sobre las capacidades del código que contiene, y sobre cómo llamar a sus métodos. Los objetos que resultan pueden entonces ser invocados desde otro programa (u objeto CORBA) desde la red. En este sentido CORBA se puede considerar como un formato de documentación legible por la máquina, similar a un archivo de cabeceras pero con más información.

CORBA utiliza un lenguaje de definición de interfaces (IDL) para especificar los interfaces con los servicios que los objetos ofrecerán. CORBA puede especificar a partir de este IDL la interfaz a un lenguaje determinado, describiendo cómo los tipos de dato CORBA deben ser utilizados en las implementaciones del cliente y del servidor.

| Características | Servicios Web XML | CORBA |
|--|--------------------------|---|
| Protocolo de comunicaciones | SOAP, HTTP | IIOP |
| Ubicación de la Funcionalidad (Servicio) | IRL | URL |
| Descripción | WSDL | IDL ²⁵ |
| Búsqueda | UDDI | Servicio de Nombres y repositorio de interfaces |

TABLA No 1, Semejanzas entre Servicios Web XML y CORBA

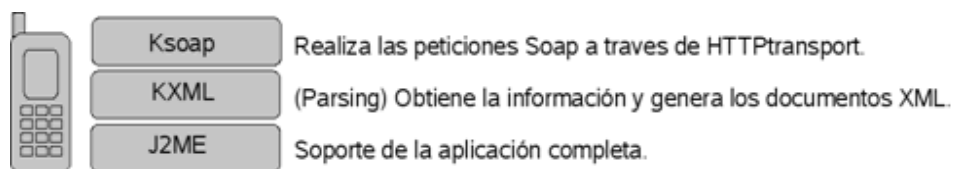
²⁵ IDL Interface Definition Language: como su propio nombre indica un lenguaje de especificación de interfaces que se usa como parte de la tecnología CORBA. Ofrece la sintaxis necesaria para definir los métodos que queremos invocar remotamente. Una vez tengamos esta interfaz creada deberemos pasarla por un compilador de interfaces que generará el proxy o stub cliente y el skeleton o stub servidor: Definición Tomada de <http://es.wikipedia.org/wiki/IDL>

1.10 Servicios Web XML y los Dispositivos Móviles

Para el caso de los dispositivos móviles compatibles con J2ME cabe aclarar que esta plataforma no cuenta con soporte nativo del protocolo SOAP aún, como tampoco de *parsing* de XML por lo tanto es necesario utilizar librerías que se encuentran actualmente disponibles para el manejo del protocolo conocida como **Ksoap** que utiliza a su vez otras librerías para el la interpretación de XML **KXML**, esta herramienta relativamente liviana en lo referente a la cantidad de memoria que utiliza (41 Kb) y el desempeño que deben tener las aplicaciones que utilizan estas librerías recordando que los dispositivos móviles presentan limitaciones en sus recursos de hardware como en velocidad de procesamiento, memoria, batería, ancho de banda para comunicación de datos.

Ksoap es un proyecto de código abierto cuyo objetivo es habilitar el consumo de servicios Web a través de dispositivos móviles compatibles con J2ME. Dado que un WS puede ser invocado enviando un mensaje XML al servidor web en espera de solicitudes, el mensaje puede ser creado sin ningún inconveniente desde la aplicación sin la necesidad de usar la librería, pero sería un trabajo no escalable y perdería su mantenibilidad. La principal ventaja que ofrece Ksoap es que puede crear un mensaje XML (*Parsing*) con simples llamados a las clases que envuelven y facilitan y ocultan la complejidad de la creación del mensaje de la petición de un servicio remoto. Asimismo la respuesta SOAP que se obtiene de un servicio contiene información en sintaxis XML que se extrae a través del también XML parser inmerso en Ksoap.

Es una ventaja adicional el hecho que Ksoap conserve ese enlace débil entre la aplicación cliente y el servicio Web, en el caso de crear una aplicación que origine el mensaje de petición para un WS determinado se estaría atando la aplicación al servicio perdiendo las virtudes de la arquitectura de los servicios Web.



ESQUEMA No 8, Estructura de KSOAP

2. TECNOLOGÍA EMPLEADA

En este punto se presenta los diferentes lenguajes, herramientas, sistemas operativos y conceptos que fueron utilizados en la ejecución práctica de este proyecto

2.1 SOFTWARE DE DESARROLLO

2.1.1 Microsoft .NET Framework²⁶

.NET Framework es una plataforma informática que simplifica el desarrollo de aplicaciones en un entorno altamente distribuido como es Internet.

El diseño de .NET Framework está enfocado a cumplir los objetivos siguientes:

- Proporcionar un entorno coherente de programación orientada a objetos, en el que el código de los objetos se pueda almacenar y ejecutar de forma local, ejecutar de forma local pero distribuida en Internet o ejecutar de forma remota.
- Proporcionar un entorno de ejecución de código que reduzca lo máximo posible la implementación de software y los conflictos de versiones.
- Ofrecer al programador una experiencia coherente entre tipos de aplicaciones muy diferentes, como las basadas en Windows o en el Web.
- Basar toda la comunicación en estándares del sector para asegurar que el código de .NET Framework se puede integrar con otros tipos de código.



FIGURA No 8²⁷, Logo del entorno de desarrollo del Visual Studio .NET 2003

.NET Framework contiene dos componentes principales: Common Language Runtime y la biblioteca de clases de .NET Framework.

²⁶ ms-help://MS.NETFrameworkSDKv1.1.ES/cpguidenf/html/cpovrintroductiontonetframeworksdk.htm

²⁷ Imagen tomada del sitio Web oficial de Microsoft <http://www.microsoft.com>

Common Language Runtime es el fundamento de la tecnología. El motor de tiempo de ejecución se puede considerar como un agente que administra el código en tiempo de ejecución y proporciona servicios centrales, como la administración de memoria, la administración de subprocesos y la interacción remota, al tiempo que aplica una seguridad a los tipos y otras formas de especificación del código que garantizan su seguridad y solidez.

La biblioteca de clases, el otro componente principal de .NET Framework, es una completa colección orientada a objetos de tipos reutilizables que se pueden emplear para desarrollar aplicaciones que abarcan desde las tradicionales herramientas de interfaz gráfica de usuario (GUI) o de línea de comandos hasta las aplicaciones basadas en las innovaciones más recientes proporcionadas por ASP.NET, como los *formularios Web* (Web Forms) y los *servicios Web XML*. (Xml Web Services).

2.1.2 Net Beans 4.1

NetBeans IDE es un entorno de desarrollo (no un lenguaje de programación como podría pensarse) para aplicaciones JAVA desde aplicaciones empresariales J2EE hasta aplicaciones para dispositivos móviles en J2ME pasando por aplicaciones Web y servicios Web XML, que provee un gran conjunto de características y un entorno de trabajo más productivo al no tener la necesidad de buscar plug-ins para realizar tareas convencionales en la plataforma JAVA.

NetBeans es un entorno multiplataforma que puede ser instalado y utilizado en un sin número de sistemas operativos, como las diferentes versiones de Windows, Linux, Solaris SPARC, MAC OS X y Solaris x86 lo que implica una total independencia del sistema donde se realiza el desarrollo.



FIGURA No 9²⁸, Logo del grupo de desarrollo de NetBeans

²⁸ Imagen tomada del sitio Web oficial de NetBeans <http://netbeans.org>

NetBeans es un entorno de uso libre, puede ser descargado sin ningún costo dado que es un producto soportado por una prospera y creciente comunidad de desarrolladores Java. Todas las fuentes están disponibles y la plataforma NetBeans puede servir como base para las aplicaciones propias. No existen restricciones en las redistribuciones que se hagan.

El Paquete de movilidad de Netbeans (NetBeans Mobility Pack) provee un conjunto de herramientas para construir aplicaciones J2ME. Contienen un diseñador visual que permite arrastrar y colocar elementos, resuelve el problema que la fragmentación de dispositivos y puede ser usada para crear soluciones End-to-End.

Un ambiente bastante agradable y útil para el desarrollo de aplicaciones móviles por que facilita el trabajo con ayudas visuales como despliegue de información tanto de referencia como de las características principales, propiedades, métodos de los objetos mientras se escribe. Además NetBeans provee características como texto predictivo y manejo de errores de sintaxis en tiempo real que agilizan el proceso de codificación y aprendizaje del lenguaje.

Un sistema de gestión de proyectos basado en Apache Ant que garantiza no quedar bloqueado usando el IDE. Permite compartir fácilmente el proyecto con un grupo de desarrollo, asimismo construir, ejecutar e implementar los proyectos en un servidor remoto conectado al equipo local donde se encuentra instalado el IDE.

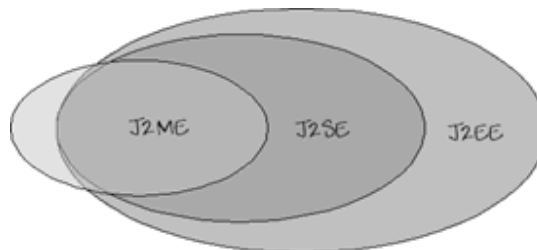
La versión utilizada para el desarrollo es NetBeans IDE 4.1 que tiene una gran cantidad de herramientas integradas y un gran desempeño, y que permite la personalización del entorno para mayor comodidad y adaptación de uso del usuario.

2.1.3 J2ME (Java 2 MicroEdition)

La sigla correspondiente a Java 2 Micro Edition, que corresponde a la versión de JAVA orientada a dispositivos móviles en cuyas características se destacan la conectividad inalámbrica y baja capacidad de procesamiento y memoria, tales como teléfonos celulares, PDA's, telemetría en automóviles, entre otros.

La tecnología software Java esta orientada de acuerdo a las capacidades de hardware, por ejemplo, se puede realizar aplicaciones en J2EE para aprovechar las capacidades inmensas que ofrecen los grandes servidores, utilizar J2SE para crear aplicaciones de escritorio, pero como los dispositivos móviles no encajan en estas categorías se hace necesario crear una nueva edición con menor funcionalidad pero que se adapte a las

necesidades de estos dispositivos livianos. Es por esto que se podría decir que J2ME una versión ultra liviana de J2SE dado que es un subconjunto de funcionalidades y librerías de esta última.



ESQUEMA No 9, Diagrama de conjuntos de la plataforma Java.

Máquinas Virtuales J2ME

Una máquina virtual de Java (JVM) es un programa encargado de interpretar código intermedio (bytecode) de los programas Java precompilados a código máquina ejecutable por la plataforma, efectuar las llamadas pertinentes al sistema operativo subyacente y observar las reglas de seguridad y corrección de código definidas para el lenguaje Java. De esta forma, la JVM proporciona al programa Java independencia de la plataforma con respecto al hardware y al sistema operativo subyacente. Las implementaciones tradicionales de JVM son, en general, muy pesadas en cuanto a memoria ocupada y requerimientos computacionales. J2ME define varias JVMs de referencia adecuadas al ámbito de los dispositivos electrónicos que, en algunos casos, suprimen algunas características con el fin de obtener una implementación menos exigente.

2.2 SISTEMAS OPERATIVOS

2.2.1 Windows XP Service Pack 2 Professional Edition

Windows XP es el sistema operativo basado en win32 enfocado a los usuarios finales, actualmente se distribuye en 4 versiones, XP HOME enfocada a usuarios domésticos que no exigen todo el potencial de la maquina, XP PROFESSIONAL enfocada a usuarios más avanzados, el cual al poseer esquemas de seguridad más desarrollados hace que sea el sistema operativo más adecuado para el desarrollo de esta aplicación. Las otras versiones del Windows XP son XP MEDIA CENTER EDITION especializada en tecnologías multimedia, XP TABLE PC que es el sistema operativo enfocado a la nueva generación de dispositivos Table PC.



FIGURA No 10²⁹, Logo del sistema operativo Windows XP

Windows XP Professional fue utilizado como sistema operativo de desarrollo debido a los múltiples beneficios de seguridad que posee, entre estos se cuentan:

Centro de seguridad de Windows

Mantiene el PC actualizado con los últimos estándares de seguridad para combatir el creciente número de virus y software mal intencionados distribuidos por Internet.

Corta Fuegos (firewall)

Windows XP Professional en su edición Service Pack 2 instala en el equipo un corta fuegos que se encuentra activo desde el momento en que se inicia Windows, esta mejora en el sistema operativo evita que otros equipos intenten hacerse con el control de la maquina, además de gestionar que programas que están instalados en el equipo local pueden acceder a Internet.

²⁹ Imagen tomada del sitio Web oficial de Microsoft <http://www.microsoft.com>

Escritorio Remoto

El escritorio remoto que sólo se encuentra en la versión de Windows XP Professional permite que se pueda acceder remotamente a otro PC basado en Windows, dejando la posibilidad de trabajar con los archivos del equipo de trabajo en la comodidad de la casa.

Unirse a una Red o dominio

Windows XP está diseñado para conectarse con redes de servidores de forma fácil mediante un asistente de conexión que simplifica el proceso de adición a un dominio. Gracias a su alto perfil de seguridad y estabilidad, Windows XP es el sistema operativo de la máquina de desarrollo donde se encuentra la aplicación de lado del servidor, a pesar que el equipo final donde va a quedar instalado la aplicación es Windows 2003 Server, Windows XP se comporta bien en el entorno de consulta y consumo de servicios Web.

2.2.2 Windows 2003 Server Standard Edition (W2K3)

Windows 2003 Server es la edición de sistemas operativos de Windows enfocada al manejo eficiente de datos, su programación y comportamientos están enfocados al repositorio, mantenimiento y consulta de datos.

Entre sus características se cuenta su alto nivel de seguridad, al exigir identidades a los usuarios que requieran de sus servicios, el uso de contraseñas y usuarios unificados son otros métodos de seguridad que soporta Windows 2003 Server.



ESQUEMA No 10³⁰, Esquema de seguridad basado en Corta-Fuegos de Windows 2003 Server

³⁰ Imagen tomada y modificada del sitio Web de Windows 2003 Server
<http://www.microsoft.com/windowsserversystem/>

De W2k3 existen actualmente 4 versiones que se acoplan a las tareas que debe realizar el equipo servidor, ellas son:

Windows 2003 Server Web Edition: Esta versión es aconsejable para equipos que hospeden servicios Web y sitios Web, es decir, para tener un servidor que proporcione páginas Web. Es el más bajo de la gama y sólo existe en versión OEM³¹.

Windows 2003 Standard Edition: Es la versión estándar de W2K3, el que sirve tanto para administración de una red local y usuarios como para servicios Web y Hosting. Incluye la posibilidad de conexiones a escritorio remoto, algo que resulta de muchísima utilidad.

Windows 2003 Enterprise Edition: Esta versión es muy parecida a la estándar, la diferencia es que en esta versión se tiene realmente un servidor con todos los servicios, sea para pequeños o para grandes volúmenes de datos, porque con la versión Standard el manejo de grandes volúmenes bloquea el sistema constantemente.

Windows 2003 Datacenter Edition: Esta es la versión más alta de la gama esta enfocada a trabajar en escenarios donde se necesita responder a un gran número de peticiones, reservada para grandes servidores debido al costo de su licencia, Windows 2003 DataCenter Edition maneja múltiples procesadores y memoria.

El Servidor de Biblioteca conocido como alcatraz, es el lugar donde se encuentra la interfaz gráfica y reside el servidor Web, el número de peticiones que llegan a este servidor es reducido, ya que se utiliza exclusivamente para responder las peticiones de los clientes de biblioteca, por esta razón se instaló como sistema operativo la versión estándar de W2K3.

Como detalles de este sistema operativo se cuenta que es el SO más estable de la familia de Microsoft, tiene una excelente documentación así como la instalación, la gran cantidad de funciones extras es otro de sus fuertes, pero no todo es bueno el W2K3, su punto débil es la configuración, la cual requiere un alto grado de conocimiento sobre SO, lo que lo hace difícil ponerlo en marcha.

³¹ OEM, Original Equipment Manufacturer, Marca que se le da al software licenciado que instalan los fabricantes de hardware al momento de vender sus equipos directamente de fábrica.

2.2.3 Windows Mobile 2003

Windows Mobile 2003 es un sistema operativo basado en la API de Win32, pero enfocada para gestionar las características de los equipos portátiles, entre estos se incluyen los Pda's y los teléfonos inteligentes.

Actualmente existen dos versiones de WinMobile 2003, conocida como WinMobile PPC es el sistema operativo de las PDA's, se enfoca a manejar los limitados recursos de este tipo de equipos, aún con estas restricciones WinMobile para PDA's crea un entorno Windows en el equipo, permitiendo al usuario realizar las tareas básicas que se pueden ejecutar en las versiones adultas de Windows, además de gestionar la conectividad inalámbrica si el equipo tiene instalada una tarjeta de red.



FIGURA No 11³², Entorno gráfico del SO Windows Mobile 2003

Windows Mobile 2005 originalmente llamado el magneto fue lanzado en el año 2005, es la versión mejorada que Microsoft lanza de su versión móvil de Windows 2003, el nuevo ambiente esta enfocado al entorno .net y entre sus nuevas características se cuentan:

- *Una nueva versión de su motor gráfico.*
- *Windows Media 10 Player Mobile.*
- *Actualizado a los nuevos estándares de la IEEE sobre la conectividad inalámbrica WIFI y Bluetooth.*
- *Versiónes Pocket de el procesador de texto Word, la hoja de calculo Excel.*

³² La imagen pertenece al emulador de la PDA, el cual es instala con el entorno de desarrollo de Microsoft.NET 2003

- *Trae instalado la versión de PowerPoint Mobile, para crear animaciones en este tipo de dispositivos.*

Para el desarrollo de la aplicación cliente escogimos este sistema operativo es debido a que su programación esta basada en .net y sus programas soportan el nuevo estándar de Servicios Web, además de poseer conexión WIFI para entrar en cualquier red inalámbrica que soporte este protocolo, instancia necesaria para que el cliente que esta instalado en la PDA se pueda comunicar con el servidor que se encuentra en Internet.

2.2.4 LINUX SUSE 10

SUSE Linux es una de las más conocidas distribuciones de Linux, establece una informática segura, fiable y completa, Incluye un sistema operativo Linux fácil de instalar que permite navegar por Internet, enviar correo electrónico, organizar fotos digitales, reproducir películas y música, y crear documentos y hojas de cálculo. Incluso puede utilizarlo para albergar un blog o sitio Web, crear una red doméstica y desarrollar sus propias aplicaciones.

Independientemente de que sea la primera vez que se utiliza Linux o se utilice diariamente, SUSE Linux 10.0 ofrece las funciones que se necesitan para la conectividad e informática. SUSE Linux también incluye más de 1.000 aplicaciones de código abierto líderes a nivel mundial. Desde las aplicaciones más consolidadas como Firefox y OpenOffice.org a los "first looks" ("primeros vistazos") a las tecnologías emergentes como Beagle y Xen, todas aportan un verdadero valor.



FIGURA No 12, Logo del sistema operativo de Linux SUSE, de la empresa Novell

Como es Linux, su sistema es inherentemente seguro, prácticamente inmune a los gusanos y virus de Internet y el correo electrónico. Por supuesto, se incorporan sólidas funciones de seguridad.

Aunque no está destinado para implantaciones empresariales, SUSE Linux 10.0 ofrece evaluadores corporativos con un excelente dominio de tecnologías como XEN (virtualización), OpenOffice 2.0 (paquete de productividad), Beagle (búsqueda en equipos de escritorio) y Mono (entorno de desarrollo de código abierto). Estas tecnología, una vez perfeccionadas y admitidas, pueden aparecer en futuras ofertas de Novel Enterprise Linux.

2.2.5 SYMBIAN OS

Symbian OS es un sistema operativo de código abierto para celulares y dispositivos móviles. Nokia, Sony, Motorola, Fujitsu y otras empresas, se cuentan entre los fabricantes de hardware que apoyan e instalan Symbian en sus equipos.

La estructura de Symbian esta basada en un micro Kernel el cual contiene un programador de tareas y un manejador de memoria, estos componentes prestan los servicios a las librerías del sistema y del usuario.

También hay un gran subsistema de redes y comunicaciones con tres servidores principales. ETEL (EPOC telefonía), ESOCK (EPOC sockets) y C32 (responsable de la comunicación), cada uno de estos servidores tiene un esquema basado en los modelos de comunicación inalámbrica Bluetooth, Irda, USB.

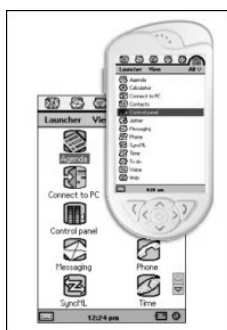


FIGURA No 13³³, Symbian UIQ 7.0, la interfaz de Symbian para los móviles de última generación

³³ Imagen tomada de http://www.quesabesde.com/PDA/noticias_detalle.asp?noticia=379

Symbian soporta la instalación de la maquina virtual de Java para dispositivos móviles, conocida como Kilo-Virtual Machine (KVM), esta combinación permite al dispositivo ejecutar programas basados en el estándar J2ME de Java, el cual soporta la creación de clientes basados en la arquitectura de Servicios Web.

2.3 HARDWARE

En este punto se listan todos los equipos que de alguna manera permitieron el desarrollo de este proyecto, sea en la fase de diseño como en la puesta en marcha.

2.3.1 Equipos para el Desarrollo de las aplicaciones

La aplicación se desarrollo en dos equipos diferentes conectados en red a través de la red LAN de la Universidad Industrial de Santander, sus especificaciones técnicas son:

Equipo de Desarrollo para la aplicación Servidor:

Hardware

- Equipo Dell Optiplex GX520
- Procesador Intel Pentium 4 HD, 3.0GHZ
- 512 MB de RAM
- Monitor 17" LCD
- Disco Duro de 80 GB
- Unidad de CD-RW 48x
- Mouse y Teclado
- Tarjeta de Red 10/100

Software

- Sistema Operativo Windows XP Service Pack 2
- Servidor Web IIS 5.0
- .NET Framework 1.1
- Visual Studio .NET
- Internet Explorer 6.0

Equipo de Desarrollo para la aplicación Cliente:

Hardware

- Equipo Portátil ACER
- Procesador Intel Pentium 4 HD, 2.8GHZ
- 512 MB de RAM
- Monitor 14.5" LCD
- Disco Duro de 80 GB
- Unidad de DVD-RW 24x
- Tarjeta de Red 10/100
- Tarjeta de Conexión inalámbrica WIFI
- Mouse

Software

- Sistema Operativo LINUX SUSE 10
- NetBeans 4.1
- Java Virtual Machine 5.0
- Emulador de Celular Nokia Serie 80

2.3.2 Equipos para la implantación

La aplicación en su fase final posee dos componentes, el componente servidor quedara instalado en el IIS del servidor de biblioteca conocido como alcatraz, por otra parte se crearon dos clientes, uno de esos clientes residirá en un celular Nokia el cual es compatible con J2ME, el otro en una PDA con Windows Mobile como sistema operativo. Sus especificaciones técnicas son las siguientes:

Equipo que hospeda la aplicación Servidor:

Hardware

- Equipo DELL PowerEdge 4600
- 2 Procesadores Intel XEON 2.4GHz
- 4GB de RAM
- Monitor 19" LCD
- Disco Duro de 200 GB
- Unidad de CD-ROM 48x
- Mouse y Teclado
- Tarjeta de Red 10/100

Software

- Sistema Operativo Windows 2003 Server Standard 5.2
- Servidor Web IIS 6.0
- IBM Informix Client – SDK 2.90
- .NET Framework 1.1
- Internet Explorer 6.0

Equipos que hospedan la aplicación Cliente:**CELULAR****Hardware**

- 128KB de memoria no volátil
- 32KB de memoria volátil
- Procesador de 16 bits
- Consumo bajo de batería
- Servicio de datos por parte del operador de la red celular

Software

- Sistema Operativo Symbian 5
- Kilo-Virtual Machine 1.0
- Equipo Celular compatible con J2ME

PDA**Hardware**

- 16MB de memoria no volátil
- 32MB de memoria volátil
- Procesador de 32 bits 200Mhz
- Consumo bajo de batería
- Tarjeta de conexión inalámbrica WIFI

Software

- Sistema operativo Windows Mobile 2003

2.4. MANEJO Y ALMACENAMIENTO DE DATOS

El manejo de la información es uno de los puntos clave de este proyecto, a continuación se describen los manejadores de base de datos que están presentes en los diferentes donde residen las aplicaciones.

2.4.1 Informix

Informix es uno de los cuatro grandes gestores de bases de datos que existen en el mercado, DB2 de IBM, SQL Server de Microsoft y Oracle son los otros motores de datos que comparten casi la totalidad del mercado, en su interior Informix es un gestor creado por Informix Software Inc que incluye un RDBMS³⁴ basado en SQL, además de ser un lenguaje de cuarta generación con herramientas para la inclusión de queries en programas de aplicación.

Entre sus características más importantes se cuenta su robustez, Informix se instala y funciona correctamente bajo ambientes NT y Unix.

- Dispone de herramientas gráficas que permiten asistir tanto al proceso de instalación como al de administración del sistema, sin necesidad de que el administrador tenga grandes conocimientos en administración de equipos.
- Posee una consola centralizada desde la cual el administrador del sistema puede realizar el control sobre sus bases de datos remotas como locales, además de ocupar menos memoria y recursos que otros manejadores de bases de datos.
- Proporciona tablas que forman la interfaz de monitoreo del sistema.
- Tiene la capacidad de relacionar datos que se encuentran en múltiples espacios físicos.
- Su mayor debilidad es el alto costo de las licencias para implantarlo a pequeñas y medianas empresas.

³⁴ (RDBMS) RELATIONAL DATABASE MANAGEMENT SYSTEM es un sistema manejador de bases de datos basado en el modelo entidad-relación creado por Edgar F. Codd

Actualmente Informix es el manejador de base de datos de la biblioteca de la Universidad Industrial de Santander, base de datos en la cual esta enfocada parte de la aplicación que se esta creando, a su debido tiempo se explicará mediante que método el programa servidor accede a los datos almacenados en el servidor de biblioteca y enviados al programa cliente, para que este mediante su interfase lo visualice.

2.4.2 SqlServer CE

SqlServerCe es el motor de base de datos para Pocket's, su creación y funcionamiento esta basado en el proveedor de datos de .NET Compact Framework el cual fue desarrollado con tecnología .NET de Microsoft.

Un proveedor de datos de .NET Compact Framework para SQL Server CE describe una colección de clases utilizada para tener acceso a una base de datos de SQL Server CE, los dispositivos basados en Windows CE o Windows Mobile obtienen mediante la invocación de este gestor un entorno administrado de base de datos similar a los grandes repositorios, tales como Informix u Oracle.

Con SqlServerCe, se pueden crear bases de datos de SQL Server CE en un dispositivo móvil y establecer conexiones con las bases de datos de SQL Server que se encuentran en el dispositivo o en un servidor remoto.

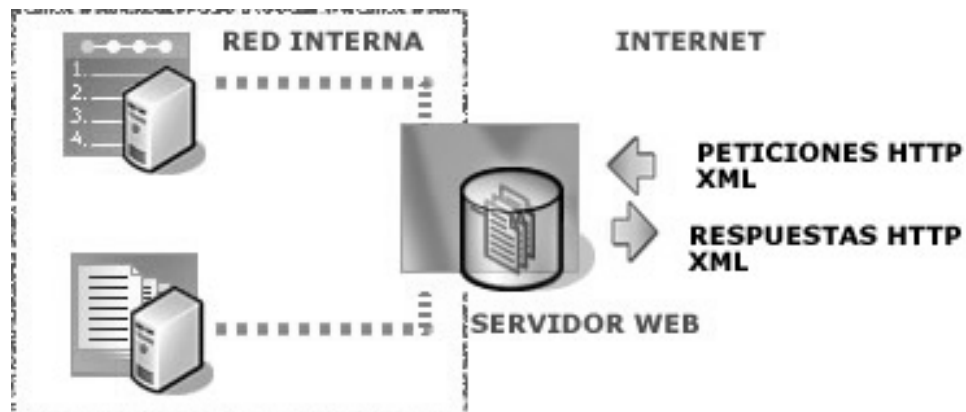
Características:

- Aunque su entorno y funcionamiento esta basado en los gestores de datos de mayor capacidad, SqlServerCe esta enfocado a funcionar sobre dispositivos con características limitadas.
- El tamaño de las bases de datos sobre esta tecnología esta limitada a Megas.
- El grupo de sentencia SQL que soporta esta limitado a los querys de consulta más básicos.

2.5 SERVIDOR WEB

2.5.1 Conceptos de Servidor Web³⁵

Un servidor Web es un programa que implementa el *protocolo HTTP* (*hypertext transfer protocol*). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas Web o páginas HTML (*hypertext markup language*): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos.



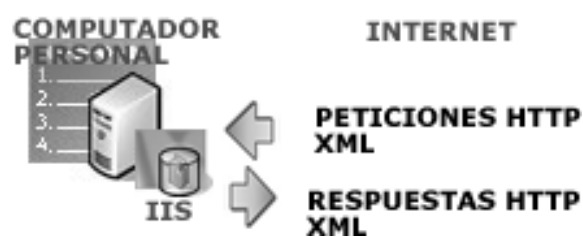
ESQUEMA No 11, Esquema que muestra la disposición lógica del Servidor Web en una red local LAN

Un servidor Web se encarga de mantenerse a la espera de *peticiones HTTP* llevadas a cabo por un *cliente HTTP* que solemos conocer como *navegador*. El navegador realiza una petición al servidor y éste le responde con el contenido que el cliente solicita. A modo de ejemplo, al teclear *www.direccion.com* en nuestro navegador, éste realiza una petición HTTP al servidor de dicha dirección. El servidor responde al cliente enviando el código HTML de la página; el cliente, una vez recibido el código, lo interpreta y lo muestra en pantalla. Como vemos con este ejemplo, el cliente es el encargado de interpretar el código HTML, es decir, de mostrar las fuentes, los colores y la disposición de los textos y objetos de la página; el servidor tan sólo se limita a transferir el código de la página sin llevar a cabo ninguna interpretación de la misma.

³⁵ Definición tomada del sitio Web http://es.wikipedia.org/wiki/Servidor_web

2.5.2 Servidor IIS (Internet Information Service - IIS)

Internet Information Service es un servidor Web altamente integrado a la familia de sistemas operativos basados en Windows NT, permite al equipo actuar como servidor tanto de páginas Web como de servicios Web XML, para poder responder peticiones de los SW, el servidor IIS debe instalar el .NET Framework el cual es un componente adicional a la instalación por defecto.



ESQUEMA No 12, Disposición del Servidor IIS en un equipo personal con Windows como sistema operativo

IIS viene como aplicación opcional en las otras versiones del sistema operativo Windows, la versión 6.0 de IIS se instala por defecto con el sistema operativo Windows 2003 Server, y esta combinación de SO y Servidor Web es la que se encuentra instalada en el servidor Alcatraz de la biblioteca, que es el lugar virtual donde va a quedar instalada el componente Servidor de la aplicación.

3. METODOLOGÍA PARA EL DESARROLLO DEL SOFTWARE

En este aparte se presenta las metodologías utilizadas para llevar a cabo la implementación de los conceptos teóricos estudiados anteriormente y su resultado en un desarrollo tecnológico que sea palpable a la comunidad.

3.1 Escogiendo la metodología de desarrollo

El desarrollo de una aplicación basándose en *la ingeniería del software*³⁶ no es una tarea fácil, prueba de ello son las innumerables propuestas metodológicas que existen actualmente y que inciden en el proceso de desarrollo. Por una parte existen las metodologías tradicionales que se centran especialmente en el control del proceso, estableciendo rigurosos controles a los procesos a seguir, las actividades a realizar, los documentos a generar y las herramientas a utilizar.

Estas metodologías han resultado ser efectivas en la mayoría de proyectos, en algunos casos no han servido de mucho, y en otros proyectos su utilización es prácticamente inaplicable. Una posible mejora a estas metodologías sería poner más restricciones a los puntos débiles del proceso, ayudando a reforzar los documentos que deben pactar el desarrollador y el usuario, pero al final lo único que se lograría será un proceso más extenso y complicado que dejaría en un segundo plano el desarrollo de la aplicación, desarrollo que debe ser tenido en cuenta como el eje central en el desarrollo a través del tiempo de la metodología que se ha de seguir³⁷.

³⁶ Término propuesto por Bauer en la Conferencia de Garmish, en 1968: “Establecer y usar principios de ingeniería orientados a obtener software de manera económica, fiable y que funcione eficientemente sobre máquinas reales”

³⁷ No se puede centrar los esfuerzos exclusivamente a seguir una metodología y a realizar documentos sin tener en cuenta que la evolución del proyecto es el eje central del mismo.

| Metodologías Tradicionales | Metodologías Ágiles |
|--|--|
| Cierta resistencia a los cambios | Especialmente preparados para cambios durante el proyecto |
| Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo | Basadas en heurísticas provenientes de prácticas de producción de código |
| Impuestas externamente | Impuestas internamente (por el equipo desarrollador) |
| Proceso mucho más controlado, con numerosas políticas/normas | Proceso menos controlado, con pocos principios |
| Existe un contrato prefijado | No existe contrato tradicional o al menos es bastante flexible |
| El cliente interactúa con el equipo de desarrollo mediante reuniones | El cliente es parte del equipo de desarrollo |
| Grupos grandes y posiblemente distribuidos | Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio |
| La arquitectura del software es esencial y se expresa mediante modelos | Menos énfasis en la arquitectura del software |

TABLA No 2: La tabla recoge esquemáticamente las principales diferencias de las metodologías ágiles con respecto a las tradicionales (no ágiles). Estas diferencias que afectan no sólo al proceso en sí, sino también al contexto del equipo así como a su organización.

3.2 Metodología de Programación Extrema



Copyright © 2003 United Feature Syndicate, Inc.

La programación extrema (que en adelante se denominara XP) es una disciplina de desarrollo de software "relativamente" nueva. Tiene características totalmente diferentes a la planificación tradicional de cualquier proyecto de software; Si bien usa un sistema iterativo de desarrollo, el paradigma es totalmente diferente a lo "estándar".

No existe una metodología Universal de desarrollo de software que se adapte a todos los procesos de desarrollo, debido a las características especiales de cada proyecto, en consecuencia se ha de escoger la metodología que mejor se adapte a los objetivos del proyecto, siendo necesario realizar algunas modificaciones para aplicarlo al desarrollo de este proyecto.

Características de XP:

La programación extrema da por supuesto que es imposible prever todo antes de comenzar a programar; es imposible o si lo fuera es demasiado costoso e innecesario, ya que muchas veces se gasta demasiado tiempo y recursos en cambiar la documentación de la planificación para que se parezca al código. Para evitar esto, XP intenta implementar una forma de trabajo donde se adapte fácilmente a las circunstancias.

Básicamente consiste en trabajar estrechamente con el cliente, haciendo pequeñas iteraciones (mini-entregas), cada dos semanas, donde no existe mas documentación que el código en sí³⁸; cada versión contiene las modificaciones necesarias según el cliente vaya retroalimentando el sistema (por eso es necesaria la disponibilidad del cliente durante todo el desarrollo).

Para suplir la falta de requisitos, casos de uso, y demás herramientas; XP utiliza **historias de usuarios**, la historia de usuario es una frase corta que representa alguna función que realizara el sistema. Cada historia de usuario no puede demorar en desarrollarse más de una semana, si así lo requiera, debe segmentarse.

Es requisito para XP definir un **estándar en el tipo de codificación**, esto hace que los programadores tengan definido ya el estilo de programación y no que cada uno programe a su estilo.

Al ser un modelo iterativo e incremental, la metodología de programación extrema visto globalmente se basa en tres pasos fundamentales del modelo de prototipado clásico, escuchar al cliente, construir el prototipo y probar el prototipo con el cliente, pasos que están íntimamente ligados al modelo de programación extrema.

³⁸ Este es el punto más débil de la metodología de programación extrema, debido a que la administración y mantenimiento de la aplicación por un equipo diferente al desarrollador podría llegar a ser complicado y caótico.

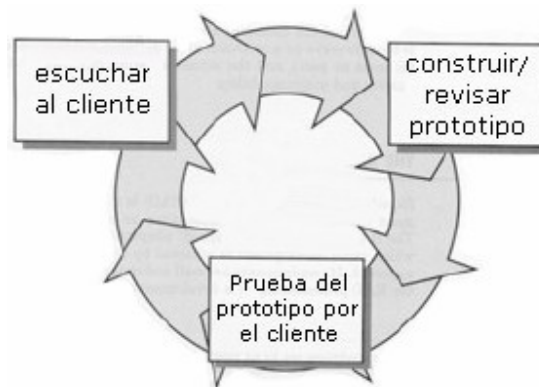


FIGURA No 14, Modelo de desarrollo del Prototipado Clásico³⁹

La **PRUEBA EN CADA ITERACIÓN** es más que importante; de eso se trata este paradigma de programación, corregir mientras se programa. De esta forma se van cubriendo todos los baches de cada versión que aparezca.

El código no es de nadie, todo el equipo puede manipular el código que existe, de esta forma cada pareja puede mejorar cada sección de código que utiliza, esto requiere de una prueba del mismo y la re-implementación en el sistema general.

Cada dos semanas se entrega una versión al cliente, que lo verifica, realiza el feedback y se continúa el desarrollo; este ciclo continua hasta que el sistema cumpla con las expectativas del cliente, acto que concluirá el proyecto.

Los pilares fundamentales de la metodología de Programación Extrema son:

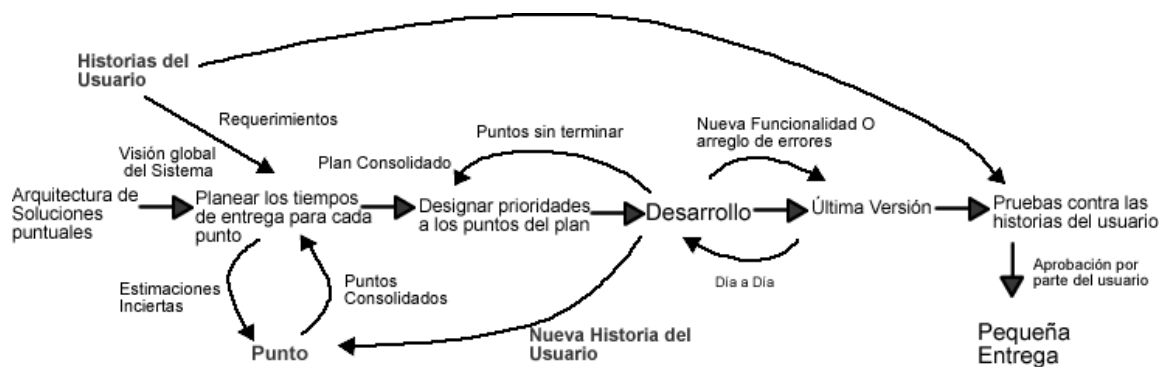
- Desarrollo iterativo e incremental
- Pruebas unitarias continuas
- Programación por parejas
- Frecuente interacción del equipo de programación con el cliente
- Corrección de todos los errores
- Refactorización del código
- Propiedad del código compartida
- Simplicidad en el código

³⁹ Pressman, Roger. Ingeniería del Software, Un enfoque práctico. McGraw Hill, 5ª edición. Madrid, 2002

3.3 El modelo de desarrollo de la aplicación

El modelo de desarrollo se centra en realizar pequeñas entregas del producto al usuario donde están resueltos los requerimientos hechos por el, requerimientos que están consignados en las historias de usuario.

El modelo se inicia con la visualización de las historias de usuario por parte del equipo desarrollador con el fin de crear una visión global del sistema, en estos documentos están plasmados los requerimientos que la división de servicios de biblioteca espera de la aplicación.



ESQUEMA No 13, Modelo de desarrollo de la Metodología de Programación Extrema⁴⁰

⁴⁰ Imagen tomada del sitio oficial de extreme programming en Internet <http://www.extremeprogramming.org/map/project.html>

ETAPA 2

DESARROLLO

En este capítulo se presenta la forma de cómo afrontar la preparación del proyecto, se comienza con la definición del problema a resolver, se especifican los objetivos del proyecto y se establece una línea de desarrollo del software.

4.1 Definición del problema

La coordinación de servicios al público de la biblioteca central de la Universidad Industrial de Santander ha observado con detenimiento el desarrollo de las nuevas tecnologías de la información, y ha visto en ellas la posibilidad de acercar a sus usuarios los servicios que actualmente presta a la comunidad, entre los servicios con los que se cuentan están la compra, renovación, archivado, catalogación y préstamo de material bibliográfico que adquiere la Universidad Industrial de Santander.

Como primer acercamiento a estas nuevas tecnologías la biblioteca busca llevar el servicio de búsqueda de libros a los dispositivos móviles que posean los usuarios, con esta propuesta se busca que la comunidad en general pueda acceder a información de cuales materiales bibliográficos posee la universidad y su disponibilidad para sacarlo en calidad de préstamo, esto con el fin de darles la oportunidad a los usuarios de biblioteca la posibilidad de consultar el catálogo bibliográfico de la universidad desde cualquier punto geográfico donde el usuario cuente con un punto de acceso inalámbrico a la red Internet, sin que sea necesario su desplazamiento hasta las dependencias de la Universidad.

Aprovechando el hecho de que Bucaramanga posee la primera red de acceso inalámbrico a Internet desde cualquier punto del área metropolitana, la aplicación de búsqueda de libros debe aprovechar el potencial que se deriva del uso de esta tecnología, al permitir que los usuarios hagan uso de este medio para acceder a la información de la biblioteca.

4.1.1 Síntomas, causas y consecuencias del problema

| Síntomas | Causas | Consecuencias |
|--|--|--|
| El desplazamiento hasta las dependencias de la biblioteca o hasta alguna Terminal fija con acceso a Internet que tiene que realizar el usuario para llevar a cabo la consulta. | El modulo de consulta actual se encuentra en la Web y sus controles están enfocados a ser utilizados por terminales fijos. | El usuario tiene que desplazarse hasta una Terminal fija para realizar la consulta. |
| El buscador de la biblioteca disponible en Internet ⁴¹ , no admite una interfaz en dispositivos móviles. | El diseño del buscador de la biblioteca esta diseñado para los navegadores de escritorio ⁴² , y no contempló la posibilidad de efectuar búsquedas desde otros dispositivos con recursos más limitados. | Los usuarios deben realizar la búsqueda desde dispositivos que tengan instalados estos navegadores de escritorio. |
| La espera en las terminales de consulta se hace evidente en horas o temporadas de alta congestión. | El número de estudiantes que demandan terminales de consulta no es constante a través del tiempo, aumenta o disminuye según la temporada académica. | Algunos usuarios desisten de acceder a la Terminal de consulta. |
| La información bibliográfica esta limitada y no esta disponible para todos los potenciales usuarios. | El formato actual de implementación no contempla el uso de información de acceso público de otra forma distinta a la lectura en páginas Web dado que sólo se encuentra representada en formato HTML para ser interpretada por navegadores Web. | La biblioteca central limita su cobertura en la cantidad de usuarios, equipos y sistemas que acceden a la información. |

TABLA No 3, Tabla que muestra los síntomas causas y consecuencias del problema planteado

⁴¹ LIBRUIS Primer prototipo de consulta al sistema de biblioteca vía Web de la Universidad Industrial de Santander

⁴² Internet Explorer (IE), Netscape Navigator, Mozilla, FireFox y otros.

4.2 OBJETIVOS

OBJETIVO GENERAL

Diseñar y desarrollar una aplicación para la consulta de información bibliográfica correspondiente a la biblioteca central de la Universidad Industrial de Santander, con el fin de permitir a los usuarios llevar a cabo estas consultas a través de dispositivos móviles.

OBJETIVOS ESPECÍFICOS

- Diseñar una arquitectura Software basada en Servicios Web que permita la consulta bibliográfica de los volúmenes registrados en la base de libros de la biblioteca central de la Universidad Industrial de Santander. Esta consulta utilizará parámetros como el tipo de documento (tesis, libro, analítica) y otros como autor, título, materia, ISBN, número de Inventario y año de edición.
- Diseñar y desarrollar una aplicación (Servicio Web) que responda a las peticiones de consulta bibliográfica de los programas cliente.
- Diseñar y desarrollar una aplicación para dispositivos móviles⁴³ que permita al usuario consumir⁴⁴ el servicio Web de consultas bibliográficas de la biblioteca Central de la Universidad Industrial de Santander.
- Publicar el Servicio Web en los servidores UDDI gratuitos disponibles para esto, para permitir que usuarios registrados puedan acceder a la información bibliográfica de la biblioteca central.

⁴³ PDAs (Asistente Personal Digital) y teléfonos celulares Compatibles con tecnología JAVA 2 MICRO EDITION.

⁴⁴ El término consumo se refiere a la utilización del servicio Web por parte de un cliente.

4.3 JUSTIFICACIÓN Y ALCANCE DEL PLAN PROPUESTO

DESCRIPCIÓN DE OBJETIVOS

El proyecto está orientado hacia dos tecnologías en constante crecimiento y que han tenido una gran acogida en la industria del software en el siglo XXI, que son los servicios Web y Los dispositivos móviles.

Los servicios Web son una tecnología impulsada por grandes empresas en el ámbito del software como son Microsoft, IBM, Sun, además por la empresa líder mundial en teléfonos celulares NOKIA, entre otras.

Se utiliza la tecnología de servicios Web por que gracias a su baja cohesión permite ser consumido no sólo por la aplicación en el dispositivo móvil sino también por:

Otras aplicaciones en diferentes lenguajes o con diferentes plataformas.

Distintos servicios Web residentes en otros servidores.

Usuarios desde un navegador de Internet.

Otra de las grandes ventajas que admite esta tecnología es la escalabilidad hacia consultas con un mayor alcance, como por ejemplo hacia la consulta de otras bibliotecas ajenas a la de la institución a través del mismo servicio, sin tener que cambiar necesariamente la aplicación que lo consume, en nuestro caso particular la aplicación residente en el dispositivo móvil.

Existen dos aspectos principales que garantizan un mercado viable para la implementación de la aplicación tanto dentro del campus universitario como para usuarios externos de este:

El crecimiento vertiginoso que ha tenido el número de usuarios que utilizan telefonía móvil y que por ende están en la capacidad de utilizar los servicios de transmisión de datos.

“De acuerdo con las cifras presentadas por la Superintendencia de Industria y Comercio a septiembre de este año el número de usuarios se elevó a 18’331.450, lo

que representa un crecimiento del 75 por ciento en los nueve primeros meses del año."⁴⁵

El hecho de que las empresas de telecomunicaciones están promoviendo el consumo masivo de transmisión de datos, permiten que este tipo de servicios se encuentre disponible para la comunidad universitaria en general.

4.4 VIABILIDAD E IMPACTO

Este trabajo de investigación cuenta con documentación limitada, teniendo como referencia la información disponible al interior de la universidad, en la biblioteca existe una tesis de maestría⁴⁶ en informática que describe los servicios Web basados en el protocolo SOAP que sirve como una primer aproximación a los servicios Web. Para subsanar esta deficiencia se cuenta con información y trabajos relacionados a nivel internacional, lo cual nos brinda el soporte necesario para la investigación y creación de la aplicación.

Se dispone de recursos técnicos para el soporte del desarrollo de la solución software dada como son los lenguajes de programación, manejadores de base de datos, computadores, Internet, etc.

Se cuenta con los equipos requeridos, como son los servidores de la división de servicios de información⁴⁷ que alojarían el servicio Web, para realizar las pruebas debidas.

En el ámbito de los servicios Web no existe en la universidad precedente alguno de la utilización de esta tecnología debido a su reciente creación y especificación, pero se busca a través de la investigación a fondo, conocer las ventajas y desventajas que éstos presentan y en qué casos se deben utilizar, para dar el soporte teórico en castellano a las posibles aplicaciones futuras.

⁴⁵ www.eltiempo.com/econ/2005-10-12/ARTICULO-PRINTER_FRIENDLY-_PRINTER_FRIENDLY-2565431.html

⁴⁶ Programación de Servicios Web basados en protocolos SOAP, Oswaldo Augusto García.

⁴⁷ Entidad encargada al interior de la Universidad Industrial de Santander, de la administración y seguridad de la información que utilizan los diferentes servidores de la universidad.

Además el proyecto hace parte de un grupo de trabajos pioneros en Computación Móvil en la Escuela, por lo que el proyecto dejará un aporte significativo en el área para futuros desarrollos con esta tecnología.

Por otro lado se busca prestar un mejor servicio a la comunidad usuaria de la biblioteca central de la Universidad Industrial de Santander dándole el don de la ubicuidad a la información requerida.

El objetivo de esta fase es obtener una clara comprensión del problema a resolver, abstraer las necesidades del usuario y derivar de ellas las funciones que debe realizar el sistema. Los requisitos del usuario definen los servicios que el sistema debe proporcionar y las restricciones y condiciones de uso de los mismos. Es decir, en esta etapa se trata de expresar la estructura de la solución así como los beneficios y la funcionalidad del software.

Se pretende intensificar la recopilación de requerimientos indagando los requisitos del software para obtener como resultado la comprensión de la naturaleza de los programas que han de construirse.

Para esto es necesario tener en cuenta dos tipos de requisitos:

Requisitos funcionales que están ligados a los resultados que desea obtener el usuario a partir de los datos que este ingrese al sistema, además de plantear las funcionalidades o características operativas que debería/podría ofrecer el sistema.

Requisitos no funcionales que incluyen restricciones sobre los funcionales y aspectos de calidad del sistema⁴⁸, como por ejemplo, mantenibilidad –facilidad para que el sistema evolucione y se modifique según las necesidades futuras que se presenten, escalabilidad –posibilidad de incrementar sustancialmente el número de usuarios u otros parámetros, facilidad de uso, confiabilidad, portabilidad, y disponibilidad, que no pueden ligarse a funciones concretas dentro del sistema.

⁴⁸ León Serrano, Gonzalo. Ingeniería de Sistemas de Software. Primera Edición. Madrid, 1996.

4.5 ESPECIFICACIÓN Y ANÁLISIS DE REQUERIMIENTOS

A continuación se presentan los requerimientos que los usuarios hacen al equipo desarrollador, en los cuales están plasmadas las características que los usuarios esperan del sistema.

4.5.1 Requerimientos de la comunidad

Historia de Usuario.

Son la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

Los siguientes párrafos son las historias de usuario que describen los requerimientos que hace al sistema la coordinación de Servicios al público de biblioteca:

- La coordinación de servicios al público de la biblioteca central busca llevar a la plataforma inalámbrica, el servicio de búsqueda de libros y disponibilidad de los mismos, la aplicación debe permitir a los usuarios realizar la consulta de cuales documentos posee el archivo de biblioteca, y su disponibilidad para ser consultados o sacarlos de la biblioteca en calidad de préstamo.
- El sistema debe permitir al usuario buscar cuales documentos tiene a su disposición la biblioteca utilizando el método de búsqueda de coincidencias a través de palabras claves que el desee buscar.
- El sistema debe permitir discriminar la búsqueda por autor, titulo, materia, además del criterio del tipo de documento a buscar. (Libro, Tesis, analítica)

- El sistema debe Permitir que el usuario realice búsquedas de libros a partir del número ISBN⁴⁹ de este, proporcionándole los datos básicos así como su disponibilidad de sus ejemplares para el préstamo.
- El sistema debe permitir que el usuario contando con el número de inventario que la biblioteca le ha asignado al ejemplar, buscar cual es el libro que corresponde a ese ejemplar.
- La aplicación debe permitir que a partir de un número de inventario del ejemplar ver el estado de la disponibilidad de los otros ejemplares que pertenecen al libro, además de los datos básicos de este.

4.6 Implementación

El objetivo de la implementación es crear una herramienta software que sea el resultado de la puesta en marcha del proceso de desarrollo. Básicamente se busca el cumplimiento de los objetivos propuestos, el uso de la arquitectura, además de verificar que cada uno de los puntos propuestos en el diseño detallado se haya cumplido.

4.6.1 Vista conceptual

La vista conceptual se usa para definir los elementos funcionales y no funcionales del sistema, además de la visión que tiene los usuarios de la aplicación, esta vista muestra los sistemas y módulos que tiene la aplicación, además de la funcionalidad de cada uno de ellos.

La primera parte de la vista conceptual consiste en dividir el sistema en subsistemas y sus relaciones, esta relación entre los subsistemas se registrará bajo el modelo de repositorio, el cual dice que cada subsistema mantiene su propia base de datos y pasa datos explícitos a los otros subsistemas.

⁴⁹ ISBN International Standar Book Number, Número Internacional Estándar del libro, es un código único internacional que identifica y particulariza a un libro de los otros.

Basado en esta primera descomposición del sistema se procede a la descomposición modular, donde los subsistemas son descompuestos en módulos. Un módulo es un componente del sistema que proporciona servicios a otros componentes y no se puede considerar como parte separada del sistema.

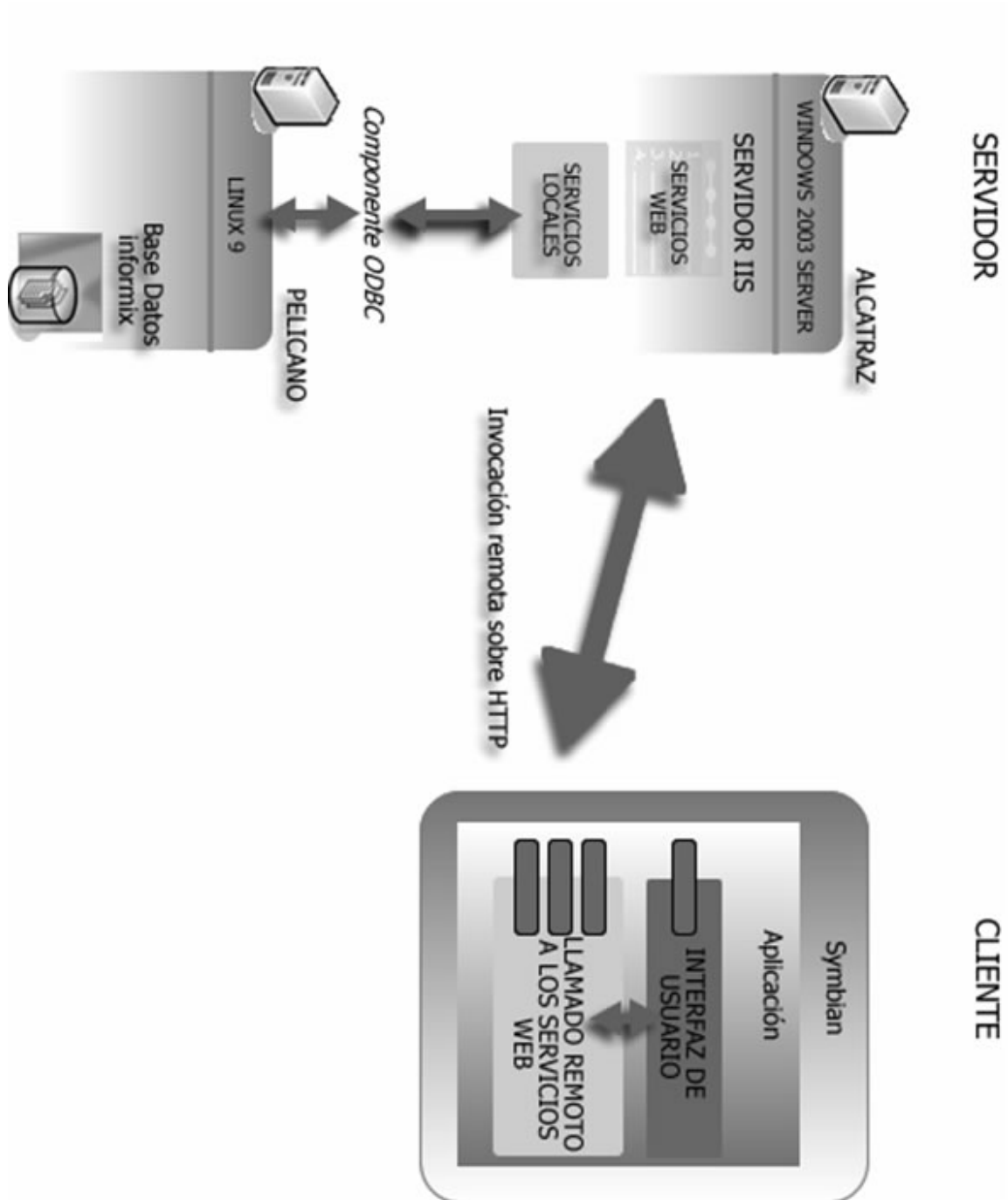
Estructuración Interna del sistema

La aplicación consiste en dos subsistemas la *aplicación para el dispositivo móvil* y la *aplicación para el servidor*, estos subsistemas se comunican a través de otro sistema intermedio de mayor envergadura que puede llegar a ser Internet o intranet.

Tomando como punto de referencia la aplicación cliente (verde) el sistema muestra en pantalla el modulo conocido como la interfaz de usuario, mediante la manipulación de este modulo el cliente accede a las funciones programadas en su dispositivo para obtener la información de la biblioteca.

Una vez el cliente haya alcanzado unos de los módulos de consulta (General, ISBN, Número de Inventario) y decida realizar una consulta, el sistema creara un documento XML con su petición, el cual será enviado al servidor Alcatraz para que este equipo lo conteste de la manera más adecuada.

Cuando Alcatraz haya realizado la consulta a las bases de datos residentes en Pelicano, creara un documento XML con la respuesta, y esta será enviada de vuelta al cliente, el cual mediante la interfaz de usuario mostrara el resultado en la pantalla del dispositivo.



ESQUEMA No 14, Vista Conceptual de la aplicación, en el se muestran los diferentes componentes que intervienen, en la parte superior se encuentran los servidores de Biblioteca, Base de datos y Servidor Web, y en la parte inferior se encuentra el componente cliente de la aplicación, la flecha bidireccional central es el medio que usan para comunicarse, en este caso Internet y la red Celular

4.6.2 Vista Física

Esta vista muestra la distribución física de los equipos que conforman la aplicación, es decir, los equipos hardware que componen la solución. Para interpretar el gráfico hay que localizar primero los equipos cliente, PDA y Celular (derecha de la imagen). Estos equipos tienen instalada en su interior el programa cliente, cada uno de ellos en su respectivo lenguaje.

El usuario haciendo uso de este programa, realiza peticiones a los servicios Web XML residentes en el servidor Alcatraz (izquierda Azul), donde este a su vez consulta la base de datos (PELICANO) en busca de los libros, una vez encontrado lo que se busca se realiza el proceso contrario de alcatraz al cliente enviándole la respuesta a su petición.

Por ejemplo cuando el cliente hace una consulta por ISBN de un libro de biblioteca, la aplicación cliente hace un proceso interno de serialización a lenguaje XML de los parámetros de su petición. Una vez creado el documento XML a partir de esta, el programa localiza el equipo en Internet que hospeda el servicio Web que le da respuesta a su consulta por ISBN, esta localización se hace mediante la dirección Web del servidor que el cliente previamente conoce, se le concede conocer esta dirección al cliente con el fin de liberarlo del trabajo de buscar al servidor en Internet.

En este punto el cliente tiene el documento XML con la petición de su libro, además de conocer la dirección y el servicio Web XML que responde a su petición, con estos dos datos el cliente envía su petición por el medio que tenga disponible (WIFI-GPRS) al servidor Alcatraz, que es el equipo que posee el servicio Web XML capaz de responderle.

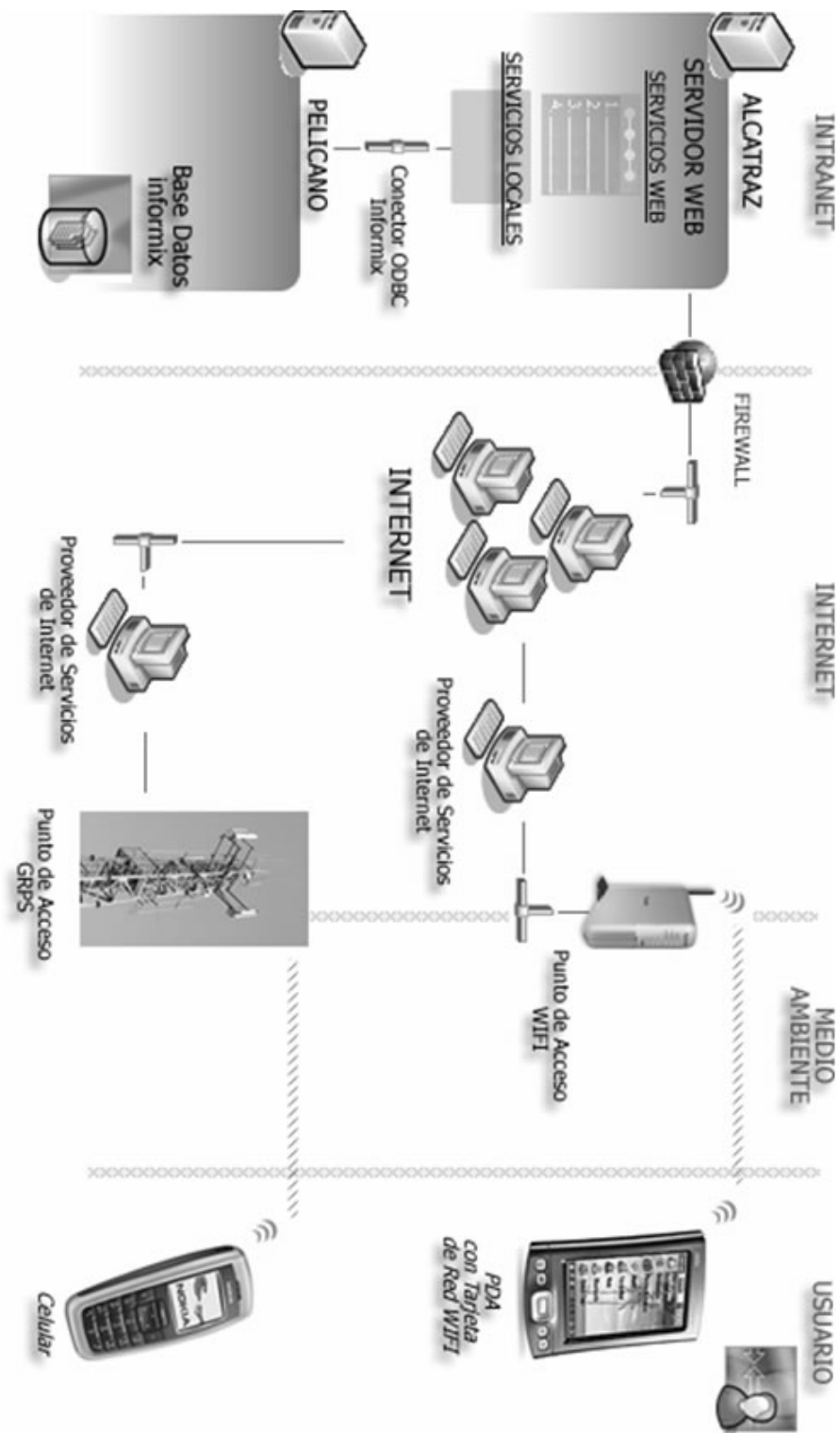
El recorrido del documento inicia en el equipo del cliente, una vez creado el documento XML con la petición, el dispositivo se comunica con su proveedor de acceso a Internet, y le envía vía medio inalámbrico el documento XML, el proveedor de servicios se encarga de cambiar el documento de medio inalámbrico a la red Internet, una vez en Internet el documento XML se dirige hacia Alcatraz, el cual es el equipo identificado con la dirección IP que se le fue asignado que buscara.

Cuando el documento XML llega al servidor Web del equipo Alcatraz, el servicio Web que se invocó en la petición lee el documento en formato XML que le llegó, y realiza los procesos internos necesarios para su contestación, en este caso específico utiliza un conector ODBC⁵⁰ informix para leer los datos de los libros del servidor Pelicano, que es el lugar físico donde se encuentran las la base de datos de biblioteca.

Una vez leídos los datos necesarios el servicio Web XML crea un objeto para contestar la petición del cliente, una vez creado el objeto este se le pasa al serializador XML que reside en le Servidor Web, el cual lo traduce a formato XML.

Ya traducido el nuevo documento XML se realiza el recorrido inverso, con la finalidad de hacer llegar al cliente la respuesta a su petición inicial.

⁵⁰ ODBC son las siglas de Open DataBase Connectivity, un estándar de acceso a Bases de Datos desarrollado por Microsoft Corporation.



ESQUEMA No 15, Vista física de la aplicación, Esta vista muestra la distribución física de los equipos que conforman la aplicación. De arriba hacia abajo se cuenta desde el servidor donde reside el servicio Web XML hasta el cliente del servicio, representados por la PDA y el Celular

4.6.3 Vista Lógica

La vista lógica muestra las estructuras internas de la aplicación y su comportamiento ante los diferentes requerimientos, permitiendo hacer un primer análisis de los procesos de cada una de las aplicaciones (cliente y servidor).

Tomando como referencia el gráfico No X se puede observar que el componente de Alcatraz conocido como el servidor Web IIS 6.0 es el encargado de alojar el servicio Web XML,



ESQUEMA No 16, Diagrama lógico de los métodos existentes en el servicio Web XML

Cliente Pocket

FormPrincipal

CrearBaseDatos () :void
 CrearTablas () :void
 BorrarBaseDatos () :void
 ExistenLibros (int) :bool
 LeerDireccionIp () :string
 BorrarCrearBdCrearTablas () :void
 MenuFormularios :void

FormGeneral

Consultar_Click () :void
 Borrar_Click () :void
 InvocarSwBusquedaP (string, 3 int) :void
 EscribeFormListado (int) :void
 BorraTit_EscribeNuevosTit () :void
 ReescribePalClave () :void
 ConsultarIp () :string
 ValidarTexto (string) :string
 ValidarOpciones (6 bool) :int

FormISBN

ValidarFormato (string) :void
 VectorEjeToStringEje (string[]) :string
 ConsultarIp () :string

FormNroInventario

Consultar_Click () :void
 ValidarFormato (string) :int
 VectorEjeToStringEje (string[]) :string
 ConsultarIp () :string

FormLibrosConsultados

FormLibrosConsultados_Load :void
 ListaTitulos_SelectIndexChanged (string) :void
 Consultar_click () :void
 Borrar_Click () :void
 Consultarlibros () :void

FormIP

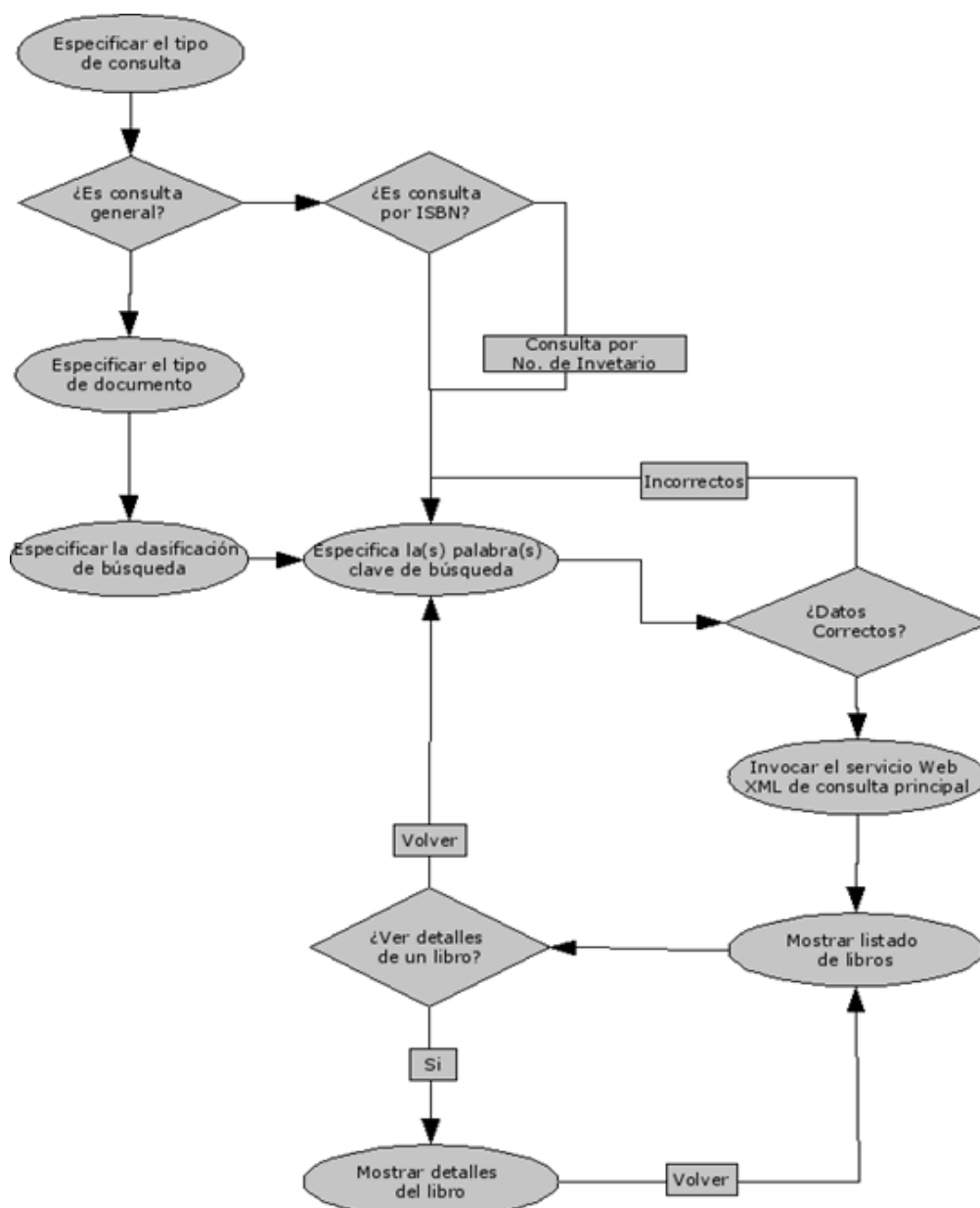
FormIp () :void
 Cambiar_Click () :void
 ConsultarIp () :string
 Prueba_Click () :void

ESQUEMA No 17, Diagrama de métodos del programa cliente para el SO Windows CE

4.7 Diseño Detallado de las aplicaciones

4.7.1 Esquema de funcionamiento de los módulos de Consulta

La biblioteca de la Universidad Industrial de Santander ofrece el servicio de consultas bibliográficas a través de la página de Internet <http://alcatraz.uis.edu.co/biblioteca/> donde se pueden realizar la búsqueda especificando ciertos parámetros. Estos mismos parámetros son especificados para la consulta a través del dispositivo móvil.



ESQUEMA No 18, Desarrollo del Flujo de consulta bibliográfica de las aplicaciones en los diferentes clientes

Como se puede apreciar en el diagrama se requieren estos parámetros para iniciar una consulta. Inicialmente se escoge un tipo de consulta que puede ser de tres tipos consulta general, por ISBN o por el número de inventario. En la consulta general existen combinaciones de tipos de filtros de búsqueda:

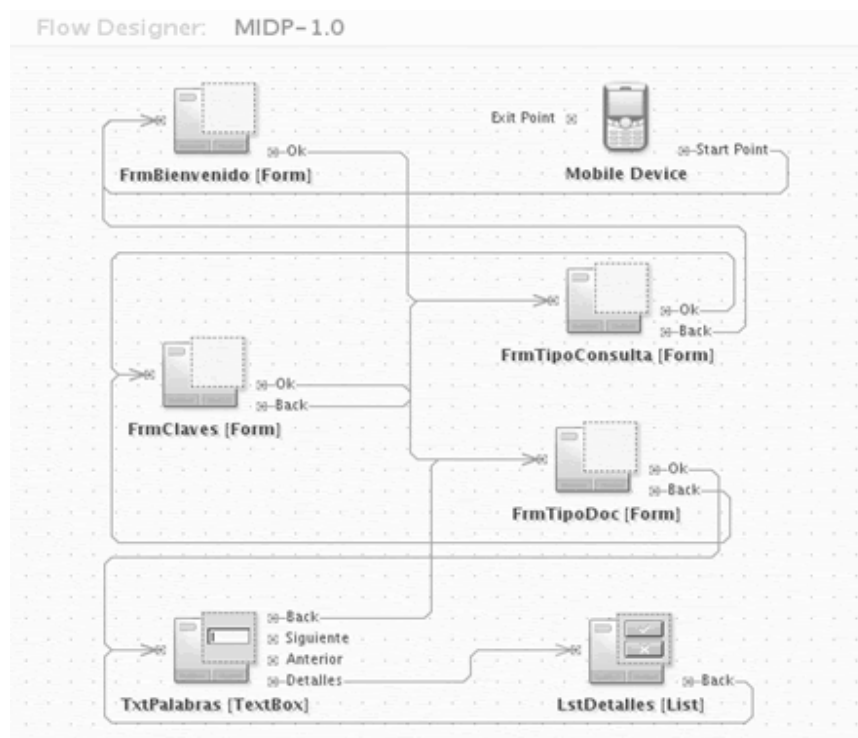
Tipo de documento

Que especifica la clase de documento que se va a buscar, puede ser libro, tesis o analítica y debe ser escogida al menos una opción.

Clasificación de búsqueda

Especifica el tipo de palabras que se está buscando, es decir, si corresponde al autor, título o materia y al igual que la opción anterior debe tomarse al menos una de las opciones.

Una vez especificados los parámetros de búsqueda la aplicación invoca el servicio Web obteniendo de esta manera el resultado de la consulta donde se muestra sólo los títulos de los libros que concuerdan con las características especificadas.



ESQUEMA No 19. Diagrama de flujo del programa cliente J2ME en el diseñador de aplicaciones Java MicroEdition de NetBeans

El diseño de la aplicación J2ME esta basado en el diagrama de flujo anterior que fue creado en la herramienta NetBeans 4.1 que ofrece la capacidad de crear interfaces arrastrando y colocando objetos de una forma totalmente visual.

A continuación se muestran las imágenes de las aplicaciones cliente del servicio Web XML de consultas bibliográficas, en ellas se presentan los formularios de listado de libros y los detalles del libro, cada grupo de imágenes son de las respectivas aplicaciones cliente.

Ciente J2ME

Pantalla con el listado de libros



Pantalla con los detalles de un libro



FIGURA No 15-16, Imágenes correspondientes a la interfase del cliente J2ME, la primera imagen es el listado de resultados después de haber realizado la consulta, la segunda imagen muestra los detalles del libro que se escogió en el listado de libros

Cliente Windows Mobile

Pantalla con el listado de libros



Pantalla con los detalles de un libro



FIGURA No 17-18, Imágenes correspondientes a la interfase del cliente Windows Mobile

| Nro Inventario | Título del documento | Tipo de documento | Año de Edición |
|----------------|--|-------------------|----------------|
| 82555 | CALCULO CON GEOMETRIA ANALITICA / EDWIN J. PURCELL, DALE VARBERG. | LIBRO | 1993 |
| 104952 | CALCULO CON GEOMETRIA ANALITICA / Edwin J. Purcell, Dale Varberg | LIBRO | 1993 |
| 42757 | BERKELEY PHYSICS COURSE. | LIBRO | 196 |
| 14806 | PRINCIPLES OF MICROWAVE CIRCUITS. | LIBRO | 1965 |
| 28716 | CURSO DE FISICA DE BERKELEY. | LIBRO | 1970 |
| 75027 | CALCULO CON GEOMETRIA ANALITICA / Edwing J. Purcell y Dale Varberg | LIBRO | |
| 128999 | CALCULO | LIBRO | 2001 |
| 23574 | CHEMISTRY & CHEMICAL REACTIVITY / JOHN C. KOTZ, KEITH F. PURCELL. | LIBRO | 1991 |
| 20571 | BERKELEY PHYSICS COURSE | LIBRO | 1996 |

FIGURA No 19, Pantalla de consulta del cliente ASP.NET del servicio Web XML

A partir de esta consulta se puede obtener detalles más trascendentales del material bibliográfico que son:

Número de clasificación, Título, Autor Principal, Tipo de material bibliográfico, Ubicación y estado de los ejemplares.

4.7.2 Esquema de funcionamiento de los módulos de Configuración del Cliente

En el diseño de la aplicación se tuvieron en cuenta otros factores diferentes a la búsqueda de los libros, estos factores tienen que ver más con permitir que la funcionalidad de la aplicación no se vea afectada por cuestiones técnicas de la misma, vale la pena decir que las decisiones que se tomaron al momento de diseñar la aplicación están encaminadas al ahorro de costos, y al ahorrar en lo posible el uso del canal de datos inalámbrico, que además de ser costoso en cuestión de presupuesto, sus velocidades de transmisión son relativamente bajas, por estas razones la aplicación de consulta bibliográfica tiene ciertos parámetros que el usuario puede configurar para mejorar el desempeño y la rentabilidad de la aplicación.

En la aplicación se pueden especificar parámetros de configuración que se utilizan para adaptar la aplicación cliente a las capacidades del dispositivo. Las siguientes son los parámetros que se pueden especificar de manera manual:

Ubicación del servicio Web XML

URL de la ubicación del servicio Web que es definida por la universidad y que puede variar en un momento determinado por cambios técnicos del servidor de biblioteca, entre estos cambios están el cambio de dirección IP, el cambio de carpeta en la cual se aloja el servicio Web XML, e incluso el cambio de nombre del propio servicio Web XML.

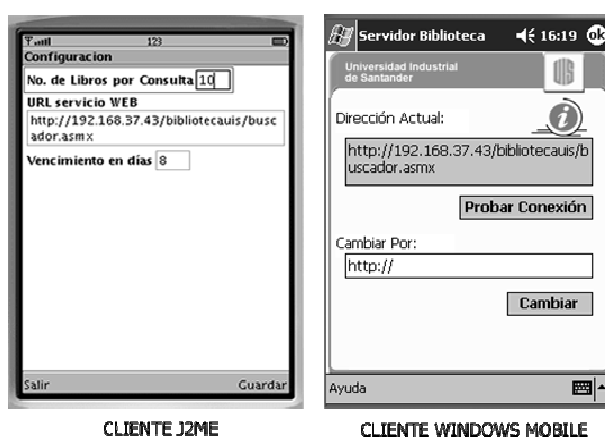


FIGURA No 20, Pantallas de la aplicación cliente donde el usuario puede cambiar varios parámetros de la aplicación, tales como la dirección IP del Servidor de biblioteca, la pantalla de la izquierda corresponde al cliente J2ME, mientras que el de la derecha es la del cliente de Windows Mobile

Cantidad de registros

El número de registro de material bibliográfico puede exceder las capacidades del dispositivo o sencillamente se hace innecesario transmitir toda la información en una sola llamada al servicio. Es por esto que se hace una división por lotes de los resultados, donde en cada uno de los lotes de resultados debe haber un número de registros máximo que devuelve el servidor. De acuerdo a las capacidades y/o gustos del usuario se puede establecer una cantidad óptima de registros para que las consultas sean más ágiles y económicas en términos de transmisión de datos.

Borrado de registros antiguos

La aplicación guarda los registros de los resultados de las consultas realizadas por el usuario en la memoria no volátil del dispositivo, con el fin de obtener los resultados de las consultas más utilizadas de manera más rápida, ya que una vez invocado el servicio no se necesita volver a consumirlo mientras la información permanezca en el dispositivo. Esta información puede llegar a ser obsoleta transcurrido un tiempo determinado por esta razón se establece el número de días que la información permanece disponible en el dispositivo, y una vez que se haya cumplido el límite fijado de días, la aplicación se encarga de borrar los registros no actualizados.

Guardar datos del libro en la base de datos local

Como se menciona anteriormente la aplicación cliente para J2ME guarda los registros de los resultados de las consultas en la memoria no volátil del dispositivo de manera automática.



FIGURA No 21, Pantalla donde se muestran los detalles del libro con la opción de guardar resaltada (cliente Windows Mobile)

El caso del programa cliente de Windows Mobile, el usuario cuenta con la opción de guardar los datos del libro que consulto, esta opción se lleva a cabo haciendo clic sobre el botón guardar presente en la pantalla de detalles del libro, al momento de ejecutar esta opción el programa guarda los datos del libro dentro de la base de datos local.

Esto permite que el usuario haciendo uso de la opción "Otras -> libros consultados" observe un listado de todos los libros que haya consultado y guardado hasta el momento, sin que sea necesario estar conectado a un punto de acceso a Internet. Una vez el usuario este haciendo uso de esta pantalla, encontrará las opciones de ver los detalles del libro que previamente había consultado, o de borrar la información del libro que previamente había guardado.

4.7.3 Esquema de funcionamiento de la Aplicación Servidor

Es la aplicación que se aloja en el servidor Web de biblioteca y su trabajo es responder los requerimientos de los clientes del servicio Web XML.

En este punto se explica el esquema entidad-relación de la base de datos de biblioteca, las tablas más relevantes para la consulta bibliográfica, además de explicar brevemente el desarrollo de una consulta bibliográfica

La base de datos de biblioteca posee en su actualidad 128 tablas para guardar toda la información relacionada con las actividades de la biblioteca, como el caso de uso de esta aplicación es una consulta bibliográfica, únicamente se muestran en el esquema las tablas más relevantes de la consulta.

Los nombres de las tablas así como sus campos poseen un esquema de nombres bastante particular, al inicio puede ser complicado crearse una visión de la base de datos, pero una vez se haya familiarizado con los nombres el diagrama de relaciones se hacen más evidentes al lector.

El esquema de nombres de las tablas se basa en utilizar una palabra clave seguida de las tres primeras letras de la relación que guarda dicha tabla por ejemplo:

PalabrasAutTitMat: Palabras que se relacionan con los autores, títulos y materias del material bibliográfico.

AutTitMat_PalAutTit: Relación entre el código de la palabra y el autor, título o materia del material bibliográfico.

AutoresTitMat: Tipo de relación entre el número de identificación y el material bibliográfico.

MatBib_Aut: Relación de autor entre número de identificación y el material bibliográfico.

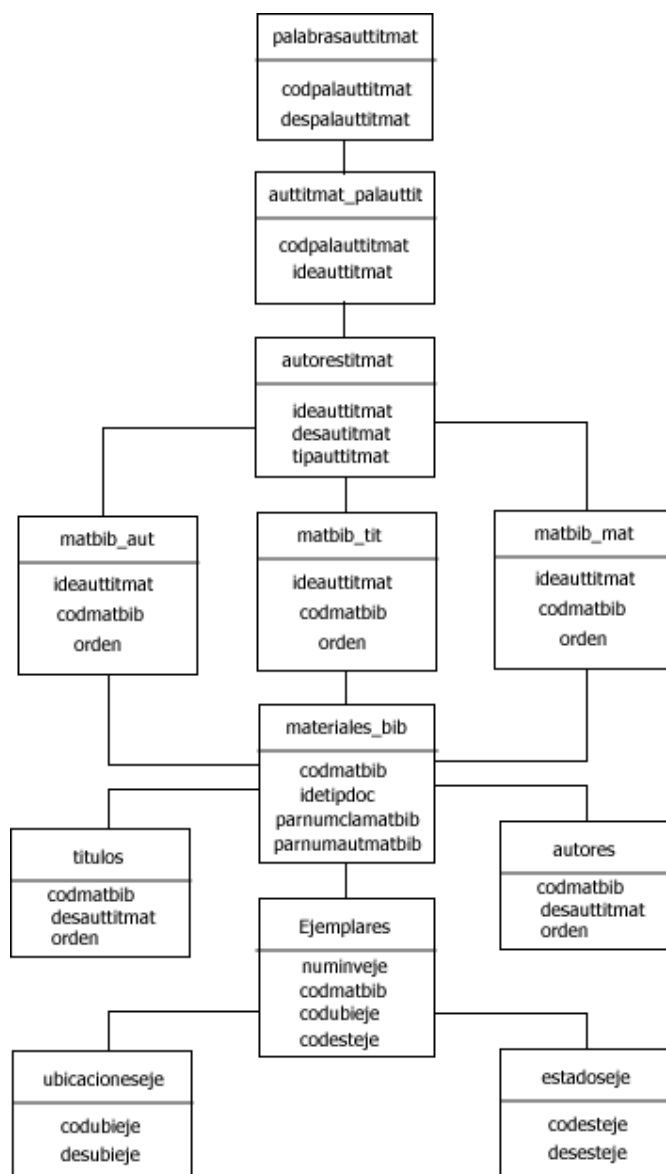
MatBib_Tit: Relación de título entre número de identificación y el material bibliográfico.

MatBib_Mat: Relación de tipo materia entre número de identificación y el material bibliográfico.

MaterialesBib: Esquema del material bibliográfico.

Ejemplares: Esquema de los ejemplares del material bibliográfico.

4.7.4 Diagrama Entidad–Relación de la base de datos de biblioteca



ESQUEMA No 20, Esquema de las tablas más relevantes de la base de datos del servidor de Biblioteca

Para entender el esquema hay que situarse en la parte superior del esquema, la primera tabla que aparece se conoce como PalabrasAutTitMat⁵¹, la cual guarda la relación entre la palabra y un código interno que se le asigna por parte de biblioteca.

⁵¹ Tabla con las palabras que se relacionan con los autores, títulos y materias del material bibliográfico.

Una vez hallado el código de la palabra, se busca en la tabla AutTitMat_PalAutTit⁵² cuales números de identificación (ideauttitmat) tiene relación con el código de la palabra.

Teniendo los números de identificación se buscan en la tabla AutoresTitMat que tipo de relación (relación de titulo, de autor o de materia) tienen con el material bibliográfico. Obtenidos estos datos se procede a buscar en cada tabla en particular⁵³ el código del material bibliográfico (codmatbib).

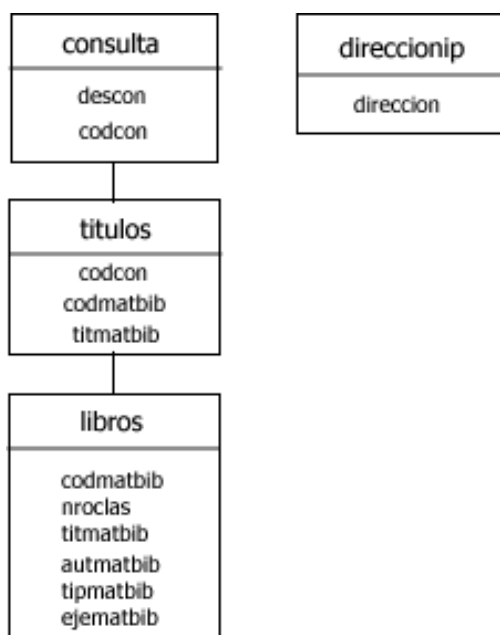
Esta lista de codmatbib es el resultado de la búsqueda del material bibliográfico, sin tener aún en cuenta el filtro del tipo de documento, hay que recordar que los materiales bibliográficos están catalogados en libros, Tesis y material seriado (analíticas), así que dependiendo del criterio de búsqueda que digito el usuario, hay que enfrentar esta tabla con la tabla de materiales_bib para tomar como resultado únicamente los codmatbib del tipo de documento bibliográfico que se están buscando.

Una vez obtenido los codmatbib depurados se proceden a buscar en sus respectivas tablas el titulo de libro y el nombre del autor principal, además de sus ejemplares y la disponibilidad de sacarlos a préstamo.

⁵² Tabla que guarda la relación entre el código de la palabra y los materiales bibliográficos.

⁵³ MatBib_Aut si la relación entre el número de identificación y el código de la palabra es por autor, MatBib_Tit por títulos o MatBib_Mat si es por materias.

4.7.5 Diagrama Entidad-Relación del Cliente Windows Mobile



ESQUEMA No 21, Esquema de la base de datos local del cliente Windows Mobile

El diagrama entidad-relación anterior corresponde a la distribución de las tablas de la base de datos que reside en la PDA, tanto su creación, inserción y consulta de datos se hace a través del motor de base de datos SqlServerCe, el cual es una versión para dispositivos móviles del gestor de bases de datos de Microsoft.

En el esquema se muestran cuatro tablas, las tres tablas de la columna izquierda corresponden a los datos de consulta, a los títulos consultados, y la última corresponde a los libros en sí, estas tablas están relacionadas directamente con el programa de esta manera:

Cuando el cliente ingresa al módulo de consulta general y digita un texto de consulta y elige las opciones de consultar por y tipo de documento, el sistema automáticamente codifica las opciones (las transforma en un número) y guarda los dos datos (texto y código de consulta) en la base de datos.

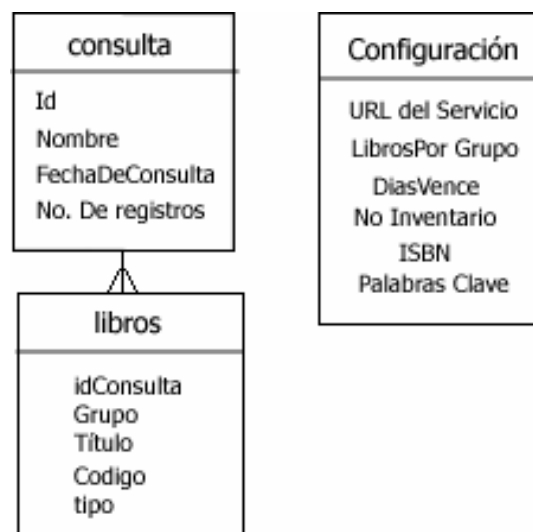
La tabla títulos se escribe con los nombres de los libros que devuelve la consulta al programa servidor.

Una vez obtenida la lista y el usuario quiera ver detalles del material bibliográfico, habrá de presionar el botón *detalles del libro*, con lo cual la aplicación le mostrará en el formulario los detalles del libro, allí esta la opción de guardar los datos del libro, será decisión del usuario guardar los datos del libro, los cuales quedaran registrados en la tabla libros de la base de datos local.

4.7.6 Diagrama Entidad-Relación del cliente J2ME

Java 2 Micro Edition no especifica un motor de bases de datos dentro del dispositivo pero si provee una serie herramientas base para el uso de archivos, estos archivos están conformados por registros con un índice único, con los cuales se pueden hacer versiones livianas que simulan una base de datos.

A continuación la estructura de los archivos para mantener en copia dura la información del material bibliográfico dentro del dispositivo móvil.



ESQUEMA No 22, Diagrama Entidad-Relación de la base de datos que se creó para el cliente J2ME

4.7.7 Vista a las Pantallas de la aplicación Cliente



FIGURA No 22, Pantalla principal de la aplicación cliente, en ella se muestran los diferentes menús de acceso a la aplicación, El menú principal de la aplicación se puede observar en la parte baja de la pantalla

El menú correspondiente a las consultas se encuentra desplegado, utilizando esta interfase el usuario puede ingresar a los demás formularios donde se encuentran las opciones explicadas en el diseño detallado de la aplicación.

Las opciones como otras, administración y ayuda poseen accesos a otras aplicaciones del sistema

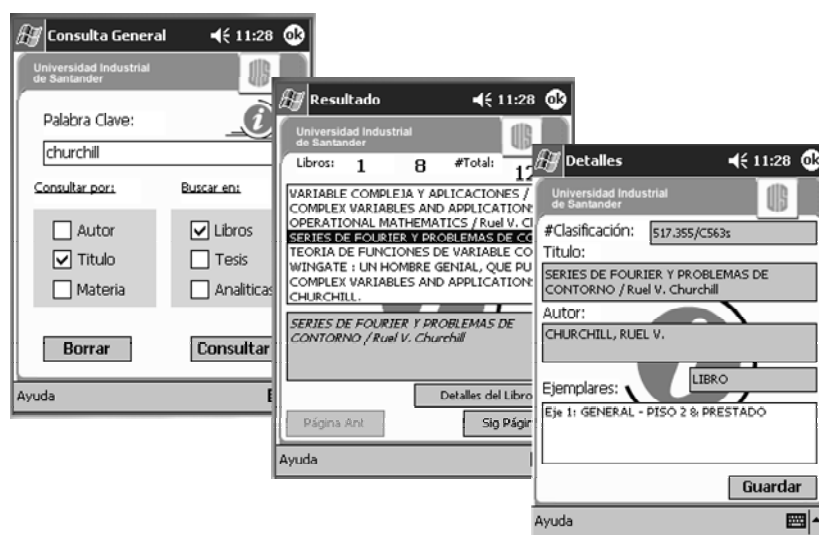


FIGURA No 23, de Izquierda a derecha. Formularios que corresponden a la búsqueda general de la aplicación cliente Windows Mobile (Búsqueda General, Resultado, Detalles)

4.8 Pruebas técnicas

Basados en los fundamentos de la programación extrema se realizaron las pruebas sobre las aplicaciones desarrolladas tanto en el servidor como en los clientes asegurando de tal manera el buen funcionamiento de las mismas, evaluando la calidad y el rendimiento del producto final.

La evaluación efectuada al sistema en general y a su lógica interna, asegura que las sentencias están ejecutándose correctamente y que los procesos externos son funcionales, garantizando que las entradas definidas producen los resultados esperados.

Las pruebas de unidad fueron una herramienta clave para el control de los cambios. Se utilizaron programas no interactivos, para correr en lotes y probar las clases componentes del sistema. Con estos programas se verificaba que el resultado de los métodos Web fuera el correspondiente a un a entrada de valores específica. Las pruebas unitarias reducen este riesgo dado que avisarán al instante si algo deja de funcionar. Es por esto que se optó por tomar el hábito de ejecutar todas las pruebas del sistema tras hacer un cambio.

Las pruebas que fueron realizadas por diferentes usuarios pueden clasificarse en tres categorías:

Desarrolladores: los gestores del análisis, diseño, desarrollo e implantación del conjunto de aplicaciones, son el primer nivel de detección de fallas, estos evalúan la arquitectura, la programación implementada, la integración y la adaptación del sistema al medio.

Auditoria: La auditoria del sistema definida como la evaluación del correcto funcionamiento de cada uno de los servicios implementados por parte del director del proyecto, quien como concedor de los procesos puede detectar fallas que se hayan filtrado al nivel de los desarrolladores.

Usuario: El usuario, evalúa la adaptación del sistema a las necesidades que se habían planteado en la definición del mismo. Detectando fallas de interfaz de usuario y de procedimiento.

Utilizando la herramienta SOAPScope se realizaron las pruebas unitarias para observar el comportamiento de la comunicación a través de los mensajes SOAP. El análisis de los mensajes errados creados en el dispositivo cliente, comprobando la sintaxis de los mismos a través de la comparación contra el descriptor de servicios WSDL.

En cada una de las pruebas se tuvieron en cuenta cada uno de los criterios a continuación:

Integridad de Interfaz: se analizaron las interfaces, su usabilidad, la ubicación de los objetos en el espacio, los colores utilizados que mantuviesen armonía con los colores institucionales, y que se amoldaran a la estructura global de las aplicaciones.

Validez funcional: pruebas correspondiente al correcto funcionamiento de las aplicaciones implicando los parámetros de Entrada-Salida y comparándolos con los resultados obtenidos de la aplicación Web disponible actualmente en el portal de la Universidad Industrial de Santander <http://alcatraz.uis.edu.co/biblioteca>

Desempeño: Se llevaron a cabo pruebas diseñadas para verificar los límites de rendimiento establecidos en tiempo de diseño para la obtención de los datos, inicialización de las aplicaciones, tiempo respuestas del servicio Web a las aplicaciones cliente, comportamiento de las aplicaciones en el dispositivo móvil.

Se utilizaron diferentes tipos de dispositivos móviles para las pruebas entre ellos:

- Motorola V3
- Nokia 3220
- Nokia 6020
- HP iPac

Utilizando la red WIFI abierta de rectoría para la PDA, asimismo para los teléfonos celulares las redes GRPS de las diferentes compañías de teléfonos celulares de Colombia.

ETAPA 3

CONCLUSIÓN

5 Conclusiones

La computación móvil abre las puertas a innumerable cantidad de servicios que puede ofrecer nuestra institución tanto a la comunidad universitaria como a agentes externos a ella, proporcionando agilidad y facilidad de acceso a la información.

Se espera que el uso de la aplicación sea masivo, dado el alto numero de usuarios que utilizan dispositivos con capacidad de conexión a Internet y el impulso que le están dando las compañías de telecomunicación inalámbrica al uso de Internet en dispositivos móviles.

El desarrollo de aplicaciones para dispositivos móviles implica un gran reto para el programador ya que se debe pensar en solucionar el problema pero a la vez esa solución debe ser lo más liviana posible dada las capacidades de procesamiento que se esperan de un dispositivo de recursos limitados. Además por el hecho de no contar con un motor de bases de datos al interior de dispositivo (en el caso de J2ME) se dificulta el almacenamiento y manipulación de la información.

Debido a que la implementación de las aplicaciones se realizó con base en estándares y componentes software reutilizables, el sistema es de fácil mantenimiento y depuración, lo cual le permite ser actualizable según los requerimientos y necesidades tanto del usuario final como del futuro desarrollador.

La capacidad de integración que proveen los servicios Web le permite a los diferentes estamentos de la universidad establecer una simbiosis sin perder independencia entre ellos, ofreciendo y consumiendo servicios Web XML como forma de comunicación con el fin de agilizar los trámites y papeleos necesarios para cumplir las tareas que implican diferentes dependencias.

Por ejemplo, dado que la información más completa y actualizada se encuentra en los servidores de la división de servicios de información, esta entidad podría ofrecer servicios de consulta de datos de los estudiantes a los diferentes centros de estudios o

cualquier otro sin la necesidad de saber que plataforma utiliza el centro de estudios para consultarla. Así existen gran cantidad de servicios que son necesarios para agilizar los procesos al interior de la universidad.

También se pueden ofrecer servicios Web XML a entidades externas, ya sea públicos como el de consulta bibliográfica o privados como el de actualizar la lista de estudiantes que han pagado la liquidación desde la entidad bancaria directamente, este servicio requeriría métodos de encriptación de los mensajes y claves de acceso para mantener seguridad y por ende la viabilidad.

Debido a que la investigación aplicada de la tecnología de los servicios Web XML y las diferentes aplicaciones fueron realizadas al interior de la división de servicios de información nos permitió a los desarrolladores interactuar con el ambiente laboral empresarial de manera directa, no sólo aplicando los conocimientos de aprendido en el recorrido de la carrera sino también estableciendo lazos sociales con las personas al interior de esta gran empresa.

La investigación de nuevas tecnologías como son los servicios Web y aplicaciones móviles nos permitió dar un vistazo a una pequeña porción de la gran cantidad de líneas de investigación que ofrece la ingeniería de sistemas, y como esta tiene una evolución y ampliación constantemente activa en cada una de ellas.

6 Recomendaciones

Para que la aplicación tenga aceptación por parte de la comunidad universitaria se requieren que se realicen campañas promoviendo la utilización de las consultas bibliográficas, ya sea a través del sitio Web o circulares informativas para una socialización de los usuarios con la herramienta. Asimismo campañas de instalación y uso de la aplicación para facilitar el acceso a ella.

Se podría escalar la aplicación del servicio Web XML de consultas de materiales bibliográficos a una estructura mas compleja en la cual el cliente invoca un servicio principal y este a su vez accede al resto de servicios ubicados virtualmente en otras bibliotecas de manera transparente para el cliente, de esta manera no se requiera una aplicación diferente para cada biblioteca.

La capacidad de procesamiento, memoria y conectividad de los dispositivos móviles está aumentando considerablemente es por esto que se recomienda realizar aplicaciones que ofrezcan servicios adicionales que aprovechen todas las ventajas de las tecnologías de comunicación inalámbrica.

Debido al buen desempeño de la aplicación desarrollada con software de libre distribución, se considera importante incentivar el aprendizaje y uso de este tipo de herramientas, de manera que se aumenten tanto conocimientos como estrategias de desarrollo y se puedan disminuir costos en el desarrollo de los proyectos.

Se recomienda utilizar la tecnología de los servicios Web XML como mecanismo de integración de los diferentes sistemas de información de las entidades al interior del campus. Además ofrecer servicios a entidades externas utilizando metodologías para la seguridad de la información en la comunicación.

7 Bibliografía

Gómez Florez, Luís Carlos.

Planeación de Proyectos Informáticos, Grupo de investigación en sistemas y tecnología de la información (STI), Escuela de Ingeniería de Sistemas e Informática, Universidad Industrial de Santander. Bucaramanga, Colombia, 2001

Guía para la planeación y desarrollo de proyectos en el área de la informática.

Dawson, Christian W.; Quetglás, Gregorio Martín.

El proyecto fin de carrera en ingeniería informática: Una guía para el estudiante, Madrid, Pearson educación S.A., 2002, 169 Páginas

Guía para el estudiante que busca realizar proyectos en ingeniería informática enfocados a proyectos de grado de escuelas que imparten educación superior.

Preparation of Papers for IEEE TRANSACTIONS and JOURNALS (March 2004),

England, IEEE Documents Department, 2004, 12 páginas

Formato para presentar trabajos de investigación ante los diferentes sitios Web que se encargan de publicar trabajos de investigación en comunidades relacionadas con el desarrollo de Software.

Pressman, Roger S.

Ingeniería del Software un enfoque práctico, edición No. 5, España, Mc Graw Hill, 2002, 601 Páginas

Herramienta guía en el desarrollo de proyectos informáticos de Calidad, sus capítulos ayudan al estudiante a planear las principales características del sistema, tales como objetivos, complejidad y estructura interna de la aplicación.

Sergio Andrés Contreras, Diana Marcela Zambrano

SEIS SIGMA en el desarrollo de proyectos de información tecnológica de la división de servicios de información y su aplicación en un modulo de la interfase Web del sistema de la biblioteca UIS,

Colombia, Universidad Industrial de Santander, 2004

Oswaldo Augusto García Mora,

Programación de Servicios Web Basados en protocolos SOAP, Colombia, Universidad Industrial de Santander, 2003, 76 páginas

Tesis de Maestría en Ingeniería informática que presenta al lector una primera aproximación al entendimiento de los servicios Web XML, y su importancia en el contexto de la programación actual.

Scout Short,

Building XML Web Services for the Microsoft .NET Platform, Estados Unidos, Microsoft Press, 2002, 426 páginas

En este libro se describe los primeros pasos para desarrollar aplicaciones informáticas basadas en la arquitectura de Servicios Web, las reglas para el diseño, transporte, y codificación de aplicaciones enfocadas a la herramienta de desarrollo de software de Microsoft .NET Framework⁵⁴.

James Kao,

Developer's guide to building XML-based Web Services with the Java 2 Platform, Enterprise Edition (J2EE), Estados Unidos, The MiddleWare Company, 2001, 21 páginas

La descripción de como desarrollar servicios Web en la plataforma de J2EE, encuentra su lugar en esta guía para el desarrollador de servicios Web XML para la plataforma de Sun⁵⁵.

Doug Tidwell, James Snell, Pavel Kulchenko,

Programming Web Services with SOAP, Estados Unidos, O'Reilly, 2001, 225 páginas

Programando Servicios Web con SOAP presenta en sus páginas como construir aplicaciones distribuidas basadas en la Web, usando los protocolos SOAP, WSDL y UDDI⁵⁶. Además de aprender sobre el estándar XML utilizado como lenguaje para la definición de documentos.

⁵⁴ .NET Framework, Visual C#, Visual Studio, y Windows son marcas registradas de Microsoft Corporation en Estados Unidos y otros Países.

⁵⁵ Sun Microsystems, J2EE, J2ME y Java son marcas registradas por Sun Microsystems y pertenecen a sus respectivos dueños.

⁵⁶ SOAP, Simple Object Access Protocol; WSDL, Web Service Definition Language; UDDI, Universal Description, Discovery and Integration

Rick Dobson,

Programming Microsoft SQL Server 2000 with Visual Basic .NET, Estados Unidos, Microsoft Press, 2002, 530 páginas

En este libro Rick Dobson muestra su experiencia de cinco años como programador de base de datos para la empresa Microsoft, presenta sus experiencias y las principales dificultades que se pueden presentar al trabajar sobre el manejador de bases de datos de SQL Server 2000 utilizando la herramienta Visual Basic .NET como guía de referencia.

Erick T. Ray.

Learning XML, Estados Unidos, Release Team [oR], 2001, 276 páginas

En esta guía se explica los por menores del lenguaje de etiquetas extensibles, su uso como estándar de descripción de documentos, además de su propósito final de escribir documentos que puedan ser intercambiados en medios homogéneos como la intranet u heterogéneos como la Internet, sin ningún tipo de problemas de compatibilidad.

Kim Topley,

J2ME in a Nutshell, Estados Unidos, O'Reilly, 2002, 519 páginas

Este libro presenta ante el lector un esquema de cómo implementar y programar dispositivos móviles compatibles con el lenguaje de programación J2ME de Java.

Sun Microsystems,

J2ME Web Services –Specification Version 1.0 White Paper, Estados Unidos, Sun Microsystems, 2004, 18 páginas

Este documento presenta una vista general del campo de los servicios Web y provee información de la especificación de Servicios Web para J2ME, Versión 1.0

7.1 Sitios Web disponibles para consultar información

<http://msdn.microsoft.com/vstudio/>

http://www.tetrattech.com/images/service/communication/Optic_Wire.jpg

<http://wikipedia.org>

ms-help:

//MS.NETFrameworkSDKv1.1.ES/cpguidenf/html/cpovrintroductiontonetframeworksdk.
htm

<http://www.microsoft.com/downloads/details.aspx?FamilyId=74473FD6-1DCC-47AA-AB28-6A2B006EDFE9&displaylang=en>

<http://www.microsoft.com/latam/technet/articulos/200204/art22/default.asp>

http://www.microsoft.com/mspress/latam/Pages/WinXP_756.htm

<http://www.microsoft.com/spanish/MSDN/estudiantes/ssoo/default.asp>

<http://www.microsoft.com/windowsserver2003/default.mspx>

<http://www.microsoft.com/windowsxp/pro/evaluation/whyupgrade/top10.mspx>

http://www.ciao.es/Windows_273938_3

http://www.quesabesde.com/PDA/noticias_detalle.asp?noticia=379

<http://www.100cia.com/enciclopedia/Informix>

<http://www.monografias.com/trabajos13/trsqlinf/trsqlinf.shtml#HISTOR>

<http://www.gui-innovations.com/html/remotesqlce.html>

<http://www.microsoft.com/WindowsServer2003/iis/default.mspx#EQC>

8. Anexos

Anexo 1

Instalación y configuración del ambiente de desarrollo

El ambiente de desarrollo se llevo a cabo en dos computadores conectados en red. En el primer equipo se trabajo bajo el ambiente de Windows XP SP2 el programa Servidor y el cliente Windows Mobile, en el segundo equipo bajo Linux SUSE 10 se trabajo el cliente J2ME de la aplicación, los pasos que se enumeran a continuación se basan en equipos de desarrollo con características similares.

Equipo de desarrollo con SO Windows XP Service Pack 2.

Instalar el Servidor IIS en el sistema operativo, este proceso se realiza a través de la instalación de los componentes adicionales de Windows utilizando el CD o DVD de instalación del SO. Se debe instalar el servidor con todos sus componentes adicionales, en especial las extensiones de servidor de Frontpage 2000 y el servicio de World Wide Web.

Descargar e instalar el .NET Framework 1.1.

El proceso de descarga se puede realizar desde el sitio Web de Microsoft, o desde el CD de instalación de los prerrequisitos que viene con el ambiente de desarrollo de .NET Framework.

Instalar todos los componentes software del CD de prerrequisitos de Microsoft Visual Studio .NET.

Instalar el Ambiente de desarrollo Microsoft Visual Studio .NET, en el proceso se puede escoger que lenguajes de programación instalar en el equipo de desarrollo, en este proyecto se opto por instalar todos los lenguajes pero únicamente se utilizo C# para el desarrollo de las aplicaciones.

Después de finalizada la instalación de .NET se procede a instalar el archivo ODBC Informix Client 2.9, el cual es el controlador informix que se utiliza para conectar la base de datos que se encuentra en el equipo Pelicano, con este que se comporta como el equipo de desarrollo de las aplicaciones.

Anexo 2

Instalación y configuración del Servidor

EL sistema operativo instalado en el equipo servidor es Windows 2003 Server Standard Edition, el proceso de instalación del SO se hace de forma estandar. En este proceso el instalador se encarga de instalar el servidor IIS y el .net Framework, dos componentes esenciales en la aplicación.

Después de instalar el SO se procede a instalar el componente ODBC de informix, el componente utilizado en la aplicación es el IBM informix client sdk 2.9, aunque versiones posteriores del sdk no deben presentar problemas. La descarga se puede realizar desde el sitio oficial de IBM en la Web, una vez instalados se procede a configurar el componente ODBC de informix en el DSN del sistema, (los detalles de la conexión se omiten por razones de seguridad).

Para realizar este proceso se necesita que el instalador conozca algunos conceptos básicos del servidor de IIS.

Una vez realizados los pasos anteriores se procede a instalar el servicio Web en el Servidor Web del equipo como se muestra en el anexo 6.⁵⁷

Opcional

Si el sistema operativo no es de la familia de Windows 2003 Server, se debe instalar todos los componentes software del CD de prerequisites de Microsoft Visual Studio .NET, en el se incluye el .NET framework, el cual es el motor de los servicios Web XML.

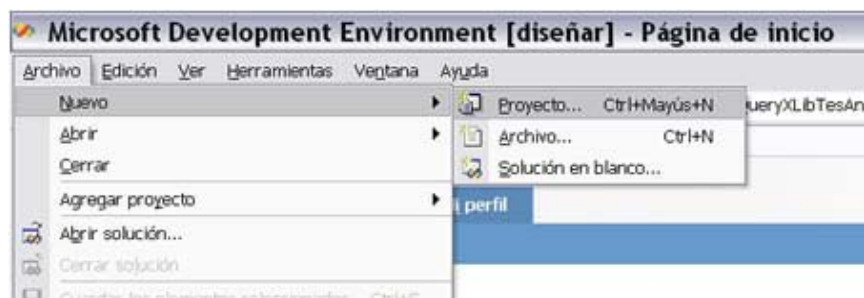
Si desea depurar trabajar con el código de la aplicación, debe instalar el Ambiente de desarrollo Microsoft Visual Studio .NET, en el proceso se puede escoger que lenguajes de programación instalar en el equipo, en este proyecto se opto por instalar todos los lenguajes pero únicamente se utilizo C# para el desarrollo de las aplicaciones.

⁵⁷ Anexo 6 – Implementar servicios Web XML en el servidor

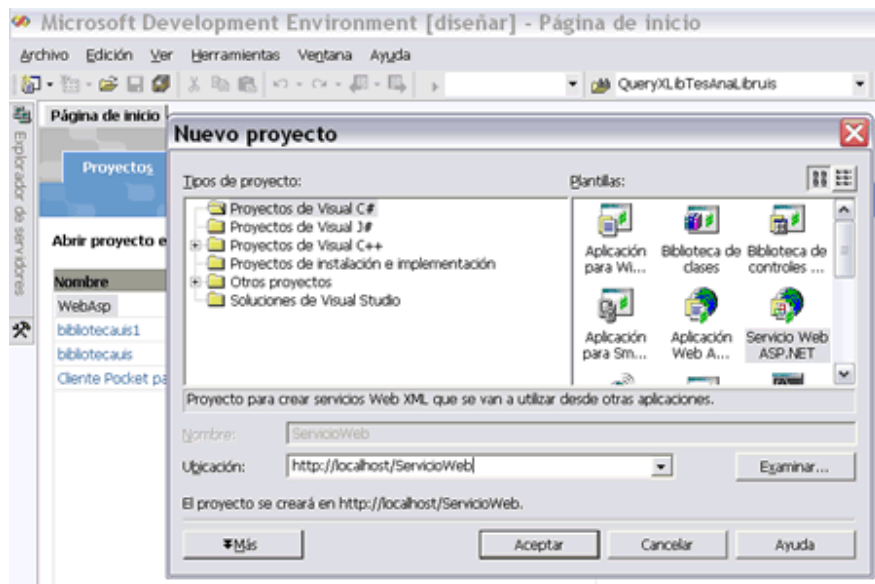
Anexo 3

Crear un servicio Web XML en .NET para servidores IIS

El primer paso que se realiza en la creación de un servicio Web XML, es la creación de un nuevo proyecto denominado Servicio Web ASP.NET del menú Archivo>nuevo>proyecto.



El lenguaje que escoge para realizar la programación del servicio Web se deja escoger a gusto del programador, para estos ejemplos se ha escogido a visual C# como herramienta de programación.

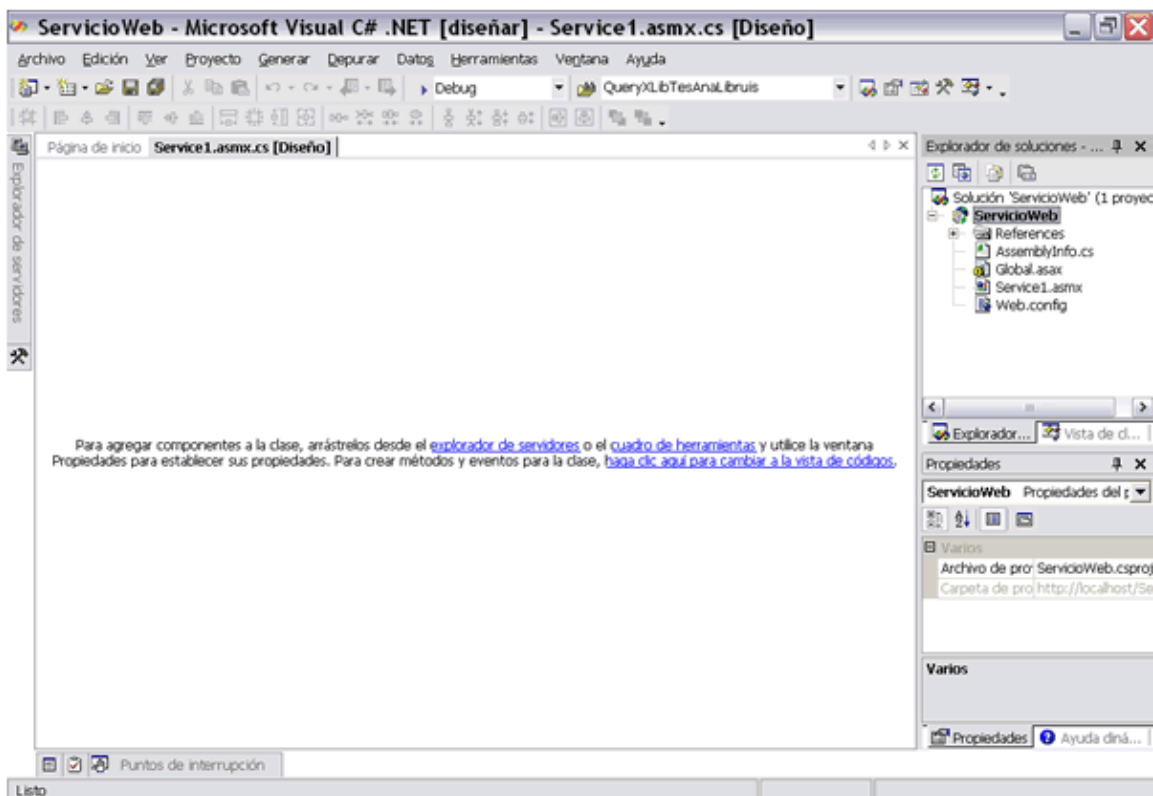


El valor de la caja de texto llamada ubicación es el único valor que se puede editar en este momento, donde ubicación es la dirección Web donde van a quedar los archivos del servicio Web XML, tanto el código fuente como el Servicio Web (.asmx) en si.

Debe tener en cuenta que debe haber estar instalado y corriendo un servidor de Internet information Server (IIS) 5 o superior en la dirección que se digita en ubicación.

En el ejemplo se va a crear un servicio Web que resida en el servidor IIS local del equipo de desarrollo, la palabra ServicioWeb representa la carpeta en el servidor donde va a residir la aplicación.

Una vez creado el proyecto de servicio Web de Asp.Net se presenta la siguiente pantalla.

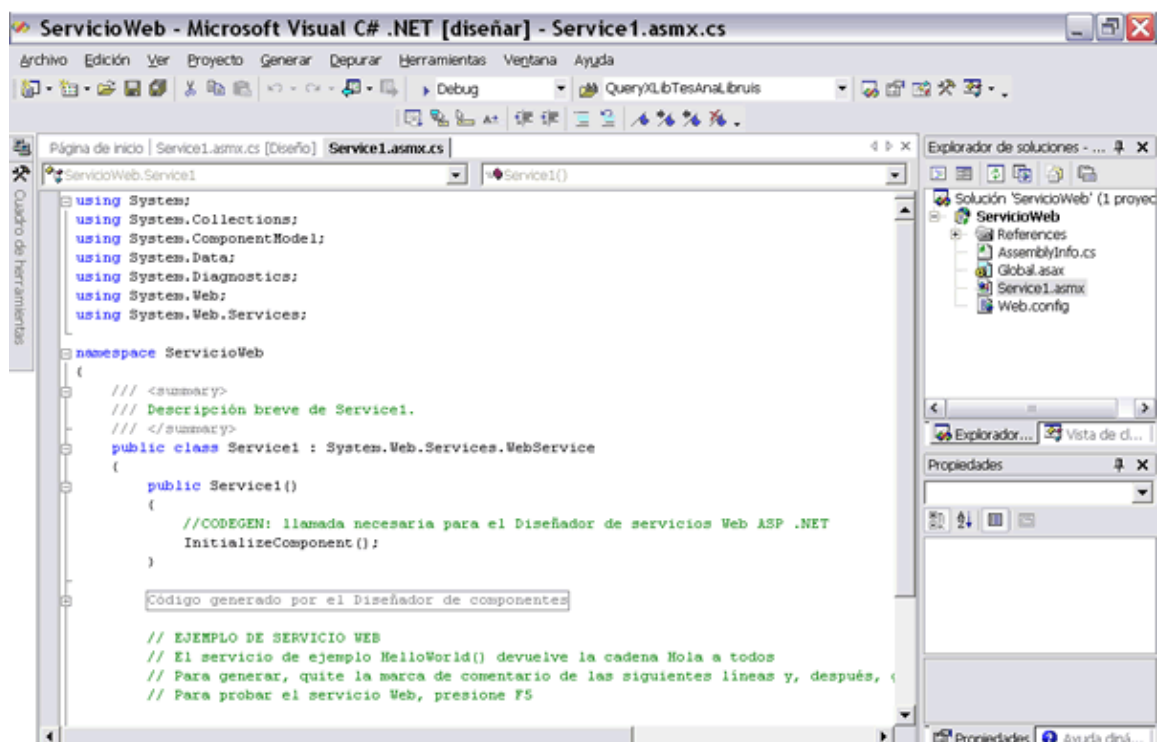


Cabe aclarar que los servicios Web no están diseñados para interactuar con personas clientes, están diseñados para que se comporten como servidores de procesos para otros programas, esta es la razón por la cual al servicio web en su pantalla de diseño que es la que se muestra en la gráfica anterior, no es un formulario, y no se le puede

agregar cajas de texto, etiquetas, botones de radio u otro componente de formularios Web.

Vamos al explorador de soluciones que se encuentra en la parte derecha de la pantalla, allí se pueden ver listados los componentes básicos del servicio Web, escogemos el archivo Service1.aspx, que es el archivo del servicio Web en si, damos clic derecho sobre el, y se escoge la opción ver código.

Esta operación se realiza con el fin de ver el código fuente del servicio Web XML.



```

using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Diagnostics;
using System.Web;
using System.Web.Services;

namespace ServicioWeb
{
    /// <summary>
    /// Descripción breve de Service1.
    /// </summary>
    public class Service1 : System.Web.Services.WebService
    {
        public Service1()
        {
            //CODEGEN: llamada necesaria para el Diseñador de servicios Web ASP .NET
            InitializeComponent();
        }

        Código generado por el Diseñador de componentes

        // EJEMPLO DE SERVICIO WEB
        // El servicio de ejemplo HelloWorld() devuelve la cadena Hola a todos
        // Para generar, quite la marca de comentario de las siguientes líneas y, después,
        // Para probar el servicio Web, presione F5
    }
}

```

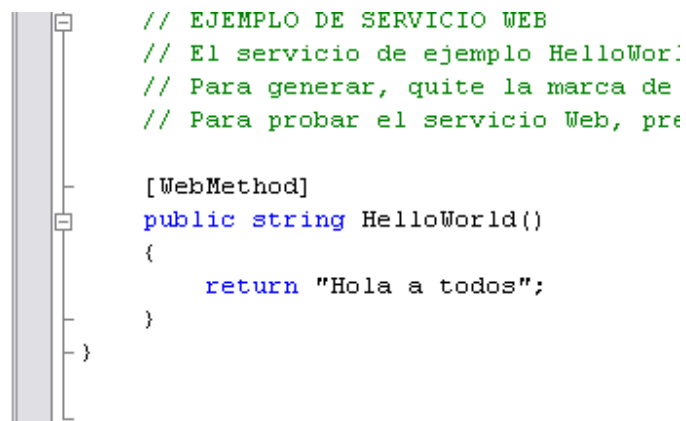
En la pantalla anterior se pueden observar las librerías que utiliza el servicio Web para su funcionamiento, a continuación se puede leer la línea namespace ServicioWeb, que corresponde al espacio de carpeta del servidor donde esta guardado el servicio web XML, hay que recordar que el nombre que se le dio al proyecto fue ServicioWeb y que este también corresponde a la carpeta que se crea en el servidor donde reside el servicio Web.

A continuación de la declaración del namespace se declara la clase pública `Service1` que al momento de heredar de la librería `Webservice` sus características, le da vida al servicio Web XML.

A continuación se implementa la clase de servicio Web XML que define los métodos y tipos de datos visibles para los clientes del servicio. Por último, se agrega la lógica del servicio Web XML a dichos métodos para procesar las solicitudes del servicio Web XML y devolver respuestas.

Inicialmente el servicio Web XML carece de interactividad, ya que no existe ningún proceso Web que interactúe con las peticiones de los clientes, es por eso que a modo de ejemplo, se presenta en la parte inferior de la hoja del código fuente y en color verde (comentario) el ejemplo más básico de un proceso Web denominado hola mundo.

Se procede a quitar las barras dobles (`//`) que acompañan al código fuente del método Web, el código fuente debe quedar como en la siguiente imagen, y no debe tener el color verde característico de los comentarios.



```
// EJEMPLO DE SERVICIO WEB
// El servicio de ejemplo HelloWorld
// Para generar, quite la marca de
// Para probar el servicio Web, pre

[WebMethod]
public string HelloWorld()
{
    return "Hola a todos";
}
```

En estos momentos el servicio Web posee un método Web que esta listo para recibir peticiones, este método esta diseñado para que al momento de invocarlo no reciba parámetros y responda con un texto cuyo valor siempre es Hola a todos.

Para Probar la interactividad del Servicio Web, compilamos el archivo utilizando la combinación de teclas `CTRL+MAYS+B`, y se ejecuta la aplicación con la tecla `F5`.

La ejecución del servicio Web se realiza en el explorador de Internet como se ve en la imagen siguiente.



En el explorador se muestran la lista de métodos que se encuentran disponibles, para este caso solo se lista el método HelloWorld, para probar la funcionalidad de este método se da clic en el título del método.



Como el método Web esta programado para enviar un mensaje y que no recibiera parámetros por valor, lo único que se debe hacer es darle clic en el botón invocar para que se emule una petición al servicio Web, la respuesta del servidor a esta petición en el método HelloWorld es el mensaje SOAP en formato XML que se ve a continuación.



Cuando se crea un servicio Web XML, este no hereda de forma automática la comunicación con ellos a través de la Web, para usar esta opción se debe aplicar el atributo `WebMethod` a los métodos públicos, al aplicar este atributo los métodos pasan a ser métodos del servicio Web XML y se pueden establecer comunicación con ellos a través de la Web.

Los métodos Web constituyen una de las partes fundamentales de los servicios Web XML. Para establecer comunicación con el servicio Web XML, se le deben enviar mensajes SOAP, es decir los clientes envían una solicitud SOAP a un Servicio Web y este responde con un mensaje SOAP, los datos son serializados a formato XML antes de realizar su travesía a través de la Web.

Para agregar más funcionalidad al servicio Web XML se deben agregar más métodos Web como el del ejemplo anterior, para tener una idea de cómo implementarlos se presenta a continuación unos ejemplos de métodos Web válidos.

```
[WebMethod]
public DataSet BusquedaLibruis(string textoClave, int codBusqueda)
{
    DataSet consulta = new DataSet();
```

```

//CONSULTA A LA BASE DE DATOS Y LLENADO DEL DATASET

Return consulta;
}

```

Este método Recibe dos parámetros, un texto y un entero, el método internamente realiza la consulta a la base de datos pertinente llena el dataset y lo devuelve al cliente que haga una petición al método `BusquedaLibruis`.

```

[WebMethod] //DEVUELVE EL NOMBRE DEL MATERIAL A PARTIR DEL NRO ISSN
public string CodMatBib_ISSN(string nroISSN)
{
    ...
    String nombreMaterial = new string();
    ...
    Return nombreMaterial;
}

```

Este método Recibe un parámetro, y devuelve un string, hay que tener en cuenta que el tipo de variable que se devuelve con la instrucción `return`, debe tener el mismo formato del tipo de dato que devuelve el método.

```

[WebMethod]
public string PruebaConexion()
{
    Lógica del Servicio Web...
    Return "En Línea";
}

```

Este método se comporta como el ejemplo de hola mundo, no recibe ningún parámetro y devuelve un texto con el valor en línea.

Una vez compilado y depurado se procede a montar el servicio Web XML en el servidor, para llevar a cabo este proceso remítase al anexo 6 Implementar Servicios Web XML en el Servidor.

Anexo 4

Manejo de los objetos sesión y aplicación en los servicios Web XML

Los servicios Web XML heredan todos los métodos de la clase `WebService`, incluidos los objetos `session` y `application`.

El objeto *aplicattion* proporciona un medio para almacenar datos a los que pueden acceder todo el código que en la aplicación, por otra parte el objeto *session* permite el almacenamiento de datos por cada sesión del cliente. Los datos del objeto *session* sólo esta disponible cuando el valor de la propiedad *EnableSession* del atributo *WebMethod* se establece a *true*.

Para los heredar las características del objeto aplicación, el servicio Web XML sólo debe implementar la clase `WebService` y de forma automática heredara sus características.

Crear y obtener acceso a los variables de session.

Realice los pasos del Anexo 3 (Declarar un servicio Web XML)

La línea `using System.Web.Services;` agrega la referencia para usar los métodos de los servicios Web XML.

La línea `public class buscador: System.Web.Services.WebService`

Permite que la clase `buscador` derive las propiedades de los servicios Web.

A esta altura el servicio Web XML puede usar el objeto aplicación para almacenar datos, si el programador desea utilizar las características del objeto `session`, debe ingresar las siguientes líneas.

Defina un método público en el servicio Web XML y establezca a *true* la propiedad *EnableSession* del atributo *WebMethod*.

```
[ WebMethod (EnableSession=true) ]
    public string PruebaConexion()
    {
        Lógica del Servicio Web...
    }
```

Las variables de session se crean en la lógica del Servicio Web como se muestra a continuación:

```
Session["miVariableSession"] = 1;
```

En el ejemplo anterior se crea una variable de session llamada miVariableSession a la cual se el asigna el valor de 1. Las variables de session no tienen un tipo de datos definido, pero para utilizarlas en programación primero se debe forzar al tipo de dato que se va a utilizar, una vez se realice la operación se puede guardar el resultado en la variable de session.

```
Session["miVariableSession"] = (int)Session["miVariableSession"] + 19;
```

El valor que se guardara en miVariableSession es de 20 el cual estará en un tipo de dato no definido.

Crear y obtener acceso a los variables de application

Se realiza como en el tema anterior los pasos del Anexo 3 (Declarar un servicio Web XML) y se define un método del servicio Web XML, en la lógica del servicio se crean las variables de estado (application) para recuperarlas posteriormente.

```
Application["miVariableApp"]= 1
```

En el ejemplo anterior se crea una variable de estado llamada miVariableApp a la cual se el asigna el valor de 1. Su uso y acceso es similar a las variables de session.

En la lógica del servicio Web se acceden a las variables de aplicación de la siguiente manera:

```
Application["miVariableApp"] = (int)Application["miVariableApp"] + 19;
```

Anexo 5

Atributos aplicables a los servicios Web XML

En la tabla siguiente se enumeran los atributos que se le aplican a un servicio Web XML y clientes de los servicios Web.

| Atributo | Descripción |
|----------------------------|---|
| <i>WebMethod</i> | Este atributo se aplica a un método para convertirlo en un método de servicio Web XML |
| <i>WebService</i> | Este atributo se aplica a una clase que implementa un servicio Web XML para especificar información adicional acerca del servicio Web XML, como el espacio de nombres XML predeterminado |
| <i>WebServiceBinding</i> | Este atributo se aplica a una clase que implementa un servicio Web XML o a una clase de proxy para especificar los enlaces implementados por el servicio Web XML |
| <i>SoapDocumentMethod</i> | Este atributo se aplica a un método de servicio Web XML o a un método de una clase de proxy para especificar que espera recibir mensajes SOAP basados en Document. |
| <i>SoapDocumentService</i> | Este atributo se aplica a una clase que implementa un servicio Web XML o a una clase de proxy para especificar que, de forma predeterminada, los métodos de servicio Web XML de la clase esperan recibir mensajes SOAP basados en Document. |
| <i>SoapRpcMethod</i> | Este atributo se aplica a un método de servicio Web XML o a un método de una clase de proxy para especificar que espera recibir mensajes SOAP basados en RPC. |
| <i>SoapRpcService</i> | Este atributo se aplica a una clase que implementa un servicio Web XML o a una clase de proxy para especificar que, de forma predeterminada, los métodos de servicio |

| | |
|-----------------------|---|
| | Web XML de la clase esperan recibir mensajes SOAP basados en RPC. |
| <i>SoapHeader</i> | Este atributo se aplica a un método de servicio Web XML o a un método de una clase de proxy para especificar que puede procesar un encabezado SOAP determinado. |
| <i>SoapExtension</i> | Un atributo derivado de esta clase se aplica a un método de servicio Web XML o a un método de una clase de proxy para especificar que una extensión SOAP debe ejecutarse con el método de servicio Web XML. |
| <i>MatchAttribute</i> | Representa los atributos de una coincidencia encontrada utilizando la coincidencia de modelos de texto. Sólo es válido para los clientes de servicios Web XML. |

TABLA No 4⁵⁸, Atributos que se le aplican a la definición de los servicios Web XML para cambiar su desempeño

⁵⁸ Información tomada del sitio Web de Microsoft <http://www.microsoft.com>

Anexo 6

Implementar Servicios Web XML en el Servidor

Para implementar un servicio Web en el servidor y hacerlo visible a los clientes, debe copiar en una carpeta del servidor el archivo .asmx además de los archivos de configuración del servicio Web XML.

En la aplicación contenida en esta tesis se generó un servicio Web XML con el nombre `buscador.asmx`, este archivo y los archivos de configuración deben ser copiados en un directorio virtual o en una carpeta del servidor, a continuación se muestra una estructura de carpetas típica de un servidor donde se puede copiar el servicio Web XML.

```

\Inetpub
  \Wwwroot
    \biblotecauis
      Buscador.asmx
      Buscador.disco <opcional>
      Web.config
      Buscador.wSDL <opcional>
    \Bin
  
```



FIGURA No 24, Disposición de las páginas que corresponde a la aplicación que reside en el servidor de desarrollo, en ella se ven los diferentes componentes de la aplicación servidor, el servicio Web XML tiene como nombre `buscador.asmx`, el archivo descriptor `buscador.wSDL` y el archivo de descubrimiento tiene el nombre de `buscador.disco`

Elementos publicados

Directorio: c:/inetpub/wwwroot/bibliotecauis/

Actúa como directorio raíz del servicio Web XML. *Los demás archivos se colocan en este directorio.* El directorio debe tener el indicador de aplicación Web de los Servicios de Internet Information Server (IIS).

Archivo: buscador.asmx

Actúa como la dirección URL base para los clientes que llamen al servicio Web XML. El nombre del archivo puede ser cualquier nombre de archivo válido.

Archivo: buscador.disco

(Opcional) Actúa como un mecanismo de descubrimiento para el servicio Web XML. El archivo .disco no se crea automáticamente para un servicio Web XML. Se puede utilizar un generador de wsdl que tiene el motor de .net, lo que hay que hacer es probar el SW en el explorador de internet del servidor y a la dirección del SW se le agrega el texto ?disco.

Al ejecutar esto se crea un archivo .disco del servicio Web XML.

Archivo: buscador.wsdl

(Opcional) En este archivo se proporciona información acerca de las funciones del servicio Web XML, lo que incluye los métodos de servicio Web XML que implementa, sus parámetros y los tipos devueltos.

Archivo: Web.config

Si debe reemplazar los valores predeterminados de configuración, puede incluir un archivo Web.config. Los servicios Web XML utilizan el archivo de configuración para permitir la personalización y extensibilidad del sistema.

Por ejemplo, puede suministrar un archivo Web.config específico del servicio Web XML si se requiere autenticación para el servicio, pero no para otras aplicaciones Web del sistema.

Directorio \Bin

Contiene los archivos binarios del servicio Web XML. Si la clase del servicio Web XML no se encuentra en el archivo .asmx, el ensamblado que contiene la clase debe estar en el directorio \Bin.

Anexo 7

Probar un servicio Web XML desde el explorador

Una vez publicado el servicio Web XML en el servidor se puede realizar pruebas desde el explorador para probar su funcionalidad, para ello publique el servicio Web en el servidor (anexo 6), una vez implementado puede probarlo a través del explorador de Internet del mismo servidor. Para esta prueba sólo se permite el invocar el servicio Web de manera local.

- Escriba la dirección URL en la barra de dirección del servicio Web XML que se va a probar.

http://Servidor/Aplicacion/NombreServicio.asmx

Servidor:

Nombre del servidor en el que se implemento el servicio Web XML.

Aplicación:

Nombre del directorio virtual y el resto de la ruta de acceso de la aplicación Web.

NombreServicio.asmx:

Nombre del archivo .asmx del servicio Web XML.

- Aparecerá el archivo de descripción HTML del servicio Web XML



FIGURA No 25, Vista de la ejecución del servicio Web XML desde el explorador del servidor

- La página de descripción HTML del servicio Web XML muestra todos los métodos de servicio Web XML admitidos por un servicio determinado.
- Conecte haciendo clic con el método de servicio Web XML que desee y escriba los parámetros necesarios.



FIGURA No 26, Vista de la ejecución de un método del servicio Web XML en el explorador del servidor

- En respuesta a esta solicitud, el servidor mostrará la página de respuesta donde muestra el documento XML que le enviaría al cliente cuando este realice una petición del mismo tipo.

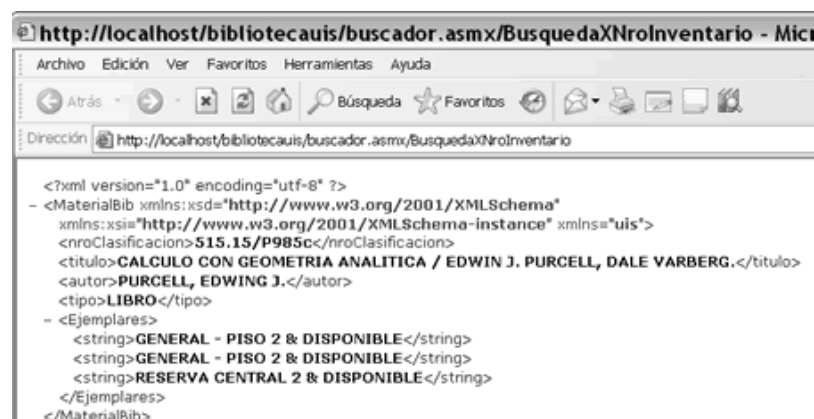


FIGURA No 27, Vista de la respuesta en formato XML que el servidor le enviaría al cliente cuando este lo invoque con los mismos parámetros con los cuales se le hizo la solicitud

Anexo 8

Acceder a la descripción del servicio Web XML (WSDL)

Una vez que conoce la dirección URL de un servicio Web puede utilizarla para tener acceso a una página Web denominada página de ayuda del servicio. En esta página se proporciona información acerca de las funciones del servicio Web XML, lo que incluye los métodos de servicio Web XML que implementa, sus parámetros y los tipos devueltos.

Para acceder al WSDL del servicio Web XML se debe incluir en la dirección URL del servicio Web el parámetro ?WSDL para poder ver la descripción del mismo.

- Escriba la dirección URL en la barra de dirección del servicio Web XML que se va a probar.

http://Servidor/Aplicacion/NombreServicio.asmx

Servidor:

Nombre del servidor en el que se implemento el servicio Web XML.

Aplicación:

Nombre del directorio virtual y el resto de la ruta de acceso de la aplicación Web.

NombreServicio.asmx:

Nombre del archivo .asmx del servicio Web XML.

Se agrega el texto ?wsdl al final de la dirección, la dirección resultante es:

http://Servidor/Aplicacion/NombreServicio.asmx?WSDL

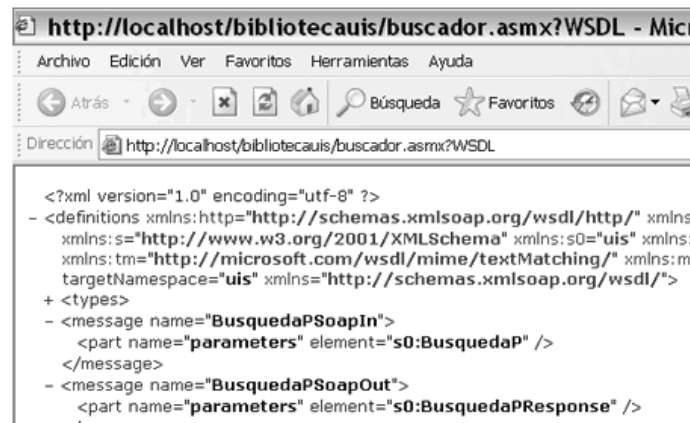


FIGURA No 28, Vista del archivo .WSDL (descripción del servicio Web XML) desde el explorador del Servidor

Al visitar esta dirección el resultado es el documento XML conocido como buscador.wsdl, en el cual se proporciona información acerca de las funciones del servicio Web XML, lo que incluye los métodos del servicio Web XML que implementa, sus parámetros y los tipos devueltos.

Anexo 9

Controlar excepciones en los servicios Web XML

Las excepciones en las que incurren los métodos Web XML se envían de vuelta al cliente en forma de error SOAP, Un error de SOAP es un elemento XML <Fault> contenido en un mensaje SOAP que especifica cuándo se produjo un error, El elemento XML <Fault> de SOAP contiene información como la cadena y el origen de la excepción.

Un servicio Web XML puede iniciar una excepción específica del problema, como `ArgumentOutOfRangeException`, o una excepción `SoapException` el error de SOAP se convierte en una excepción `SoapException` y los datos de la excepción se colocan en la propiedad `Message`. Por lo tanto, un cliente puede configurar un bloque `Try/Catch` para detectar una excepción `SoapException`.

Iniciar excepciones desde un servicio Web XML

La propagación de errores a un cliente se realiza mediante el inicio de excepciones.

Los tipos de excepciones que se manejan son:

Iniciando una excepción `SoapException`.

Iniciando una excepción `SoapHeaderException`.

Iniciando una excepción específica del problema.

| Excepción | Manejo de la excepción |
|---|--|
| Excepción distinta de <code>SoapException</code> y <code>SoapHeaderException</code> | Un método de servicio Web XML detecta un caso de excepción e inicia la excepción específica, como <code>argumentOutOfRangeException</code> para el cliente. Un cliente de ASP.NET recibe una excepción <code>SoapException</code> con los datos serializados en texto en la propiedad <code>Message</code> . |
| <code>SoapException</code> | Un método de servicio Web XML detecta un caso de excepción e inicia una excepción <code>SoapException</code> . Además, proporciona detalles adicionales relativos al problema. |

| | |
|---------------------|--|
| | Para proporcionar la información adicional, el método de servicio Web XML llena la propiedad Detail. Un cliente recibe la excepción SoapException con información adicional. |
| SoapHeaderException | Un método de servicio Web XML detecta un caso de excepción al procesar un encabezado SOAP. De acuerdo con la especificación de SOAP, el método de servicio Web XML debe iniciar una excepción SoapHeaderException para el cliente. Un cliente recibe la excepción SoapHeaderException. |

TABLA No 5, Excepciones que son lanzadas al momento de consumir los servicios Web XML por parte de los clientes

Para detectar una excepción iniciada por un método de servicio Web XML

En un bloque Try/Catch, detecte la excepción SoapException (todas las excepciones iniciadas por un método de servicio Web XML se inician como excepciones SoapException).

En el ejemplo de código siguiente de un cliente que llama a un método de servicio Web XML, se detecta la excepción iniciada por dicho método.

```

Try
{
    String titulo = ServicioWeb.Buscar(12345);
}
catch(SoapException error)
{
    MessageBox.Show(error.Message);
}

```

Anexo 10

Como dar visibilidad al servicio Web XML

El uso de los servicios Web XML se hace a través de la red mediante protocolos estándar, pero no sirve de nada realizar servicios Web XML sino existen clientes que los consuman. Por esta razón se debe dar visibilidad al servicio Web publicándolo en los servidores UDDI de Internet, para que sea descubierto por potenciales clientes.

Para llevar a cabo esta tarea hay que determinar si existen directorios de servicios Web XML donde se pueda publicar el servicio, para conocer que proveedores proporcionan este servicio se debe consultar las siguientes direcciones, donde residen los servidores UDDI de las tres grandes compañías que están prestando (al momento de escribir este libro) el servicio de registro de servicios Web XML.

IBM: <https://uddi.ibm.com/beta/registry.html>

Microsoft: <http://uddi.beta.microsoft.com>

SAP: <http://udditest.sap.com>

- Para el protocolo UDDI ver 2 sólo basta con ingresar al sitio Web de la empresa prestadora del servicio UDDI.
- Una vez allí registrarse como usuario del sitio.
- Ya realizado el proceso de registro de usuario, se dirige a la sección de registro de los servicios Web donde se debe especificar la dirección Web (URL) de la definición del servicio Web (.WSDL), y una breve descripción de lo que realiza el servicio (descripcion.html).
- Estos datos describen el servicio Web, además que el potencial usuario podrá saber donde encontrar el servicio Web y los datos con los cuales este trabaja.

El siguiente proceso que debe realizar el cliente es crear la referencia Web al servicio que acaba de consultar, este proceso se explica en el anexo 11⁵⁹.

⁵⁹ Anexo 11, Como los clientes buscan los servicios Web XML

Anexo 11

Como los clientes buscan los servicios Web XML

Para un cliente que desea consumir un servicio Web XML y no posee ningún tipo de información de cómo encontrarlos, los datos que necesita pasarle y otros detalles técnicos, debe realizar los siguientes pasos para encontrarlos y finalmente invocarlos:

1. Primero se ha de determinar si existe un servicio Web XML acorde a los que se esta buscando. Para este propósito se ha de consultar un directorio UDDI, como <https://uddi.ibm.com/beta/registry.html> o <http://uddi.microsoft.com>, para conocer que proveedores proporcionan servicios Web XML y sus funciones específicas. En el directorio se encuentra la dirección URL donde reside el servicio Web XML del proveedor.

| DIRECTORIO DE WEBSERVICES | | Exhibidos 1-30 31-60 61-90 91-120 121-124 |
|---|--|---|
| Inicio > Artículos > DIRECTORIO DE WEBSERVICES [124]: | | |
| <ul style="list-style-type: none"> • Buscadores [4] • Proveedores de Contenido [42] • Gráficos [1] • Matemáticas y funciones [7] • Utilidades [39] • Miscelánea [1] • Ubicación Geográfica [18] • Comunicaciones [11] | | |
| Popular | | 7/31/2003 |
| Hits: 143 ★★★★★☆ Comentarios[0] Votos [1] Descripción: envia mensajes sms | | |
| BANCO CENTRAL Popular | | 6/29/2003 |
| Hits: 478 ★★★★★☆ Comentarios[0] Votos [1] Descripción: Demuestra tarifas noruegas del banco central diariamente. Se puede probar. | | |
| BUSCA DIRECCIONES FISICAS Popular | | 5/7/2003 |
| Hits: 582 ★★★★★☆ Comentarios[0] Votos [1] Descripción: Encuentra direcciones de residencia. Se puede probar. | | |
| BUSCADOR DE CODIGOS ZIP Popular | | 6/8/2003 |
| Hits: 782 ★★★★★☆ Comentarios[0] Votos [1] Descripción: ubica codigos postales y genera listas de ellos por millas de radio. Se puede probar | | |

FIGURA No 29⁶⁰, Directorio donde se muestran servicios Web XML disponibles en Internet

2. Obtenida la dirección URL del SW del proveedor, se llama al descubrimiento de servicios Web XML para obtener los detalles específicos acerca del servicio Web XML que está disponible en esa dirección URL. La información de cada servicio se devuelve al cliente en forma de una descripción de servicio, que consiste en un documento XML que describe el servicio Web XML en Lenguaje XML basado

⁶⁰ Imagen tomada del sitio Web de Willy, <http://willydev.net>

en el formato de descripción de servicios Web (WSDL). En la descripción de servicio se ofrecen detalles concretos sobre cómo comunicar con un servicio Web XML, como el cliente debe invocarlo para consumirlo, datos necesarios para crear una referencia Web al servicio Web XML.

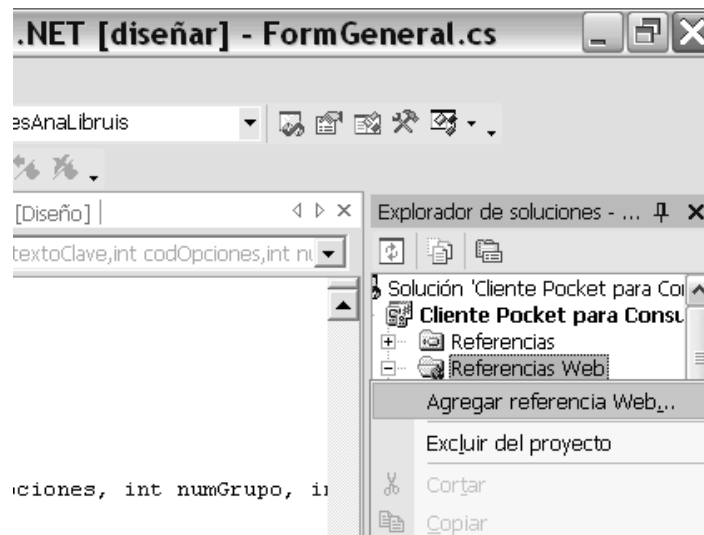


FIGURA No 30, Imagen que muestra en un programa cliente la localización en el proyecto donde se agrega la referencia Web al servicio Web en el entorno de .NET



FIGURA No 31, Pantalla que muestra los métodos Web que posee el servicio Web al cual se le está agregando la referencia

- Una vez creada la referencia Web se debe crear una instancia (objeto) de la referencia Web, los métodos de este objeto son los mismos métodos Web XML que se implementaron en el servicio Web XML, sólo que ahora aparecen como si fueran métodos locales.



FIGURA No 32, Pantalla que muestra como crear un objeto a partir de la clase creada de la referencia al servicio Web XML

- Invoke el método en la clase Proxy correspondiente del servicio Web XML con el cual desea comunicarse.

Anexo 12

Seguridad en los Servicios Web XML

La seguridad es un aspecto clave en el desarrollo de una aplicación, y toma un valor mucho mayor si su desempeño esta enfocada a intercambiar información usando la red de Internet como plataforma.

En la seguridad de los servicios Web XML existen dos conceptos claves de seguridad que hay que tener en cuenta:

La autenticación y la autorización.

La autenticación es validar un usuario en función de sus credenciales (nombre de usuario, contraseña) con el fin de determinar si es quien dice que es.

La autorización es determinar si el usuario autenticado tiene o no acceso a ciertos recursos.

En los servicios Web XML existen varias opciones para la autenticación del cliente, lo difícil es decidir cual es la más indicada, la elección debe estar basada en la curva de seguridad y rendimiento.

Los datos del cliente que se deben validar pueden ser enviados de dos maneras:

Usando credenciales del cliente

Encabezado SOAP

Usando credenciales del cliente:

Bajo este método el nombre del usuario y la contraseña, viajan con la carga útil del mensaje, estos datos se le denominan credenciales del cliente, para ser acceder a su valor existen métodos predefinidos.

Encabezado SOAP:

Este método se utiliza cuando se quiere desarrollar un método de autenticación personalizado, los datos a validar viajan en el encabezado SOAP, fuera de la carga útil del mensaje.

En la siguiente tabla se muestran las principales opciones de autenticación que están disponibles para autenticar las peticiones de los clientes.

| Opción de autenticación | Descripción |
|-------------------------|---|
| Básica | Puede utilizarse para la identificación no segura de los clientes, puesto que el nombre de usuario y la contraseña se envían en cadenas de texto sin cifrar codificadas en base 64. En este tipo de autenticación, las contraseñas y los nombres de usuario se codifican pero no se cifran. Un usuario informático con intenciones maliciosas que disponga de una herramienta de supervisión de red puede interceptar los nombres de usuario y las contraseñas. |
| Básica sobre SSL | Puede utilizarse para la identificación segura de clientes en escenarios de Internet. El nombre de usuario y la contraseña se envían a través de la red con cifrado SSL (Secure Sockets Layer, Nivel de sockets seguros), en lugar de como texto sin cifrar. Es relativamente sencillo de configurar y eficaz para escenarios de Internet. Sin embargo, el uso de SSL reduce el rendimiento. |
| Implícita | Puede utilizarse para la identificación segura de clientes en escenarios de Internet. Utiliza hash para transmitir las credenciales de los clientes de forma cifrada ya que la contraseña no se transmite en texto sin cifrar. Además, la autenticación implícita puede funcionar a través de servidores proxy. Sin embargo, no se admite en muchas plataformas. |
| Integrada de Windows | Utiliza NTLM o Kerberos. Utiliza un intercambio criptográfico con el explorador Web Microsoft Internet Explorer del usuario. |
| Certificados de cliente | Puede utilizarse para la identificación segura de clientes en escenarios de Internet y de intranet. Requiere que cada cliente obtenga un certificado de una entidad emisora de certificados de confianza mutua. Opcionalmente, los certificados se asignan a cuentas de usuario, que IIS utiliza para autorizar el acceso al servicio Web XML. |
| Formularios | No se admite en los servicios Web XML. Se trata de un sistema que redirige las solicitudes no autenticadas a un formulario HTML mediante redirección en el cliente http. La mayoría de los clientes de servicios Web XML no desearán proporcionar credenciales mediante una IU. Se debe |

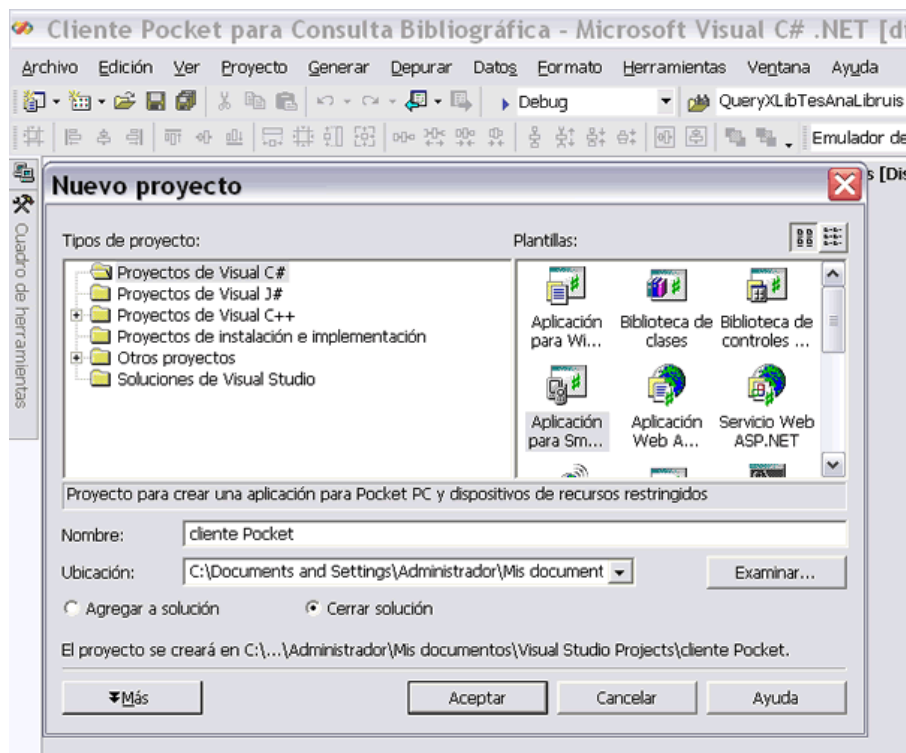
| | |
|---------------------------------|---|
| | proporcionar una alternativa para no usar esta opción con el fin de evitar el formulario de inicio de sesión. |
| Encabezados SOAP: personalizada | Puede utilizarse para escenarios de Internet seguros y no seguros. Las credenciales de los usuarios se pasan en el encabezado SOAP del mensaje SOAP. Con independencia de la plataforma que aloje el servicio Web XML, el servidor Web proporciona una implementación de autenticación personalizada. |

TABLA No 6, Opciones de autenticación en los servicios Web XML

Anexo 13

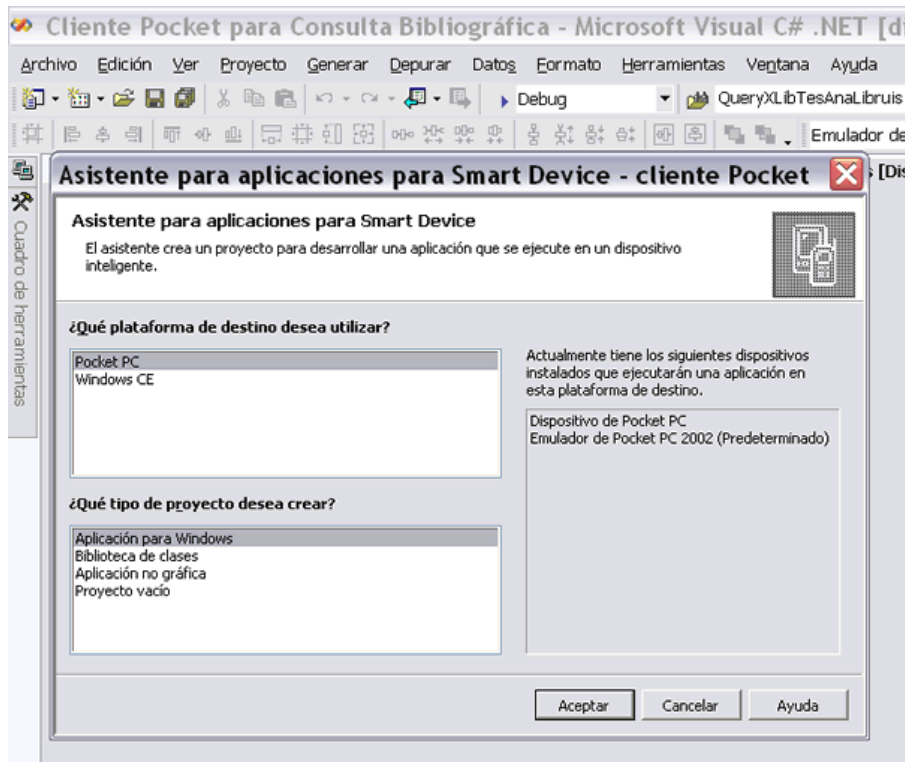
Crear el programa Cliente de Windows Mobile del servicio Web XML del servidor de Biblioteca

El primer paso que se realiza en la creación del cliente de Windows Mobile, es la creación de un nuevo proyecto denominado Aplicación para SmartDevice del menú Archivo>nuevo>proyecto, a este proyecto cliente se le llamará Cliente Pocket, la ubicación puede ser cualquier carpeta en el equipo desarrollador.

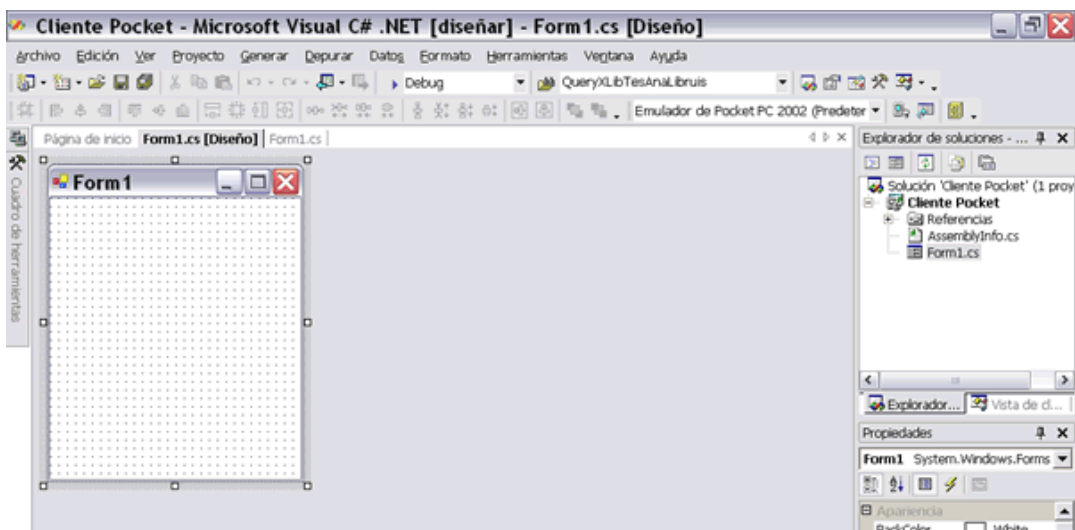


El lenguaje que escoge para realizar la programación del servicio Web se deja escoger a gusto del programador, para estos ejemplos se ha escogido a visual C# como herramienta de programación.

En la siguiente pantalla se muestran las plataformas disponibles donde se desea que pueda correr el programa y el tipo de proyecto a crear, por defecto se escoge un proyecto para pocket PC que sea una aplicación para Windows.



Una vez creado el proyecto de Cliente de Windows Mobile se presenta la siguiente pantalla en la cual se un proyecto para un dispositivo basado en Windows Mobile el cual solo se compone de un único formulario llamado Form1.

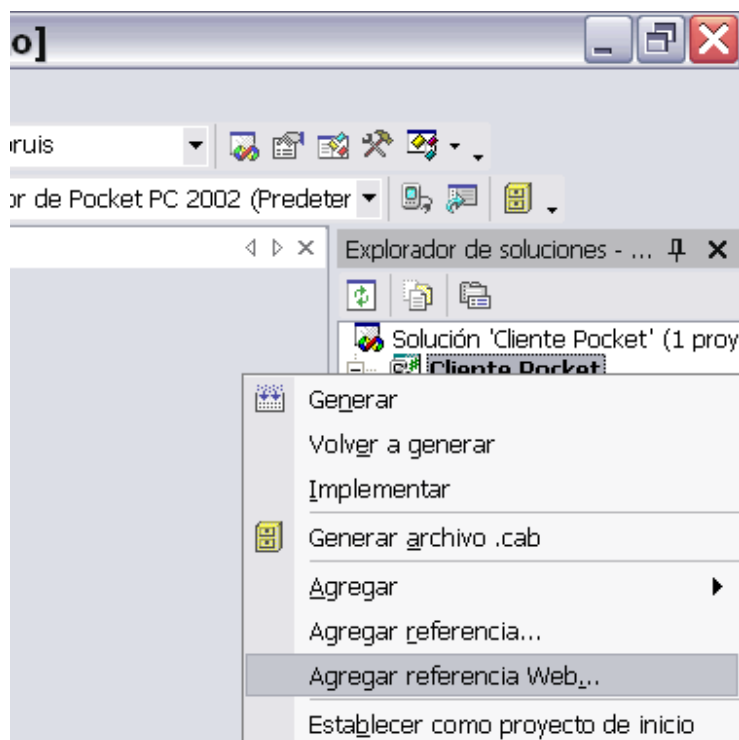


La programación de este tipo de proyectos se realiza como cualquier proyecto de Formularios en la herramienta de .net, donde si se desea modificar el código fuente del formulario, lo que hay que realizar es escoger en el explorador de soluciones el formulario, darle clic derecho en el y escoger la opción ver código.

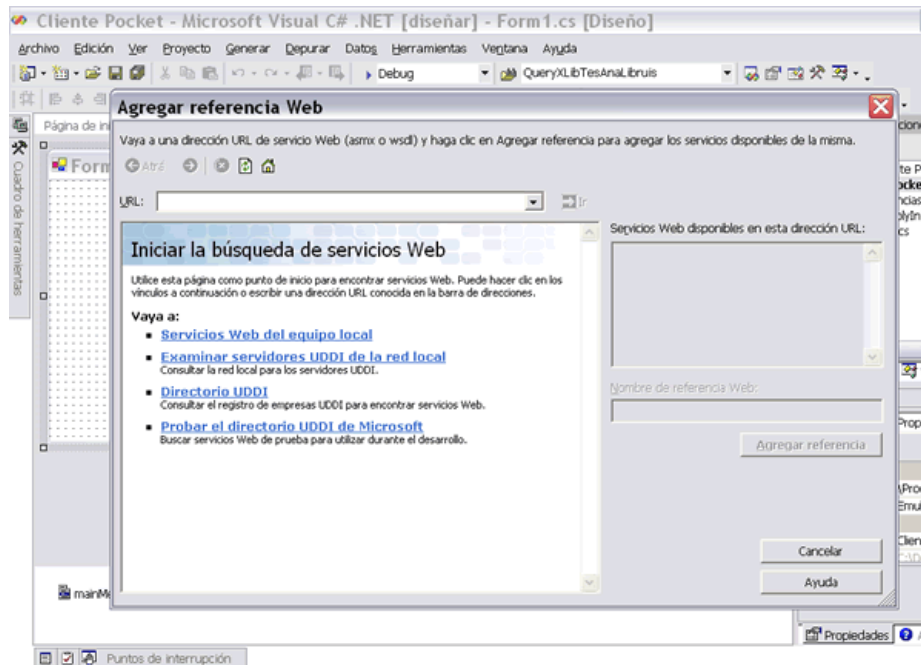
Esta es una de los comandos básicos que se realizan al crear aplicaciones clientes basadas en formularios, pero en este manual veremos como crear una referencia Web al servidor de Biblioteca para consumir los servicios que este tiene a disposición.

El primer paso es crear una referencia Web al servidor de Biblioteca, para realizar este paso lo primero que se debe saber es la URL del servidor donde esta corriendo el servicio Web, que para el caso de estudio que se esta realizando es <http://alcatraz.uis.edu.co/bibliotecasw/buscador.asmx>.

Para crear una referencia Web el primer paso es localizar el explorador de soluciones del proyecto, una vez allí se da clic derecho sobre el nombre del proyecto (cliente Pocket), en el menú emergente se escoge la opción agregar referencia Web, el proceso se muestra en la siguiente gráfica.



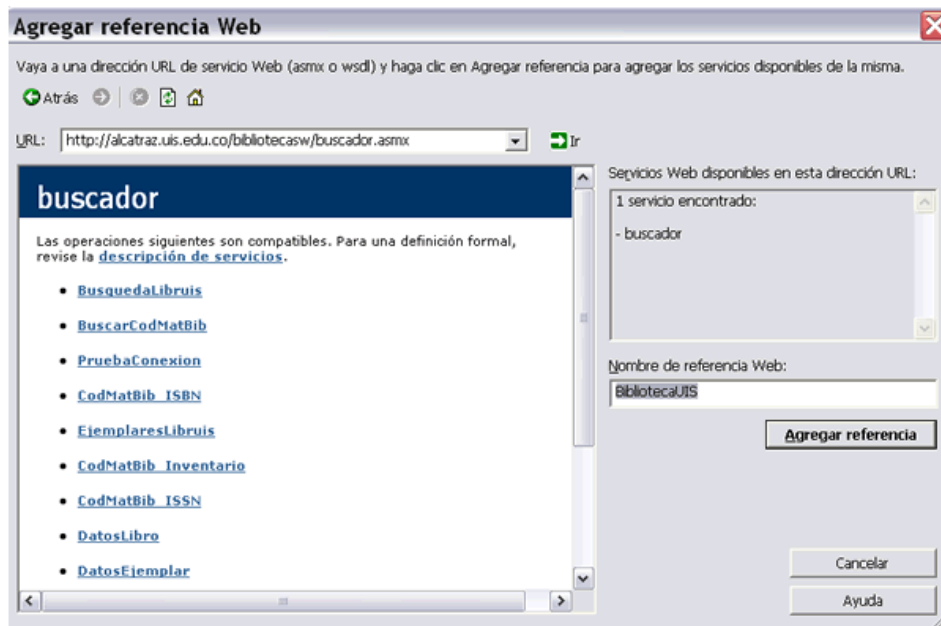
Una vez realizado el clic en la opción agregar referencia Web se presenta la pantalla donde se agregan las referencias Web a los servicios Web disponibles en la red.



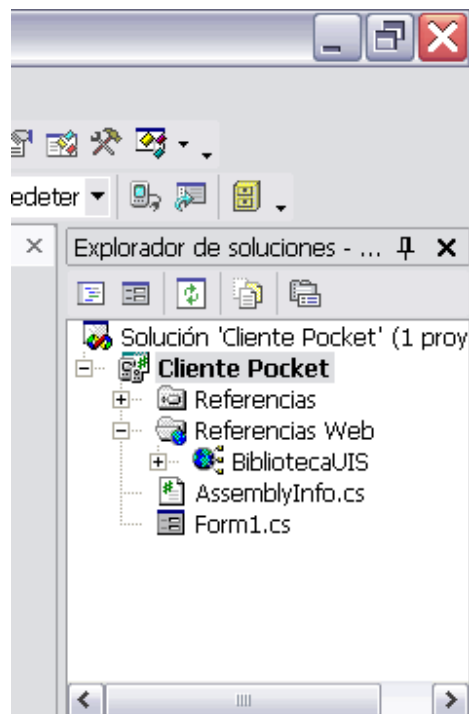
En esta pantalla se crea la referencia Web al servicio Web residente en alcatraz, para realizar la copia del archivo descriptor que reside en el servidor.

Para realizar este proceso se debe ingresar la dirección URL del servicio Web, que para este caso es <http://alcatraz.uis.edu.co/bibliotecasw/buscador.asmx>, y se clic en el botón ir para navegar hasta el servicio y ver que métodos tiene implementados.

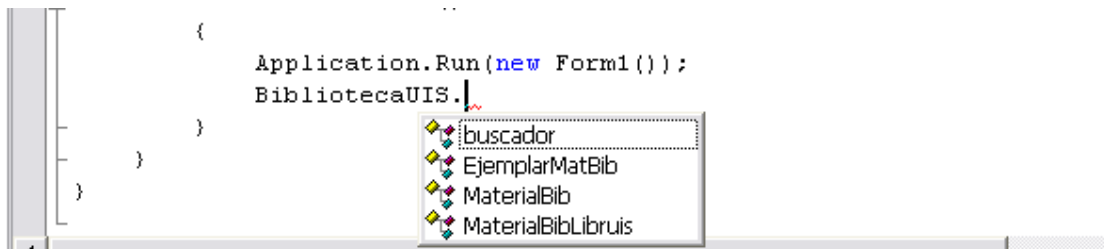
En la gráfica siguiente se ve en el lado izquierdo la lista de métodos disponibles para el servicio Web buscador en biblioteca.



En la caja llamada nombre de referencia Web se le da un nombre a la referencia Web que la identifique en la programación del proyecto, en este caso se llamará BibliotecaUIS, ya que es una referencia Web al servidor de Biblioteca, se da clic en agregar referencia.



En el explorador de Windows se puede observar la referencia Web marcada con el nombre BibliotecaUIS, a partir de esta referencia se crean objetos a partir de las clases del servidor de biblioteca.



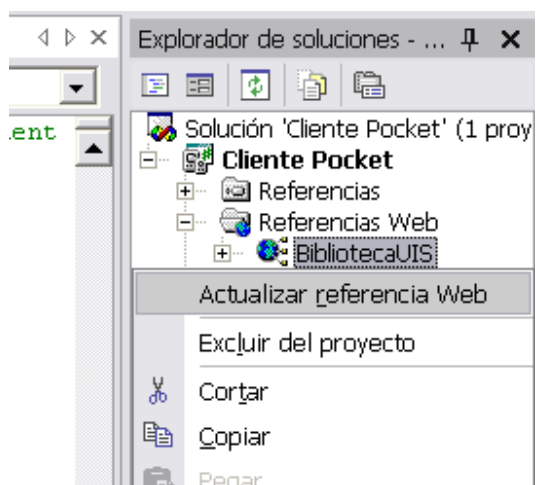
Por ejemplo para acceder a los métodos Web como búsqueda por ISBN del buscador de biblioteca, se debe crear un objeto del buscador e implementar los métodos a partir de este objeto, a continuación se observa una serie de líneas de código que explican estos pasos.

```

static void Main()
{
    Application.Run(new Form1());
    BibliotecaUIS.buscador servicioWeb = new BibliotecaUIS.buscador();
    BibliotecaUIS.MaterialBib libro = new BibliotecaUIS.MaterialBib();
    libro = servicioWeb.BusquedaXIsbn("12345678-9");
}
}

```

En la imagen anterior se ve como se crean dos objetos, el primero denominado ServicioWeb hace referencia a los métodos del servicio Web de biblioteca, la segunda referencia es para crear un objeto libro para guardar el resultado de la consulta al método Web, este proceso es necesario debido a que el resultado de la consulta al método Web de biblioteca devuelve un *objeto material bibliográfico*.



Para consumir otros servicios Web que están implementados en otros servidores, se debe agregar referencias Web a estos servicios, y repetir los pasos anteriores. Pero si se lo que se desea es agregar más métodos Web al servicio Web buscador.asmx instalado en biblioteca, y al cual ya esta creada la referencia en este cliente, no es necesario crear la referencia Web de nuevo, el paso que hay que realizar es actualizar la referencia Web existente, paso que se muestra en la imagen anterior.

Anexo 14

Instalar el programa Cliente de Windows Mobile en el dispositivo Móvil compatible con Pocket PC

Primero hay que descargar el ActiveSync 3.8 o la versión que soporte el dispositivo Pocket al cual se le va instalar la aplicación, ActiveSync es la encargada de sincronizar los datos del equipo de desarrollo con los datos que se encuentran en el Pocket PC.

https://www.microsoft.com/windowsmobile/downloads/eula_activesync38_1034.mspx?productid=34 es la dirección Web donde se puede descargar la versión ActiveSync 3.8, después de descargarla se debe instalar en el equipo de desarrollo donde se encuentra la aplicación que se va a instalar en la PDA.

Los requisitos para instalar el ActiveSync en el equipo de desarrollo son:

- Microsoft Windows XP, Windows 2000, Windows Millennium Edition, Windows NT Workstation 4.0 con SP6 o posterior o Windows 98
- Microsoft Outlook 98 o posterior para sincronizar el correo electrónico, el calendario, los contactos, las tareas y las notas al PC de escritorio o al portátil (se recomienda Outlook 2003)
- Microsoft Internet Explorer 4.01 con SP1 o posterior
- Unidad de disco duro con espacio disponible en disco que oscile entre 12 y 65 MB (los requisitos reales dependen de la selección de funciones y de la configuración del sistema actual del usuario)
- Puerto de comunicaciones disponible de 9 ó 25 clavijas (se necesita un adaptador par el puerto de comunicaciones de 25 clavijas), puerto de infrarrojos o puerto USB (disponibles únicamente para Windows 98, Windows Me, Windows 2000 y Windows XP)
- Tarjeta gráfica VGA o adaptador de gráficos de vídeo compatible de 256 colores o más
- Teclado
- Microsoft Mouse o dispositivo compatible



Después de instalar el ActiveSync, se debe conectar el cable del equipo, encender el dispositivo y colocar el dispositivo en el soporte, activeSync detectara automáticamente el dispositivo Pocket PC, una vez realizado el proceso se procede a copiar la aplicación desde el equipo de desarrollo al dispositivo Pocket PC.

