

Entornos para el desarrollo de aplicaciones móviles

Martínez Castro, José M.
Instituto Tecnológico de Chilpancingo
Chilpancingo, Guerrero, México
jmmtzc@hotmail.com

Cuevas Valencia, René E.
Universidad Autónoma de Guerrero
Unidad Académica de Ingeniería
Chilpancingo Guerrero; México
reneecuevas@hotmail.com

Martínez Castro, Ricardo N.
Centro de Bachillerato Tecnológico Industrial y
de Servicios No. 73
Rio Bravo, Tamaulipas, México
rmartincas@hotmail.com

Fecha de recepción: 20 de diciembre de 2011
Fecha de aceptación: 10 de febrero de 2012

Resumen

El esquema de desarrollo de las aplicaciones móviles es una tecnología que se encuentra en auge, aun cuando los dispositivos móviles existen desde hace más de 15 años, la perspectiva de los visionarios de las tecnologías de la información, y la disminución de costos de fabricación, ha permitido que tales dispositivos estén al alcance de los usuarios. Aun cuando parte de la teoría para el desarrollo de aplicaciones de esta tipo ya se encuentra plasmada en diversos documentos, se considera relevante plantear los nuevos usos que pueden darse a esta tecnología; lo anterior derivado por los avances tecnológicos: características, rendimiento, esquemas de comunicación, sistemas operativos, lenguajes de programación, almacenamiento y recuperación de datos, entre otras.

Palabras Clave: Dispositivos móviles, gestión conexión – desconexión, android, black berry, windows mobile.

Abstract

The pattern of development of mobile applications is a technology that is booming, even when mobile devices have been around for over 15 years, the visionary perspective of information technology, and lower manufacturing costs, has allowed such devices are available to users. Although part of the theory for the development of applications of this type is already reflected in various documents, is considered relevant to raise the new uses that can be given to this technology, the above derivative by technological advances: features, performance, layout communication, operating systems, programming languages, data storage and retrieval, among others.

keywords: Mobile device management on - off, android, black berry, windows mobile.

1. Introducción

Los dispositivos móviles poseen en la actualidad capacidades similares, en ocasiones superiores, a las computadoras personales de los que se iniciaron en el mundo de la computación en la década de los 80s. Lo que obliga a retomar el concepto de la *optimización de recursos* que en aquella época era de vital importancia, almacenamiento y espacio de memoria resurgen por sí mismos como los nuevos retos a vencer respecto de los diseñadores; sin dejar de lado el rendimiento de los procesadores que la Ley de Moore no nos permite olvidar.

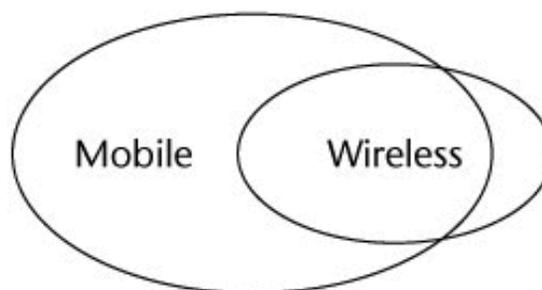
1.1. Inalámbrico vs Móvil

Como similitudes podemos referir que el medio de comunicación utilizado no está confinado a un medio guiado o cable.

Mallick [1] establece una diferencia entre los desarrollos de aplicaciones inalámbricas en relación a las aplicaciones móviles. No existe

una relación bidireccional entre ellas, ya que no todas las aplicaciones inalámbricas son móviles y viceversa (Figura 1).

Figura 1. Inalámbricas vs Móviles.



La diferencia se puede identificar en la *cobertura* de las aplicaciones, esto es, la necesidad de establecer una comunicación continua con la contraparte correspondiente; por ejemplo: en una red de área local inalámbrica existirá comunicación entre los nodos siempre y

cuando se encuentren dentro de la cobertura, de forma similar ocurrirá con las redes donde se requiera del enlace de comunicación constante entre los elementos de la red.

Por otra parte las aplicaciones móviles requerirán del canal de comunicaciones (**inalámbrico o alambico**) sólo en los casos donde la comunicación con los otros nodos sea necesaria, esto es, podrá continuar la operación del dispositivo aún sin la conexión permanente con la red.

Esto puede semejarse al uso del FTP y del Email para transferencia de datos; mientras el primero requiere de un canal de comunicación permanente entre emisor y receptor, el segundo solo requiere de la dirección del receptor para el envío de los datos.

1.2. Comunicación

Respecto a las formas de comunicación de los dispositivos móviles pueden referirse a los diferentes esquemas de comunicación entre PCs (Figura 2):

- Wifi (802.11: a, b, g, n, h ,i)
- Bluetooth
- Infrarojo
- Serial (USB, RS-232)

Figura 2. Comunicación Inalámbrica.



1.3. Tipo de Dispositivos

Dentro los tipos de dispositivos inalámbricos se pueden encontrar:

- Computadoras de escritorio y portátiles (Laptop, Notebook, Netbook)
- Tablets
- PDAs
- Smartphone

Los cuales se comunican con tarjetas de red: integradas, adaptadores o periféricos (USB, PCI, PCMCIA).

2. Diseño

Lo realmente importante en las aplicaciones móviles es la recomendación de no abordar el desarrollo *completo* de un sistema alrededor de los dispositivos móviles, ya que éstas deben formar parte de un sistema completo, esto es, serán un módulo de captura y/o consulta de información.

Existen diversas formas de diseñar los sistemas en red, sin embargo un elemento importante a considerar en este diseño es una característica inherente a las redes inalámbricas, esto es, la intermitencia en las comunicaciones.

La tendencia actual en el desarrollo de aplicaciones móviles trata de asumir condiciones mínimas de operación requeridas como son: conexión a la red de comunicaciones, recursos limitados (almacenamiento, energía).

Se hará una reflexión mayor sobre la red de comunicaciones a través del siguiente esquema de gestión de la red.

2.1. Limitantes en las redes de comunicación

En los países desarrollados no es posible concebir espacios donde no exista cobertura de alguna red de comunicaciones: celular, WiFi, lo cual implicar la posibilidad de conectarse a Internet por medio de redes tolerantes a fallas, donde los periodos de desconexión podrían considerarse mínimos.

Sin embargo en los países en desarrollo, uno de las oportunidades de mejora es precisamente la aplicación de la infraestructura de comunicaciones, donde no existe; y donde existe mejorarla en el sentido amplio del concepto optimización (tolerante a fallos, rendimiento, ancho de banda, disponibilidad) y principalmente el **costo de acceso al Internet**; mismo que en la actualidad no es accesible para todos los usuarios, aun cuando existen empresas proveedoras del servicio de Internet inalámbrico de banda ancha a través de los enlaces de telefonía celular.

2.2. Gestión conexión / desconexión

Una respuesta a esta limitante se encuentra en el uso limitado de la red de comunicación a través del uso de la Arquitectura Middleware (intermediario) [2] (Figura 3).

Donde: los *clientes inteligentes* (alámbricos o inalámbricos) se comunican al *servidor de sincronización* a través del *muro de fuego* para acceder a la *base de datos* en tiempos específicos de la ejecución de la aplicación.

En referencia a los datos se debe establecer en el nivel de visión [3], ya que solo se tomaran en cuenta las tablas que tengan relación con las acciones a realizar en el dispositivo móvil; denominándolas: *catálogo*, a aquellas tablas de consulta; *datos*, a aquellas tablas cuyo contenido será capturado como resultado de la operación de los dispositivos.

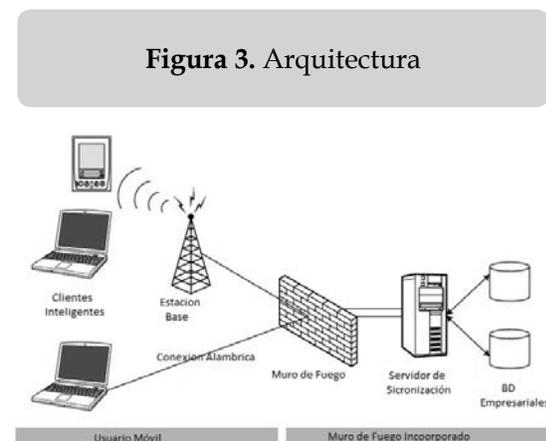
Se requiere establecer dos estados del *cliente inteligente*: conexión y desconexión, que hacen referencia directa a la comunicación con el *Middleware*, y como secuencia de enlace se propone la siguiente:

Cliente		Intermediario
		Listo
Solicita Catalogo	→	Recibe solicitud
	←	Envía Catalogo
Captura Datos		
Envía Datos	→	Recibe Datos
		Actualiza Datos

3. Implementacion

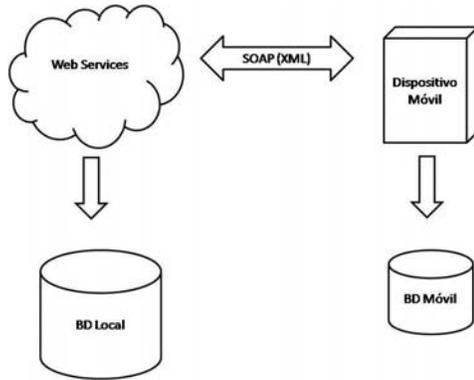
A este respecto se han realizado ejercicios de implementación de aplicaciones móviles [4][5][6] con los siguientes elementos de software:

- Cliente: Win Mobile, RIM, Android
- Intermediario; Web services, ASP.Net, IIS, Casini.
- Servidor: SMBD SQL Server.



Tomando en cuenta el esquema de conexión descrito en la (figura 4).

Figura 4. Esquema de Conexión.

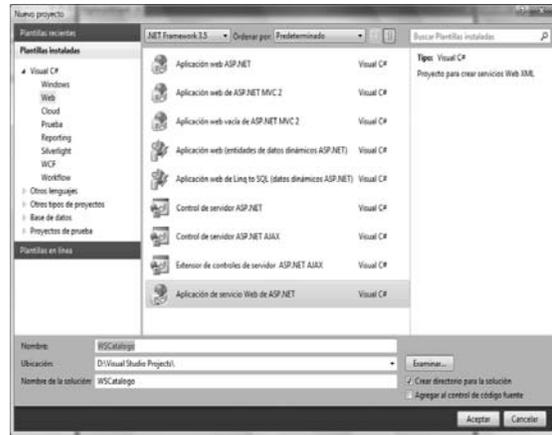


A continuación se muestran ejemplos de aplicaciones desarrolladas en los diferentes sistemas operativos para dispositivos móviles.

3.1. Middleware

Se inicia implementando un esquema de BD en SQL Server y accediéndolo por medio de un Web Service desarrollado en ASP.Net [7] (Figura 5).

Figura 5. Web Services.



El cual implementa los servicios (Web Methods) de consulta de *catálogos* y actualización de *datos* (Figura 6).

Figura 6. Implementación Acceso a Base de Datos.

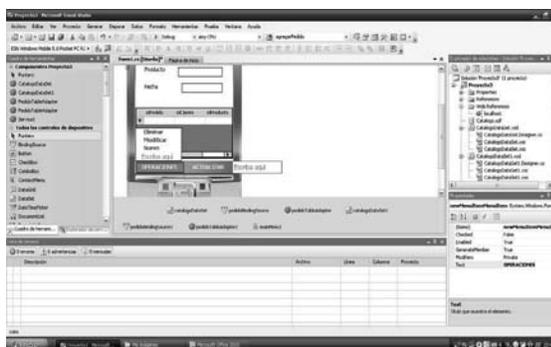


Una vez creado el servicio web se publica en IIS o Casini como Web Application Server (Figura 7).

Figura 7. Servicio web en ejecución.



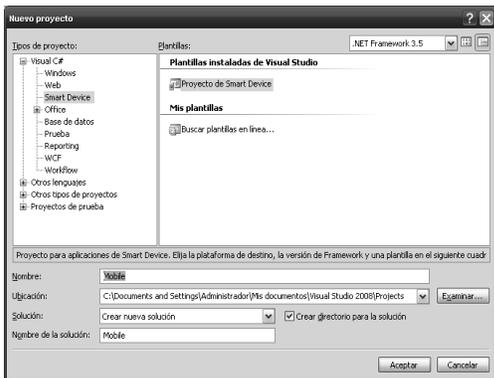
Figura 9. Creación de la Interfaz.



3.2. Win Mobile

Para la creación de una aplicación en este sistema operativo se requirió de la instalación del Visual Studio Dot Net 2008 [8], de donde se seleccionó la plantilla para crear un proyecto Smart Device (Figura 8).

Figura 8. Proyecto Smart Device.



Mediante la barra de herramientas se construye interfaz de la aplicación (Figura 9).

Una vez creada se procede a programar las acciones a realizar en los botones correspondientes, por ejemplo la inserción de un registro nuevo se realiza con las siguientes instrucciones:

```
private void menuItem3_Click_1(object sender, EventArgs e)
{
    pedidoBindingSource1.AddNew();
    Proyecto3.PedidoEditViewDialog
    pedidoEditViewDialog = Proyecto3.
    PedidoEditViewDialog.Instance(this.
    pedidoBindingSource1);
    pedidoEditViewDialog.ShowDialog();
    this.pedidoTableAdapter1.Update(this.
    catalogoDataSet1.Pedido);
    this.pedidoBindingSource1.MoveFirst();
}

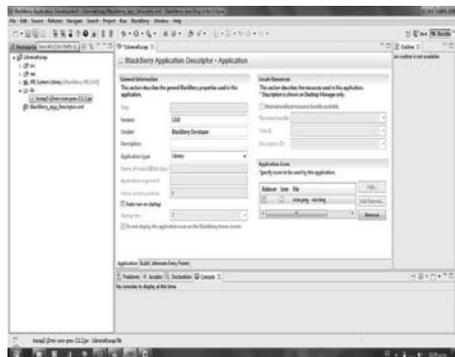
```

Para finalizar se realizan las pruebas de funcionamiento de la aplicación (Figura 10).

Figura 10. Ejecución de la Aplicación.



Figura 11. Proyecto vacío para la Librería KSOAP.



3.3. RIM

En los requerimientos para desarrollar aplicaciones móviles para el sistema operativo Research In Motion (Investigación en Movimiento) se encuentra la instalación de los siguientes elementos de software [9]:

- Plug-in BlackBerry para el entorno de desarrollo Eclipse [10].
- Simulador BlackBerry [9].
- MDS (Movil Data Service)
- Librería Ksoap [11].

Como requisito para el inicio se debe agregar la librería KSOAP en un proyecto vacío de BlackBerry, para su posterior referencia en el proyecto de la aplicación (Project References), al realizar la pruebas correspondientes se identificó este procedimiento como válido, ya que al hacer referencia directa a la librería mencionada, esta no fue reconocida (Figura 11).

En Eclipse se selecciona el proyecto de BlackBerry para iniciar el desarrollo de nuestra aplicación (Figura 12).

Figura 12. Proyecto Blackberry.



Se crean los objetos que servirán de interfaz de la aplicación (Figura 13).

Figura 13. Interfaz con objetos a utilizar.

Continuamos con la codificación en Lenguaje de Programación Java para especificar las acciones a realizar por parte de los objetos contenidos en la interfaz. Por ejemplo agregar un nuevo registro.

```
agregar_Boton = new ButtonField ("AGREGAR",
Field.FOCUSABLE);
agregar_Boton.setChangeListener(new
ClickBotonAgregar());
```

```
publicclass ClickBotonAgregar implements
FieldChangeListener{
```

```
publicvoid fieldChanged(Field field, int
context) {
```

```
objFunc.insertarPedidos(
Integer.parseInt(idPedido_Field.
getText()),
String.valueOf(comboClientes.
getSelectedIndex()+1).toString(),
String.valueOf(comboProductos.
getSelectedIndex()+1).toString(),
fecha_Field.getText(),
cantidad_Field.getText()
);
```

```
cargaListaPedidos();
}
}
```

```
publicvoidinsertarPedidos(
int id,
String cliente,
String producto,
String fecha,
String cantidad)
{
try{
```

```
URI myURI =
URI.create ("file:///SDCard/Databases/" +
"BDPedidos.db");
db = DatabaseFactory.open(myURI);
Statement st =
db.createStatement
("INSERT INTO \"pedidos\"
( \"id\", \"cliente\",
\"producto\", \"fecha\",
\"cantidad\")
VALUES ('"+id+"','" + cliente +
',' + producto + ',' +
fecha + ',' + cantidad +
');");
ejecutar(st);
db.close();
}catch ( Exception ex ){
Dialog.alert("Error al insertar: " +ex.
getMessage() );
ex.printStackTrace();
}
```

Para realizar las pruebas de funcionamiento es necesario ejecutar el simulador de BlackBerry, el almacenamiento la BD local que se crea con la aplicación, se debe simular una tarjeta SD multimedia y montarla en el simulador, se crea una carpeta en cualquier ruta de la computadora, y se configura el menú *Change SD card*; se agrega el directorio de la SD card, que será la ruta donde se creó la carpeta; se monta la "nueva SD card" al simulador seleccionando la opción *Mount Selected* (Figura 14).

Figura 14. Simulación del Almacenamiento Local para la BD.

Para ejecutar la aplicación en el simulador es necesario cargar la aplicación generada con extensión *.COD desde el menú *File*, en la opción *Load BlackBerry* (Figura 15).

Figura 15. Ejecución de la Aplicación.



En la (Figura 16) se observa parte del funcionamiento de una interfaz de la aplicación.



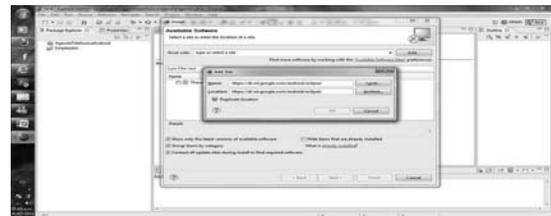
3.4. Android

Para desarrollar una aplicación Android, es necesario descargar, instalar y actualizar la versión más reciente del SDK del sistema operativo [12], así también el JDK de Java.

De la misma forma que en el apartado anterior se utiliza Eclipse como entorno de desarrollo [10], para ello se debe configurar el *plugin* de Android, tomando en cuenta los siguientes pasos:

- En el menú *Help*, de seleccionar la opción *Install New Software*.
- Seleccionar el botón *Add*, e ingresar la siguiente URL <https://dl-ssl.google.com/android/eclipse/> (Figura 15).

Figura 15. Configuración de Eclipse



Una vez en el entorno se selecciona el tipo de proyecto a desarrollar y la versión de Android a utilizar (Figura 16).

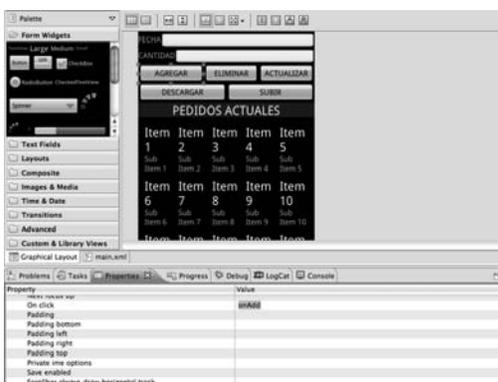
Figura 16. Seleccionar Proyecto y versión de Android.





Se está en condiciones de a crear la interfaz grafica, para ello se debe modificar el archivo `res>layout>main.xml` y colocar las etiquetas, cuadros de texto, botones y la grilla (Figura 17).

Figura 17. Interfaz de Proyecto en Android.



Se procede a definir las acciones a ejecutar en los botones de la interfaz, por ejemplo agregar un registro a la base de datos.

```
public void onAdd(View boton) {
    objetoFunciones.AgregarBaseDatos(
        Integer.valueOf(clave.getText().
            toString()),
        clientes[cliente.
            getSelectedItemPosition()][0],
        productos[producto.
            getSelectedItemPosition()][0],
        fecha.getText().toString(),
        cantidad.getText().toString())
}
```

```
);
mensaje("Añadido a la Base de Datos");
actualizar();
clave.setText("");
fecha.setText("");
cantidad.setText("");
}

public void AgregarBaseDatos (
    int clave, String empleado, String
    producto,
    String fecha, String cantidad
){
    SQLiteDatabase db = sqlhelper.
    getWritableDatabase();
    db.execSQL(
        "INSERT INTO pedidos
        (clave, empleado, producto, fecha,
        cantidad)
        VALUES
        ('" + clave + "','" + empleado + "','" +
        producto + "','" + fecha + "','" +
        cantidad + "') "
    );
    db.close();
}
```

Se ejecuta la aplicación con la finalidad de verificar el correcto funcionamiento (Figura 18).

Figura 18. Configuración de Eclipse



4. Conclusiones

Se describieron los pasos generales para el desarrollo de aplicaciones móviles, sin embargo aún pueden generarse un número mayor de para el desarrollo de estas tecnologías, detallar en los sistemas operativos aplicaciones de mayor complejidad, interacción con los periféricos de las tablets y smartphones, así como investigar el proceso de comercialización de las aplicaciones en el market de cada sistema operativo.

Del documento presentado se destaca que ya ha sido implementado por alumnos de tres instituciones de educación superior en el Estado de Guerrero: Instituto Tecnológico de Chilpancingo, Universidad Autónoma de Guerrero y Universidad Americana de Acapulco, queda a disposición la versión completa de

Queda pendiente el mundo IOS, ya que los elementos de prueba para su desarrollo están en proceso de elaboración, en cuanto se concluyan los trabajos correspondientes se contará con la evidencia documental para abordar el tema.

5. Referencias

- [1] **Mallick M.**, (2003), *Mobile and Wireless Design Essentials*, ISBN:0471214191 John Wiley & Sons.
- [2] **Gonzales S., Juan**, (2001), *Apuntes del Curso Sistemas Distribuidos II*, Maestría en Ciencias en Ciencias Computacionales con especialidad en Sistemas Distribuidos del Centro Nacional de Investigación y Desarrollo Tecnológico, Cuernavaca, Morelos, México.
- [3] **Silberschatz & et all**, (2002), *Fundamentos de Bases de Datos*, Cuarta edición., McGraw-Hill / Interamericana de España, S. A. U.
- [4] **Martínez C., J.M.**, (2012), *Apuntes del Curso: Desarrollo de Aplicaciones para Tecnologías Móviles*, Ingeniería en Sistemas Computacionales especialidad en Desarrollo de Software del Instituto Tecnológico de Chilpancingo, Guerrero, México.
<http://jmmc.itchilpancingo.edu.mx/moodle/course/view.php?id=93>
- [5] **Martínez C., J.M.**, (2012), *Apuntes del Curso: Tecnologías Móviles*, Ingeniería en Computación de la Unidad Académica de Ingeniería de la Universidad Autónoma de Guerrero, Chilpancingo, Guerrero, México.
- [6] **Martínez C., J.M.**, (2012), *Apuntes del Curso: Cómputo Móvil*, Ingeniería en Computación de la Facultad de Ingeniería de la Universidad Americana de Acapulco, Guerrero, México.
- [7] **Microsoft** (2012), *Centro de desarrollo de ASP.NET*, <http://msdn.microsoft.com/es-mx/aa336522>.
- [8] **Microsoft** (2012), *Microsoft Visual Studio Dot Net*, <http://msdn.microsoft.com/es-mx/vstudio/aa718325>
- [9] **RIM**, (2012), *Blackberry Developers*, <http://us.blackberry.com/developers/javaappdev/>
- [10] **Eclipse Foundation**, (2012), *Eclipse Project*, <http://www.eclipse.org/>
- [11] **KSOAP**, (2012), *KSoap 2.1.2*, <http://sourceforge.net/projects/ksoap2/files/ksoap2/2.1.2/>
- [12] **Google**, (2012), *Android Developers*, <http://developer.android.com/index.html>