



Facultad de Administración

Escuela de Ingeniería de Sistemas

*“DISEÑO DE UN SISTEMA PARA MANEJO DE
UNA BASE DE DATOS EN ACCESS A TRAVÉS
DE MENSAJERIA SMS IMPLEMETADO EN
VISUAL BASIC”*

Trabajo de graduación previo a la obtención
del Título de Ingenieras de Sistemas

Nora Patricia Sarmiento Molina
Gabriela Alexandra Vanegas Arízaga

Director: Ing. Fabián Carvajal

Cuenca, Ecuador
2007

DEDICATORIA

Nosotras dedicamos esta monografía:

A Dios principalmente, por habernos dado salud y vida para poder haber terminado otra etapa muy importante en nuestras vidas.

A nuestros padres, por su comprensión, por su paciencia y por que se han esforzado para que día a día sigamos superándonos.

Y finalmente, a nuestros hermanos y familiares por el apoyo incondicional que nos han brindado en el transcurso del desarrollo de esta monografía.

AGRADECIMIENTO

Va un sincero agradecimiento a todo el personal docente de la Universidad del Azuay por habernos inculcado los conocimientos necesarios para lograr nuestra meta y de una manera especial a nuestro director de tesis el Ing. Fabián Carvajal por su dirección, paciencia, entrega y valiosos consejos que nos hicieron posible alcanzar los objetivos de nuestra monografía.

Finalmente queremos agradecer a todas y cada una de las personas que de una o de otra forma hicieron posible la culminación de este proyecto.

INDICE DE CONTENIDOS

<i>DEDICATORIA</i>	2
<i>AGRADECIMIENTO</i>	3
<i>INDICE DE CONTENIDOS</i>	4
<i>ABSTRACT</i>	8
<i>INTRODUCCION</i>	9
<i>CAPITULO I: ESTUDIO DE LA DE TELEFONÍA MÓVIL Y LOS MENSAJES DE TEXTO (SMS)</i>	11
1.1. Introducción	11
1.2. Telefonía móvil	11
1.3. Evolución de los sistemas de comunicación móvil	12
1.4. Importancia de los sistemas de comunicación móvil	14
1.5. Servicios de mensajes de texto SMS	14
1.5.1 Las ventajas del SMS	15
1.5.2 Desventajas de los SMS	15
1.6. Conclusión:	16
<i>CAPITULO II: DESCRIPCIÓN DEL SOFTWARE QUE UTILIZAREMOS PARA LA IMPLEMENTACIÓN.</i>	17
2.1. Introducción	17
2.2. Definición de Visual Basic 6.0	17
2.3. Origen y Desarrollo de Visual Basic 6.0:	18
2.4. Características de Visual Basic 6.0	19
2.5. Detección de Puertos	20
2.6. El Control MSCOMM	20
2.6.1 Conexión con MSCOMM:	21
2.7. Conexión con la Base de Datos de Access:	23
2.8. Conclusión	23
<i>CAPITULO III: ESPECIFICACIÓN DETALLADA DEL FUNCIONAMIENTO DEL SISTEMA</i>	25
3.1. Introducción	25
3.2. Descripción General del Sistema	25
3.3. Módulo Servidor	26
3.3.1 Diseño y esquema	26
3.3.2 Envío y recepción de mensajes al dispositivo remoto	28
3.3.3 Puerto USB	28
3.3.4 Comandos AT	30
3.3.5 Envío de mensajes modo texto	32
3.3.6 Recepción de mensajes	33
3.3.7 Actualización de la base de datos	34

3.3.8	Eliminación de Clientes _____	35
3.4	Módulo Cliente _____	35
3.4.1	Diseño y esquema _____	35
3.5	Casos de Uso _____	39
3.5.1	Descripción de actores _____	40
3.5.2	Descripción del caso de uso _____	41
3.5.3	Detalle del caso de uso _____	41
3.6	Conclusiones _____	41
<i>CAPITULO IV: IMÁGENES Y GRÁFICOS PARA FACILIDAD DE ENTENDIMIENTO DEL USUARIO.</i> _____		43
4.1	Introducción _____	43
4.2	Manual de usuarios que manejan el Sistema(Servidor) _____	43
4.2.1	Requerimientos para el Sistema: _____	43
4.2.1.1	Requerimientos de Hardware: _____	43
4.2.1.2	Requerimientos de Software: _____	44
4.2.2	Configuración del Control de Comunicaciones (MSCOMM): _____	44
4.2.3	Seleccionar el puerto de Comunicación: _____	45
4.2.4	Abrir el Puerto COM: _____	46
4.2.5	Configuración del MSCOMM: _____	46
4.2.6	Limpiar Pantalla: _____	46
4.2.7	Comandos AT: _____	46
4.3	Manual de Clientes _____	47
4.3.1	Registro de Usuario y Contraseña: _____	47
4.3.2	Consulta del Artículo: _____	47
4.3.3	Pedido del Artículo: _____	47
4.4	Conclusiones: _____	48
<i>CONCLUSIONES</i> _____		49
<i>GLOSARIO</i> _____		51
<i>ANEXOS</i> _____		54

Todos los criterios vertidos en esta monografía son de responsabilidad de sus autoras:

Patricia Sarmiento M.

Gabriela Vanegas A.

RESUMEN

Nuestro proyecto se basa en la realización de pedidos para una ferretería mediante la utilización de mensajes SMS, detallándolo a continuación: Primeramente tenemos como fundamento esencial una base de datos ya creada con información que va a ser implementada en esta ferretería incorporada en el Gestor de Base de Datos Access 2003, luego estudiamos la necesidad imperiosa que afecta al cliente cuando esta en algún lugar y necesita conocer información detallada de cierto producto, a sabiendas de todo lo mencionado anteriormente empezamos a desarrollar el sistema en el lenguaje de programación Visual Basic 6.0, teniendo como argumentos: ser un usuario registrado dentro de nuestra empresa, conocer el código de nuestros productos para poder realizar la consulta; y, debemos contar con el módem de un celular el cual nos va a servir como enlace con el servidor. Nuestro sistema esta desarrollado para poder registrarse como usuarios, obtener la información detallada de sus productos y poder hacer pedidos mediante la utilización de un celular.

ABSTRACT

Our project describes the process of ordering from a hardware store via SMS messages as follows. First, we have a database as an essential foundation, which has been already created with information that will be implemented in this hardware store by incorporating it to the Access 2003 Database. Then we analyze the client's urgent need to obtain detailed information of a specific product.

Based on the above mentioned, we began to develop the system using the Visual Basic 6.0 programming language for the following reasons. First, it is a user that is registered in our company. Besides, it knows the codes of our products to process the search, and finally, we can count on the modem of a cellular phone that will serve us as a link with the server. Our system is developed to be able to register as users, obtain detailed information of the products, and be able to place orders via cell phone.



Ruth W. ...

INTRODUCCION

En la actualidad, el uso de los mensajes cortos de los teléfonos celulares (SMS) está siendo utilizado de forma muy extensa. La facilidad de acceso al servicio, las posibilidades en la comunicación con otras personas y la sencillez de su concepción están haciendo que este medio de comunicación continúe creciendo. Pero los SMS no sólo pueden utilizarse en la comunicación entre personas. Puede obtenerse un gran servicio de su utilización para comunicar con agentes automáticos que respondan a mensajes enviados automática o manualmente.

Teniendo en consideración que la telefonía móvil y a su vez los mensajes SMS han tenido un gran avance en la actualidad desarrollaremos un sistema de envío de mensajes cortos a través de un teléfono celular para obtener información de un producto de ferretería con lo cual otorgamos al usuario la facilidad y comodidad de consultar e inclusive de realizar pedidos en cualquier momento y de manera inmediata.

Uno de los motivos principales que se llevó a realizar esta monografía es por que el envío de mensajes cortos a través del teléfono móvil se ha convertido en una herramienta esencial e indispensable de comunicación, basada fundamentalmente en el criterio de la inmediatez. Las personas necesitan comunicar de manera instantánea aquello que en ese mismo momento es considerado como importante.

Por lo tanto este sistema puede ayudar al usuario a obtener la información del producto que necesite de una manera eficiente y rápida para el lugar en el que se encuentre en ese instante que necesite.

Debido al auge que ha tendido últimamente la telefonía móvil en nuestra sociedad y porque no decir en el mundo entero, se ha considerado la necesidad de realizar esta monografía que describe el diseño y la implementación de un sistema para el pedido de artículos de una ferretería mediante mensajes de texto.

El sistema se ha dividido en dos partes para su mejor entendimiento la primera se ha denominado SERVIDOR que es la encargada de realizar la configuración, el envío,

la recepción de mensajes y la actualización de la base datos implementada en Access este sistema se ha desarrollado en Visual Basic utilizando un teléfono móvil conectado a un ordenador a través de un puerto USB. La segunda parte se le denominó cliente y es el que inicia la conversación con el servidor es el que envía los mensajes para realizar el pedido

CAPITULO I: ESTUDIO DE LA DE TELEFONÍA MÓVIL Y LOS MENSAJES DE TEXTO (SMS)

1.1. Introducción

Este capítulo contiene una visión general de la telefonía móvil incluyendo una breve descripción de la importancia que ha tenido la telefonía en el mundo y su evolución y desarrollo en los últimos años, considerando que esto es lo mas importante para el entendimiento de porqué se realizó el trabajo y también se comentará en breves rasgos de los mensajes de texto SMS (*Short Message Service*) siendo esta una de las principales herramientas para el desarrollo de este proyecto exponiendo sus ventajas y desventajas.

1.2. Telefonía móvil

La telefonía móvil, también llamada telefonía celular, básicamente está formada por dos grandes partes: una red de comunicaciones (o red de telefonía móvil) y los terminales (o teléfonos móviles) que permiten el acceso a dicha red.

La red de telefonía móvil consiste en un sistema telefónico en el que mediante la combinación de una red de estaciones transmisoras-receptoras de radio (repetidores ó también llamados estaciones base) y una serie de centrales telefónicas de conmutación, se posibilita la comunicación entre terminales telefónicos portátiles (teléfonos móviles) o entre terminales portátiles y teléfonos de la red fija tradicional. El teléfono móvil consiste en un dispositivo de comunicación electrónico con las mismas capacidades básicas de un teléfono de línea telefónica convencional. Además de ser portátil es inalámbrico al no requerir cables conductores para su conexión a la red telefónica.

El teléfono móvil o celular fue inventado en 1947 por la empresa norteamericana *AT&T*, pero no se hizo portátil de manera práctica hasta 1983 cuando Motorola culmina el proyecto *DynaTAC 8000X*, que es presentado oficialmente en 1984. El *DynaTAC 8000X* pesaba cerca de 1 Kg., tenía un tamaño de 33.02x4,445x8,89

centímetros (13 x 1,75 x 3,5 pulgadas), y rendía una hora de comunicación y ocho horas en stand-by, con pantalla de LED.

Con el paso del tiempo los ordenadores han ido haciéndose más y más pequeños desde aquellos gigantes de acero y válvulas hasta los ordenadores personales de la actualidad, y estamos viendo ahora a la aparición de pequeños dispositivos del tamaño de la palma de la mano creciendo día a día en un afán de alcanzar la posibilidad de comparación con los ordenadores de sobremesa.

A la vez, las comunicaciones se han ido haciendo cada vez más importantes, por no decir imprescindibles. Muchas personas, ya no conciben un ordenador sin conexión a otros. El ordenador aislado poco a poco va convirtiéndose en una máquina del pasado.

Por otra parte, en los últimos años, el acercamiento de la telefonía móvil a todo tipo de gente mediante precios más o menos accesibles ha propiciado que nuestra conciencia de "estar conectados" aumente, y si esta es permanente, on-line, mucho mejor.

Uno de los servicios ofertados por estas operadoras en los teléfonos celulares de tecnología GSM es el envío y recepción de mensajes cortos, los conocidos SMS.

1.3. Evolución de los sistemas de comunicación móvil

Los sistemas de comunicaciones móviles han aumentado durante los últimos años. En el caso concreto de los servicios de telefonía se ha superado incluso el número de abonados de la telefonía fija. De hecho, la ITU ¹ ya reflejaba unas previsiones de crecimiento mayor para la telefonía móvil en su informe sobre el Desarrollo Mundial de las Telecomunicaciones del año 2002. Previsiones que se han cumplido, puesto que la edición del informe del 2006 ya refleja que el número de abonados a la telefonía móvil es superior y sigue en claro aumento, mientras que las redes fijas crecen en menor medida y en algunos países incluso están decreciendo.

¹International Telecommunication Union, Telefonía Movil

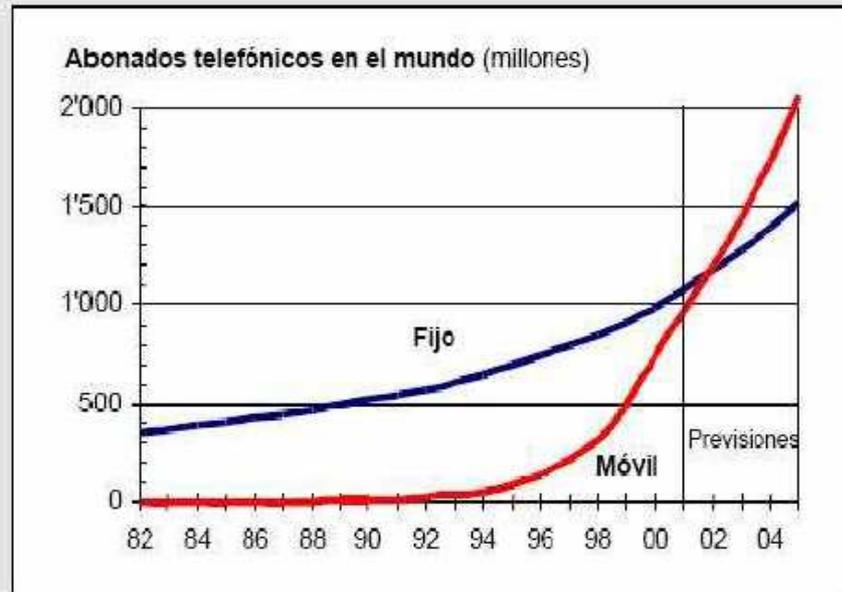


Fig. 1.1 Comparación entre usuarios de telefonía fija y móvil

Se puede observar que este crecimiento espectacular de la telefonía móvil empieza a mediados de los 90. No se debe a una casualidad, ya que éste es el momento en que aparecen los sistemas de telefonía móvil de segunda generación (2G).

Los sistemas 2G permiten la transmisión de datos, pero su baja velocidad y el hecho que estuvieran orientados a conexión (se factura por tiempo de conexión) hizo que el principal servicio fuera el de voz. Sin embargo, el servicio de datos basado en mensajes cortos (SMS) ha representado un gran éxito y actualmente es una de las más importantes fuentes de ingresos de las operadoras.

Este escenario empezó a cambiar con la introducción de los sistemas “2,5G”.

Estas redes están actuando de transición entre la 2G y las redes de tercera generación 3G. Con esta evolución los servicios de datos se han ido desarrollando y están propiciando el cambio de hábitos de los usuarios en lo que respecta al uso del terminal móvil. De esta forma, cada vez están adquiriendo más importancia los servicios de datos.

Actualmente los sistemas de tercera generación proporcionan una velocidad de transmisión mayor y un servicio de datos mejorado. Es decir, que el diseño de estos

sistemas se ha hecho con la intención de proporcionar acceso a servicios y aplicaciones multimedia.

1.4. Importancia de los sistemas de comunicación móvil

Por la necesidad de que tenemos las personas de estar comunicados se presenta telefonía móvil, dispositivos que nos permiten poder estar comunicados en cualquier lugar y como su nombre lo indica aún en movimiento, sea en el coche, en otra ciudad, en la calle o caminando.

La telefonía móvil en la sociedad a logrado ser de vital importancia ya que nos permite estar comunicados constantemente y además en la actualidad proporciona otros servicios como Internet, juegos, radio, etc., logrando que los usuarios quieran tener cada vez mas servicios adicionales y ocasionando que este se vuelva indispensable en la vida de una persona.

1.5. Servicios de mensajes de texto SMS

El *Short Message Service* (SMS) es la capacidad de enviar y recibir mensajes de texto desde y hacia teléfonos móviles. El texto se puede componer de letras, o números o una combinación de ellos. Los SMS se crearon cuando se incorporaron al estándar *Global System for Mobiles* (GSM).

El servicio de mensajes cortos o SMS es un servicio disponible en los teléfonos móviles que permite el envío de mensajes cortos (también conocidos como mensajes de texto, o más coloquialmente, textos o mensajitos). SMS fue diseñado originariamente como parte del estándar de telefonía móvil digital GSM, pero en la actualidad está disponible en una amplia variedad de redes, incluyendo las redes 3G.

El éxito de SMS, parece provenir de la sencillez y facilidad de manejo, por un lado, y de que "hay alguien al otro lado" con quien realizar el acto de la comunicación. Estos dos factores han provocado dicho éxito, aún teniendo en contra el precio del servicio en algunos casos y las limitadas características de esta comunicación.

1.5.1 Las ventajas del SMS

En la actualidad los mensajes SMS tienen toda una serie de ventajas de acuerdo a sus características:

Almacenaje y envío

Esto significa que en el caso de que el envío y el receptor no estén disponibles, el mensaje es almacenado.

Confirmación de entrega

Esto significa que el usuario sabe que el mensaje corto ha llegado a su destino.

1.5.2 Desventajas de los SMS

Tamaño limitado de los mensajes

El mensaje corto está limitado a 140 octetos debido a las limitaciones de la Mobile Application Part (MAP). Sería preferible un mensaje que tenga varias veces esta magnitud.

Estructura inflexible del mensaje

La estructura del protocolo SMS Protocol Data Unit como está definido en el estándar GSM 03.40 es inflexible. El Data Coding Scheme, Origination Address, Protocol Identifier y otros campos de cabecera están fijos y esto ha limitado el número de escenarios posibles. Incluso el uso de algunas características impide que se usen otras.

Almacenaje y Envío

Todos los mensajes SMS pasan por el centro de mensajes del proveedor. Mucho se ha discutido sobre la posibilidad de enviar mensajes sin pasar por esos centros pero sin éxito hasta la fecha

Las posibilidades de comunicación mediante "mensajes cortos" SMS son muchas y muy variadas, pero siempre limitadas por las características de estos mensajes, 160

caracteres, muy baja velocidad (en comparación con las líneas telefónicas convencionales), duración limitada (24 ó 48 horas normalmente, si no se entregan antes son cancelados), no es un servicio garantizado (el mensaje suele llegar pero no hay garantía de ello, ni que lleguen en el orden en que se han enviado) y posibilidad de comunicación sólo entre teléfonos celulares GSM entre los que haya "visibilidad" (que los operadores de los dos teléfonos, emisor y receptor, tengan convenio de intercambio de mensajes). Existen muchas especificaciones de formato de mensaje para los servicios prestados a través de SMS que les dotan de gran potencia y complejidad. Pero es en el uso básico con un sistema de enlace sencillo donde se están obteniendo los mejores resultados, tanto de cantidad de mensajes enviados como de servicios que se están utilizando.

En todo caso, estas posibilidades resultan suficientes aprovechadas de forma adecuada, y una de esas formas es tener uno de los lados de la comunicación gobernado por un servicio automático que se encargue de responder a las peticiones recibidas desde múltiples teléfonos celulares.

1.6. Conclusión:

- La telefonía móvil ha permitido mantener en total comunicación a las personas estableciéndose como una de las principales maneras de comunicarse.
- El éxito de SMS, parece provenir de la sencillez y facilidad de manejo, por un lado, y de que "hay alguien al otro lado" con quien realizar el acto de la comunicación. Estos dos factores han provocado dicho éxito, aún teniendo en contra el precio del servicio en algunos casos y las limitadas características de esta comunicación.
- La mensajería *Short Message Service* o los SMS no sólo pueden utilizarse en la comunicación entre personas. Puede obtenerse un gran servicio de su utilización para comunicar con agentes automáticos que respondan a mensajes enviados automática o manualmente.

CAPITULO II: DESCRIPCIÓN DEL SOFTWARE QUE UTILIZAREMOS PARA LA IMPLEMENTACIÓN.

2.1. Introducción

En este capítulo se presentará una breve descripción del lenguaje de programación Visual Basic 6.0, las características generales, su incursión en la programación, su origen, desarrollo, conceptos básicos, herramientas, entorno y algunos conceptos más que necesitaremos para el desarrollo de esta monografía, también analizaremos las ventajas y desventajas de este lenguaje.

2.2. Definición de Visual Basic 6.0

Visual Basic 6.0 es una excelente herramienta de programación que permite crear aplicaciones propias (programas) para *Windows* o *Windows NT*. Sus aplicaciones en Ingeniería son casi ilimitadas: representación de movimientos mecánicos o de funciones matemáticas, gráficas termodinámicas, simulación de circuitos, etc.¹

Este programa permite crear ventanas, botones, menús y cualquier otro elemento de *Windows* de una forma fácil e intuitiva. El lenguaje de programación que se utilizará será el *Basic*.¹

Microsoft Visual Basic 6.0 es un entorno integrado de desarrollo que forma parte de Microsoft Visual Studio 6. Fue la mejor herramienta para desarrollar aplicaciones para Windows, en cuanto a su productividad: hacer un programa para Windows con Visual Basic era mucho más rápido y fácil que hacerlo con Visual C++.

Las principales características del lenguaje Visual Basic son:

1. Soporta la Programación Estructurada.

¹ GARCIA DE JALON, Javier. RODRIGUEZ, José Ignacio. BRAZALEZ, Alfonso. *Aprenda Visual Basic 6.0*. San Sebastián: 1999. p. 14

2. Soporta la Programación Basada en Objetos. No la Programación Orientada a Objetos, ya que sus clases no permiten la herencia, sino que está basada en el modelo de componentes
3. Integración con objetos ActiveX
4. Rápida curva de aprendizaje.
5. Destrucción automática de objetos
6. Soporte al modelo de eventos.
7. Soporte al modelo de propiedades.

Sus principales desventajas son:

1. Pobre mecanismo de control de errores. No soporta las excepciones, que fueron adoptadas en Visual Basic .NET.
2. Dificultad para el desarrollo de estructuras de datos complejas.
3. Necesidad de colaboración con componentes desarrollados en otros lenguajes para tareas no triviales, o que simplemente son imposibles de realizar con este lenguaje.
4. El código generado es menos eficiente que el código generado de otros lenguajes, como C++.

2.3. Origen y Desarrollo de Visual Basic 6.0:

Visual Basic es uno de los tantos lenguajes de programación que podemos encontrar en la actualidad. Dicho lenguaje nace del BASIC que fue creado en su versión original en el *Dartmouth College*, con el propósito de servir a aquellas personas que estaban interesadas en iniciarse en algún lenguaje de programación. Luego de varias modificaciones, en 1978 se estableció el BASIC estándar.²

La sencillez del lenguaje ganó el desprecio de los programadores avanzados por considerarlo "un lenguaje para principiantes".

El primero que salió fue GW-BASIC, luego apareció QuickBASIC y actualmente se lo conoce como Visual Basic y la versión más reciente es la 6 que se incluye en el paquete Visual Studio 6 de Microsoft. Esta versión combina la sencillez del BASIC con un poderoso lenguaje de programación Visual que juntos permiten desarrollar

programas sofisticados de 32 bits para Windows. Esta agrupación de sencillez y estética permitió ampliar mucho más los privilegios de Microsoft, ya que el lenguaje sólo es compatible con Windows, un sistema operativo de la misma empresa. Visual Basic es una perfecta alternativa para los programadores de cualquier nivel que deseen desarrollar aplicaciones compatibles con Windows.

2.4. Características de Visual Basic 6.0

Es un lenguaje de fácil aprendizaje pensado tanto para programadores principiantes como expertos, guiado por eventos, y centrado en un motor de formularios que facilita el rápido desarrollo de aplicaciones gráficas. Su sintaxis, derivada del antiguo BASIC, ha sido ampliada con el tiempo al agregarse las características típicas de los lenguajes estructurados modernos. Se ha agregado una implementación limitada de la programación orientada a objetos (los propios formularios y controles son objetos), aunque sí admite el polimorfismo mediante el uso de los Interfaces, no admite la herencia.

No requiere de manejo de punteros y posee un manejo muy sencillo de cadenas de caracteres. Posee varias bibliotecas para manejo de bases de datos, pudiendo conectar con cualquier base de datos a través de ODBC (Informix, DBase, Access, MySQL, SQL Server, PostgreSQL ,etc) a través de ADO.

Es utilizado principalmente para aplicaciones de gestión de empresas, debido a la rapidez con la que puede hacerse un programa que utilice una base de datos sencilla, además de la abundancia de programadores en este lenguaje.

El compilador de Microsoft genera ejecutables que requieren una DLL para que funcionen, en algunos casos llamada MSVBVMxy.DLL (acrónimo de "MicroSoft Visual Basic Virtual Machine x.y", siendo x.y la versión) y en otros VBRUNXXX.DLL ("Visual Basic Runtime X.XX"), que provee todas las funciones implementadas en el lenguaje. Además existen un gran número de bibliotecas (DLL) que facilitan el acceso a muchas funciones del sistema operativo y la integración con otras aplicaciones. Sin embargo esto sólo es una limitación en sistemas obsoletos, ya que las bibliotecas necesarias para ejecutar programas en Visual Basic vienen de serie en todas las versiones de Windows desde Windows 2000.

2.5. Detección de Puertos

Utilizamos las siguientes instrucciones para la detección de puertos:

```
ccsize = LenB(cc)
DetectaPortaCOM = GetDefaultCommConfig("COM" + Trim(Str(port)) +
Chr(0), cc, ccsize)
SetStatus "Verificando los puertos Disponibles...", true
```

2.6. El Control MSCOMM



En nuestro sistema vamos a utilizar el control MSCOMM porque nos proporciona comunicaciones serie para que nuestra aplicación pueda transmitir y recibir datos a través del puerto USB.

Con este control establecemos la comunicación con el teléfono móvil, es la interfaz para poder acceder a los mensajes que recibe.

Este control nos proporciona dos formas diferentes de tratamiento de las comunicaciones:

- Las comunicaciones controladas por eventos son un método muy poderoso para el tratamiento de interacciones con el puerto serie. En muchas situaciones deseará que se le notifique cuándo tiene lugar un evento. Se utiliza el evento *OnComm* de éste control para interceptar y tratar estos eventos de comunicaciones; también detecta y trata los errores en las comunicaciones. En la propiedad *CommEvent* puede ver una lista completa de todos los eventos y errores posibles en las comunicaciones.

- También puede sondear los eventos y errores si comprueba el valor de la propiedad CommEvent después de cada función crítica de su programa. Esta alternativa es preferible si la aplicación es pequeña y autónoma.

Cada uno de los controles MSComm que use corresponde a un puerto. Si necesita tener acceso a más de un puerto en su aplicación, debe usar más de un control MSComm. La dirección del puerto y la dirección de la interrupción pueden cambiarse desde el Panel de control de Windows.

El control MSComm tiene muchas propiedades importantes como son:

Propiedades	Descripción
CommPort	Establece y devuelve el número del puerto de comunicaciones.
Settings	Establece y devuelve la velocidad en baudios, paridad, bits de datos y bits de parada en forma de cadena.
PortOpen	Establece y devuelve el estado de un puerto de comunicaciones. También abre y cierra un puerto.
Input	Devuelve y quita caracteres del búfer de recepción.
Output	Escribe una cadena de caracteres en el búfer de transmisión.

2.6.1 Conexión con MSCOMM:

Para poder realizar la conexión con el teléfono móvil procedemos a ejecutar las siguientes instrucciones para configurar primordialmente el MSCOMM:

Mscomm1.**Handshaking** = Comnone (Elijo Ignorar Cualquier Protocolo A Nivel Hardware.)

MscComm1.**Inbuffersize** = 1024 (Definen Cuanta Memoria Es Designada Para Los Buffers De Recepción.)

MscComm1.**Rthreshold** = 1 (Números De Caracteres Que Deben Encontrarse En El Buffer)

MscComm1.**Eofenable** = False (Cuando Un Carácter De Fin De Archivo (End Of File) Fue Encontrado Durante Una Entrada De Datos)

MscComm1.**Outbuffersize** = 1024 (Definen Cuanta Memoria Es Designada Para Los Buffers De Transmisión)

MscComm1.**Sthreshold** = 1 (Determina Cuantos Caracteres Deben Transmitirse Antes De Generar Un Evento OnComm)

MscComm1.**Nulldiscard** = False (Hace Que El Caracter Nulo Se Tome Como Un Carácter Mas)

MscComm1.**Rtsenable** = True (Indica Al Modem Que Deseamos Enviar Datos)

MscComm1.**Dtreable** = True (Indica Al Modem Que El Terminal Esta Preparado Para Recibir Datos)

MscComm1.**Settings** = "9600,N,8,1" (Determina A Qué Velocidad Se Harán Las Comunicaciones Y Otros Parámetros)

MscComm1.**Inputlen** = 0 (Leer Todos Los Caracteres Quedando El Buffer Vacío)

MscComm1.**Inputmode** = Cominputmodebinary (Habilitamos Para Que Reciba Tipos De Datos Caracteres No Imprimibles)

Primeramente para realizar la comunicación con el teléfono móvil debemos utilizar el evento **OnComm**, luego si queremos realizar la recepción de un mensaje utilizamos las siguientes instrucciones:

```
Select Case MScComm1.CommEvent
```

```
Case comEvReceive
```

```
SMESSAGE = StrConv(MScComm1.Input, vbUnicode)
```

Para saber que recibe el modem utilizamos lo siguiente:

```
Private Sub ReceiveResponse(sResponse As String)
```

```
txtOutput = txtOutput & "Respuesta Recibida" & vbCrLf & sResponse & vbCrLf  
& "Fin de Rspuesta" & vbCrLf
```

```
txtOutput.SelStart = Len(txtOutput)
```

```
txtOutput.Refresh
```

End Sub

2.7. Conexión con la Base de Datos de Access:

Para poder acceder a una Base de Datos en Access mediante Visual Basic 6.0 es muy fácil simplemente se debe ejecutar las siguientes instrucciones:

```
***** Colocamos una variable para colocar la Base de Datos *****  
Public nombre_variable as Database  
***** Abrimos la Base de Datos *****  
Set nombre_variable = OpenDatabase(App.Path & nombre_base_datos)
```

Mediante la utilización de registros nosotros accedemos a las tablas correspondientes, con siguientes pasos:

a) Definimos las variables :

+ Definimos la variable para incorporar el registro.

Dim **REGISTRO** As Recordset

+ Definimos la variable para asignar el Select.

Public **SQL** As String

b) Utilizamos la siguiente sentencia para acceder a una tabla:

SQL = "Select * from nombre_tabla"

Set **REGISTRO** = nombre_variable_base.OpenRecordset(**SQL**)

If **REGISTRO**.RecordCount = 1 Then

Instrucciones a Ejecutar

End If

2.8. Conclusión

El lenguaje de programación Visual Basic es un lenguaje de fácil aprendizaje pensado tanto para programadores principiantes como expertos, guiado por eventos, y centrado en un motor de formularios que facilita el rápido desarrollo de aplicaciones gráficas.

La facilidad del lenguaje permite crear aplicaciones para Windows en muy poco tiempo. En otras palabras, permite un desarrollo eficaz y menor inversión en tiempo que con otros lenguajes.

También este lenguaje tiene algunos inconvenientes como que no existe forma alguna de exportar el código a otras plataformas fuera de Windows, dificultad para el desarrollo de estructuras de datos complejas entre otras pero en definitiva diremos que este lenguaje de programación es muy útil y sencillo.

CAPITULO III: ESPECIFICACIÓN DETALLADA DEL FUNCIONAMIENTO DEL SISTEMA

3.1. Introducción

En este capítulo daremos a conocer de una manera muy detallada en que consiste el sistema, dividiendo en dos partes para su mejor entendimiento servidor y cliente, en las mismas explicaremos por separado su esquema, diseño, funcionamiento y los medios requeridos para su implementación.

3.2 Descripción General del Sistema

El proyecto pretende realizar la implementación de un sistema de envío y recepción de mensajes SMS para acceder a la base de datos y obtener la información de un producto de ferretería específico que el cliente quiera pedir en ese momento.

El sistema consta principalmente de los siguientes puntos:

- Un sistema de seguridad de acceso a la base de datos constituida por un nombre de usuario y un password, con lo cual el usuario podrá acceder a la información almacenada en la base de datos, también le facilitará al cliente la posibilidad de registrarse en caso de que este no estuviera registrado
- El cliente realizará la consulta del artículo ingresando el código del mismo el cual el sistema le enviará la información como:
 - o Código
 - o Descripción
 - o Cantidad
 - o Precio.

- Una vez que el cliente ingrese el código del artículo este deberá ingresar la cantidad que desee pedir, si el cliente ingresa una cantidad superior a la que existe este le pedirá que ingrese nuevamente una cantidad
- El sistema se encargará de actualizar la base de datos con la nueva cantidad de artículos
- El sistema también elimina al cliente de la cola de espera que luego de un cierto tiempo para ser exactos 1 minuto no sigue con la realización de pedidos.

El objetivo de este proyecto es brindar al cliente la comodidad, facilidad y rapidez para que este realice un pedido en el lugar en el que este se encuentre simplemente con la utilización de mensajes de texto y con el código del artículo que quiera pedir.

El sistema se ha estructurado en dos partes. La primera, denominada Servidor, es la aplicación: es la encargada de la configuración, el envío, la recepción de mensajes y la actualización de la base de datos. La segunda parte se ha denominado Cliente y es el que envía el mensaje con el artículo que desea pedir

Los detalles de la implementación de los módulos Servidor y Cliente se verán a continuación.

3.3 Módulo Servidor

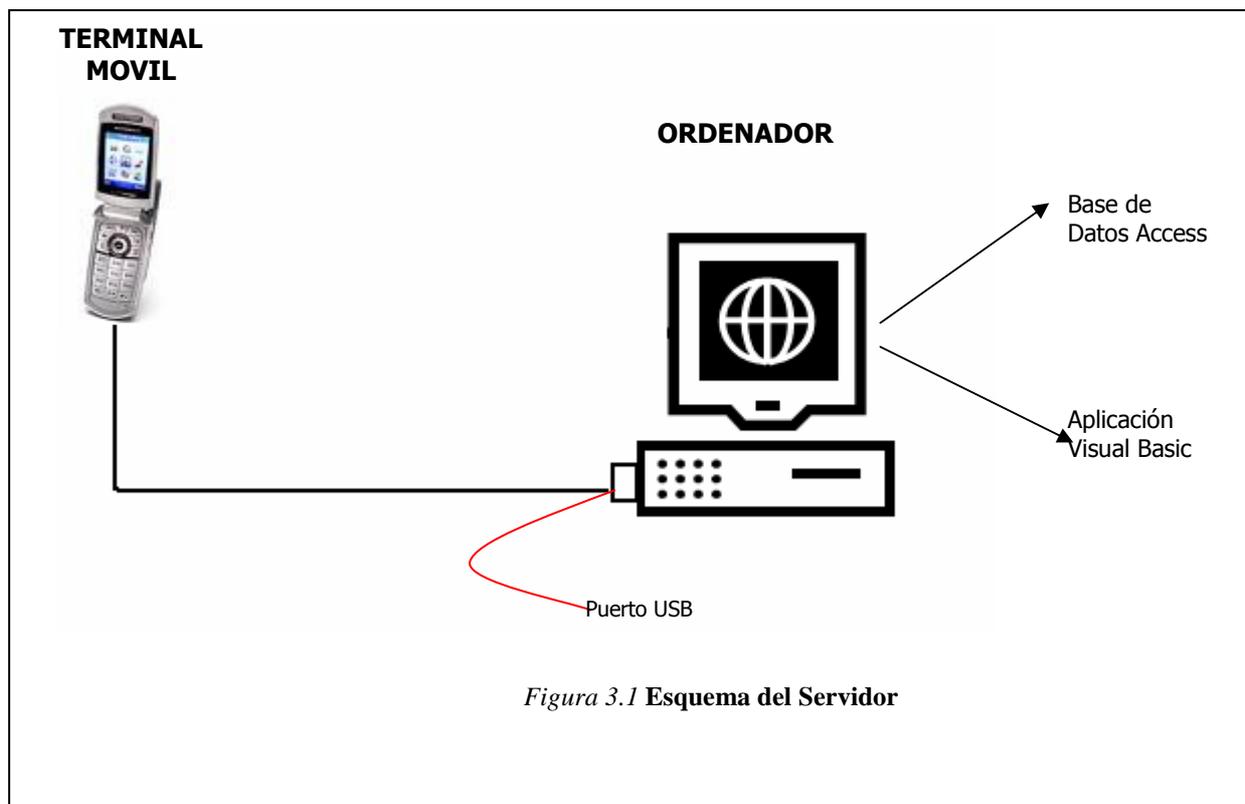
3.3.1 Diseño y esquema

El módulo denominado Servidor es el encargado de la configuración, el envío y la recepción de mensajes así como de la actualización de la base de datos y el terminal en donde se recibe el mensaje esta implementado en el módulo (Cliente). Este módulo es capaz de soportar varios clientes que realizan un pedido al mismo tiempo.

Para su desarrollo se necesita un terminal móvil que disponga de mensajes de texto, el cual debe estar conectado a un ordenador y la aplicación desarrollada en visual Basic y una base de datos Access en donde se encuentra la información referente al

producto. En el ordenador se ejecutará el programa encargado de controlar el terminal móvil de modo que mande un mensaje de texto a un número predeterminado (en nuestro caso será el número del móvil que realiza el pedido) en caso de que se haya recibido un mensaje por parte del cliente

Para el desarrollo del módulo Servidor se puede utilizar cualquier teléfono móvil que tenga una forma de envío de mensajes de modo texto y conexión a un PC.

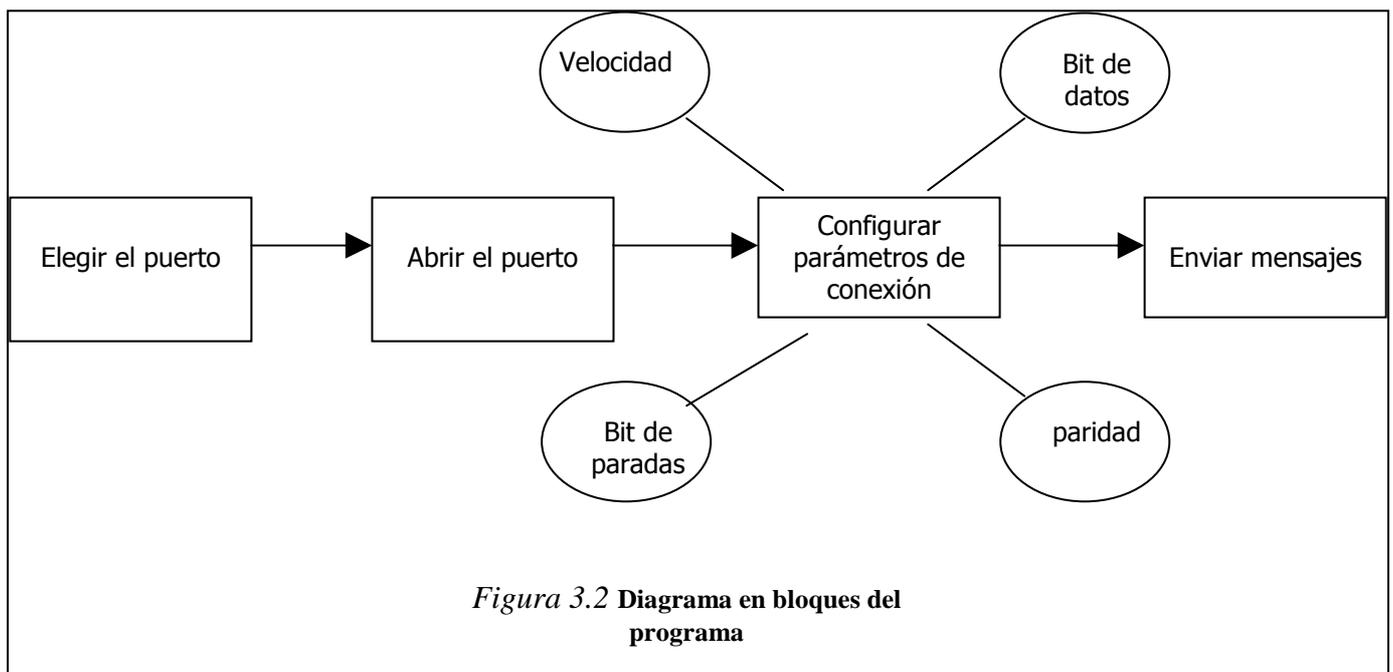


Como podemos observar en la figura 3.1 la conexión entre el ordenador y el teléfono celular se hace a través del puerto USB.

Para que el teléfono móvil funcione el ordenador deberá contener los drivers del móvil que estemos utilizando.

3.3.2 Envío y recepción de mensajes al dispositivo remoto

Hemos desarrollado una aplicación capaz de enviar y recibir mensajes desarrollada en visual Basic considerando este una excelente herramienta de programación como lo vimos en el capítulo anterior, como muestra la figura a continuación el primer paso consiste en elegir y abrir el puerto con el que queremos trabajar, después tendremos que configurar algunos parámetros como velocidad, los bits de datos, los bits de paradas y la paridad y finalmente se envía y se recibe el mensajes mediante comandos AT.



3.3.3 Puerto USB

USB Universal Serial Bus es una interface plug&play entre la PC y ciertos dispositivos tales como teclados, mouses, scanner, impresoras, módems, placas de sonido, cámaras, teléfonos móviles, etc.

Una característica importante es que permite a los dispositivos trabajar a velocidades mayores, en promedio a unos 12 Mbps, esto es más o menos de 3 a 5 veces más rápido que un dispositivo de puerto paralelo y de 20 a 40 veces más rápido que un dispositivo de puerto serial.

La comunicación entre el ordenador y el Terminal en este caso el teléfono celular que enviará y recibirá los mensajes se realiza por uno de los puertos USB que tenga el ordenador.

El primer paso consiste en elegir y abrir el puerto por donde queremos que se establezca la comunicación con el celular, podemos escoger cualquier puerto siempre y cuando este no este ocupado, se deberá configurar los parámetros que vimos anteriormente como la velocidad, los bits de datos, los bits de parada y la paridad.

En la figura 3.3 veremos los parámetros que hemos escogido para configurar el puerto



Figura 3.3 Parámetros de Configuración

Los valores de los parámetros que se han escogido para configurar el puerto son los siguientes:

- **Baud Rate:** la velocidad 9600 bps (bits por segundo)
- **Data Bits:** 8 bits de datos
- **Parity:** sin paridad
- **Stop Bits:** 1 bit de parada.

Una vez que hemos abierto y configurado el puerto tenemos que mandar comandos para poder enviar y recibir los mensajes, los mensajes que se tendrá que enviar están ya establecidos en la aplicación.

Esto se lo hace a través de los comandos AT que son instrucciones que entienden los teléfonos móviles pero mas detalladamente veremos a continuación.

3.3.4 Comandos AT

Los comandos AT son instrucciones codificadas que conforman un lenguaje de comunicación entre el hombre y un terminal módem. Los comandos AT se denominan así por la abreviatura de *attention*.

El juego de comandos AT fue desarrollado en 1977 por Dennis Hayes como un interfaz de comunicación con un módem para poder configurarlo y proporcionarle instrucciones, tales como marcar un número de teléfono. Más adelante, fueron las compañías Microcomm y US Robotics las que siguieron desarrollando y expandiendo el juego de comandos hasta hacerles universales.

La telefonía móvil GSM también ha adoptado como estándar este lenguaje para poder comunicarse con sus terminales. De esta forma, todos los teléfonos móviles GSM poseen un juego de comandos AT específico que sirve de interfaz para configurar y proporcionar instrucciones a los terminales.

La estructura necesaria para ejecutar estos comandos es sencilla: cualquier comando que se quiera introducir debe ir siempre precedido de AT.

Aunque inicialmente estos comandos se desarrollaron para comunicarse con un módem, la telefonía móvil también ha adoptado este lenguaje. Los teléfonos móviles poseen un grupo específico de comandos AT que permiten configurarlos. La diferencia respecto a los comandos de los módems es que en este caso no se trata de un grupo de instrucciones universales, sino que existen comandos propios para cada fabricante (aunque los comandos básicos son los mismos en todos los terminales).

También hay que tener en cuenta que algunos móviles no permiten ejecutar todo el juego de comandos AT. Por lo tanto, cuando se pretende trabajar con los comandos AT específicos, se requiere buscar el juego de comandos del fabricante del terminal y comprobar que el teléfono soporta los que se van a necesitar.

En nuestro caso, se ha empleado un terminal Motorola y es necesario utilizar los comandos referentes al envío y recepción de mensajes de texto. Estos son algunos de los comandos que detalla el fabricante y que serán necesarios para desarrollar el programa:

CMGF: Comando para seleccionar el formato del mensaje. Existen dos formatos posibles: el modo binario PDU y el modo texto. Para especificar el formato se debe especificar un "0" para el modo PDU, o escribiendo "1" para el modo texto. En nuestro caso utilizaremos en modo texto lo cual detallaremos posteriormente

CSCA: Se introduce este comando para indicar el número del centro de mensajería de la compañía telefónica. Este centro se conoce como SMSC y para el envío de cualquier mensaje mediante comandos AT es imprescindible especificar este número.

CMGS: Para enviar el mensaje. Al escribir el comando se debe detallar el número del destinatario. Después de ejecutarlo se escribe el mensaje que se desea enviar.

CMGL: Informa de los posibles estados de un mensaje en la memoria que el teléfono puede soportar como.

0 / "REC UNREAD": Almacenado en Bandeja de entrada y sin leer.

1 / "REC READ": Almacenado en Bandeja de entrada y leído.

2 / "STO UNSENT": Almacenado en Bandeja de salida y sin enviar.

3 / "STO SENT": Almacenado en Bandeja de salida y enviado.

4 / "ALL": Todos los mensajes almacenados.

AT+CMGR: Permite leer mensajes SMS de la bandeja de entrada.

Estos comandos son suficientes para los requerimientos de nuestra aplicación que es el envío y la recepción de mensajes.

3.3.5 Envío de mensajes modo texto

El modo texto es mucho más sencillo de desarrollar que el modo PDU, pero hay que tener en cuenta que algunos terminales no soportan este formato.

Para realizar el programa que envíe el mensaje en modo texto se necesitan ejecutar algunos de los comandos vistos anteriormente, en el que sólo hay que indicar el número de teléfono al que queremos enviar el mensaje y el contenido del mensaje.

El primer comando que se ejecuta en el programa es CMGF con el valor 1, para indicarle que se enviará un mensaje en modo texto seguido de un retorno de carro y un salto de línea.

MSComm1.Output = "AT+CMGF=1" & Chr\$(13) & Chr(10)

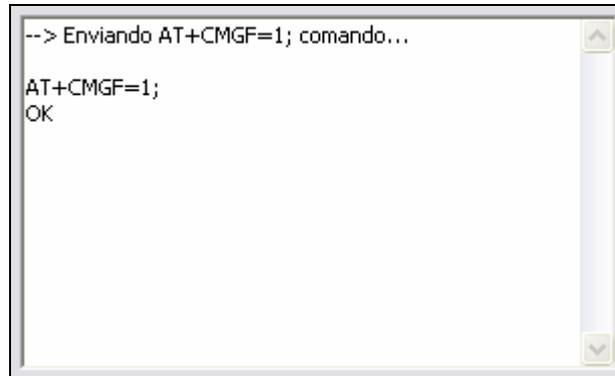


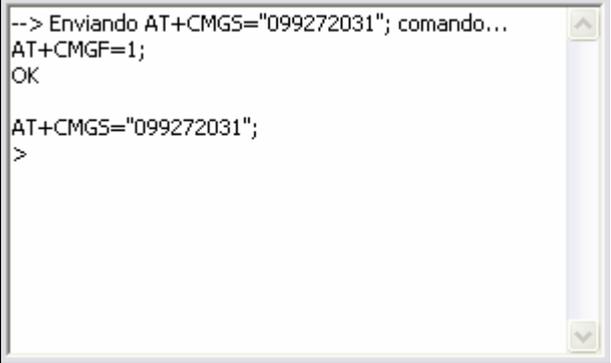
Figura 3.4 Cuadro de Resultado del comando CMGF

Como vemos en la figura 3.4 después de ejecutar el comando el modem nos devuelve OK, esto quiere decir que el teléfono móvil si soportó modo texto

Finalmente se escribe el comando referente al envío de mensajes (CMGS) especificando el número del destinatario. Al ejecutar el comando se escribe el mensaje que se desea enviar. Una vez escrito hay que presionar (Ctrl Z) para enviarlo.

MSComm1.Output = "AT+CMGS=" & Chr(34) & telefono_destinatario & Chr(34) & Chr\$(13)

MSComm1.Output = "MENSAJE" & Chr(26)



```
--> Enviando AT+CMGS="099272031"; comando...
AT+CMGF=1;
OK
AT+CMGS="099272031";
>
```

Figura 3.5 Cuadro de Resultado del comando CMGS

Como vemos en la figura 3.5 el modem responde enviando el carácter “>” que indica que se puede escribir el mensaje que se quiere enviar.

Si el mensaje se ha enviado correctamente, devuelve la cadena “+CMGS:<nr>” seguida del OK.

3.3.6 Recepción de mensajes

El sistema desarrollado interactúa con el cliente recibiendo mensajes dependiendo de las necesidades de cada cliente para ello necesitamos utilizar algunas instrucciones que detallaremos a continuación para su mejor entendimiento:

SMESSAGE = StrConv (MSComm1.Input, vbUnicode)

Recibimos el mensaje con la instrucción *MSComm1.Input*, luego la convertimos en *vbUnicode* que es un carácter estándar de la Organización internacional de estándares ISO y utiliza un esquema de códigos de 16-bits (2 bytes) que permite 65.536 espacios de caracteres distintos; y, todo esto almacenamos en la variable de tipo String *SMESSAGE* la cual la utilizaremos para los diferentes requerimientos del usuario.

3.3.7 Actualización de la base de datos

Una vez que el cliente envía el mensaje con el código y la cantidad que desea pedir el último paso que tiene que hacer este módulo es la actualización de la base de datos de manera automática cuando el cliente envíe el mensaje con la cantidad del artículo.

La base de datos esta desarrollada en Access y contiene 3 tablas:

- Artículos
- Usuarios
- Estado

Las mismas que están estructuradas de la siguiente manera:

Artículos: Se encuentra toda la información necesaria del artículo.

- Código ***Texto(5)*** : *Código del artículo con el cual el cliente puede hacer el pedido*
- Descripción ***Texto(50)***: *Nombre del artículo*
- Cantidad ***Número***: *Cantidad disponible para hacer el pedido*
- Precio ***Moneda***: *Precio que tiene el artículo*

Usuarios: Se encuentra la información relacionada con el cliente

- Usuario ***Texto(10)***: *Nombre de usuario que eligió el cliente*
- Password ***Texto(10)***: *Contraseña del cliente para poder realizar el pedido*

Estado: Esta la información para poder realizar el pedido

- Número ***Texto(12)***: *Numero de teléfono del cliente que esta realizando el pedido*
- Estado ***Número***: *Sirve para saber en que estado se encuentra el cliente como registro, consulta y pedido*
- Artículo ***Texto(5)***: *Código del artículo que se esta pidiendo*
- Hora ***Fecha/Hora***: *Hora que el cliente está realizando cada uno de los pasos, sirve para eliminar el cliente cuando este deje de hacer los procedimientos.*

3.3.8 Eliminación de Clientes

El sistema tiene la posibilidad de eliminar a los clientes que luego de un cierto tiempo en este caso un minuto no han seguido con el proceso de pedido de artículos es decir, por ejemplo si el cliente ha ingresado al sistema con su usuario y password correctamente y no ha realizado la consulta de artículos el sistema automáticamente le elimina de la tabla auxiliar ESTADO así si el cliente desea realizar la consulta este tendrá nuevamente que ingresar al sistema.

3.4 Módulo Cliente

3.4.1 Diseño y esquema

El módulo denominado Cliente es el terminal móvil en donde el cliente puede enviar y recibir mensajes.

La aplicación que se encuentra en el módulo servidor es capaz de soportar más de un cliente al mismo tiempo como vimos anteriormente.

El cliente tiene que enviar primeramente un mensaje con el usuario y la contraseña y si no lo tiene puede registrarse, debiendo enviar el nombre de usuario un espacio y la contraseña.



Figura 3.6 Imagen del teléfono móvil con el nombre de Usuario y Contraseña

El servidor le responderá confirmando o pidiendo que ingrese nuevamente el usuario y contraseña:



Figura 3.7 Imagen del teléfono móvil confirmación de usuario



Figura 3.8 Imagen del teléfono móvil cuando es incorrecto

Luego que el servidor responda confirmando su usuario deberá enviar el código del artículo que desea pedir



Figura 3.7 Imagen del teléfono móvil con el código del Artículo

Si el artículo es incorrecto el servidor responderá:

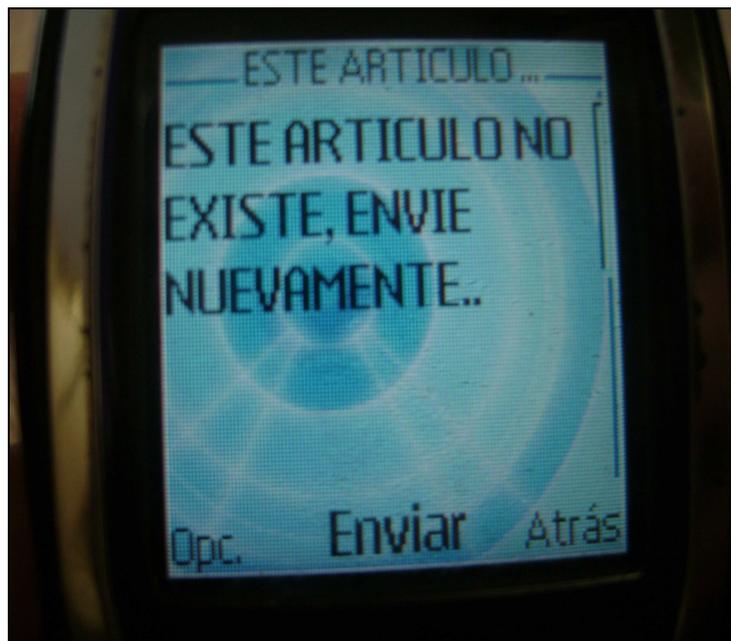


Figura 3.8 Imagen del teléfono móvil articulo no existe

Si el artículo que el cliente ha ingresado es correcto el sistema le envía las características de este producto como nombre, cantidad y precio para que el cliente pueda realizar el pedido que desee

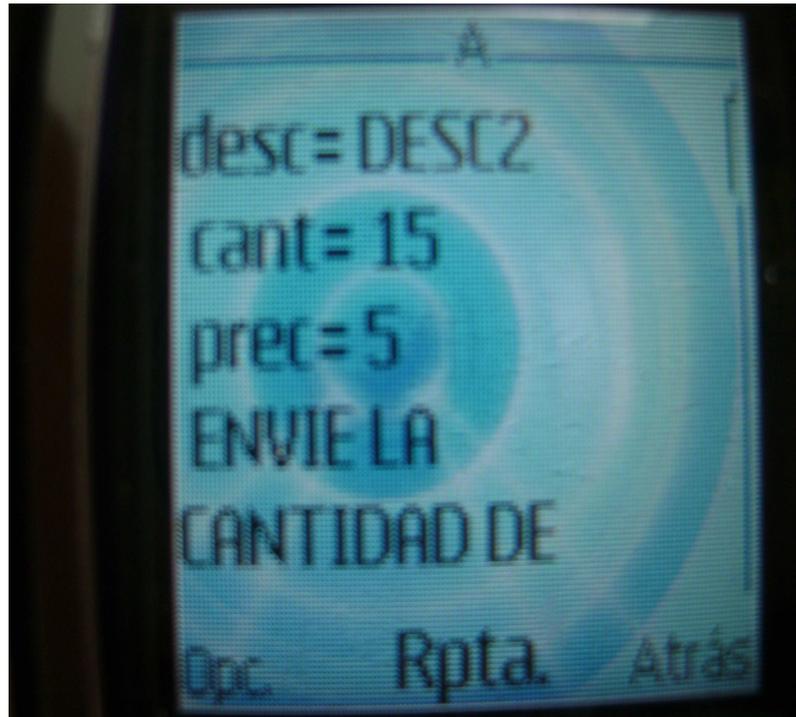


Figura 3.9 Imagen del teléfono móvil con la información del artículo

El cliente envía el mensaje con la cantidad de artículos que desee hacer la consulta.



Figura 3.10 Imagen del teléfono móvil con la cantidad que desea pedir

El sistema también valida si la cantidad que el cliente desea pedir es disponible si no envía un mensaje notificándole al cliente

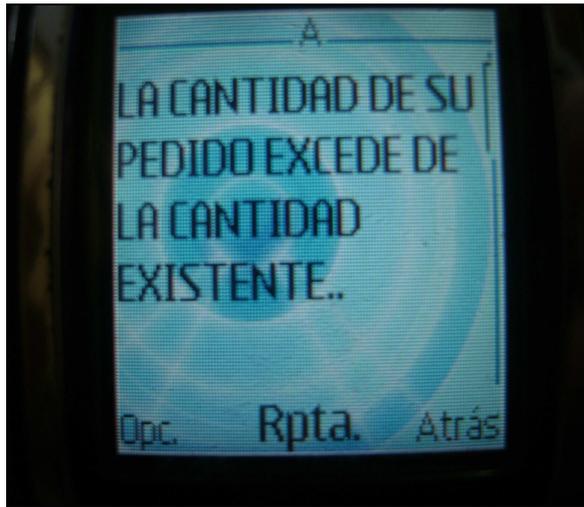


Figura 3.11 Imagen del teléfono móvil con la validación de la cantidad

3.5 Casos de Uso

Para un mejor entendimiento de los dos módulos relacionados realizaremos el caso de uso (*Anexo 1*) correspondiente con sus respectivos actores y relaciones.

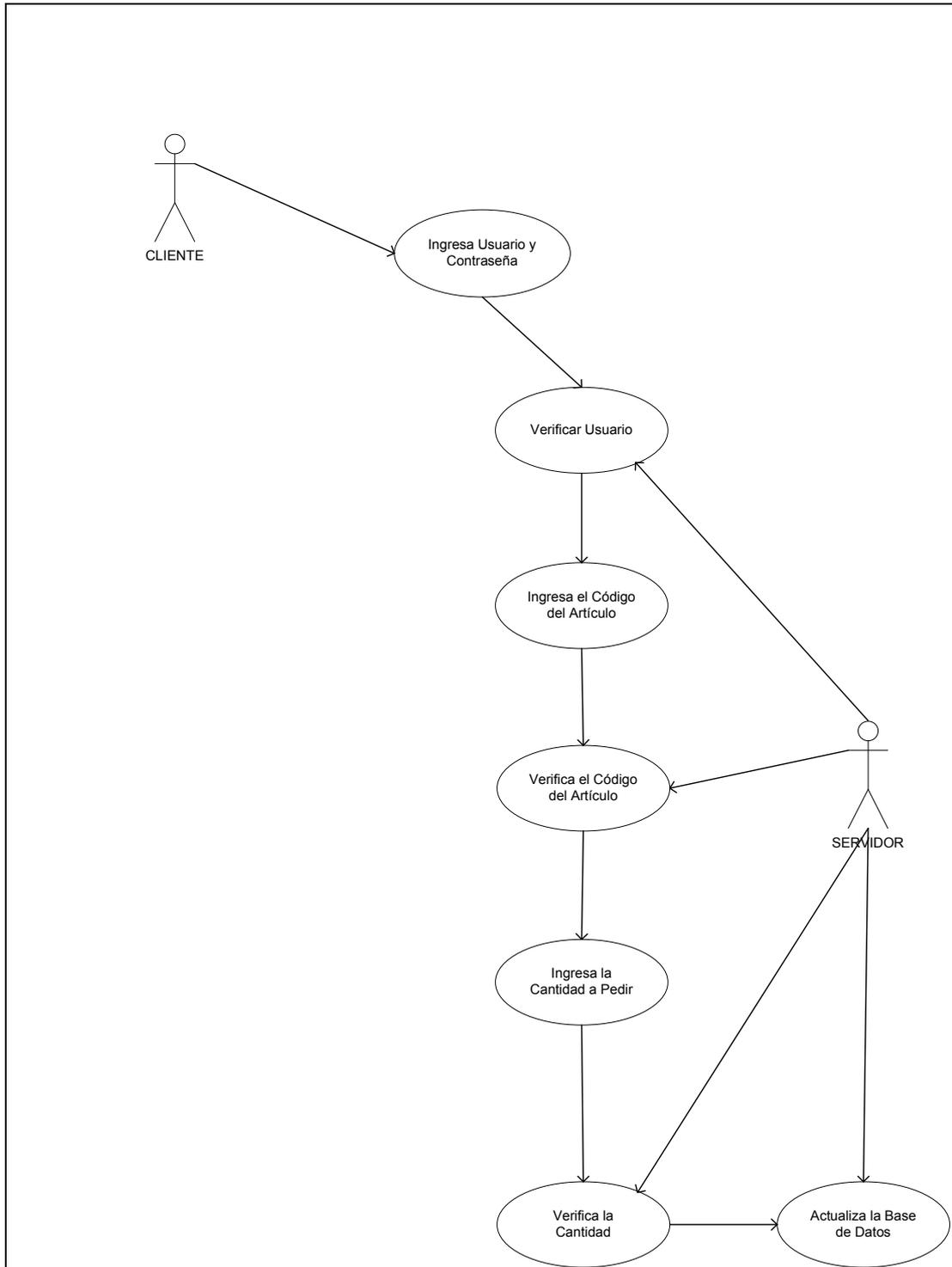


Figura 3.9 **Caso de Uso 1: Pedido de Artículo**

3.5.1 Descripción de actores

Cliente: Encargado de realizar el pedido mediante un mensaje SMS utilizando un celular.

Servidor: Encargado de configurar las opciones y estructuras del sistema así como de inicializar los parámetros del sistema.

3.5.2 Descripción del caso de uso

Caso de uso 1	Realizar Pedidos
Actor:	Cliente, Servidor
Descripción:	Se ingresan el nombre de usuario y contraseña y se realiza el pedido.
Prioridad:	Obligatorio
REQUISITOS ASOCIADOS	
R.1.1	El sistema permitirá ingresar el nombre de usuario y la contraseña.
R.1.2	El sistema verificará si existe el usuario.
R.1.3	El sistema permitirá ingresar el código del artículo si el usuario se encuentra registrado.
R.1.4	El sistema verificará el código del artículo.
R.1.5	El sistema permitirá ingresar la cantidad que el cliente quiera realizar el pedido.
R.1.6	El sistema verificará la cantidad del pedido.
R.1.7	El sistema actualizará la base de datos.

3.5.3 Detalle del caso de uso

En nuestro caso de uso el sistema permite que el cliente ingrese el código del usuario y la contraseña haciendo que el servidor valide si el usuario existe o no, si existe el cliente debe ingresar el código tomando en consideración que el cliente debe conocer el código exacto del artículo que queremos pedir y luego el servidor verificar si el código del artículo pertenece a nuestra Base de Datos, si pertenece entonces ingresamos la cantidad que queremos pedir y el servidor actualiza automáticamente nuestra Base de Datos; caso contrario si no existe el usuario, el servidor crea al nuevo usuario para que pueda realizar el pedido.

Para un mejor entendimiento adjuntaremos el código fuente en donde tenemos detallados todos los procedimientos, funciones, etc. utilizados para el Desarrollo de este Sistema. (*Anexo 3*)

3.6 Conclusiones

Como nos pudimos dar cuenta el sistema pretende facilitar al cliente para que realice un pedido en el lugar en el que éste se encuentre y de una manera muy rápida con

simplemente enviar un mensaje de texto y el servidor le responderá lo cual se convertirá en una conversación interactiva entre el servidor y el cliente.

Para que este proyecto funcione necesitamos de parte del servidor un ordenador y un teléfono móvil que soporte modo texto conectados a través de un puerto USB y de la parte del cliente un teléfono móvil.

El sistema es muy simple de utilizar y se podrá mantener una conversación interactiva con el cliente y este podrá realizar las diferentes actividades que necesita de una forma sencilla, lo cual le ahorra tiempo y dinero

CAPITULO IV: IMÁGENES Y GRÁFICOS PARA FACILIDAD DE ENTENDIMIENTO DEL USUARIO.

4.1 Introducción

En este capítulo brindaremos la información necesaria para que el usuario tenga facilidad en el manejo del sistema, mediante la utilización de imágenes las cuales detallarán los principales puntos de la Aplicación.



Figura 4.1: Pantalla Principal

4.2 Manual de usuarios que manejan el Sistema(Servidor)

4.2.1 Requerimientos para el Sistema:

4.2.1.1 Requerimientos de Hardware:

- Una computadora
- Un Teléfono Celular (modo Texto) *Ver Descripción en Anexo 2*
- Un Cable USB para conexión del Teléfono con la PC

4.2.1.2 Requerimientos de Software:

- Instalado Visual Basic en la PC.
- Drivers del Teléfono móvil a utilizar.
- Una Base de Datos en Microsoft Office Access

4.2.2 Configuración del Control de Comunicaciones (MSCOMM):

4.2.2.1 Definición del MSCOMM:

El control *Communications* permite agregar tanto una funcionalidad sencilla de comunicaciones de puerto serie, como una funcionalidad avanzada para crear una herramienta de comunicaciones completa controlada por eventos.

El control *Communications* proporciona una interfaz con un conjunto estándar de comandos de comunicaciones. Permite establecer una conexión con un puerto serie, conectar con otro dispositivo de comunicaciones (por ejemplo, un módem), emitir comandos, intercambiar datos, y supervisar y responder a varios eventos y errores que se pueden producir durante una conexión serie.

4.2.2.2 Configuración del MSCOMM:



Figura 4.2: Configuraciones del MSCOMM

Baub Rate: Indica la velocidad en baudios, los valores posibles pueden ser:

50 100 110 300 600 1200 2400 4800 9600 14400 19200 y
28800

El valor por defecto es **9600**.

Parity: Es usada en la validación de datos, los valores posibles para paridad son:

- N - No envía bit de paridad ni hace comprobación de paridad en la recepción.
- O - Envía y comprueba paridad, con el criterio de paridad IMPAR
- E - Envía y comprueba paridad, con criterio de paridad PAR

El valor por defecto es “**N**”.

Data Bits: Define el número de bits que representa un paquete de datos, los valores para el parámetro Bits de Información pueden ser:

- 7 - Se envían / reciben 7 bits por trama de información.
- 8 - Se envían / reciben 8 bits por trama de información
- 5 - Se envían / reciben 5 bits por trama de información.

El valor por defecto es **8**.

Stop Bits: Indica cuando un paquete de datos fue recibido, los valores para el parámetro Bits de parada pueden ser:

- 1 - Se envía un bit de parada
- 2 - Se envían 2 bits de parada

El valor por defecto es **1**.

4.2.3 Seleccionar el puerto de Comunicación:

Para saber en que puerto estamos trabajando hacemos clic en el botón , el cuál nos muestra los puertos que estamos utilizando;

dentro del sistema en la opción  seleccionamos o ingresamos el número del puerto correspondiente que vamos a utilizar.

4.2.4 Abrir el Puerto COM:

Para poder abrir el puerto COM simplemente hacemos clic en el botón:

ABRE PUERTO COM

4.2.5 Configuración del MSCOMM:

Hacemos clic en el botón de **CONFIGURACION** y en la aplicación esta definido los parámetros de configuración del MSCOMM:

```
MSComm1.Output = "AT+CMGF=1" & Chr$(13) & Chr$(10)
MSComm1.Output = "AT+CNMI=3,1,0,0,0" & ";" & Chr$(13) & Chr$(10)
MSComm1.Output = "AT+CPMS=?" & ";" & Chr$(13) & Chr$(10)
MSComm1.Output = "AT+CPMS?" & ";" & Chr$(13) & Chr$(10)
```

4.2.6 Limpiar Pantalla:

Para limpiar la pantalla en donde se ejecuta los distintos comandos damos clic en el botón:

LIMPIAR

4.2.7 Comandos AT:

Podemos enviar comandos AT para comprobar el buen funcionamiento del módem, colocando o escogiendo del siguiente combo **COMANDO** y escribiendo cualquiera de los comando que vimos en el capítulo 3.2.4, para ejecutar el comando damos un Enter o hacemos clic en el botón **ENYIA UN COMANDO**.

4.3 Manual de Clientes

4.3.1 Registro de Usuario y Contraseña:

1. Si el Cliente ya se encuentra registrado debe enviar su nombre usuario y contraseña correspondiente.
2. Si el Cliente no se encuentra registrado debe enviar el nombre de usuario y contraseña con el que desea registrarse.
3. Si el Cliente envía un nombre de usuario o contraseña incorrecta, el sistema le notificará enviando un mensaje, preguntando: Si desea registrarse o que vuelva a ingresar su nombre de usuario y contraseña.

4.3.2 Consulta del Artículo:

Una vez que el cliente se encuentra registrado, éste deberá enviar mediante un mensaje de texto el código del artículo que desea consultar, con el cuál el sistema le responderá con un mensaje mostrando la siguiente información:

- Código del artículo
- Descripción
- Cantidad
- Precio

4.3.3 Pedido del Artículo:

El usuario deberá enviar un mensaje con la cantidad del artículo que desea pedir, validando el sistema si la cantidad que pide es inferior o igual a la existente, e igualmente actualizando la Base de Datos, con lo cual el sistema mediante un mensaje nos indica que la Base de Datos fue actualizada satisfactoriamente; y, nos muestra el siguiente cuadro con la información actual:

DATOS DEL CLIENTE :		RECIBIRA LA INFORMACION SIGUIENTE:	
NUMERO DE TELEFONO:	<input type="text" value="099272931"/>	Descripcion:	<input type="text" value="CHAPAS DE BAÑO"/>
LO QUE ENVIARON:	<input type="text" value="COD02"/>	Cantidad:	<input type="text" value="9"/> Precio: <input type="text" value="4,5"/> U\$

Figura 4.3: Cuadro de Resultado después de realizar el pedido

4.4 Conclusiones:

Como podemos observar el manejo del sistema es sencillo lo único que necesita el usuario que va a manejar el sistema es cumplir con todos los requerimientos tanto de software como de hardware mencionados anteriormente y el cliente debe seguir uno a uno los pasos descritos en este capítulo.

El objetivo de este capítulo es dar una ilustración tanto al usuario como al cliente del manejo de la aplicación demostrando que tiene una interfaz muy amigable y no necesariamente tiene que ser un experto para manejarlo.

CONCLUSIONES

1. En esta monografía se ha descrito el proceso de desarrollo del sistema de envío y recepción de mensajes para realizar el pedido de un artículo determinando dependiendo de las necesidades del cliente Cada uno de los módulos que lo componen se han diseñado y explicado por separado y, posteriormente, se han unido para su funcionamiento.
2. El módulo denominado Servidor se ha creado una aplicación en visual Basic capaz de realizar la configuración, el envío, la recepción de mensajes y la actualización de la base de datos implementada en Access, utilizando un teléfono móvil conectado a un ordenador a través de un puerto USB. Esta comunicación permite que el programa gestione el envío y recepción de mensajes de texto por un puerto determinado.
3. El programa realizado se gestiona el envío y recepción de mensajes mediante la utilización de comandos AT específicos para telefonía móvil. Existen dos formatos posibles para enviar los mensajes: el modo texto y el modo binario PDU, nosotros decidimos utilizar el modo texto ya que esta es la forma mas sencilla de enviar mensajes.
4. La aplicación se encarga de conectarse a la base de datos Access y realizar la actualización correspondiente dependiendo de los pedidos que realice cada cliente.
5. El programa controla el acceso simultáneo de los clientes, permitiendo que varios clientes realicen sus pedidos al mismo tiempo, dependiendo de que operadora nos de acceso al sistema móvil.

6. Respecto al módulo Cliente, este es el que inicia la comunicación con el servidor enviando la información referente a su registro y posteriormente la información referente al artículo y al pedido que necesita en ese momento, con lo cual el servidor irá verificando cada uno de los mensajes emitidos por el cliente

7. Obteniendo como resultado a todo esto no solamente una aplicación que realiza pedidos si no un sistema confiable, seguro y apto para las necesidades del cliente.

GLOSARIO

SMS: Short Message Service

GSM: *Global Systema for Mobile communications* (Sistema Global para las Comunicaciones Móviles), anteriormente conocida como "*Group Special Mobile*" (GSM, Grupo Especial Móvil) es un estándar mundial para teléfonos móviles digitales. El estándar fue creado por la CEPT y posteriormente desarrollado por ETSI 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).

3G: Es una abreviatura para tercera-generación de telefonía móvil. Los servicios asociados con la tercera generación proporcionan la posibilidad para transferir tanto voz y datos.

MAP: Mobile Application Part

ITU: International Telecommunication Union

BASIC: Beginner's All-purpose Symbolic Instruction Code

CLIENTE: Es la persona que realiza el pedido

SERVIDOR: Es la persona encargada en la configuración del sistema

AT: ATtention.

PDU: Protocol Data Unit

SMSC: Short Message Service Center

ISO: International Standards Organization

BIBLIOGRAFIA

BRAZALES, Alfonso. GARCÍA DE JALÓN, Javier. RODRÍQUEZ, Jose Ignacio. *Aprenda Visual Basic 6.0*.
(http://www.wikilearning.com/el_entorno_de_visual_basic_6_0-wkccp-16856-12.htm)

CABRAL., Carlos. *SMS a través de RS232*. 2003

HERNÁNDEZ, Andrés. *Pasarelas de mensajería SMS*.
(<http://es.tldp.org/Presentaciones/200103hispalinux/seco/pdf/pasarelas-sms.pdf>)

RODRÍGUEZ, Alberto. *Envío de Mensajes SMS desde Fox*.
(<http://www.w3.org/TR/REC-html40>).

RODRÍGUEZ, Alberto. *Envío de Mensajes SMS desde Fox*.
(<http://www.fpress.com/revista/Num0304/art.htm>)

SUAREZ, Luis. *Curso de Visual Basic*.
(<http://www.telecable.es/personales/jrubi/index.htm?curso.htm>)

ETSI. European Telecommunications Standards Institute. *Global System for Mobile communications*. 1999

Comunicación móvil. (<http://comunicacionmovil.blogspot.com/>)

Monografías. *Comunicaciones Móviles*.
(<http://www.monografias.com/especiales/comunicamov/>)

MOTOROLA PERSONAL COMMUNICATIONS. *ISU AT Command Reference*. 2000

Ontrak control systems. *Sending Commands using VISUAL BASIC to ADR Interfaces*. (<http://ontrak.net/visual.htm>)

Solotuweb.com. *Visual Basic 6.0*.
(<http://www.solotuweb.com/recursos~t~Visual%20Basic%206.0~id~28.html>)

Wikipedia. *TELEFONIA MOVIL*.
(http://es.wikipedia.org/wiki/Telefon%C3%ADa_m%C3%B3vil)

3Com. *Comandos AT*. <http://www.usr.com/support/3cp3056/3cp3056-spanish-ug/atcoms.htm>

ANEXOS

Anexo 1: Casos de Uso

El modelo de casos de uso describe la funcionalidad propuesta del nuevo sistema. Un caso de uso representa una unidad discreta de interacción entre un usuario (humano o máquina) y el sistema. Un Caso de Uso es una unidad simple de trabajo significativo; por ejemplo, "Validarse en el sistema", "Registrarse en el sistema" y "Crear un pedido" son todos casos de uso. Cada caso de uso tiene una descripción que describe la funcionalidad que se construirá en el sistema propuesto. Un caso de uso puede "incluir" la funcionalidad de otro caso de uso o "extender" a otro caso de uso con su propio comportamiento.

Típicamente los casos de usos están relacionados a 'actores'. Un actor es una entidad humana o máquina que interactúa con el sistema para desarrollar un trabajo significativo.

Anexo 2: Descripción de Modo Texto

El modo texto es mucho más sencillo de desarrollar que el modo PDU, pero hay que tener en cuenta que algunos terminales antiguos no soportan este formato. Para realizar el programa que envíe el mensaje en modo texto se necesitan ejecutar los tres comandos vistos anteriormente. El primer comando que se le pasa al programa es **CMGF** con el valor 1, para indicarle que se enviará un mensaje en modo texto. Cuando el programa responde con OK se puede escribir el siguiente comando. El siguiente paso será indicar el número del centro de mensajería de la compañía telefónica para eso escribimos el comando **CSCA**. Finalmente se escribe el comando referente al envío de mensajes (**CMGS**) especificando el número del destinatario. Al ejecutar el comando se escribe el mensaje que se desea enviar. Una vez escrito hay que presionar (Ctrl Z) para enviarlo.

Anexo 3: Código Fuente

```
***** DECLARACION DE VARIABLES *****  
Option Explicit  
Dim SMESSAGE As String  
Dim AUXUSUARIOS As String  
Dim VARARTICULO As String  
Dim VARUSUARIO As String  
Dim VARPASSWORD As String  
Dim VARCODIGO As String  
Dim DESCRIPCION As String  
Dim CANTIDAD As String
```

```

Dim PRECIO As String
Dim MEMORIA As String
Dim STR2 As String
Dim STR3 As String
Dim STR1 As String
Dim INFOMSG As String
Dim STRAUX2 As String
Dim STRAUX3 As String, AUXFECHA As String, FECHA As String, TEL As String
Dim POS As Integer, TELEFONO As String, AUXTELEFONO As String
Dim CADENA As String, CAMPOS As String
Dim SQL As String
Dim NUMMSG As String
Dim T As String
Dim NUEVO As Boolean
Dim BAND4 As Boolean
Dim BAND7 As Boolean
Dim BAND8 As Boolean
Dim USUARIOEXISTE As Boolean
Dim BANDERA As Boolean
Dim CONTAR As Integer
Dim AUXCANTIDAD As Integer
Dim I As Integer
Dim j As Integer
Dim VARESTADO As Integer
Dim TAMAÑO As Integer
Dim JJ As Long
Dim BAND2 As Integer
Dim TAMUSUARIO As Integer
Dim TAMPASSWORD As Integer
Dim N As Integer
Dim M As Integer
Dim AUXHORA As Date
Dim HORAUSUARIO As Date
Dim HORACONSULTA As Date
Dim REGISTRO As Recordset
Dim REGISTRO1 As Recordset
Dim REGISTRO2 As Recordset

Private Sub cboComm_Change()
On Error GoTo ERRPUERTO
    MSComm1.CommPort = cboComm.Text
    If Err Then

        Else
            SetStatus "CAMBIANDO PARA EL PUERTO COM " & cboComm.Text, True
        End If

ERRPUERTO:
End Sub

Private Sub cboComm_Click()
    Call cboComm_Change
End Sub

Private Sub cmdClear_Click()
    txtOutput.Text = ""
End Sub

Private Sub ListaPortasCom()
    Dim I As Integer
    cboComm.Clear
    SetStatus "VERIFICANDO LOS PUERTOS DISPONIBLES...", True
    For I = 1 To 6
        If DetectaPortaCOM(I) Then
            cboComm.AddItem I
            SetStatus "COM" & I & " ENCONTRADO", False
        End If
    Next
    cboComm.ListIndex = 0
End Sub

Private Sub cmdConfiguracion_Click()
On Error GoTo ERRORES
    MSComm1.Output = "AT+CMGF=1" & Chr$(13) & Chr$(10)
    MSComm1.Output = "AT+CNMI =3,1,0,0,0" & ";" & Chr$(13) & Chr$(10)
    MSComm1.Output = "AT+CPMS=?" & ";" & Chr$(13) & Chr$(10)
    MSComm1.Output = "AT+CPMS?" & ";" & Chr$(13) & Chr$(10)

```

```

ERRORES:
End Sub

Private Sub cmdGetInfo_Click()
On Error GoTo ERRPORT
Dim sLastString As String
Dim sOutput As String
Dim sBuffer As String
Dim I As Long
With MSComml
'***** ABRIMOS EL PUERTO *****
If cmdCOMStatus.Caption = "ABRIR PUERTA COM" Then Call cmdCOMStatus_Click
'***** ENVIA UN COMANDO *****
sOutput = cboCommand.Text
setStatus "Enviando " & sOutput & " comando...", True
.Output = sOutput & Chr$(13)
End With
Exit Sub
ERRPORT:
End Sub

Private Sub SaveSettings()
Dim SSETTINGS As String
SSETTINGS = cboBaubRate.Text
SSETTINGS = SSETTINGS & "," & cboBaubRate.Text
SSETTINGS = SSETTINGS & "," & cboParity.Text
SSETTINGS = SSETTINGS & "," & cboDataBits.Text
SSETTINGS = SSETTINGS & "," & cboStopBits.Text
MSComml.Settings = SSETTINGS
End Sub

Private Sub cmdCOMStatus_Click()
On Error GoTo ERRSTATUS
'***** CERRAMOS EL PUERTO *****
If cmdCOMStatus.Caption = "CERRAR PUERTO COM" Then
setStatus "FECHANDO PUERTO COM " & MSComml.CommPort, True
MSComml.PortOpen = False
cmdCOMStatus.Caption = "ABRE PUERTO COM"
Else
'***** ABRIMOS EL PUERTO *****
setStatus "ABRIENDO PUERTO COM " & MSComml.CommPort, True
MSComml.PortOpen = True
cmdCOMStatus.Caption = "CERRAR PUERTO COM"
End If
Exit Sub
ERRSTATUS:
End Sub

Private Sub Form_Load()
Dim I As Integer
Dim SSETTINGS As String
'***** INICIALIZAMOS LAS VARIABLES *****
BAND8 = False
BAND7 = False
BANDERA = False
BAND2 = 0
TAMAÑO = 0

'***** CONFIGURACION DE LA PROPIEDAD SETTINGS *****
SSETTINGS = MSComml.Settings
cboBaubRate.Text = Left$(SSETTINGS, 4) ' INDICAN LA VELOCIDAD DEL PUERTO COM
cboParity.Text = Mid$(SSETTINGS, 6, 1) ' LA PARIDAD EN LA VALIDACION DE DATOS
cboDataBits.Text = Mid$(SSETTINGS, 8, 1) ' INDICAN LOS BITS DE INFORMACION
cboStopBits.Text = Mid$(SSETTINGS, 10, 1) ' INDICAL LOS BITS DE PARADA
cboCommand.ListIndex = 0

'***** CONFIGURACION DEL PUERTO *****
If MSComml.PortOpen = False Then 'VERIFICA SI EL PUERTO ESTÁ CERRADO
MSComml.Handshaking = comNone 'IGNORAR CUALQUIER PROTOCOLO A NIVEL HARDWARE.
MSComml.InBufferSize = 1024 'CANTIDAD DE MEMORIA DESIGNADA A LOS BUFFERS
MSComml.RThreshold = 1 'NUMEROS DE CARACTERES DENTRO EN EL BUFFER
MSComml.EOFEnable = False 'CARÁCTER DE FIN DE ARCHIVO ENCONTRADO EN LA
ENTRADA DE DADOS.
MSComml.OutBufferSize = 1024 'CANTIDAD DE MEMORIA DESIGNADA A LOS BUFFERS.
MSComml.SThreshold = 1 'CANTIDAD DE CARACTERES DEBEN TRANSMITIRSE

```

```

        MSComm1.NullDiscard = False      'HACE QUE EL CARACTER NULO SEA UN CARÁCTER MAS
        MSComm1.RTSEnable = True         'INDICA AL MODEM QUE DESEAMOS ENVIAR DATOS
        MSComm1.DTREnable = True         'MODEM EL CUAL VAMOS A RECIBIR LOS DATOS
        MSComm1.Settings = "9600,N,8,1" 'VELOCIDAD DE COMUNICACIONES Y PARÁMETROS
        MSComm1.InputLen = 0              'LEER TODOS LOS CARACTERES
        MSComm1.InputMode = comInputModeBinary 'RECIBA TIPOS DE CARACTERES NO
IMPRIMIBLES
    End If

    ' ***** OBTENER PUERTOS COM *****
    DoEvents
    ListaPortasCom
End Sub
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As
Single)
    imgSalirl.Visible = True
    imgSalir.Visible = True
End Sub

Private Sub imgSalir_Click()
Dim MENSAJE As String
On Error GoTo ERRORSALIR
' ***** SALIR DEL SISTEMA *****
MENSAJE = MsgBox("ESTA SEGURO QUE DESEA SALIR DEL SISTEMA..?", vbYesNo, "S A L I
R")
If MENSAJE = vbYes Then
    Call ABRIRBASEDATOS
    SQL = "select * from ESTADO"
    Set REGISTRO2 = DATOS.OpenRecordset(SQL)
    While REGISTRO2.EOF = False
        REGISTRO2.Delete
        REGISTRO2.MoveNext
    Wend
    End
End If

ERRORSALIR:

End Sub
Private Sub imgSalirl_MouseMove(Button As Integer, Shift As Integer, X As Single, Y
As Single)
    imgSalirl.Visible = False
End Sub

Private Sub MSComm1_OnComm()
On Error Resume Next
Select Case MSComm1.CommEvent
    ' ***** SE ACTIVA CUANDO TIENE UNA ENTRADA EL MODEM DEL CELULAR *****
    Case comEvReceive
        SMESSAGE = StrConv(MSComm1.Input, vbUnicode)
        INFOMSG = SMESSAGE
        BAND8 = False
    End Select
    SetStatus (SMESSAGE), False
    CortaCadena
End Sub
Private Sub ReceiveResponse(sResponse As String)
' ***** MOSTRAMOS LO QUE RECIBE EL MODEM *****
txtOutput = txtOutput & "RESPUESTA RECIBIDA" & vbCrLf & sResponse & vbCrLf & "FIN
DE RESPUESTA." & vbCrLf
txtOutput.SelStart = Len(txtOutput)
txtOutput.Refresh
End Sub
Private Sub SetStatus(sStatus As String, bOperation As Boolean)
    txtOutput = txtOutput & IIf(bOperation, "--> ", "") & sStatus & vbCrLf
    txtOutput.SelStart = Len(txtOutput)
    txtOutput.Refresh
End Sub

Private Sub cmdGetAllPorts_Click()
Dim NumPorts As Long
Dim I As Integer
' ***** MOSTRAMOS LOS PUERTOS DISPONIBLES *****
SetStatus "OBTENIENDO TODOS LOS PUERTOS DISPONIBLES...", True
For I = 1 To 16
    If DetectaPortaCOM(I) <> 0 Then
        txtOutput.Text = "COM" & I
    End If
End For

```

```

        End If
    Next
End Sub

' ***** FUNCION PARA CONSULTAR LOS ARTICULOS DE LA BASE DE DATOS *****
Sub Consultar()
Call ABRIRBASEDATOS
DESCRIPCION = ""
CANTIDAD = ""
PRECIO = ""
SQL = "select * from ARTICULOS WHERE CODIGO = ' " & UCase(txtEnvia.Text) & "' "
Set REGISTRO = DATOS.OpenRecordset(SQL)
If REGISTRO.RecordCount = 1 Then
    BANDERA = False
    DESCRIPCION = REGISTRO!DESCRIPCION
    CANTIDAD = REGISTRO!CANTIDAD
    PRECIO = REGISTRO!PRECIO
    CAMPOS = "desc= " & REGISTRO!DESCRIPCION & Chr(13)
    CAMPOS = CAMPOS & " cant= " & REGISTRO!CANTIDAD & Chr(13)
    CAMPOS = CAMPOS & " prec= " & REGISTRO!PRECIO & Chr(13)
    CAMPOS = CAMPOS & " ENVIE LA CANTIDAD DE ARTICULOS QUE DESEA PEDIR: " &
Chr(13)
    EnviarSMS
Else
    BANDERA = True
    EnviarSMS
End If
End Sub

Sub EnviarSMS()
MSComm1.Output = "AT+CMGF=1" & Chr$(13) & Chr(10)
Sleep (300)
DoEvents
If InStr(TELEFONO, "+593") Then
    TEL = "0" & Mid(TELEFONO, 5, Len(TELEFONO))
Else
    TEL = TELEFONO
End If
MSComm1.Output = "AT+CMGS=" & Chr(34) & TEL & Chr(34) & Chr$(13) & Chr(10)
Sleep (300)
DoEvents
If BANDERA = False Then
    MSComm1.Output = CAMPOS & Chr(26) & Chr(13)
    Sleep (300)
    DoEvents
    txtDescripcion.Text = DESCRIPCION
    txtCantidad.Text = CANTIDAD
    txtPrecio.Text = PRECIO
    Frame2.Visible = True
Else
    MSComm1.Output = "ESTE ARTICULO NO EXISTE, ENVIE NUEVAMENTE.. " & Chr(26) &
Chr(13)
    Sleep (300)
End If
DoEvents
End Sub

Sub CortaCadena()
'PARA LA LECTURA DEL LUGAR DE ALMACENAMIENTO Y POSICION DEL MENSAJE RECIEN LLEGADO
If Mid(INFOMSG, 3, 5) = "+CMTI" Then ' VERIFICA LA LLEGADA DE UN NUEVO MENSAJE
    INFOMSG = Trim(SMESSAGE) ' ELIMINA LOS ESPACIOS EN BLANCO
    MEMORIA = Trim(Mid(INFOMSG, 11, 2)) ' MUESTRA EL TIPO DE MEMORIA
    JJ = Len(Mid(INFOMSG, 15, 20)) - 2 ' LONGITUD DE LOS CARACTERES
    NUMMSG = Trim(Mid(INFOMSG, 15, JJ)) 'SELECCION DE LOS CARACTERES
    Sleep (1000) ' ESPERA EL TIEMPO DEFINIDO
    DoEvents
    MSComm1.Output = "AT+CMGR =" & NUMMSG & ";" & Chr$(13) & Chr$(10)
    Sleep (1000)
    On Error Resume Next
    MSComm1_OnComm
End If
If Not NUEVO Then
    If Mid(INFOMSG, 3, 5) = "+CMGR" Then
        CADENA = CADENA & INFOMSG
        NUEVO = True
    End If
Else

```

```

CADENA = CADENA & INFOMSG
End If
' MOSTRAMOS LOS ULTIMOS CARACTERES DEL MENSAJE
If Right(CADENA, 6) = vbCrLf + "OK" + vbCrLf Then
    ' ***** MOSTRAMOS LO QUE CONTIENE EL MENSAJE *****
        STR2 = vbCrLf + vbCrLf
        STR3 = vbCrLf + "OK" + vbCrLf
        STR1 = Mid(CADENA, InStr(CADENA, STR2) + 3)
        STRAUX2 = Mid(CADENA, InStr(CADENA, STR2) + 3, InStr(STR1, STR3) - 1)
        STRAUX3 = Mid(STRAUX2, InStr(STRAUX2, vbCrLf) + 2, Len(STRAUX2) -
(InStr(STRAUX2, vbCrLf) + 1)) 'mensaje recibido
    ' ***** SACAMOS EL NUMERO DE TELEFONO DEL QUE NOS ENVIA EL MENSAJE *****
        POS = InStr(CADENA, ",")
        AUXTELEFONO = Mid(CADENA, POS + 3, Len(CADENA))
        TELEFONO = Mid(AUXTELEFONO, 1, InStr(AUXTELEFONO, Chr(34)) - 1)
    ' ***** OBTENEMOS LA FECHA EN LA QUE NOS ENVIO EL MENSAJE *****
        AUXFECHA = Mid(AUXTELEFONO, InStr(AUXTELEFONO, Chr(34)) + 4,
Len(AUXTELEFONO))
        FECHA = Mid(AUXFECHA, 1, InStr(AUXFECHA, Chr(34)) - 1)
    ' ***** COLOCAMOS EL ESTADO EN EL QUE SE ENCUENTRA *****
        ESTADO
        CADENA = ""
        NUEVO = False
        Exit Sub
End If
txtEnvia.Text = STRAUX3
txtTelefono.Text = TEL
End Sub

' ***** FUNCION PARA ACTUALIZAR LA BASE DE DATOS *****
Sub ACTUALIZACION()
Call ABRIRBASEDATOS
SQL1 = "select * from ESTADO WHERE NUMERO = '" & TELEFONO & "'"
Set REGISTRO = DATOS.OpenRecordset(SQL1)
If REGISTRO.RecordCount = 1 Then
    VARCODIGO = REGISTRO!ARTICULO
End If
txtEnvia = VARCODIGO
SQL = "select * from ARTICULOS WHERE CODIGO = '" & txtEnvia & "'"
Set REGISTRO1 = DATOS.OpenRecordset(SQL)
If REGISTRO1.RecordCount > 0 Then
    If UCase(txtEnvia.Text) = REGISTRO1!codigo And AUXCANTIDAD <= REGISTRO1!CANTIDAD
Then
        txtCantidad.Text = REGISTRO1!CANTIDAD - AUXCANTIDAD
        SQL1 = "update ARTICULOS set CODIGO = '" & UCase(txtEnvia.Text) & "',
CANTIDAD=" & txtCantidad.Text & "'"
        DATOS.Execute SQL1
        MsgBox "LA CANTIDAD EXISTENTE FUE ACTUALIZADA SATISFACTORIAMENTE.."
    Else
        EXCEDECANTIDAD
        MsgBox "LA CANTIDAD DE SU PEDIDO EXCEDE DE LA CANTIDAD EXISTENTE.."
        Exit Sub
    End If
End If
End Sub

Sub EXCEDECANTIDAD()
MSComm1.Output = "AT+CMGF=1" & Chr$(13) & Chr(10)
Sleep (300)
DoEvents
If InStr(TELEFONO, "+593") Then
    TEL = "0" & Mid(TELEFONO, 5, Len(TELEFONO))
Else
    TEL = TELEFONO
End If
MSComm1.Output = "AT+CMGS=" & Chr(34) & TEL & Chr(34) & Chr$(13) & Chr(10)
Sleep (300)
DoEvents
MSComm1.Output = "LA CANTIDAD DE SU PEDIDO EXCEDE DE LA CANTIDAD EXISTENTE.." &
Chr(26) & Chr(13)
Sleep (300)
DoEvents
End Sub

Sub usuarios()
Dim USUARIO As String
Dim BAND3 As Boolean

```

```

VARUSUARIO = ""
VARPASSWORD = ""
BAND3 = False
TAMAÑO = Len(AUXUSUARIOS)
BAND8 = True
For I = 1 To TAMAÑO
    USUARIO = Mid(AUXUSUARIOS, I, 1)
    If USUARIO <> " " And BAND3 = False Then
        VARUSUARIO = VARUSUARIO & USUARIO
    Else
        If USUARIO <> " " Then
            VARPASSWORD = VARPASSWORD & USUARIO
        Else
            BAND3 = True
        End If
    End If
Next
Call ABRIRBASEDATOS
SQL = "select * from USUARIOS WHERE USUARIO = ' " & VARUSUARIO & "' AND PASSWORD = ' " & VARPASSWORD & "' "
Set REGISTRO = DATOS.OpenRecordset(SQL)
If REGISTRO.RecordCount > 0 Then
    MsgBox "USUARIO CORRECTO", vbInformation, "R E S U L T A D O .."
    BAND4 = False
    ENVIARMENSAJEUSUARIO
Else
    TAMUSUARIO = Len(VARUSUARIO)
    TAMPASSWORD = Len(VARPASSWORD)
    If TAMUSUARIO < 3 And TAMUSUARIO > 0 Then
        MsgBox "EL USUARIO DEBE TENER POR LO MENOS 3 CARACTERES COMO MINIMO..",
vbInformation, "R E S U L T A D O .."
        BAND2 = 1
        ENVIARUSUARIOS
    Else
        If TAMPASSWORD < 5 And TAMPASSWORD > 0 Then
            MsgBox "EL PASSWORD DEBE TENER POR LO MENOS 5 CARACTERES COMO MINIMO..",
vbInformation, "R E S U L T A D O .."
            BAND2 = 2
            ENVIARUSUARIOS
        Else
            BAND4 = True
            MsgBox "USUARIO Y PASSWORD NO REGISTRADOS.. ", vbInformation, "R E S
U L T A D O .."
            ENVIARMENSAJEUSUARIO
        End If
    End If
End If
End Sub

Sub ENVIARUSUARIOS()
    MSComm1.Output = "AT+CMGF=1" & Chr$(13) & Chr(10)
    Sleep (300)
    DoEvents
    If InStr(TELEFONO, "+593") Then
        TEL = "0" & Mid(TELEFONO, 5, Len(TELEFONO))
    Else
        TEL = TELEFONO
    End If

    MSComm1.Output = "AT+CMGS=" & Chr(34) & TEL & Chr(34) & Chr$(13) & Chr(10)
    Sleep (300)
    DoEvents
    If BAND2 = 1 Then
        MSComm1.Output = "EL NOMBRE DE USUARIO DEBE TENER POR LO MINIMO 3
CARACTERES.. INGRESE NUEVAMENTE.." & Chr(26) & Chr(13)
        Sleep (300)
    End If
    If BAND2 = 2 Then
        MSComm1.Output = "EL PASSWORD DEBE TENER POR LO MENOS 5 CARACTERES COMO
MINIMO.. INGRESE NUEVAMENTE.." & Chr(26) & Chr(13)
        Sleep (300)
    End If
    DoEvents
End Sub

Sub ENVIARMENSAJEUSUARIO()
    MSComm1.Output = "AT+CMGF=1" & Chr$(13) & Chr(10)

```

```

Sleep (300)
DoEvents
If InStr(TELEFONO, "+593") Then
    TEL = "0" & Mid(TELEFONO, 5, Len(TELEFONO))
Else
    TEL = TELEFONO
End If
MSComm1.Output = "AT+CMGS=" & Chr(34) & TEL & Chr(34) & Chr$(13) & Chr(10)
Sleep (300)
DoEvents
If BAND4 = False Then
    MSComm1.Output = "USUARIO Y PASSWORD CORRECTOS.." & Chr(26) & Chr(13)
    Sleep (300)
Else
    MSComm1.Output = "USUARIO Y PASSWORD NO REGISTRADOS DESEA REGISTRARSE
SI/NO..." & Chr(26) & Chr(13)
    Sleep (300)
End If
DoEvents
End Sub

' ***** VERIFICAMOS EL ESTADO EN EL QUE SE ENCUENTRA *****
Sub ESTADO()
Call ABRIRBASEDATOS
Dim VALOR As String
SQL = "select * from ESTADO WHERE NUMERO = '" & TELEFONO & "'"
Set REGISTRO2 = DATOS.OpenRecordset(SQL)
AUXHORA = TimeSerial(Hour(Time), Minute(Time), Second(Time))
If UCase(STRAUX3) = "SI" Then
    REGISTROUSUARIO
End If
If UCase(STRAUX3) = "NO" Then
    REGISTRO2.Delete
    Exit Sub
End If
If REGISTRO2.RecordCount = 1 Then
    If REGISTRO2!ESTADO = 2 And BAND8 = False And AUXHORA <= HORAUSUARIO Then
        HORACONSULTA = TimeSerial(Hour(Time), Minute(Time), Second(Time) + 60)
        txtEnvia.Text = STRAUX3
        Sleep (1000)
        Consultar
        BAND8 = True
        If BANDERA = False Then
            VARESTADO = 3
            SQL1 = "update ESTADO set NUMERO ='" & TELEFONO & "', ESTADO='" &
VARESTADO & "', ARTICULO='" & txtEnvia.Text & "', HORA='" & AUXHORA & "'"
            DATOS.Execute SQL1
        End If
    Else
        If REGISTRO2!ESTADO = 2 And BAND8 = False Then
            Sleep (1000)
            DEMORA
            REGISTRO2.Delete
        End If
    End If
    If REGISTRO2!ESTADO = 3 And BAND8 = False And AUXHORA <= HORACONSULTA Then
        AUXCANTIDAD = STRAUX3
        Sleep (1000)
        ACTUALIZACION
        REGISTRO2.Delete
    Else
        If REGISTRO2!ESTADO = 3 And BAND8 = False Then
            Sleep (1000)
            DEMORA
            REGISTRO2.Delete
        End If
    End If
Else
    HORAUSUARIO = TimeSerial(Hour(Time), Minute(Time), Second(Time) + 60)
    VARESTADO = 2
    VARARTICULO = ""
    SQL1 = "Insert into ESTADO values ('" & TELEFONO & "', '" & VARESTADO & "', '" &
VARARTICULO & "', '" & AUXHORA & "'" )"
    DATOS.Execute SQL1
    AUXUSUARIOS = STRAUX3
    Sleep (1000)
    usuarios

```

```

End If
End Sub

Sub DEMORA()
MSComm1.Output = "AT+CMGF=1" & Chr$(13) & Chr(10)
Sleep (300)
DoEvents
If InStr(TELEFONO, "+593") Then
    TEL = "0" & Mid(TELEFONO, 5, Len(TELEFONO))
Else
    TEL = TELEFONO
End If
MSComm1.Output = "AT+CMGS=" & Chr(34) & TEL & Chr(34) & Chr$(13) & Chr(10)
Sleep (300)
DoEvents
MSComm1.Output = "EL TIEMPO ESTIMADO A EXCEDIDO ENVIE NUEVAMENTE SU USUARIO .." &
Chr(26) & Chr(13)
Sleep (300)
DoEvents
End Sub

Sub REGISTROUSUARIO()
Call ABRIRBASEDATOS
SQL = "select * from USUARIOS WHERE USUARIO = '" & VARUSUARIO & "'"
Set REGISTRO = DATOS.OpenRecordset(SQL)
If REGISTRO.RecordCount = 1 Then
    MsgBox "NOMBRE DE USUARIO YA REGISTRADO, INGRESE UN NUEVO NOMBRE DE USUARIO.."
    , vbInformation, "R E S U L T A D O .."
    USUARIOEXISTE = True
    SQL1 = "select * from ESTADO WHERE NUMERO ='" & TELEFONO & "'"
    Set REGISTRO2 = DATOS.OpenRecordset(SQL1)
    REGISTRO2.Delete
Else
    SQL = "Insert into USUARIOS values ('" & UCase(VARUSUARIO) & "', '" &
UCase(VARPASSWORD) & "'"
    DATOS.Execute SQL
    AUXHORA = TimeSerial(Hour(Time), Minute(Time), Second(Time))
    HORAAUSUARIO = TimeSerial(Hour(Time), Minute(Time), Second(Time) + 60)
    VARESTADO = 2
    SQL1 = "Insert into ESTADO values ('" & TELEFONO & "', '" & VARESTADO & "',
'" & VARARTICULO & "', '" & AUXHORA & "'" )"
    DATOS.Execute SQL1
    USUARIOEXISTE = False
End If
REGISTRADO
End Sub

Sub REGISTRADO()
MSComm1.Output = "AT+CMGF=1" & Chr$(13) & Chr(10)
Sleep (300)
DoEvents
If InStr(TELEFONO, "+593") Then
    TEL = "0" & Mid(TELEFONO, 5, Len(TELEFONO))
Else
    TEL = TELEFONO
End If
MSComm1.Output = "AT+CMGS=" & Chr(34) & TEL & Chr(34) & Chr$(13) & Chr(10)
Sleep (300)
DoEvents
If USUARIOEXISTE = False Then
    MSComm1.Output = "USUARIO REGISTRADO INGRESE EL CODIGO DE ARTICULO" & Chr(26)
& Chr(13)
    Sleep (300)
End If
If USUARIOEXISTE = True Then
    MSComm1.Output = "ESTE USUARIO YA EXISTE, INGRESE UN NUEVO NOMBRE DE
USUARIO.." & Chr(26) & Chr(13)
    Sleep (300)
End If
DoEvents
End Sub

```



Cuenca, 24 de julio de 2007

Economista

Luis Mario Cabrera

DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACION
DE LA UNIVERSIDAD DEL AZUAY

Ciudad

De nuestras consideraciones:

Nosotros Sarmiento Molina Nora Patricia y Vanegas Arizaga Gabriela Alexandra, egresadas de la Escuela de Ingenieria de Sistemas, nos dirigimos a usted y por su digno intermedio al honorable consejo de Facultad, para solicitar la aprobacion del Diseño de Monografia con el tema "DISEÑO DE UN SISTEMA PARA MANEJO DE UNA BASE DE DATOS EN ACCESS A TRAVÉS DE MENSAJERIA SMS IMPLEMETADO EN VISUAL BASIC", requisito previo a la obtención del Titulo de Ingenieria de Sistemas para los estudiantes que hemos realizado el curso de graduación en la Universidad de Buenos Aires, así como la designación de(l) (los) Directores.

El diseño de la Monografía cuenta con el informe favorable del Director de Escuela Ing. Paúl Ochoa y del Ing. Fabián Carvajal, profesor de la facultad.

Por la favorable acogida que brinde a la presente, anticipamos nuestros agradecimientos.

Atentamente,


Código: 29801


Código: 29235

DR. ROMEL MACHADO CLAVIJO SECRETARIO DE LA FACULTAD DE
CIENCIAS DE LA ADMINISTRACION DE LA UNIVERSIDAD DEL AZUAY,

CERTIFICO Que, El H. Consejo de Facultad en sesión del 24. de Julio 2007 conoció
y aprobó la Monografía presentada por las señoritas Nora Patricia Sarmiento
Molina y Gabriela Alexandra Vanegas Arizaga como requisito previo a la obtención
del Grado de Ingeniero de Sistemas denominada DISEÑO DE UN SISTEMA PARA
MANEJO DE UNA BASE DE DATOS DE ACCESS A TRAVES DE MENSAJERIA
SMS IMPLEMENTO EN VISUAL BASIC la misma que se aprueba, y se designa
como Director de dicho trabajo al Ingeniero Fabian Carvajal y de conformidad con las
disposiciones reglamentarias el denunciante deberán presentar su trabajo de
Monografía en un plazo mínimo de treinta días y un máximo de noventa días es
decir hasta el 24 de Octubre del 2007.

Cuenca 30 de Junio del 2007.






Cuenca, 24 de julio de 2007

Economista

Luis Mario Cabrera

DECANO DE LA FACULTAD DE CIENCIAS DE LA ADMINISTRACION
DE LA UNIVERSIDAD DEL AZUAY

Ciudad

Señor Decano

Quienes suscribimos comunicamos a usted que hemos procedido a revisar el diseño de la monografía presentado por Nora Patricia Sarmiento Molina y Gabriela Alexandra Vanegas Arizaga, estudiantes de la Escuela de Ing. de Ingeniería de Sistemas con el tema "DISEÑO DE UN SISTEMA PARA MANEJO DE UNA BASE DE DATOS EN ACCESS A TRAVES DE MENSAJERIA SMS IMPLEMETADO EN VISUAL BASIC" , como requisito previo a la obtención del Título de Ingeniería de Sistemas, sobre el cual le presentamos el siguiente informe:

El diseño cumple con los requisitos metodológicos básicos exigidos por la facultad, en cuanto a la introducción, resumen del proyecto, impactos, objetivos, marco teórico, contenidos, procedimientos, recursos, cronograma y bibliografía necesaria para el desarrollo de la monografía.

Por las consideraciones anotadas, se emite un informe favorable y salvo su mejor criterio, se recomienda su aprobación.

Atentamente,

Ing. Paul Ochoa

Director de la Escuela

Ing. Fabian Carvajal

Director de Monografía

DISEÑO DE MONOGRAFIA

1 Título del Proyecto

“DISEÑO DE UN SISTEMA PARA MANEJO DE UNA BASE DE DATOS EN ACCESS A TRAVÉS DE MENSAJERIA SMS IMPLEMETADO EN VISUAL BASIC”

2 Selección y Delimitación del Tema

Contenido: El sistema abarcará aspectos importantes los cuales detallaremos a continuación:

- Estudio sobre la mensajería SMS
- Conexión a la base de datos
- Registro de usuarios a través de un *usuario* y un *password*
- Actualización de la base de datos

El sistema utilizará una base de datos Access y la información que dispondremos para la aplicación es de una ferretería

Espacio: El presente proyecto lo realizaré en la ciudad de Cuenca

Tiempo: El proyecto abarcara un plazo no mayor a 8 semanas.

3 Descripción del Objetivo de Estudio

Teniendo en consideración que la telefonía móvil y a su vez los mensajes SMS han tenido un gran avance en la actualidad desarrollaremos un sistema de envío de mensajes cortos a través de un teléfono celular para obtener información de un producto de ferretería con lo cual otorgamos al usuario la facilidad y comodidad de consultar e inclusive de realizar pedidos en cualquier momento y de manera inmediata.

4 Resumen del proyecto

El proyecto pretende realizar la implementación de un sistema de envío y recepción de mensajes SMS para acceder a la base de datos y obtener la información de un producto de ferretería específico en la estación celular del usuario.

En el sistema además incluirá un sistema de seguridad de acceso a la base de datos constituida por un nombre de usuario y un password, con lo cual el usuario podrá acceder a la información almacenada en la base de datos así como la facilidad de realizar pedidos y actualizar la base de datos.

5 Introducción

Con el paso del tiempo los ordenadores han ido haciéndose más y más pequeños desde aquellos gigantes de acero y válvulas hasta los ordenadores personales de la actualidad, y estamos asistiendo ahora a la aparición de pequeños dispositivos del tamaño de la palma de la mano creciendo día a día en un afán de alcanzar la posibilidad de comparación con los ordenadores de sobremesa.

A la vez, las comunicaciones se han ido haciendo cada vez más importantes, por no decir imprescindibles. Muchas personas, ya no conciben un ordenador sin conexión a otros. El ordenador aislado poco a poco va convirtiéndose en una máquina del pasado.

Por otra parte, en los últimos años, el acercamiento de la telefonía móvil a todo tipo de gente mediante precios más o menos asequibles ha propiciado que nuestra conciencia de "estar conectados" aumente, y si esta es permanente, on-line, mucho mejor.

Uno de los servicios ofertados por estas operadoras en los teléfonos celulares de tecnología GSM es el envío y recepción de mensajes cortos, los conocidos SMS.

El éxito de SMS, parece provenir de la sencillez y facilidad de manejo, por un lado, y de que "hay alguien al otro lado" con quien realizar el acto de la comunicación. Estos dos

factores han provocado dicho éxito, aún teniendo en contra el precio del servicio en algunos casos y las limitadas características de esta comunicación.

Las posibilidades de comunicación mediante "mensajes cortos GSM" (SMS) son muchas y muy variadas, pero siempre limitadas por las características de estos mensajes, 160 caracteres, muy baja velocidad (en comparación con las líneas telefónicas convencionales), duración limitada (24 ó 48 horas normalmente, si no se entregan antes son cancelados), no es un servicio garantizado (el mensaje suele llegar pero no hay garantía de ello, ni que lleguen en el orden en que se han enviado) y posibilidad de comunicación sólo entre teléfonos celulares GSM entre los que haya "visibilidad" (que los operadores de los dos teléfonos, emisor y receptor, tengan convenio de intercambio de mensajes). Existen muchas especificaciones de formato de mensaje para los servicios prestados a través de SMS que les dotan de gran potencia y complejidad. Pero es en el uso básico con un sistema de enlace sencillo donde se están obteniendo los mejores resultados, tanto de cantidad de mensajes enviados como de servicios que se están utilizando.

En todo caso, estas posibilidades resultan suficientes aprovechadas de forma adecuada, y una de esas formas es tener uno de los lados de la comunicación gobernado por un servicio automático que se encargue de responder a las peticiones recibidas desde múltiples teléfonos celulares.

6 Situación Actual y Futura

SITUACION ACTUAL

En la actualidad, el uso de los mensajes cortos de los teléfonos celulares (SMS) está siendo utilizado de forma muy extensa. La facilidad de acceso al servicio, las posibilidades en la comunicación con otras personas y la sencillez de su concepción están haciendo que este medio de comunicación continúe creciendo tras muchos años de encontrarse disponibles para el usuario final.

Pero los SMS no sólo pueden utilizarse en la comunicación entre personas. Puede obtenerse un gran servicio de su utilización para comunicar con agentes automáticos que respondan a mensajes enviados automática o manualmente.

SITUACION FUTURA

Luego de haber concluido el análisis y estudio de este trabajo investigativo aportaremos con conocimientos nuevos sobre la conexión de base de datos a través de mensajes SMS.

Dando a conocer las facilidades, beneficios, comodidades y rapidez para obtener información de una base de datos mediante el uso de mensajes SMS, logrando con esto dar una visión mas general de lo importante y lo útil que podría llegar a ser este servicio.

7 Justificación e Impacto

JUSTIFICACION

Uno de los motivos que se llevó a realizar este tema “DISEÑO DE UN SISTEMA PARA MANEJO DE MENSAJES SMS CON UNA BASE DE DATOS EN ACCESS A TRAVÉS DE VISUAL BASIC” es por que el envío de mensajes cortos a través del teléfono móvil se ha convertido en una herramienta esencial e indispensable de comunicación, basada fundamentalmente en el criterio de la inmediatez. Las personas necesitan comunicar de manera instantánea aquello que en ese mismo momento es considerado como importante.

Por lo tanto este sistema puede ayudar al usuario a obtener la información del producto que necesite de una manera eficiente y rápida para el lugar en el que se encuentre en ese instante que necesite.

IMPACTO TECNOLÓGICO

El uso constante de la telefonía móvil en el mundo entero va siendo cada vez más una parte importante del avance tecnológico. Es así que poco a poco las empresas y personas en general van haciendo de este tipo de servicio la mejor forma de comunicarse y mantenerse informados.

Motivos por los cuales el saber aprovechar todos los usos y servicios que podemos obtener con esta tecnología como el obtener información de una base de datos a través de lo cotidiano que es por medio de un mensaje SMS hace que esta tecnología tenga un impacto en el mundo moderno.

8 Objetivos

Objetivo General

Diseñar e implementar un sistema para manejo de una base de datos de una ferretería en Access a través de mensajería sms.

Objetivos Secundarios

- Diseñar e implementar la etapa de recepción de mensajes sms.
- Diseñar e implementar la etapa de envío de mensajes sms
- Diseñar e implementar la conexión a bases de datos Access a VB
- Diseñar e implementar la etapa de seguridad de acceso al sistema
- Diseñar e implementar la etapa de actualización de la base de datos.

Objetivos Personales

- Aumentar nuestros conocimientos en la implementación de servicios de telefonía móvil SMS implementando con una conexión a una base de datos a través de un lenguaje de programación Visual Basic.

9 Marco Teórico

Telefonía móvil mediante el uso de SMS

Lenguaje de programación Visual Basic

Gestor de Base de datos Access

10 Contenidos

1) Tutorial para el Sistema:

- i. Estudio de los servicios de telefonía móvil SMS implementando con la conexión a la base de datos.
- ii. Breve descripción del software que utilizaremos para la implementación.
- iii. Especificación detallada del funcionamiento del sistema
- iv. Imágenes y gráficos para facilidad de entendimiento del usuario.

2) Software desarrollado

- i. Recepción de mensajes
- ii. Envío de mensajes
- iii. Conexión a la base de datos
- iv. Seguridad de acceso al sistema a través de usuario y password
- v. Pedidos a través de mensajería móvil SMS
- vi. Actualización de la base de datos

11. Procedimientos Metodológicos

- Estudio de mercado para obtener la base de datos la cual utilizaremos para la conexión siguiente.
- Investigación de los sistemas SMS con conexión a la base de datos
- Desarrollo del software
- Descripción detallada del Sistema para fácil entendimiento del usuario.
- Pruebas y corrección de errores (retroalimentación)

12. Recursos Humanos y Técnicos

Recursos Humanos:

El proyecto se lo realizará con la participación de:

Investigador: Patricia Sarmiento Molina

Desarrollador: Gabriela Vanegas Arízaga

Asesores del Proyecto: Ing. Fabián Carvajal

Recursos Materiales:

Hardware: El hardware a utilizar es el siguiente:

Computadora personal Celaron 1.4

Teléfono celular Nokia modelo: 3220

Software

Sistema Operativo Windows XP

Visual Basic Versión 6.0

Gestor de base de datos Access

13. Cronograma de actividades

Para una correcta realización de la presente monografía se seguirá el orden del siguiente cronograma. El tiempo viene dado en semanas.

ACTIVIDADES	1	2	3	4	5	6	7	8
1. Estudio de mercado para obtener la base de datos la cual utilizaremos para la conexión	■							
2. Investigación de los sistemas SMS con conexión a la base de datos	■							
3. Desarrollo del software		■	■	■	■			
4 Descripción detallada del Sistema para fácil entendimiento del usuario.						■	■	
5 Pruebas y corrección de errores (retroalimentación)								■

14. Bibliografía

La siguiente es la bibliografía que nos proporcionara la ayuda necesaria para el desarrollo con éxito del tema.

Sitios de Internet:

- <http://es.tldp.org/Presentaciones/200103hispalinux/seco/pdf/pasarelas-sms.pdf>
- http://www.inproes.com/PDF/sms_info_general.pdf
- http://en.wikipedia.org/wiki/Short_Message_Service
- <http://www.masadelante.com/faq-sms.htm>
- <http://www.espaciopyme.com/monograficos/MensajesSMS.pdf>

