

**ESTUDIO, DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO SOFTWARE  
PARA LA GESTIÓN DE COMPARENDOS DE TRANSITO BASADO EN  
DISPOSITIVOS INTELIGENTES BAJO ARQUITECTURA MICROSOFT .NET**

**CARLOS JAVIER VASQUEZ HERNÁNDEZ  
JULIÁN SANTIAGO SANTOYO DÍAZ**

**UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA  
FACULTAD DE INGENIERÍA DE SISTEMAS  
ESCUELA DE CIENCIAS NATURALES E INGENIERÍA  
LÍNEA DE INVESTIGACIÓN EN TECNOLOGÍAS WEB  
BUCARAMANGA  
2005**

**ESTUDIO, DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO SOFTWARE  
PARA LA GESTIÓN DE COMPARENDOS DE TRANSITO BASADO EN  
DISPOSITIVOS INTELIGENTES BAJO ARQUITECTURA MICROSOFT .NET**



**CARLOS JAVIER VASQUEZ HERNÁNDEZ  
JULIÁN SANTIAGO SANTOYO DÍAZ**

Tesis de Grado para optar el título de  
Ingeniero de Sistemas

**Director  
Ing. ÁLVARO FERNANDO DELGADO MARIÑO**

**UNIVERSIDAD AUTÓNOMA DE BUCARAMANGA  
FACULTAD DE INGENIERÍA DE SISTEMAS  
ESCUELA DE CIENCIAS NATURALES E INGENIERÍA  
LÍNEA DE INVESTIGACIÓN EN TECNOLOGÍAS WEB  
BUCARAMANGA  
2005**

**Nota de aceptación**

---

---

---

---

Presidente del Jurado

---

Evaluador

---

Evaluador

Bucaramanga, 20 de Junio de 2005

## GLOSARIO

**ANCHO DE BANDA:** Término técnico que determina el volumen de información que puede circular por un medio físico de comunicación de datos, es decir, la capacidad de una conexión.

**API (APPLICATION PROGRAMMING INTERFACE):** Interfaz de Programación de Aplicaciones, es un conjunto de especificaciones de comunicación entre componentes software.

**ATENUACIÓN:** Pérdida de energía de la señal de comunicación.

**BOOL:** Tipo de variable la cual puede tener dos estados TRUE o FALSE.

**CACHE:** Área de la memoria o del disco duro reservada para hospedar una serie de datos a la espera de ser utilizados más tarde. El almacenamiento en la caché permite trabajar con esos datos rápidamente la siguiente vez que son usados.

**C SHARP (C#):** Es un lenguaje de programación orientado a objetos desarrollado por Microsoft.

**CDMA:** Acceso múltiple por división de código; Estándar digital que soporta velocidades de datos de alrededor de 14,4KBPS vía conmutación de paquetes y vía conmutación de circuitos. Es un método de transmisión móvil celular de espectro extendido que permite a varios usuarios compartir el mismo espectro de radiofrecuencia por asignación de un código único a cada usuario activo.

**cHTML:** Siglas de Compact hyper text markup language que es un lenguaje de hyper texto compacto que es utilizado para dispositivos móviles.

**CLDC (Connected Limited Device Configuration):** Dispositivo de configuración y conectividad limitada.

**CLI (Common Language Infrastructure):** Infraestructura de lenguaje común, es el lenguaje al cual se convierten las aplicaciones para ser ejecutadas.

**CLR (Common Language Runtime):** Lenguaje de tiempo de ejecución.

**CODIFICADOR:** Un codificador es un circuito cuya misión es presentar en la salida el código binario correspondiente a la entrada activada.

**DATAGRAMA:** Un datagrama es un fragmento de paquete que es enviado con la suficiente información como para que la red pueda simplemente encaminar el fragmento hacia el computador receptor, de manera independiente a los fragmentos restantes.

**DISPOSITIVO INTELIGENTE:** Es el que interactúa con los demás medios para detectar cualquier evento o acción que permita su respuesta.

**EMULADOR:** Programa que permite a un dispositivo realizar una función propia de otro.

**ENCAPSULACIÓN:** Se denomina encapsulación al ocultamiento del estado, es decir, de los datos miembro, de un objeto de manera que sólo se puede cambiar mediante las operaciones definidas para ese objeto.

**ESPECTRO ESPARCIDO:** Es una técnica por la cual la señal transmitida se ensancha a lo largo de una banda muy ancha de frecuencias, mucho más amplia, de hecho, que el ancho de banda mínimo requerido para transmitir la información que se quiere enviar.

**GATEWAY:** Es un dispositivo, con frecuencia un ordenador, que realiza la conversión de protocolos entre diferentes tipos de redes o aplicaciones.

**GPRS:** Las siglas GPRS corresponden a General Packet Radio Services, Servicio General de Paquetes por Radio. Se basa en la conmutación de paquetes realizando la transmisión sobre la red GSM que usamos actualmente. Al sistema GPRS se le conoce también como GSM-IP ya que usa la tecnología IP (Internet Protocol) para acceder directamente a los proveedores de contenidos de Internet.

**GSM:** Son las siglas de **G**lobal **S**ystem for **M**obile communications (sistema global para comunicaciones móviles). Es un sistema estándar para comunicación utilizando teléfonos móviles que incorporan tecnología digital. Por ser digital cualquier cliente de GSM puede conectarse a través de su teléfono con su computador (una PC o una PC portátil) y puede hacer, enviar y recibir e-mails, faxes, navegar por Internet, acceso seguro a la red informática de una compañía (LAN/Intranet), así como utilizar otras funciones digitales de transmisión de datos, incluyendo el Servicio de Mensajes Cortos (SMS).

**HERENCIA MÚLTIPLE:** Es un concepto proveniente de la programación orientada a objetos, en términos concretos, una instancia de objeto de la clase hija, poseerá todos los atributos y métodos de sus clases padres.

**HTTP:** Es el protocolo de la Web (WWW), usado en cada transacción. Las letras significan Hyper Text Transfer Protocol, es decir, protocolo de transferencia de hipertexto. El hipertexto es el contenido de las páginas Web, y el protocolo de transferencia es el sistema mediante el cual se envían las peticiones de acceder a una página Web, y la respuesta de esa Web, remitiendo la información que se verá en pantalla. También sirve el protocolo para enviar información adicional en ambos sentidos.

**HTTPS:** Versión segura del protocolo HTTP. El sistema HTTPS utiliza un cifrado basado en las Secure Socket Layers (SSL) para crear un canal cifrado (cuyo nivel de cifrado depende del servidor remoto y del navegador utilizado por el cliente) más apropiado para el tráfico de información sensible que el protocolo HTTP.

**INTERFAZ ABIS:** Interfaz utilizada en el sistema GSM de telefonía móvil que funciona a la velocidad de 2 Mbit/s, a través del cual se comunican las estaciones transceptoras (BTS) con las estaciones controladoras (BSC).

**METADATOS:** Son definidos como el dato sobre los datos, es un conjunto de elementos que poseen una semántica comúnmente aceptada, o sea tratan de representar la información electrónica tan dispersa y representan a la descripción bibliográfica de recursos electrónicos. Estos datos abarcan ámbitos tanto individuales como colectivos, también documentos, recursos de Internet e incluso objetos reales.

**MIDlets:** Aplicaciones MIDP.

**MIDP (PERFIL DE INFORMACIÓN PARA DISPOSITIVOS MÓVILES):** Set de APIs de java que generalmente se implementan sobre un dispositivo de configuración y conectividad limitada (CLDC). Este representa un ambiente de desarrollo apropiado para esta clase de dispositivos.

**MODIFICADOR UNSAFE:** Denota un contexto no seguro, que es necesario para cualquier operación que involucre a punteros.

**MODIFICADOR VIRTUAL:** En programación se utiliza para modificar un método o una declaración de propiedad.

**PDA (Personal Digital Assistant):** Asistente digital personal son computadores de bolsillo conocidos como pocket PC.

**POCKET PC:** Computadora de mano o bolsillo. Denominado de otra forma también, como PDA o (Personal Digital Assitant). Es un pequeño computador el cual ha sido diseñado para ocupar el mínimo espacio y por consiguiente ser fácilmente transportable, el cual ejecuta un sistema operativo Windows CE de Microsoft el cual proporciona aplicaciones similares a los de un computador de escritorio.

**POLIMORFISMO:** Describe múltiples y posibles estados de una única propiedad.

**RED INALÁMBRICA:** Red en la cual los medios de unión entre las estaciones no son cables.

**SIM (Subscriber Identifycation Module):** Tarjeta usada que contiene los datos de identificación del usuario de un teléfono móvil, como su número de teléfono.

**SINCRONIZACIÓN:** Proceso mediante el cual se transmite datos del PDA al computador, actualizando la información existente. La sincronización se realiza en la mayoría de los casos vía serie o USB, aunque también existen cables de sincronización.

**SISTEMA DE CONEXIÓN:** En este tipo de técnica no se debe establecer un canal dedicado para cada usuario sino que la conexión se realiza en el momento de utilización del canal, por lo tanto se pierde el concepto de facturación por tiempo, pasando a ser por utilización del canal de emisión. La vía de conexión es mucho más utilizada, ya que permite a los usuarios compartir el mismo medio. Se pueden recibir voz y datos simultáneamente.

**SMP:** sistemas de multiproceso simétrico.

**SOCKETS:** El protocolo TCP/IP gestiona el envío y la recepción de la información los denominados paquetes, que resultan de su división en trozos más pequeños.

La biblioteca que controla el envío/recepción de estos paquetes se denomina socket.

**SSL (Secure Socket Layer):** Protocolo para comunicaciones seguras, proporciona cifrado de datos, autenticación de servidores, integridad de mensajes y, opcionalmente, autenticación de cliente para conexiones TCP/IP.

**SYSTEM.OBJECT:** Admite todas las clases de la jerarquía de clases de .NET Framework y proporciona servicios de bajo nivel a las clases derivadas. Se trata de la clase base fundamental de todas las clases de .NET Framework; es la raíz de la jerarquía de tipos.

**STRING (Clase):** Clase del lenguaje de programación Java que permite manejar cadenas de caracteres.

**TDMA:** TDMA ("Time Division Multiple Access") es común en los sistemas de telefonía fija. TDMA es un concepto bastante antiguo en los sistemas de radio. En los sistemas modernos celulares y digitales, TDMA implica el uso de técnicas de compresión de voz digitales, que permite a múltiples usuarios compartir un canal común utilizando un orden temporal. La codificación de voz moderna, reduce mucho el tiempo que se lleva en transmitir mensajes de voz, eliminando la mayoría de la redundancia y periodos de silencio en las comunicaciones de voz. Otros usuarios pueden compartir el mismo canal durante los periodos en que éste no se utiliza. Los usuarios comparten un canal físico en un sistema TDMA, donde están asignados unos slots de tiempo. A todos los usuarios que comparten la misma frecuencia se les asigna un slot de tiempo, que se repite dentro de un grupo de slots que se llama trama.

**TECNOLOGIA CDMA (Code Division Multiple Access):** CDMA es una tecnología genérica que puede describirse, como un sistema de comunicaciones por radio celular digital que permite que un elevado número de comunicaciones de

voz o datos simultánea compartan el mismo medio de comunicación, es decir, utilizan simultáneamente un pool común de canales de radio, de forma que cada usuario puede tener acceso a cualquier canal de forma temporal; el canal es un trozo de espectro de radio que asigna temporalmente a un tema específico, como, por ejemplo, una llamada telefónica.

**TECNOLOGÍA WAP (Wireless Application Protocol):** WAP (protocolo de aplicaciones inalámbricas) es un protocolo basado en los estándares de Internet que ha sido desarrollado para permitir a teléfonos celulares navegar a través de Internet.

**TRANSACT - SQL:** Versión mejorada del lenguaje de bases de datos relacionales SQL.

**TRAU (Transcoding Rate Adaptation):** (Unidad Transcoder y Adaptadora de Velocidad), es el equipo en el cual se lleva a cabo la codificación y decodificación de la voz (fuente), así como la adaptación de velocidades en el caso de los datos.

**USING:** Instrucción de Visual C# que proporciona un mecanismo de liberación de recursos durante la ejecución de un programa.

**VALOR POR DEFECTO:** Valor predeterminado que se le asigna a una variable local de un programa.

**VELOCIDAD:** La velocidad de conexión puede llegar a los 115 kbps, 12 veces más que la permitida por la red actual GSM. Sin lugar a dudas GPRS permitirá que la tecnología WAP pueda ser mucho más potente, explotándola en una mayor proporción, despojándose de uno de los problemas más acuciantes, la velocidad de transmisión.

**VERSIONABLE:** Propiedad de Visual C# que permite crear nuevas versiones de tipos de datos sin que provoquen errores en un programa.

**WIRELESS (Inalámbrico):** Término referido a comunicaciones inalámbricas, en las que no se utiliza un medio de propagación físico, sino la modulación de ondas electromagnéticas, radiaciones o medios ópticos. Estas se propagan por el espacio vacío sin medio físico que comunique cada uno de los extremos de la transmisión.

**WTAI (Wireless Telephony Application Interface):** permite realizar llamadas telefónicas desde un sitio WAP.

**WTLS (Wireless Transport Layer Security):** La capa de seguridad que provee de privacidad, integridad y autenticación a los servicios WAP.

**XML:** es el acrónimo del inglés extensible Markup Language (lenguaje de marcado ampliable o extensible) desarrollado por el World Wide Web Consortium (W3C).

## RESUMEN

Tal es el avance tecnológico en los dispositivos móviles que ahora en nuestro entorno no se usan solamente para aplicaciones que tengan un uso didáctico, como juegos o consultas meteorológicas sino que representen innovaciones útiles para los usuarios que los posean, tal es el caso de paquetes de procesadores de palabra o calculadoras, los cuales son de gran valor ya que no es necesario tener un computador para realizar modificaciones a determinado archivo en determinado momento. En la actualidad el manejo de la información es un factor clave en cualquier tipo de negocio y tener una buena arquitectura que los maneje y que represente una base segura donde puedan ser almacenados sin mayores contratiempos crea la necesidad de invertir en tecnología que pueda agilizar procesos y disminuir costos. Principalmente el problema radica en el manejo erróneo que se le aplica a los datos de los comparendos de tránsito y a su almacenamiento en una correspondiente base de datos ya que tienen varias fases donde se podrían presentar alteraciones a la hora de almacenarlos y crear anomalías que luego serían difíciles de detectar. Por lo anterior se plantea en primer lugar modificar el proceso que se lleva actualmente mostrando una alternativa en el manejo de estos datos, desde su recolección hasta que sean integrados a una base de datos con una mayor rapidez. Como primera medida se pretende eliminar la toma de datos en papel de forma manual llenando el formato del comparendo de tránsito, para tal caso la alternativa planteada es procesar esos datos directamente en un dispositivo pocket el cual tendrá un formato que reciba la misma información que el formato original pero reduciendo los índices de errores, segundo se omitirá el proceso de digitar cada comparendo para su posterior ingreso a una base de datos, para este aspecto se utilizará sincronización desde el dispositivo pocket a la base de datos dando un margen de tiempo incomparable con el que se maneja actualmente y por último las consultas de los comparendos a través de un portal wap para acceder remotamente desde pocket duales o teléfonos celulares que manejen este tipo de tecnología.

## CONTENIDO

	pág.
INTRODUCCIÓN	22
1. TECNOLOGÍAS MÓVILES	27
1.1 TECNOLOGÍA WAP	27
1.2 BENEFICIOS DE UTILIZAR TECNOLOGÍA WAP	29
1.3 MODELO DE PROGRAMACIÓN WAP	29
1.4 WML	31
1.5 TECNOLOGÍA GSM	32
1.5.1 Arquitectura GSM	32
1.5.1.1 Estación Móvil (MS)	33
1.5.1.2 El Subsistema Estación Base (BSS)	34
1.5.1.3 Subsistema de Red (NSS)	34
1.5.1.4 El Centro de Operaciones y Mantenimiento (OSS)	35
1.6 TECNOLOGÍA CDMA	35
1.6.1 Ventajas de CDMA sobre GSM	36
1.6.2 Ventajas Fundamentales de CDMA	36
2. ANÁLISIS DE LOS ENTORNOS DE DESARROLLO	38
2.1 MICROSOFT	38
2.1.1 Visual Studio .NET	38

2.2 ARQUITECTURA DE UNA APLICACIÓN WEB MÓVIL	40
2.3 VISUAL C SHARP (C#)	40
2.4 SQL SERVER 2000	44
2.5 SQL SERVER 2000 WINDOWS CE EDITION VERSIÓN 2.0 (SQL SERVER CE)	47
2.6 CONCEJOS PARA EL DESARROLLO DE APLICACIONES MÓVILES	51
2.7 J2ME	53
2.7.1 Java apropiado para desarrollo en dispositivos móviles	56
2.7.2 Perfiles J2ME	56
2.7.2.1 MIDP – MIDlets	57
3. ANÁLISIS DEL SISTEMA DE COMPARENDOS ACTUAL Y SUS MODIFICACIONES PARA ADAPTARLO A UNA ALTERNATIVA TECNOLOGÍA A TRAVÉS DE DISPOSITIVOS MÓVILES	60
3.1 PROCESO DEL SISTEMA DE COMPARENDOS ACTUAL.	60
3.2 EVALUACIÓN DEL PROCESO DE SISTEMA DE COMPARENDOS	61
3.2.1 Ventajas	61
3.2.2 Desventajas	61
3.3 PROPUESTA ALTERNATIVA SOBRE CAMBIOS EN EL PROCESO ACTUAL DEL SISTEMA DE COMPARENDOS	62
3.3.1 Evaluación de la alternativa propuesta.	63
3.3.2 Ventajas	63
3.3.3 Desventajas	64
3.4 COMO HACER LAS MODIFICACIONES AL SISTEMA DE COMPARENDOS ACTUAL	64

4. DESARROLLO DE LA APLICACIÓN	67
4.1 MODELAMIENTO DEL SISTEMA	67
4.1.1 Requerimientos	68
4.1.1.1 Casos de uso	68
4.1.1.2 Casos de uso prototipo de Sistema de Comparendos	69
4.1.2 Análisis	71
4.1.2.1 Diagrama de clases	71
4.1.2.2 Diagrama de clases prototipo de sistema de comparendos	71
4.1.2.3 Diagrama de Secuencias	71
4.1.2.4 Diagramas de Secuencias prototipo de Sistema de Comparendos	71
4.1.2.5 Diagramas de Colaboración	79
4.1.2.6 Diagramas de Colaboración prototipo de Sistema de Comparendos	79
4.1.2.7 Diagramas de Actividad	87
4.1.2.8 Diagramas de Actividad prototipo de Sistema de Comparendos	88
4.1.2.9 Diagrama de Paquetes	96
4.1.2.10 Diagramas de Paquetes prototipo de Sistema de Comparendos	97
4.1.3 Diseño	100
4.1.3.1 Prototipo de registro de comparendos	101
4.1.3.2 Prototipo de gestión de base de datos	111

4.1.3.3 Prototipo de consultas de comparendos	115
4.1.4 Implementación	118
CONCLUSIONES	123
RECOMENDACIONES Y TRABAJOS FUTUROS	125
BIBLIOGRAFÍA	126

## LISTA DE FIGURAS

	pág.
Figura 1. Funcionamiento del WAP	28
Figura 2. Modelo de programación WAP	30
Figura 3. Arquitectura GSM	33
Figura 4. Arquitectura Framework de .NET	39
Figura 5. Componentes Visual Studio .NET	39
Figura 6. Conexión con la Base de Datos	45
Figura 7. Conexión con SQL Server	46
Figura 8. Acceso a datos	48
Figura 9. Arquitectura de almacenamiento	49
Figura 10. Arquitectura Java	55
Figura 11. Arquitectura MIDP	55
Figura 12. Proceso actual del sistema de comparendos	60
Figura 13. Modelo alternativo	63

Figura 14 Arquitectura Externa	65
Figura 15 Arquitectura Interna	66
Figura 16. Diagrama de Casos de Uso del Sistema	70
Figura 17. Diagrama de Clases del Sistema	73
Figura 18. Diagrama de Secuencia Actualizar Cuenta Usuario	75
Figura 19. Diagrama de Secuencia Crear Cuenta Usuario	75
Figura 20. Diagrama de Secuencia Consultar Comparendo	76
Figura 21. Diagrama de Secuencia Crear Comparendo	76
Figura 22. Diagrama de Secuencia de Digitar Cedula y Placa	77
Figura 23. Diagrama de Secuencia de Eliminar Cuenta de Usuario	77
Figura 24. Diagrama de Secuencia de Listar Cuentas de Usuario	78
Figura 25. Diagrama de Secuencia de Sincronización del Dispositivo Móvil	80
Figura 26. Diagrama de Colaboración de Actualizar Cuenta de Usuario	81
Figura 27. Diagrama de Colaboración de Consultar Comparendo	82
Figura 28. Diagrama de Colaboración de Crear Comparendo	83
Figura 29. Diagrama de Colaboración de Crear Cuenta de Usuario	84
Figura 30. Diagrama de Colaboración de Digitar Cedula y Placa	85
Figura 31. Diagrama de Colaboración de Eliminar Cuenta de Usuario	86

Figura 32. Diagrama de Colaboración de Listar Cuentas de Usuario	87
Figura 33. Diagrama de Colaboración de Sincronización de Dispositivos	88
Figura 34. Diagrama de Actividad de Actualizar Cuenta de Usuario	89
Figura 35. Diagrama de Actividad de Consultar Comparendo	90
Figura 36. Diagrama de Actividad de Crear Comparendo	91
Figura 37. Diagrama de Actividad de Crear Cuenta de Usuario	92
Figura 38. Diagrama de Actividad de Digitar Cedula y Placa	93
Figura 39. Diagrama de Actividad de Eliminar Cuenta de Usuario	94
Figura 40. Diagrama de Actividad de Listar Cuentas de Usuario	95
Figura 41. Diagrama de Actividad de Sincronización del Dispositivo Móvil	96
Figura 42. Diagrama de Paquetes de Interfaces	97
Figura 43. Diagrama de Paquetes de Controles	98
Figura 44. Diagrama de Paquetes de Entidades	99
Figura 45. Diagrama de Implementación	100
Figura 46. Interfaz de inicio	101
Figura 47. Interfaz de bienvenida	102
Figura 48. Interfaz de sincronización	103
Figura 49. Interfaz de Información	104

Figura 50. Interfaz de registro de comparendo paso 1	105
Figura 51. Interfaz de registro de comparendo paso 2	106
Figura 52. Interfaz de registro de comparendo paso 3	107
Figura 53. Interfaz de registro de comparendo paso 4	107
Figura 54. Interfaz de registro de comparendo paso 5	108
Figura 55. Interfaz de registro de comparendo paso 6	109
Figura 56. Interfaz de registro de comparendo paso 7	109
Figura 57. Interfaz de registro de comparendo paso 8	110
Figura 58. Interfaz de registro de comparendo paso 9	110
Figura 59. Interfaz de registro de comparendo paso 10	111
Figura 60. Interfaz de inicio	112
Figura 61. Interfaz de Información	113
Figura 62. Interfaz de gestión de usuarios	114
Figura 63. Interfaz de gestión de comparendos	115
Figura 64. Interfaz de inicio	116
Figura 65. Interfaz de consultas de comparendos	117
Figura 66. Interfaz de consultas de comparendos	117

## LISTA DE TABLAS

	pág.
Tabla 1. Comparativo de lenguajes para aplicaciones móviles	41
Tabla 2. SQL Server CE vs. SQL Server 2000	50
Tabla 3. Casos de Uso y Actores	70
Tabla 4. Convenciones para el nombramiento de controles	119
Tabla 5. Código fuente del método mnuCarga_Click	120
Tabla 6. Código fuente del método mnuSync_Click	121

## INTRODUCCIÓN

La gran aceptación de los dispositivos móviles como Pocket PC y telefonía celular y su gran crecimiento en su parte tecnológica incrementando la capacidad de almacenar aplicaciones, ha permitido que sea posible desarrollar entornos que integren varias tecnologías para sacar el máximo potencial de estos componentes que hoy por hoy se han convertido no en un lujo si no en una necesidad.

La adecuación de tecnología a procesos actuales permite un buen desempeño en las labores de alguna entidad o empresa que quiera dar un mejor manejo a datos o servicio al cliente y por consiguiente trae beneficios que dan fiabilidad y certeza a los clientes sobre las capacidades de una empresa que inicie un proceso de cambio, hacia el manejo de la tecnología en sus procesos.

El problema a tratar, es dar una alternativa al manejo de los comparendos de tránsito a través de dispositivos tecnológicos que permitan realizar el mismo procedimiento pero de una manera mas ágil y eficaz dando nuevas pautas en el uso de la tecnología en campos donde no se ha explotado completamente y que puede traer buenos resultados. La importancia de mejorar el sistema de comparendos actual permitiría evitar traumatismos tanto en la parte administrativa por el manejo y localización de la información y a los clientes quienes consultarían el estado de cuentas por infracciones de tránsito en cualquier parte sin tener que desplazarse hasta determinado sitio para saber el estado de cuentas de su vehículo.

Para implementar esta alternativa se proporcionaron objetivos tanto generales como específicos los cuales se plantearon en primera medida en realizar un estudio, diseño e implementación de un prototipo software para la gestión de comparendos de tránsito basado en dispositivos inteligentes bajo arquitectura

Microsoft .NET, mediante una investigación exploratoria; manejar principios, evaluar conceptos y técnicas que permitan desarrollarlo, con el objeto de agilizar el mecanismo que se lleva actualmente y disminuir los errores que éste presenta, posteriormente se hizo un estudio teniendo cuenta las directrices en el desarrollo para dispositivos inteligentes basado en la plataforma Microsoft .NET, mediante la recolección de material bibliográfico, recopilando la información a lo largo de la investigación en documentos, tutoriales y manuales de usuario e instalación, con el fin de mostrar las características técnicas de la tecnología que sirvan como guías para estudios posteriores, aparte el desarrollar un prototipo software que permita realizar la gestión de comparendos de tránsito basado en un emulador Pocket PC Phone Edition, creando una aplicación en C # que permita capturar la información de los comparendos y que pueda evidenciar el uso de la tecnología .NET en este dispositivo, para proponer una alternativa de solución al mecanismo que se realiza hasta el momento y por último implementar una gestión de consultas sobre el estado de cuentas de automotores para dispositivos inteligentes por medio de páginas wap accesibles a este tipo de tecnología para agilizar el acceso a la información de los usuarios.

Mundialmente conocemos que sobre las PDA's existen aplicaciones financieras, de ventas, juegos que hacen las diferentes empresas para realzar la capacidad del personal, algunas empresas son Mastersoft, Psiloc, Mobisol. Las nuevas características de los nuevos dispositivos inteligentes tienen a su disposición las ventajas de un dispositivo PDA pero con la particularidad de no solo ser para ingresar datos, sino también, para realizar llamadas telefónicas.

Según el modelo, éste puede contar con tecnología bluetooth para conectarse con impresoras, cámaras y cualquier dispositivo que cuente con esta característica.

También cuentan con beneficios de multimedia para reproducir música y video; pero lo más importante es que son compatibles con java y C++, esto permite al usuario poder desarrollar e implementar sus propias aplicaciones.

Según datos estadísticos de Microsoft en el 2003 se vendieron alrededor de 1.8 millones de unidades y se estima que para el 2005 se hallan vendido alrededor de 28 millones de unidades que tengan tecnología .NET. Esto se basa en que los usuarios buscan grandes facilidades e integración de tecnologías que faciliten las tareas cotidianas en un menor tiempo.

En Colombia el uso de PDA's es limitado puesto que son dispositivos de un alto precio en el mercado y del cual los usuarios no pueden sacar el mejor rendimiento, por no haber una cultura en su utilización, sin embargo son dispositivos que presentan grandes capacidades mencionando un caso en particular la implementación de comparenderas electrónicas en la ciudad de Bogotá D.C. con un plan piloto que comenzó con 70 de estos dispositivos (uno por agente de tránsito) y del cual se espera para un término de tres años que operen alrededor de 600 en toda la ciudad, con nuevas aplicaciones a corto plazo se podrá determinar la reincidencia en la infracción de determinado conductor o consultar las infracciones pendientes de cierto vehículo, esto nos da una perspectiva de la capacidad de estos dispositivos y su viabilidad en facilitar procesos. Según la secretaria de tránsito y transporte esto conlleva a un ahorro de costos ya que la información ingresa en línea de manera inmediata puesto que se instalaron centrales de acopio para dicha operación. Otro caso menos conocido es el desarrollado en parquesoft en el departamento del Cauca; es un sistema de orientación urbana para ciegos denominado Sky Vision que, mediante una PDA, alimentada con un dispositivo GPS (Global Positioning System), decirle al usuario mediante una voz sintética en qué calle y carrera está y, qué sitio de interés tiene cerca.

El planteamiento de solución para es la construcción de software, el cual tenga como base el mecanismo actual de realización de comparendos pero basado en la captura de la información sobre una PDA donde se pueda tener un control total de los datos y el cual presente una forma rápida de traspaso de datos a una base de datos donde se almacene la información consignada en el dispositivo dando un

margen de tiempo mucho menor comparada con la anterior dando como resultado un menor costo ya que el proceso de traspasar datos de los talonarios a la base de datos seria abolido. También se hará que sobre los dispositivos se puedan realizar consultas, este aspecto refuerza los mecanismos de consulta existentes en la dirección de transito, esta se hará por tecnología WAP, a través del mismo dispositivo, puesto que representa un ítem fundamental por la versatilidad en determinar rápidamente el estado de cuentas de un vehiculo o conductor.

La importancia de un estudio extenso sobre aplicaciones en dispositivos móviles en la ingeniería de sistemas es de carácter esencial, la tendencia hacia aplicaciones móviles determina el uso de nuevas tecnologías que abarcan grandes campos como programación, redes e ingeniería de software, manteniendo un constante aprendizaje sobre las ultimas herramientas y entornos de desarrollo que permiten comprender a cabalidad el próximo paso en la tecnología.

A continuación se dará una breve explicación de los capítulos que contiene este documento.

### **ANÁLISIS DEL SISTEMA DE COMPARENDOS ACTUAL Y SUS MODIFICACIONES PARA ADAPTARLO A UNA ALTERNATIVA TECNOLOGÍA**

**A TRAVÉS DE DISPOSITIVOS MÓVILES:** En este capitulo se da una descripción de lo actualmente se hace al realizar un comparendo de transito y como se puede modificar con tecnología sobre dispositivos móviles para que los procesos sean mas fáciles y óptimos.

**TECNOLOGÍAS MÓVILES:** La gran cantidad de tecnologías móviles que se desconocen hacen que se limite el uso de la tecnología en los procesos que comúnmente conocemos por tanto se hace una explicación de las tecnologías mas importantes para tener una visión mas clara ellas.

**ANÁLISIS DE LOS ENTORNOS DE DESARROLLO:** Investigar a grandes rasgos las características de los distintos lenguajes de desarrollo para la implementación en dispositivos móviles y determinar a partir de esto cuales características se adaptan mejor a las necesidades de la solución.

**DESARROLLO DE LA APLICACIÓN:** Mostrar como fue realizada la aplicación por las diferentes etapas tanto en modelamiento y codificación agregando vistas de lo implementado.

## **1. TECNOLOGÍAS MÓVILES**

Las tecnologías móviles son la que hacen posible que haya una comunicación entre estos dispositivos, para tener una idea mas clara se estudiaran algunas de ella consideradas las más importantes, las que están desarrollo continuo y las que se representan mayor relación con el objetivo propuesto. Estas tecnologías no son escasas por el contrario hay una gran variedad que representan el sentido de las comunicaciones en distintos aspectos.

### **1.1 TECNOLOGÍA WAP**

WAP son las siglas de Wireless Application Protocol. WAP es un protocolo basado en los estándares de Internet que ha sido desarrollado para permitir a teléfonos celulares navegar a través de Internet.

Con la tecnología WAP se pretende que desde cualquier teléfono celular WAP se pueda acceder a la información que hay en Internet así como realizar operaciones de comercio electrónico.

WAP es una serie de tecnologías que consisten en: WML, que es el lenguaje de etiquetas, WMLScript es un lenguaje de script, lo que vendría a ser JavaScript y el Wireless Telephony Application Interface (WTAI).

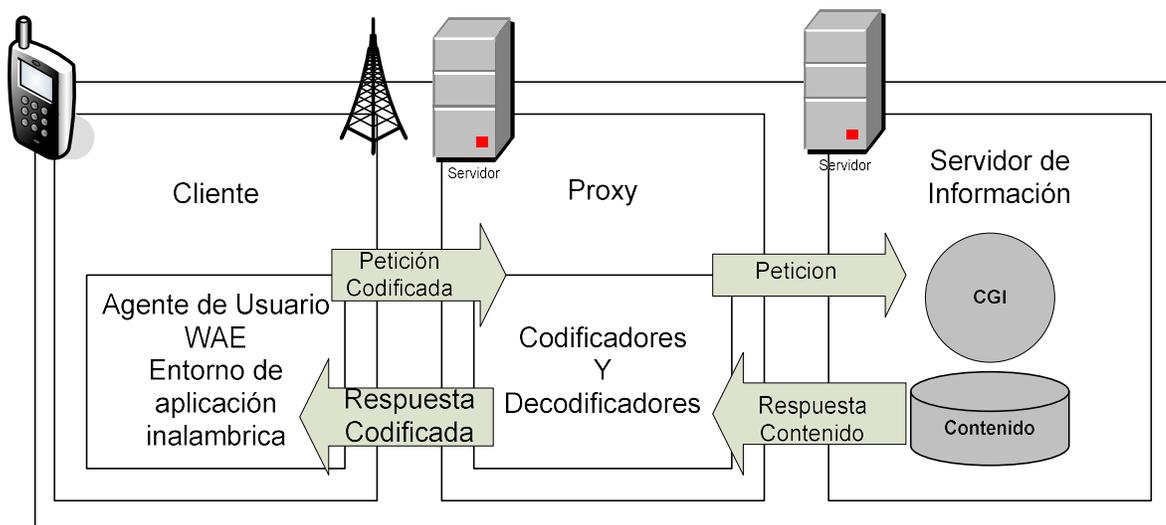
Las características principales de WML son:

- Soporte para imágenes y texto, con posibilidad de texto con formato.
- Tarjetas agrupadas en barajas. Una pagina WML es como una página HTML en la que hay una serie de cartas, al conjunto de estas cartas se les suele llamar baraja.

- Posibilidad de navegar entre cartas y barajas de la misma forma que se navega entre páginas Web.
- Manejo de variables y formularios para el intercambio de información entre el teléfono celular y el servidor.

WML es un lenguaje de marcas similar al HTML. WML es compatible con XML 1.0. Las páginas WML son llamadas barajas ya que están compuestas por cartas, un navegador WAP, solo puede mostrar una carta al mismo tiempo. <sup>1</sup>

**Figura 1. Funcionamiento del WAP**



Es una solución que proporciona acceso a la información desde cualquier terminal y en cualquier red inalámbrica. Los teléfonos celulares habilitados con la nueva tecnología WAP, son acompañantes que proporcionan información breve, oportuna e inmediata, cuando el usuario la necesita para la toma de decisiones estando en movimiento, sin requerir de una computadora. Usando cualquier dispositivo WAP puede acceder a los sistemas informáticos.

Disponible en Internet:

<sup>1</sup> <http://www.wapforum.org/what/technical.htm>

WAP es completamente independiente del operador y de la tecnología. Puede funcionar bajo GSM y GPRS por ejemplo.

## **1.2 BENEFICIOS DE UTILIZAR TECNOLOGÍA WAP**

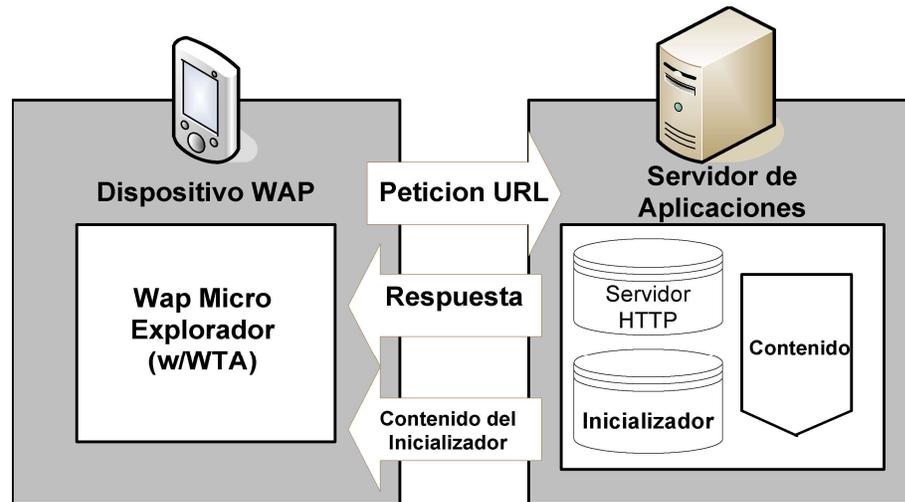
Los Beneficios de Wap son muchos pero se pueden separar en:

- Incrementa la productividad personal.
- Mejora la colaboración del grupo.
- Libertad de elección. Wap funciona independientemente del operador.
- Libertad para decidir cuando y como se accede a la información.
- Solo es necesario el móvil.
- Rapidez de actuación por parte del profesional.
- Menos papeleo y riesgo de pérdida de información.
- Control total de los servicios.
- Inmediatez de respuesta, rapidez y eficacia.

## **1.3 MODELO DE PROGRAMACIÓN WAP**

El modelo de programación WAP como muestra la figura, es un modelo WWW con pequeños cambios. Adoptando el modelo de programación WWW que provee muchos beneficios para el desarrollo de aplicaciones como un modelo ya probado y un modelo de programación familiar y la habilidad de reutilizar herramientas existentes, (Web Server, SML). Las optimizaciones y las extensiones se han hecho para nivelar las características del ambiente wireless.

**Figura 2. Modelo de programación WAP**



Las variables más significativas sobre WAP en su modelo de programación han sido:

- Inicializador
- Soporte de Telefonía (WTA)

WAP define unos componentes estándar para que sea posible la comunicación entre los terminales móviles y los servidores de red.

- **Nombramiento de modelo estándar:** Los estándares de WWW URL's son usados para identificar el contenido y el origen, también para identificar los recursos locales tales como funciones de control
- **Contenido Escrito:** Todo el contenido de WAP es dado a través de una dirección escrita específicamente. Estos permiten a los agentes de usuarios WAP acceder a la base de contenido específico por su dirección.

- Contenido en formato Estándar: Los formatos de contenido WAP están basados en tecnología WWW e incluyen información electrónica, imágenes y lenguaje script
- Protocolos de comunicaciones estándar: los protocolos WAP para comunicaciones están activados para manejar una comunicación de petición respuesta del terminal móvil a la red de servicio Web. Los tipos de contenido WAP y sus protocolos han sido optimizados para el mercado global de dispositivos sin cables.

#### **1.4 WML**

WML (Wireless Markup Language) es un lenguaje de etiquetas basado en XML y su propósito es ser usado específicamente en dispositivos con limitaciones de su ancho de banda como celulares.

WML fue diseñado con limitaciones puesto que únicamente se tuvo en mente dispositivos de poco ancho de banda y sus características incluyen:

- Una pequeña pantalla y una limitada entrada de datos por los usuarios.
- Conexión de banda estrecha
- Memoria limitada y recursos computacionales

WML incluye cuatro áreas funcionales importantes:

- La presentación de texto en varios formatos, soporta imágenes y una disposición de comandos.
- Organizado en barajas y estas en cartas, toda la información esta organizada en la colección de cartas y bajaras que exista.
- Navegación a través de cartas y enlaces para acceder a otras cartas.

- Parametrización y manejo de estados en las barajas usando un modelo de estado.

## **1.5 TECNOLOGÍA GSM**

Las siglas GSM lo cual significa Sistema Global para las Comunicaciones Móviles, conocida como Group Special Mobile es un estándar mundial para teléfonos móviles digitales. El estándar es abierto, no propietario y evolutivo (aún en desarrollo).

En GSM, una conexión se puede dedicar tanto a voz como a datos.

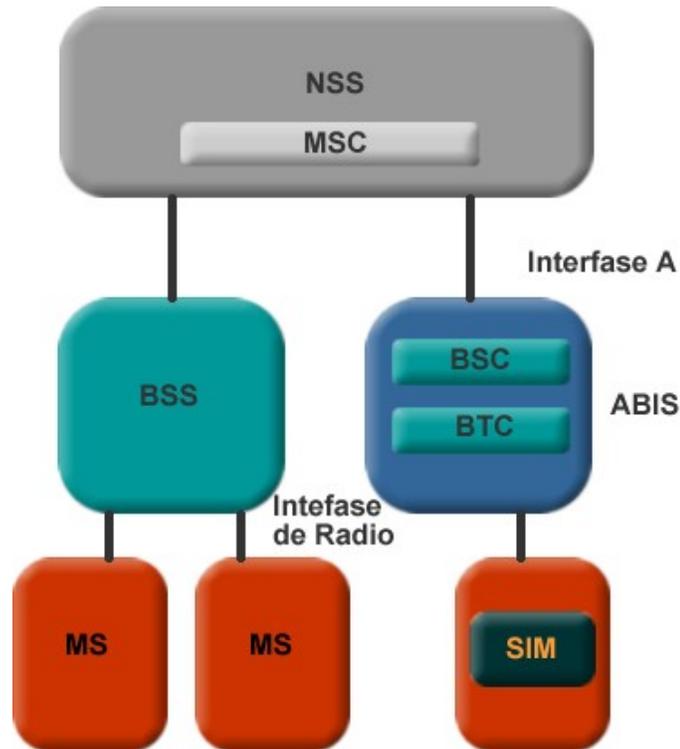
Una llamada de voz utiliza un codificador GSM específico para transmitir el sonido sobre un enlace digital de 9600 bps a la estación base.

Una conexión de datos, permite que el usuario utilice el dispositivo móvil como un módem de 9600 bps.

### **1.5.1 Arquitectura GSM**

Una breve descripción de la arquitectura interna de GSM, podemos identificar los siguientes subsistemas: Estación Móvil (Mobile Station ó MS), subsistema de la estación base ("Base Station Subsystem" ó BSS), el Subsistema de Red (Network Switching Subsystem ó NSS) que debe gestionar las comunicaciones y conectar las estaciones móviles a otro tipo de redes o a otras estaciones móviles. Además esta el Centro de Operaciones y Mantenimiento (Operation and Service Subsystem u OSS). Las MS, BSS y la NSS forman la parte operacional del sistema, mientras que el OSS proporciona los medios para que el operador los controle.

**Figura 3. Arquitectura GSM**



1.5.1.1 Estación Móvil (MS). La estación móvil representa normalmente la única parte del sistema completo que el usuario ve. Existen estaciones móviles de muchos tipos como las montadas en vehículos, y los equipos portátiles, pero quizás las más desarrolladas sean los que se encuentran en dispositivos manuales, como teléfonos.

Otra parte dentro de la estación móvil es el Módulo de Identificación Suscrito ("Subscriber Identity Module" ó SIM), que es un nombre muy limitado para las diversas funciones que este permite. El SIM es básicamente una tarjeta, que sigue las normas ISO que contiene toda la información relacionada con el suscrito almacenada en la parte del usuario de la interfaz de radio. Sus funciones, además de la capacidad de almacenar información, están relacionadas con el área de la confidencialidad.

1.5.1.2 El Subsistema Estación Base (BSS). El BSS agrupa la maquinaria de infraestructuras específicas a los aspectos celulares de GSM. El BSS está en contacto directo con las estaciones móviles a través de la interfaz de radio. Por lo tanto, incluye las máquinas encargadas de la transmisión y recepción de radio, y de su gestión. Por otro lado, el BSS está en contacto con los conmutadores del NSS. La misión del BSS se puede resumir en conectar la estación móvil y el subsistema de red, y por lo tanto, conecta al usuario del móvil con otros usuarios. El BSS tiene que ser controlado, y por tanto debe estar en contacto con el centro de operaciones y mantenimiento.

De acuerdo con la estructura de GSM, el subsistema de estación base incluye dos tipos de máquinas: el Base Transceiver Station ó Transceptor de la Estación Base, en contacto con las estaciones móviles a través de la interfaz de radio, el BSC Controlador de la Estación Base, en contacto con los conmutadores del subsistema de red o NSS.

Un BTS lleva los dispositivos de transmisión y recepción por radio, incluyendo las antenas, y también todo el procesado de señales específico a la interfaz de radio, y que se verá con posterioridad. Los BTS se pueden considerar como complejos módems de radio.

Un BSC es en definitiva un pequeño conmutador con una gran capacidad de cómputo. Su principal función, es la gestión de los canales de radio.

El concepto de la interfaz entre el BSC y el NSS se le conoce como interfaz A, y se introdujo al principio de la elaboración del estándar GSM. Solamente después se decidió estandarizar también la interfaz entre el BTS y el BSC, y se le llamó interfaz ABIS, sin tener nada que ver con la interfaz A.

1.5.1.3 Subsistema de Red (NSS). El NSS incluye las principales funciones de conmutación en GSM, así como las bases de datos necesarias para los datos de

los suscritos y para la gestión de la movilidad. La función principal del NSS es gestionar las comunicaciones entre los usuarios GSM y los usuarios de otras redes de telecomunicaciones. Dentro del NSS, las funciones básicas de conmutación están realizadas por el centro de conmutación de servicios móviles, cuya función principal es coordinar el establecimiento de llamadas hacia y desde los usuarios GSM. El Centro de Conmutación de servicios Móviles tiene interfaces con el subsistema de estación base por un lado (a través del cual está en contacto con los usuarios GSM), y con redes externas por el otro. La interfaz con las redes externas requiere un "gateway" para la adaptación, cuya función es más o menos importante dependiendo del tipo de datos y de la red a la que se accede.

1.5.1.4 El Centro de Operaciones y Mantenimiento (OSS). El centro de operaciones y mantenimiento tiene varias tareas que realizar. Todas estas tareas requieren interacciones entre algunas o todas las máquinas de la infraestructura que se encuentra en el BSS ó en el NSS y los miembros de los equipos de servicio de las distintas compañías comerciales. <sup>2</sup>

## **1.6 TECNOLOGÍA CDMA**

CDMA o Acceso múltiple por división de código en secuencia directa, es un método de modulación para transmisión de señales digitales sobre ondas radiofónicas.

CDMA es una tecnología genérica que puede describirse, como un sistema de comunicaciones por radio celular digital que permite que un elevado número de comunicaciones de voz o datos simultánea compartan el mismo medio de comunicación, es decir, utilizan simultáneamente un canal común de radio, de forma que cada usuario puede tener acceso a cualquier canal de forma temporal.

---

<sup>2</sup> [http://ceres.ugr.es/~alumnos/c\\_avila/gsm23.htm](http://ceres.ugr.es/~alumnos/c_avila/gsm23.htm)

CDMA es una técnica de acceso múltiple. En CDMA, cada comunicación se codifica digitalmente utilizando una clave de encriptación que solamente conocen los terminales involucrados en el proceso de comunicación.

La distribución celular y la reutilización de frecuencias son dos conceptos estrechamente relacionados con la tecnología CDMA; el objetivo es realizar una subdivisión en un número importante de celdas para cubrir grandes áreas de servicio. Desde un punto de vista de distribución celular, la tecnología CDMA se puede contemplar como una superación de la tradicional subdivisión celular hexagonal.

### **1.6.1 Ventajas de CDMA sobre GSM**

En el cambio hacia la 3G hay dos tendencias tecnológicas CDMA y GSM. En esta última, su próximo paso es ir al estándar GPRS (General Packet Radio Services) que vendría siendo lo que se llama generación 2.5 para finalmente llegar a 3G con W-CDMA que alcanza mayor espectro radioeléctrico. CDMA ofrece muchas ventajas de eficiencia de espectro: es más rápida en velocidad y en transmisión de datos sobre GSM actual, que tiene muchas ventajas en lo referente a la penetración de mercado y economías a escala a nivel mundial. Los operadores basan sus estrategias especialmente en ellos. Como es sabido, la tercera generación permitirá recibir y enviar información multimedios desde cualquier dispositivo móvil o fijo y permitirá velocidades desde hasta 2Mbps, las cuales estarán disponibles con CDMA2000.

### **1.6.2 Ventajas Fundamentales de CDMA**

1. Mejora el tráfico telefónico
2. Mejora la calidad de transmisión de voz y eliminación de los efectos audibles de atenuación multitrayecto
3. Reducción del número de lugares necesarios para soportar cualquier nivel de tráfico telefónico
4. Simplificación de la selección de lugares

5. Disminución de las necesidades en despliegue y costos de funcionamiento debido a que se necesitan muy pocas ubicaciones de celda
6. Disminución de la potencia media transmitida
7. Reducción de la interferencia con otros sistemas
8. Bajo consumo de energía lo cual ofrece más tiempo de conversación y permitirá baterías más pequeñas y livianas <sup>3</sup>

---

<sup>3</sup> <http://www.cdma2000.com>

## 2. ANÁLISIS DE LOS ENTORNOS DE DESARROLLO

### 2.1 MICROSOFT

#### 2.1.1 Visual Studio .NET

Visual Studio .NET es un entorno de desarrollo que permite realizar aplicaciones móviles. Proporciona a los desarrolladores la facilidad de manejar la programación en lenguajes como Visual Basic y Visual C++; y recurrir a servicios Web XML (ver figura 6).

Todos los controles de éste entorno de desarrollo automáticamente perciben las funciones del dispositivo inteligente y emiten el lenguaje de marcación correcto, permitiendo a los desarrolladores crear una aplicación móvil que funcione en diferentes dispositivos, desde teléfonos móviles basados con tecnología WAP hasta teléfonos móviles basados en cHTML (Imode y HTML para PDA's).<sup>4</sup>

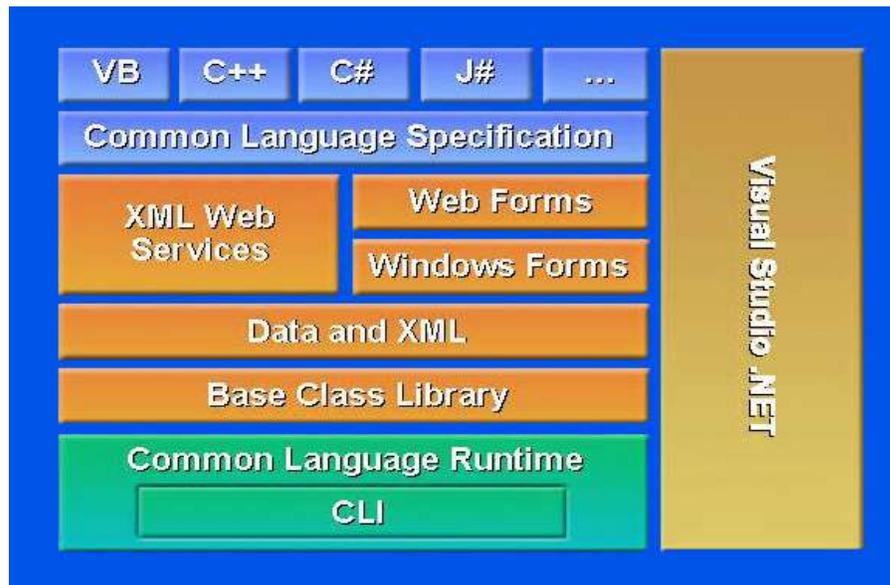
Visual Studio .NET (ver figura 7), proporciona características de crear perfiles específicos para cada forma de programación, y de ésta manera, permite a los programadores usar diferentes tipos de lenguajes en un solo programa ya que al compilar todo cambia a un lenguaje común de ejecución (CLI).

---

Disponible en Internet:

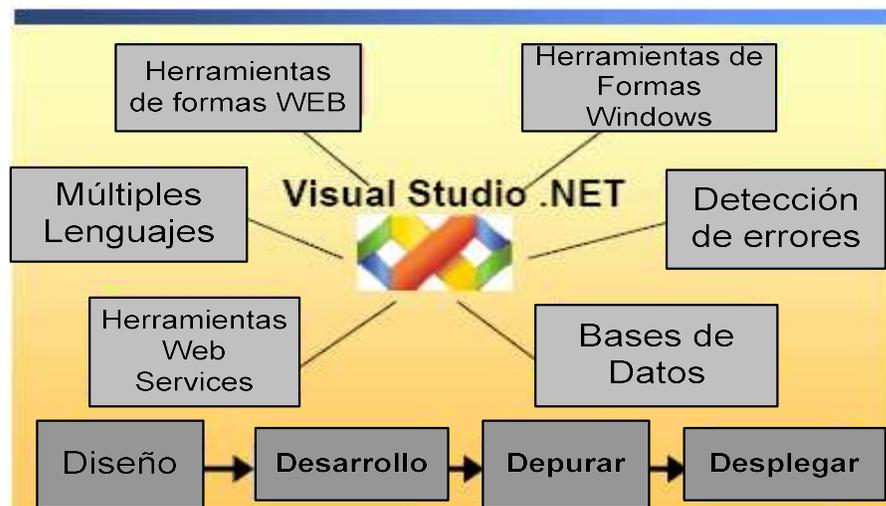
<sup>4</sup>[http://www.microsoft.com/spanish/msdn/comunidad/comunidades/aplicaciones\\_moviles/art02/default.asp](http://www.microsoft.com/spanish/msdn/comunidad/comunidades/aplicaciones_moviles/art02/default.asp)

Figura 4. Arquitectura Framework de .NET



Fuente: Microsoft Visual Studio .Net Evaluation Guide, Microsoft Corporation, Second Edition, 2002

Figura 5. Componentes Visual Studio .NET



## **2.2 ARQUITECTURA DE UNA APLICACIÓN WEB MÓVIL**

Los siguientes pasos son los que corresponden a una arquitectura móvil, cuales son los mecanismos y procesos que se presentan de acuerdo a la misma.

Para comenzar a construir una aplicación Web móvil se necesita el servicio del Internet information Server (IIS) y el Framework de .NET, esto extiende la funcionalidad de ASP .NET para facilitar el objetivo de usar tecnología de formas móviles Web, en el desarrollo se puede usar esta tecnología para dentro del framework de .NET usando servicios de ADO .NET y XML.

Una vez terminada una aplicación para este caso sobre una PDA, esta solicita una petición HTTP al servidor Web, la petición HTTP se procesa en el servidor en tres etapas principales. La primera identifica la petición del dispositivo en este caso una Pocket PC, también determina sus capacidades por ejemplo navegadores, lenguaje de etiquetas y capacidades de imágenes.

La segunda determina el contenido de las páginas móviles las cuales tienen extensión ASPX que accedieron al servidor las cuales contienen el URL y el encabezado de información, en esta etapa es compilada y guardada en el cache para posteriormente dar una respuesta, una vez compilada no es necesario hacer este nuevo proceso ya que se encuentra almacenado en la memoria cache.

La tercera etapa hace que los controles móviles ejecuten el contenido de la página, ya sea búsqueda de información u operaciones relacionadas con la aplicación en si, luego se encapsula el contenido para dar una respuesta al dispositivo.

## **2.3 VISUAL C SHARP (C#)**

C# es un lenguaje de programación orientado a objetos de la empresa Microsoft para la plataforma .NET. C# se caracteriza por que tiene las principales

características de entornos de desarrollo como C++, Java, Visual Basic o Delphi (ver tabla 1). El inventor de C# (Anders Heljsberg), fue creador de muchos otros lenguajes y entornos de desarrollo como Turbo Pascal, Delphi, Visual J++, entre otros. Lo que se busca con éste lenguaje de programación es lograr obtener la potencia de C++ con la simplicidad de lenguajes como Visual Basic.

**Tabla 1. Cuadro Comparativo de lenguajes para aplicaciones móviles**

ITEMS	C#	C++	J2ME
Modernidad	X		
Orientación a objetos	X		X
Herencia Simple	X		X
Modificador virtual	X	X	
Incluye delegados	X	X	
Sistema de tipo unificado	X		
Operadores ++ y – (prefija y posfija)	X	X	
Eficiente (modificador unsafe)	X	X	
Compatibilidad (interoperable)	X	X	X

**Las principales características de C# son:**

- Este lenguaje excluye elementos que otros lenguajes contienen y que no son necesarios en .NET.
- C# reúne elementos útiles para el desarrollo de aplicaciones móviles.

- Debido a que es un lenguaje orientado a objetos (soporta las características del paradigma de programación orientado a objetos), no acepta funciones ni variables globales, por el contrario, el código y los datos se definen con definiciones de tipos de datos, lo anterior, elimina problemas por conflictos de nombres y facilita la legibilidad del código.
- La sintaxis de C# contiene elementos del diseño de componentes, es decir, su sintaxis admite definir *propiedades*, *eventos* y atributos, en aplicaciones móviles.
- Los lenguajes de .NET contienen el recolector de basura del CLR. Esto tiene resultados en el lenguaje para exceptuar instrucciones de destrucción de objetos.
- C# contiene procedimientos que aseguran que el acceso a tipos de datos sea correcto, lo cual, evita a que se originen errores cuando se accede a la memoria de algún objeto.
- En este lenguaje se han colocado algunas restricciones en el uso de las instrucciones de control más comunes.
- Todos los tipos de datos son objetos, debido a que todos se derivan de una clase base común llamada "*System.Object*".
- Admite definir tipos de datos a través de *estructuras*, tanto para los que se apliquen las mismas optimizaciones como para los tipos de datos básicos.
- C# define el significado de la mayoría de los operadores en el momento de aplicarlos a diferentes tipos de objetos.

- C# brinda la posibilidad de agregar a los metadatos del módulo resultante de la compilación de cualquier fuente información adicional a la generada por el compilador que luego podrá ser consultada en tiempo de ejecución a través de la librería de reflexión de .NET.
- Contiene una *política de versionado*, la cual, admite implantar nuevas versiones de tipos sin tenerle miedo a que la introducción de nuevos miembros inciten errores difíciles de detectar en tipos hijos previamente desarrollados, y ya extendidos con miembros de igual nombre a los anteriormente introducidos.
- Para asegurar la seguridad, el código no admite usar punteros, además tiene cuantiosas restricciones. Sin embargo se pueden saltar las anteriores restricciones manipulando objetos a través de punteros. Para lograr lo anterior, solo basta con marcar regiones de código como inseguras (modificador *unsafe*), obteniendo la utilización de punteros, lo que puede resultar importante en situaciones donde se necesite eficiencia y velocidad de procesamiento.
- Conserva una sintaxis análoga a C, C++ o Java; la cual, incluye en el código escrito fragmentos de código de estos lenguajes (facilitando la migración a los programadores). Por otra parte, el CLR ofrece a través de los llamados *Platform Invocation Services (PInvoke)*, la posibilidad de acceder a las DLL's de la API Win32. Se puede acceder con facilidad a este tipo de funciones, mediante la capacidad de utilizar punteros en código inseguro, ya que éstas en muchas oportunidades esperan recibir o devolver punteros.<sup>5</sup>

---

Disponible en Internet: <sup>5</sup>

[http://msdn.microsoft.com/library/default.asp?url=/library/enus/dv\\_vstechart/html/vbtchVisualCTechnicalArticles.asp](http://msdn.microsoft.com/library/default.asp?url=/library/enus/dv_vstechart/html/vbtchVisualCTechnicalArticles.asp)

## 2.4 SQL SERVER 2000

SQL Server 2000 brinda agilidad en operaciones de análisis y administración de datos. Ofrece destrezas para aprovechar oportunidades que presentan en el Internet, desde el aspecto de administración de datos. SQL Server 2000 es un sistema de gestión de base de datos y análisis de datos que se centra en el desarrollo rápido de una nueva versión de aplicaciones (incluye los programas para dispositivos móviles).

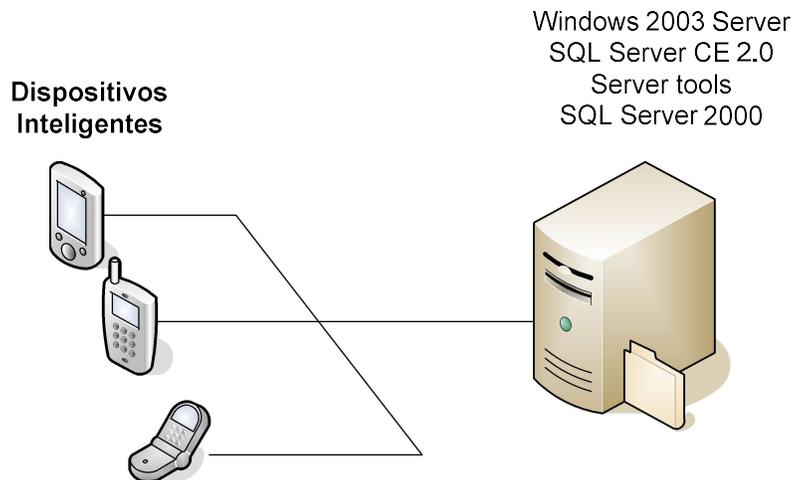
SQL Server 2000 perfecciona el rendimiento, la fiabilidad, la calidad y la facilidad de uso de SQL Server versión 7.0. El SQL Server 2000 incluye numerosas características que lo convierten en una atractiva plataforma para bases de datos de proceso transaccional en línea (OLTP), almacenamiento de datos, sincronización con dispositivos móviles y aplicaciones de comercio electrónico (ver figura 6).

También es una aplicación de administración de bases de datos enteramente dirigida hacia el Web (incluye páginas WAP), que suministra una importante compatibilidad con el lenguaje de marcado extensible (XML, Extensible Markup Language). Posibilita la realización de consultas utilizando el Internet, atravesando el servidor de seguridad (ver figura 7).

Los sistemas de multiproceso simétrico (SMP, Symmetrical Multiprocessor) son aprovechados por ésta aplicación. También incluye una fuerte administración de bases de datos, la carga de datos entre servidores y dispositivos móviles.

SQL Server 2000 tiene la opción de generar soluciones de análisis con herramientas integradas para introducirle valor a los datos. Además, puede producir procesos empresariales cimentados en los resultados del análisis y recuperar flexiblemente conjuntos de resultados personalizados de los cálculos más complejos.

**Figura 6. Conexión con la Base de Datos**



Otra característica, es la capacidad de optimizar y depurar consultas interactivamente, trasladar y transformar ágilmente datos provenientes de cualquier origen; definir y utilizar funciones como si estuvieran integradas en el Transact - SQL. Puede diseñar y codificar visualmente aplicaciones móviles de base de datos con cualquier herramienta de Visual Studio .NET. Utilizando ésta aplicación, se facilita la administración de bases de datos de forma centralizada.<sup>6</sup>

Anteriormente, SQL Server versión 7.0 utilizaba la función Servicios OLAP, ahora se llama Analysis Services de SQL Server 2000. Se ha sustituido el término Servicios OLAP por el término *Analysis Services*. Analysis Services contiene un componente de minería de datos.

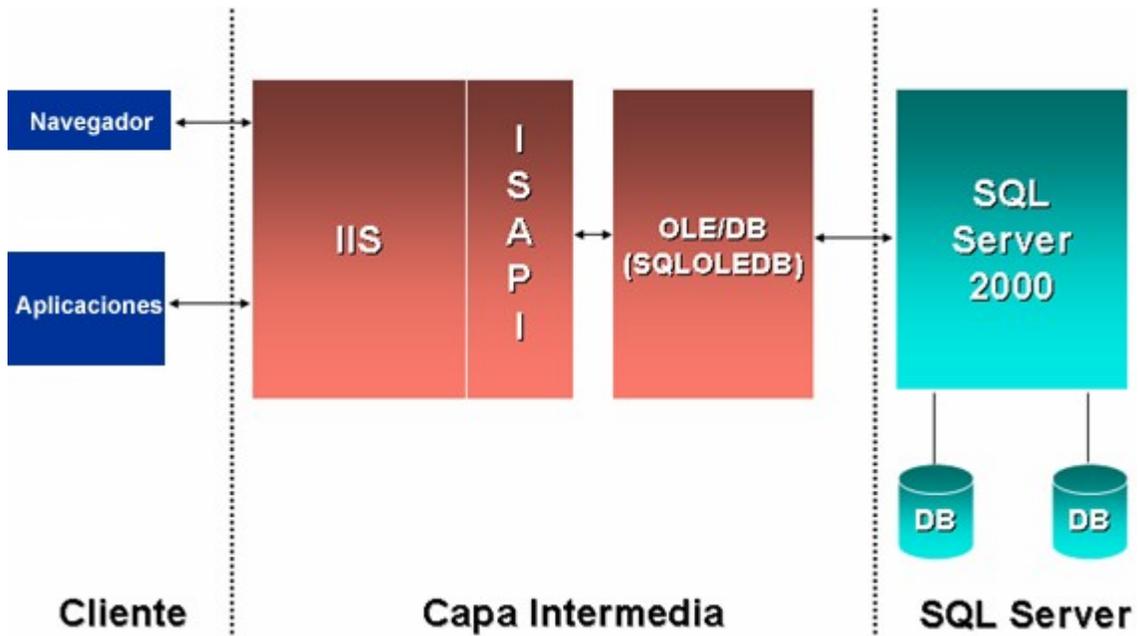
Una novedad es el Meta Data Services de SQL Server 2000, inicialmente en la versión de SQL Server versión 7.0 era el conocido componente de Repositorio. El término repositorio se usa en referencia al motor del repositorio incluido en Meta Data Services.

---

Disponible en Internet:

<sup>6</sup> <http://www.microsoft.com/sql/evaluation/overview/productguide.asp>

**Figura 7. Conexión con SQL Server**



Para trabajar con archivos de datos, las aplicaciones codifican la estructura específica de cada archivo de datos. Por otro lado, las bases de datos contienen un catálogo que las aplicaciones móviles pueden utilizar para determinar la organización de los datos. Las aplicaciones de bases de datos genéricas pueden utilizar el catálogo para presentar dinámicamente a los usuarios datos de distintas bases de datos, sin tener que depender de formatos de datos específicos.

Generalmente, una base de datos tiene dos partes principales: los archivos que almacenan la base de datos física y el software del sistema de administración de la base de datos (DBMS, Database Management System), que las aplicaciones utilizan para tener acceso a los datos. El DBMS es el responsable de mantener la estructura de la base de datos, el cual incluye, el mantenimiento de las relaciones entre los datos de la base de datos; la certeza de almacenar los datos correctamente y de no infringir reglas que definen las relaciones entre los datos; la recuperación de los datos hasta un punto coherente en caso de errores del sistema.

Cuando una aplicación móvil se conecta a una instancia de SQL Server 2000, puede hacer referencia a cualquiera de las bases de datos de esa instancia a la que el usuario tenga acceso. El componente de comunicación también admite la comunicación entre una instancia de SQL Server 2000 y una aplicación que se esté ejecutando en el mismo equipo. Puede ejecutar varias instancias de SQL Server 2000 en un único equipo.

SQL Server 2000 está diseñado para admitir el tráfico de los sitios Web. Aunque SQL Server 2000 está diseñado para funcionar como motor de almacenamiento de datos para miles de usuarios que se conectan a través de una red simultáneamente, puede funcionar también como base de datos independiente directamente en el mismo equipo de una aplicación.

Para trabajar con los datos desde una aplicación (Web, móvil y cliente / servidor) de una base de datos, debe utilizar un conjunto de comandos e instrucciones definidos por el software del DBMS.

## **2.5 SQL SERVER 2000 WINDOWS CE EDITION VERSIÓN 2.0 (SQL SERVER CE)**

SQL Server CE es una base de datos móvil compacta para el desarrollo rápido de aplicaciones en el .NET Compact Framework, que amplía las capacidades de administración de datos a los dispositivos, ideal para los ambientes móviles y wireless.

SQL Server CE 2.0 presenta las principales características de las bases de datos relacionales<sup>7</sup>, así como un analizador de consultas (QUERY) y una ayuda para las transacciones y los tipos de datos clasificados, mientras que administra los recursos del sistema.

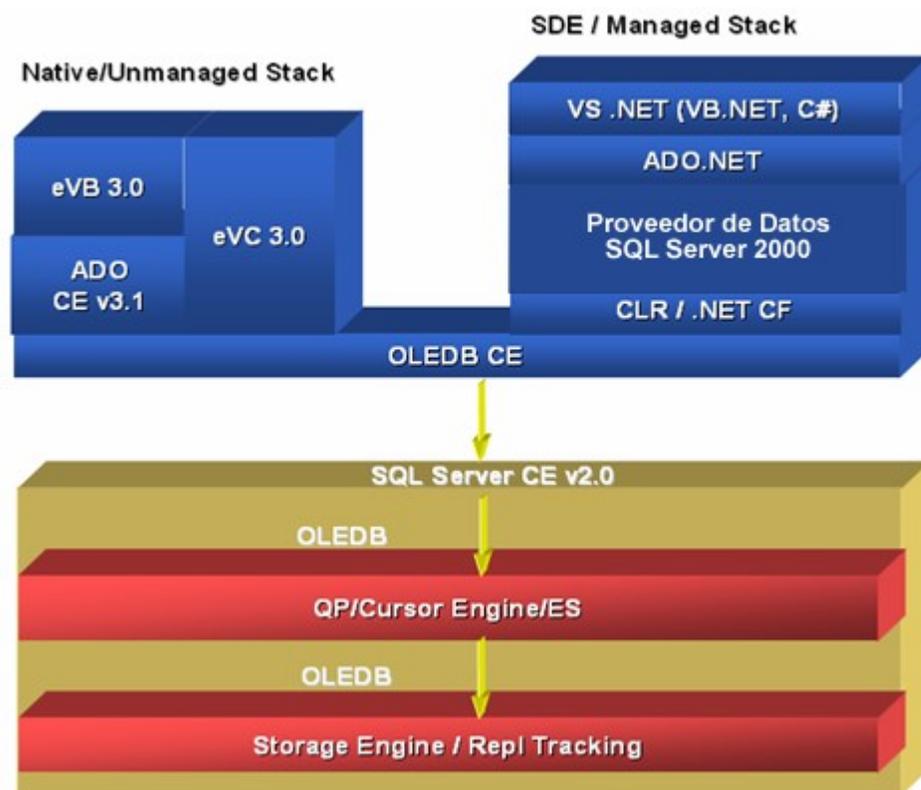
---

<sup>7</sup> <http://www.microsoft.com/sql/ce/productinfo/overview.asp>

Los datos remotos del SQL Server son entregados confiablemente. Se pueden utilizar offline y permite sincronizar más adelante con el servidor, haciendo que el SQL Server CE 2.0 pueda ser utilizado en ambientes móviles y wireless.

Esta nueva versión de SQL Server CE ofrece capacidades mejoradas de consultas y de motor de base de datos además de soluciones de conectividad mejoradas (ver figura 11). También, ofrece una base de datos sencilla apta a las aplicaciones móviles. Por otro lado cuenta con una librería dinámica para acceso a datos, la cual ofrece encriptación de 128 bits, acceso local y remoto, acceso flexible a datos; requiriendo solo 1 MB de espacio mínimo necesario<sup>8</sup>.

**Figura 8. Acceso a datos**



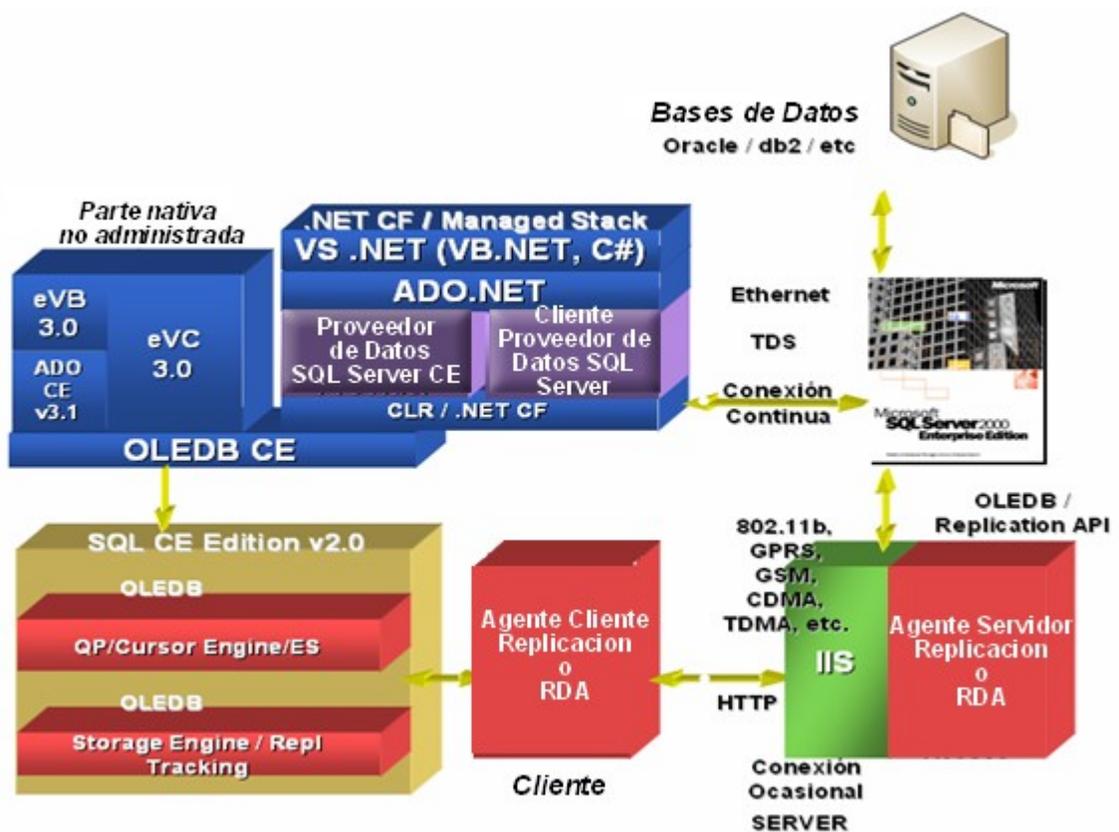
Fuente: Enterprise Solution Patterns Using Microsoft .NET, Microsoft Corporation, versión 2.0, 2002

<sup>8</sup> <http://msdn.microsoft.com/sql/sqlce/default.aspx?pull=/library/enus/dnppc2k2/html/ppcdatabase.asp>

Este tipo de bases de datos son diseñadas principalmente para movilidad, por eso permite manejar bases de datos desconectadas, las cuales suelen utilizar el Internet, Intranet y mantienen los datos actualizados por medio de la sincronización de las bases de datos.

El SQL CE es totalmente compatible con SQL Server y permite utilizar los mismos tipos de datos. En la tabla 1 podemos observar una comparación entre el SQL Server CE y SQL Server 2000.

**Figura 9. Arquitectura de almacenamiento**



Fuente: James Avery, Microsoft ASP.NET Setup and Configuration Pocket Reference, Microsoft Press 2003.

El SQL Server CE 2.0 se diseñó para poderse integrar con el .NET Compact Framework, Visual Studio .NET y Embedded Visual Tools.

**Tabla 2. SQL Server CE vs. SQL Server 2000**

SQL Server CE	SQL Server 2000
<ul style="list-style-type: none"><li>• Acceso exclusivo BD's</li><li>• Monousuario</li><li>• Comandos SQL sencillos</li><li>• Limitaciones BD's</li></ul>	<ul style="list-style-type: none"><li>• Múltiples Accesos</li><li>• Multiusuario</li><li>• Comandos SQL Complejos</li><li>• Soporte Completo</li></ul>

El SQL Server CE 2.0 amplía la frontera de la administración de datos entregando un entorno de base de datos para el rápido desarrollo. SQL Server 2000 proporciona soporte en la administración de datos y la programabilidad a través de los servidores conectados a estaciones de trabajo.

El SQL Server CE proporciona capacidades robustas de la administración de datos en los dispositivos móviles. También se asegura de que las organizaciones se pueden integrar con los sistemas existentes y de esta manera aprovechar las capacidades de desarrollo.

Debido los dispositivos están avanzando y los recursos de sistema son escasos, la base de datos compacta genera una funcionalidad esencial. El SQL Server CE utiliza la memoria lo menos posible y de manera eficaz.

El SQL Server CE 2.0 incluye actualizaciones en el *Query Analyzer*, el cual ha sido actualizado para mejorar la interfaz del usuario y generar una administración más fácil de los objetos; el *Query Processor*, que permite agilizar el código con nuevas funciones intrínsecas y consultas parametrizadas; el *Motor de almacenamiento*, incluye mejoras en la parte del acceso remoto de los datos; los *Asistentes de conectividad*, es una guía paso a paso para ayudar a instalar velozmente opciones de seguridad y de conectividad para la réplica de los datos. Finalmente, vemos

una mejora en el *ISQLW*; el cual es una herramienta de consulta en el dispositivo que admite almacenar consultas y ver el esquema en el dispositivo móvil.

## 2.6 CONSEJOS PARA EL DESARROLLO DE APLICACIONES MÓVILES

- **Conocer el entorno.** Es de carácter fundamental conocer el contexto al cual se desea aplicar una solución móvil, puesto que este da directrices de cómo debe funcionar la aplicación, es mejor descubrir por cuenta propia como se llevan a cabo los procesos que queremos optimizar y para esto hay que salirnos de nuestro campo e investigar, la mejor forma de realizar alguna operación en ciertos casos no proporciona la tecnología, esta no significa que se vaya hacer mejor. Esto nos permite que al momento de implantar algún cambio sea importante y determinante para las aspiraciones que tienen en mente.
- **Evaluar la plataforma o entorno de desarrollo a utilizar.** El desarrollo para dispositivos móviles es cada vez más fácil. Las herramientas de software convergen con las de escritorio. Hoy puedes usar Visual Studio 2003 y trabajar en C# o en VB.NET para dispositivos móviles. En Visual Studio 2005 lo podrás hacer también con C++.

La idea de que podemos empezar a trabajar en dispositivos móviles sin ningún esfuerzo no es cierta. Es necesario tomarse el tiempo para estudiar las diferencias entre la plataforma de escritorio y la móvil. Así diseñará mejor y se ahorra tiempo al final.

- **Sincronización de datos.** La gran mayoría de las aplicaciones móviles para empresas son extensiones de sistemas existentes. Aplicaciones de ventas, mantenimiento, servicio, etc. Todas deben conectarse a sistemas centrales y sincronizar la información.

Verifica como la vas a extraer y depositar en sistemas principales. Hay que determinar por medio de pruebas si los datos extraídos son los datos que realmente se necesita y si poseen todas sus caracterizas.

- **Diseño de una interfaz adecuada para dispositivos móviles.** Este punto parece sencillo pero no lo es, la costumbre a desarrollar interfaces para computadores de escritorio no nos permite ver esto con claridad, hay que tener en cuenta las limitaciones de un dispositivo móvil. Es indispensable pensar diferente cuando trabajamos en estas aplicaciones y una buena forma de hacerlo es intentar cuestionarnos de cómo cierta aplicación seria más fácil de usar.

Algunos lineamientos generales son: minimiza la captura, simplifica la navegación, estudia los controles gráficos especiales para PDA y muestra sólo la información realmente indispensable.

- **Soporte en aplicaciones móviles.** Actualizar dispositivos dispersos por todos lados no una tarea simple. Desarrolla un mecanismo automatizado de actualización y soporte o mejor aún, evalúa una opción existente que se adapte a las necesidades.
- **Pruebas.** Probar la aplicación sobre el dispositivo real las veces que se considere necesario, realizar pruebas tras prueba tratando de encontrar errores, no esperar a distribuir la aplicación para que los usuarios encuentren las fallas que no se descubrieron antes. El emulador sólo es aceptable en etapas iniciales de tu proyecto. Prueba la sincronización de datos simultánea conforme a tu carga máxima esperada.
- **Principio “simple”.** En esta clase de aplicaciones lo mas importante es la funcionalidad ya que en estos dispositivos necesitamos una respuesta rápida

mantener una simplicidad en la aplicación nos permite esta característica. Aplica el principio en todos los aspectos de la aplicación. Hay que cuestionarnos y unas buenas preguntas serian: ¿Realmente necesitamos capturar este dato? ¿Necesitamos desplegar esta información? ¿Es indispensable bajar este elemento del sistema principal? ¿Vale la pena automatizar este caso excepcional?

- **Limitaciones de memoria y velocidad de proceso.** Establece como requerimientos formales los volúmenes de información que debes manejar y los tiempos de respuesta máximos aceptables de tu aplicación para cada caso de uso. Separa tiempo en tu proyecto durante la etapa de análisis para hacer algunas pruebas de concepto que te permitan saber con certeza si es posible y de que forma. No asumas nada.
- **Escoger un dispositivo adecuado a las necesidades.** En este campo la variabilidad es mucho mayor que en el caso de PCs. Con o sin: Teclado, touch screen, wi-fi, bluetooth, cámara. Con diferentes capacidades de memoria, procesadores, opciones de expansión, etc. Lo mas importante es tratar de seleccionar un dispositivo al cual se le pueda integrar la aplicación que se desea cuando se desee.

## 2.7 J2ME

La herramienta de desarrollo java cuenta con una parte importante para el desarrollo en dispositivos móviles la cual es denominada MIDP que son un conjunto de APIs que generalmente se implementan sobre un dispositivo CLDC de configuración y conexión limitada, utilizando la plataforma Java 2 Platform Micro Edition (J2ME). MIDP, que proporciona la base para aplicaciones móviles, con altas capacidades gráficas, tiene las siguientes funciones:

- **Interfaz de usuario mejorada:** MIDP mejora la experiencia del usuario final con diversas mejoras para hacer las aplicaciones más interactivas y fáciles de utilizar. También proporciona una mayor extensibilidad y un diseño más flexible para incrementar la portabilidad de las aplicaciones a través de una amplia variedad de dispositivos con pantallas de diferentes tamaños.
- **Soporte multimedia:** MIDP permite aprovechar todas las capacidades de audio de los dispositivos, añadiendo nuevas funcionalidades de sonido como tonos, secuencias de tonos y archivos WAV.
- **Soporte de juegos:** MIDP añade un API de juegos que proporciona el aprovechamiento de las ventajas de las capacidades gráficas nativas de los dispositivos, para simplificar el desarrollo y proporcionar un mayor control de los gráficos y el rendimiento.
- **Conectividad mejorada:** MIDP añade soporte a los estándares de conectividad líderes más allá de HTTP, como por ejemplo HTTPS, datagramas, sockets y puertos serie de comunicaciones, lo que proporciona a las aplicaciones diferentes caminos para intercambiar datos.
- **Arquitectura Push:** MIDP incluye un modelo de servidor por medio del cual los MIDlets pueden ser registrados y activados cuando un dispositivo recibe información de un servidor.
- **Seguridad de principio a fin:** MIDP incorpora un robusto modelo de seguridad de principio a fin, desarrollado en estándares abiertos que protegen las redes, las aplicaciones y los dispositivos móviles. Soporta HTTPS y aprovecha los estándares existentes tales como SSL y WTLS para permitir la transmisión de datos cifrados.

Figura 10. Arquitectura Java

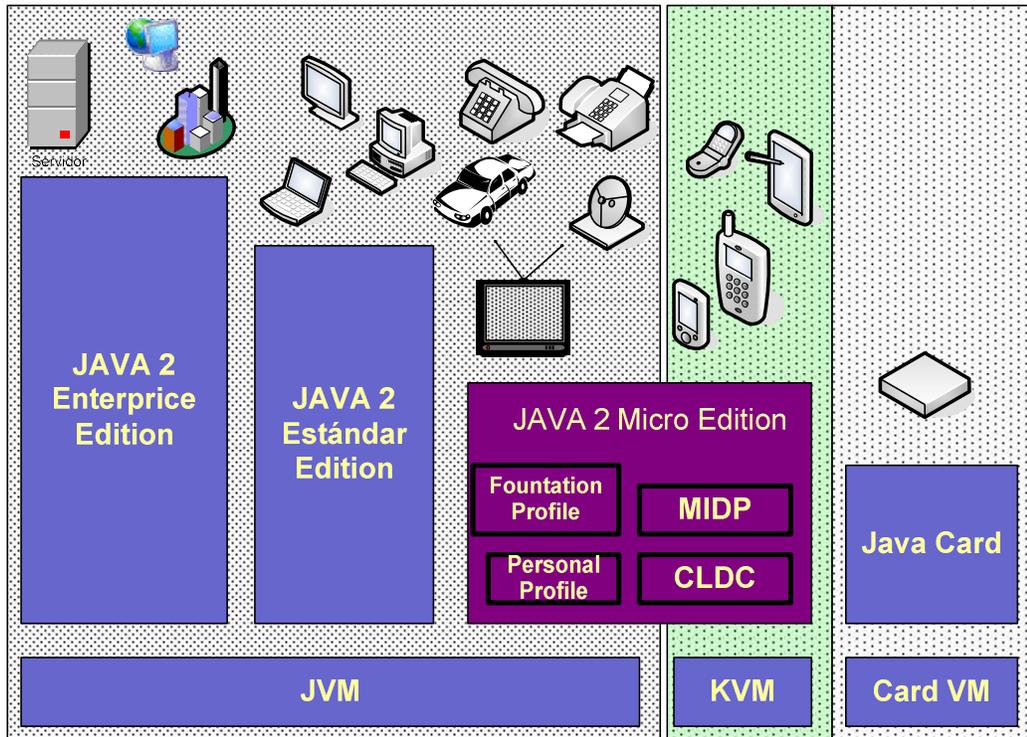
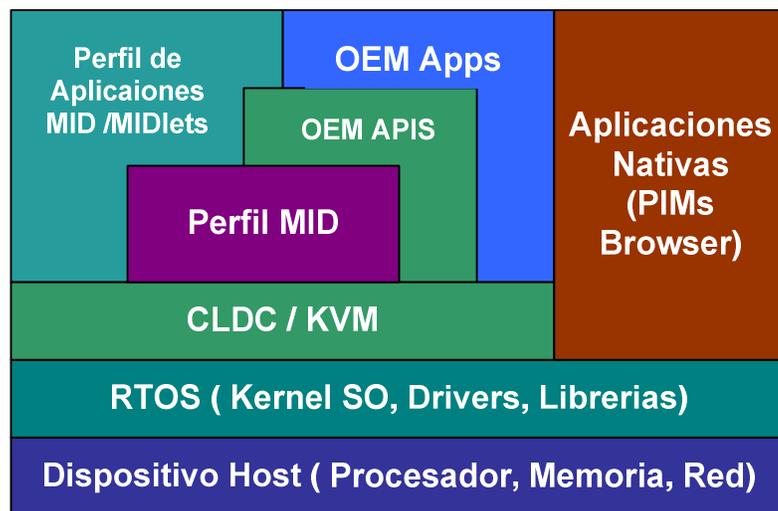


Figura 11. Arquitectura MIDP



### 2.7.1 Java apropiado para desarrollo en dispositivos móviles

Java aparte de presentar una arquitectura basada para el desarrollo de aplicaciones en dispositivos móviles también se complementa por que nos da la posibilidad de escribir una vez el programa y poder ejecutarlo en cualquier tipo de plataforma sin tener que recompilarlo de nuevo lo que se conoce con el nombre de WORA (Write Once, Run Anywhere), escríbelo una vez y ejecútalo en cualquier lugar. Hay otras razones que convierten a java en un candidato óptimo para el desarrollo de aplicaciones móviles y son: Extensión dinámica, Seguridad, Portabilidad, Fiabilidad y Código reutilizable.

Java cuenta con una edición específica para el desarrollo de aplicaciones para dispositivos embebidos y electrónicos que tiene características parecidas, estos dispositivos normalmente tienen una potencia limitada, posibilidad de conectividad a una red (normalmente sin cables) y poseen interfaces gráficos, esta se conoce como Micro Edition (J2ME).

### 2.7.2 Perfiles J2ME

Para ofrecer un completo entorno de ejecución específico para cada categoría de dispositivo, las configuraciones se deben combinar con un conjunto de API's de alto nivel, conocidas como perfiles, que definen el modelo de ciclo de vida de la aplicación, la interfaz de usuario y el acceso a las propiedades específicas del dispositivo. Algunos de los perfiles para dispositivos móviles existentes son:

- **Mobile Information Device Profile (MIDP)** Diseñado para teléfonos móviles y PDA's, incluye la interfaz de usuario, conectividad de red, almacenamiento local de datos y gestión de aplicaciones. Aporta un entorno de ejecución para Java que minimiza los consumos de memoria y de procesador.
- **PDA Profile (PDAP)** es similar al MIDP pero diseñado para PDA's que tengan mejores pantallas y más memoria de los teléfonos móviles.

2.7.2.1 MIDP – MIDlets. Las aplicaciones MIDP se denominan "MIDlets", las cuales pueden utilizar tanto las facilidades aportadas por MIDP como las APIs que MIDP hereda de CLDC, pero nunca acceden directamente al sistema operativo subyacente, por lo que no serían portables. Un MIDlet consiste en una clase Java, como mínimo, derivada de la clase abstracta MIDP, y que se ejecutan en un entorno de ejecución dentro de la máquina virtual, la cual provee un ciclo de vida bien definido controlado mediante métodos de la clase MIDlet que cada MIDlet debe implementar. También estas aplicaciones usan métodos de esta clase abstracta para conseguir servicios de su entorno. Un grupo de MIDlets que están relacionados se suelen agrupar en un MIDlet suite. Todos estos MIDlet se empaquetan, instalan, desinstalan y borran como una única entidad y comparten recursos tanto en tiempo de ejecución (se ejecutan en la misma máquina virtual, lo que implica que compartirán instancias de de todas las clases de Java cargadas en la máquina virtual), como estáticos (el almacenamiento persistente se gestiona en el nivel de suite).

Veamos seguidamente algunos detalles sobre los dispositivos que soportan MIDP. Comenzando con los requerimientos de memoria, MIDP necesita 128 KB de RAM disponible para la implementación correspondiente. A esta cantidad debemos sumarle la que necesita CLDC y como mínimo 32 Kbytes para almacenar la pila de la aplicación, tamaño que obliga al programador a tener bastante cuidado a la hora de diseñar las aplicaciones. Además, los dispositivos MIDP cuentan con 8 Kbytes como mínimo de memoria no volátil que se utiliza como almacenamiento persistente, que no se borra tras apagar el aparato (salvando el problema del cambio de batería).

Sobre las pantallas de los dispositivos, la especificación MIDP indica que ésta requiere 96 píxeles de ancho por 54 de alto y que debe soportar al menos dos colores (como es el caso de muchos teléfonos móviles, en contraposición con algunas PDA's que tienen pantallas de 160 píxeles en ambas direcciones y 65.536 colores diferentes).

Con respecto a los tipos de entradas del dispositivo, el rango es muy amplio: desde los que tienen un teclado alfanumérico completo, hasta aquellos que permiten escribir en ciertas áreas de la pantalla, pasando por los teclados de los teléfonos móviles. La especificación mínima requiere un teclado que permita marcar los números del 0 al 9, junto con el equivalente a las teclas del cursor y un botón de selección.

MIDP no asume que los dispositivos estén permanentemente conectados a una red, ni siquiera que soportan TCP/IP, pero que sí tienen algún tipo de acceso a una red. En este sentido, la especificación sí establece que soporte HTTP 1.1, bien mediante una pila de protocolos o una pasarela WAP.

También los sistemas operativos de los dispositivos tienen restricciones con respecto a MIDP. Por ejemplo, deben ofrecer un entorno de ejecución protegido donde la máquina virtual pueda correr, o algún tipo de apoyo para el acceso a una red, como puede ser el caso de un API para programar sockets, sobre el cual el protocolo HTTP se pueda implementar. Es el sistema operativo el que ofrecerá acceso al teclado y al posible dispositivo puntero, entregando los correspondientes eventos que surjan. Además, será el encargado de abstraer al MIDP la pantalla, ya que será visto por él como una matriz de píxeles, y de ofrecer un interfaz para el acceso al almacenamiento persistente.

Un aspecto muy importante a tener en cuenta es el de la seguridad. J2SE ofrece un modelo de seguridad potente y flexible, y a la vez, costoso en términos de memoria, razón por la cual CLDC y MIDP no incluyen ningún tipo de prueba en las llamadas al API de las que incluye J2SE. Para un usuario de un dispositivo móvil esto puede suponer un peligro, porque el MIDlet no está limitando en ningún sentido. Es por esto por lo que el usuario debería ser bastante cuidadoso a la hora de instalar nuevas aplicaciones.

Los MIDlets necesitan empaquetarse antes de que sean transferidos a un dispositivo para su instalación: tanto la subclase MIDlet correspondiente como las clases que requiera y el resto de archivos necesarios (como pueden ser archivos de imágenes) constituirán un único fichero JAR, incluyendo el conocido como manifiesto del JAR (contiene información de empaquetado que indica qué se almacena en el fichero JAR). Adicionalmente se emplea otro segundo fichero conocido como Java Application Descriptor (JAD). El manifiesto del JAR almacena el dispositivo, el nombre y la versión del MIDlet suite en el JAR correspondiente, así como qué archivos de clase se corresponde con cada MIDlet, información útil para instalar los MIDlets. El segundo es un fichero de texto que contiene una lista de atributos junto con su valor correspondiente. Algunos atributos están también contenidos en el manifiesto del JAR, ya que éste puede ser grande y su transferencia lenta debido a la baja velocidad del acceso a la red que suelen tener estos dispositivos, en vez de descargar el JAR completo, se descarga el fichero JAD, el cual es mucho más pequeño y rápido de transferir, se muestra por la pantalla del dispositivo y se decide si se instala o no.<sup>9</sup>

---

<sup>9</sup> [http://leo.ugr.es/~fjgc/INTRO/intro\\_8b.htm](http://leo.ugr.es/~fjgc/INTRO/intro_8b.htm)

### 3. ANÁLISIS DEL SISTEMA DE COMPARENDOS ACTUAL Y SUS MODIFICACIONES PARA ADAPTARLO A UNA ALTERNATIVA TECNOLÓGICA A TRAVÉS DE DISPOSITIVOS MÓVILES

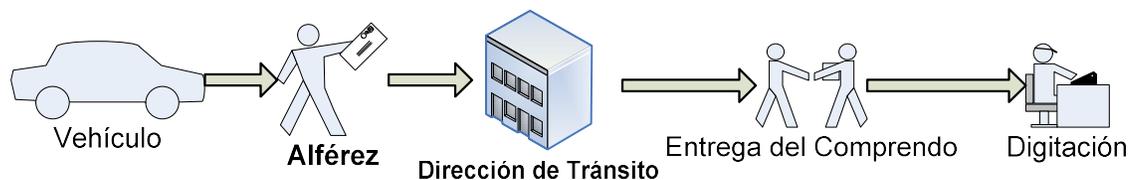
A continuación se planteara la información recopilada durante el estudio llevado a cabo al sistema de comparendos actual para determinar los procesos y mecanismos utilizados, como se plantea una mejora dando ventajas y desventajas a los cambios propuestos.

#### 3.1 PROCESO DEL SISTEMA DE COMPARENDOS ACTUAL

Tras varias visitas a la dirección de tránsito para determinar la funcionalidad del sistema de comparendos, se establecieron puntos claves para realizar modificaciones sin sufrir mayores cambios al momento de migrar a una nueva propuesta tecnológica.

El sistema de comparendos funciona de manera ordenada y básicamente su funcionamiento es el siguiente:

**Figura 12. Proceso actual del sistema de comparendos.**



Este sistema se comprende de varias etapas, de primera medida cuando se realiza un comparendo los datos de la persona infractora son registrados en un talonario o comparendera las cuales son llenadas en el momento de la infracción

manualmente por el alférez de tránsito ingresando alrededor de 30 datos correspondientes a datos personales, fecha, hora, códigos de la infracción etc.

Pasado este primer proceso se deja una copia al infractor y la otra es llevada a la dirección de tránsito donde hay una persona encargada de su recepción, luego su digitalización, para un ingreso en un sistema de base de datos donde se almacenara la información recopilada. En general estos son los procesos realizados tras la captura de un comparendo hasta su ingreso en una base de datos.

### **3.2 EVALUACIÓN DEL PROCESO DE SISTEMA DE COMPARENDOS**

Estos ítems muestran los pros y los contras significativos de este proceso pero no hay un punto de partida con el cual realizar comparaciones puesto que este proceso se maneja de igual manera en todo el país y pues tiende a tener más desventajas que ventajas.

#### **3.2.1 Ventajas**

- El proceso de dejar una copia del comparendo al infractor resulta realmente sencillo y eficaz.
- Costos a corto plazo son mucho menores. La inversión puede ser variable determinada por la cantidad de comparendos o procesos realizados en un lapso de tiempo.

#### **3.2.2 Desventajas**

- El tiempo en realizar un comparendo de esta forma es bastante largo.
- El tiempo en ser remitido a la dirección de tránsito es de acuerdo al horario del alférez y por consiguiente su ingreso al sistema de almacenamiento depende de este mismo horario.

- La integridad de los datos se ve afectada ya que el comparendo es llenado en forma manual así como su traspaso a la base de datos.
- Los costos de este proceso son elevados ya que se necesitan procesos manuales ineficaces.
- Se crea un cuello de botella al realizar cambios de turnos por la llegada masiva de alférez de tránsito a consignar los comparendos realizados durante su jornada.

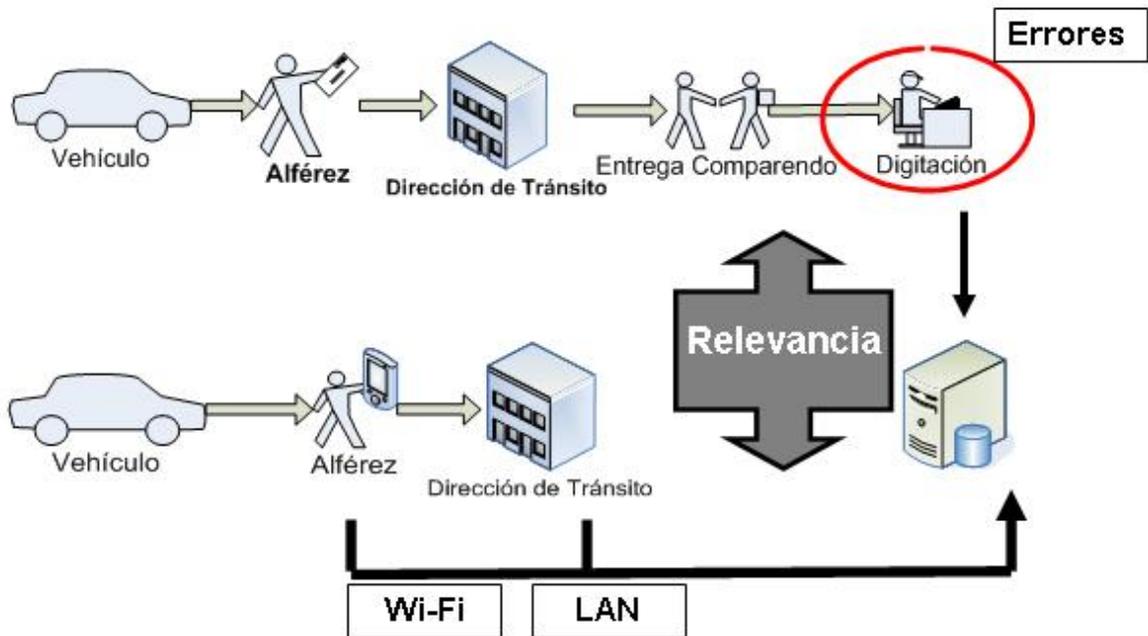
Como una apreciación especial es que los usuarios que desean consultar su estado de cuentas por infracciones de tránsito lo pueden hacer de dos maneras; una es simplemente dirigirse a la dirección de tránsito y solicitarlo o dos una consulta por medio de la página Web de la misma entidad.

### **3.3 PROPUESTA ALTERNATIVA SOBRE CAMBIOS EN EL PROCESO ACTUAL DEL SISTEMA DE COMPARENDOS**

En consecuencia tras haber realizado el análisis en el proceso actual de comparendos se llegó a plantear una alternativa académica que se adaptara más a las necesidades planteadas para el desarrollo del trabajo y que tuviera un efecto impactante al realizar los mismos procesos pero una manera rápida y donde se pudiera evidenciar que lo tecnológico nos sirve como herramienta para tener menos tiempo de respuesta y mayores beneficios en cuanto a integridad, calidad de los datos obtenidos de un comparendo y su versatilidad al momento de trasladar o migrar los comparendos a un sistema de base de datos.

El modelo planteado se representa de la siguiente manera:

**Figura 13. Modelo alternativo**



Según este modelo los principales cambios son, primero los datos son consignados en un dispositivo móvil precisamente una Pocket PC para mantener una integridad y orden mas eficiente, segundo los comparendos son sincronizados a la base de datos ahorrando el proceso de digitarlos reduciendo en tiempo y costos los procesos ya que se pasaría de minutos a segundos en esta operación.

### **3.3.1 Evaluación de la alternativa propuesta**

Esta alternativa cambia el esquema y plantea más ventajas que desventajas en comparación con el modelo actual y se ve un aumento de productividad en cuanto a tiempo lo cual significa disminución en los costos.

### **3.3.2 Ventajas**

- Se acortan los tiempos al momento de traspasar cantidad de información a la base de datos.
- Se maneja integridad en los datos
- Reducción de tiempo y costos.

- Escalabilidad y fácil mantenimiento
- Control de las aplicaciones cliente y servidor.

### **3.3.3 Desventajas**

- Para dejar una copia del comparendo al infractor se necesitaría un proceso de impresión.
- Inversión para tener una infraestructura tecnológica para que operen estos dispositivos.

En este desarrollo también se planteo realizar un proceso mas, el cual consiste en realizar las consultas del estado de cuentas por infracciones de transito sobre el mismo dispositivo móvil vía WAP dando así un nuevo elemento que complementa los ya existentes y permitiendo mayor capacidad en las consultas.

## **3.4 COMO HACER LAS MODIFICACIONES AL SISTEMA DE COMPARENDOS ACTUAL**

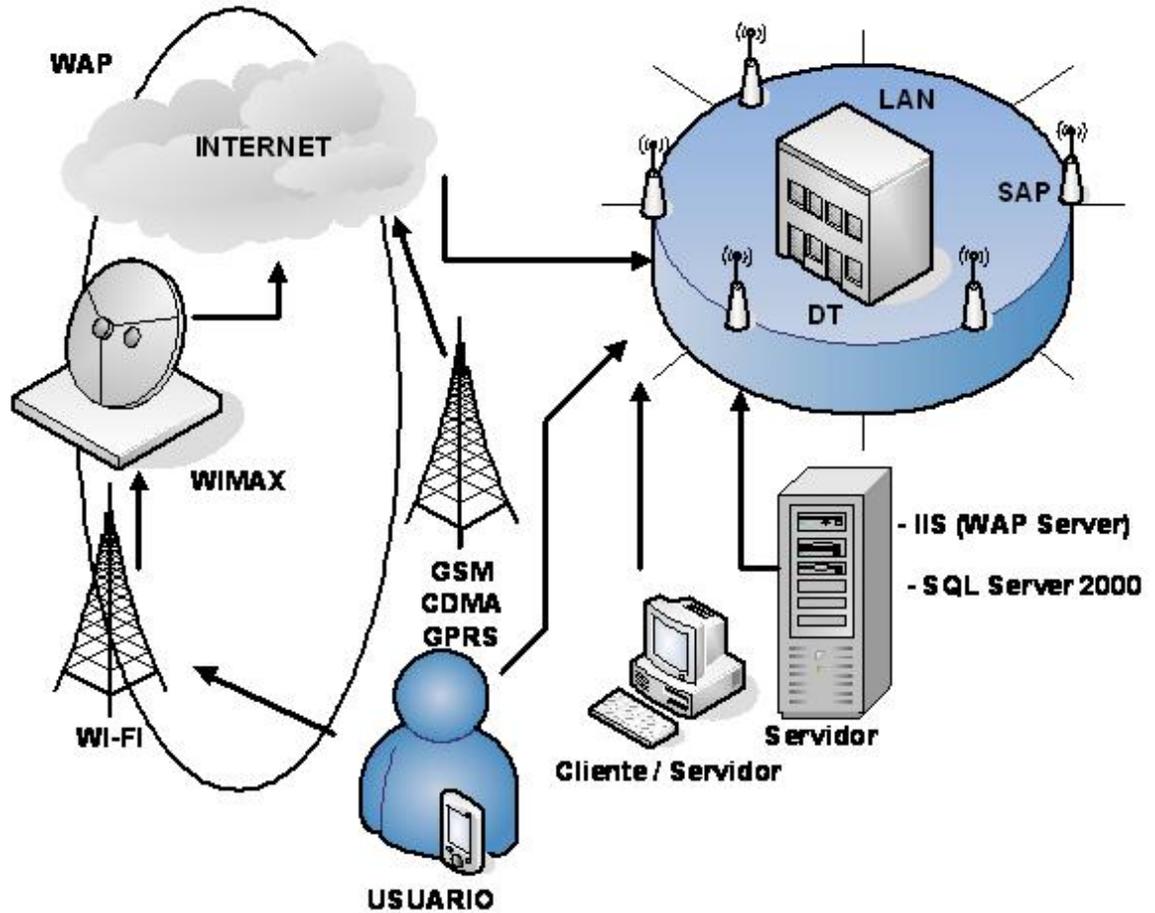
Siguiendo con lo establecido para realizar los cambios en el sistema de comparendos se plantean dos arquitecturas adaptada a los propósitos del proyecto.

En esta arquitectura lo que se pretende es evidenciar como es posible realizar las interacciones con el servidor utilizando varias tecnologías, por un lado se pueden realizar inalámbricamente y por otro lado se puede realizar por medio físico a cualquier punto de red (ver figura 14).

También esta la posibilidad que los datos sean ingresados por una aplicación la cual captura la información de los comparendos, que sean realizados de forma convencional y la cual esta conectada a la red LAN para su sincronización con el

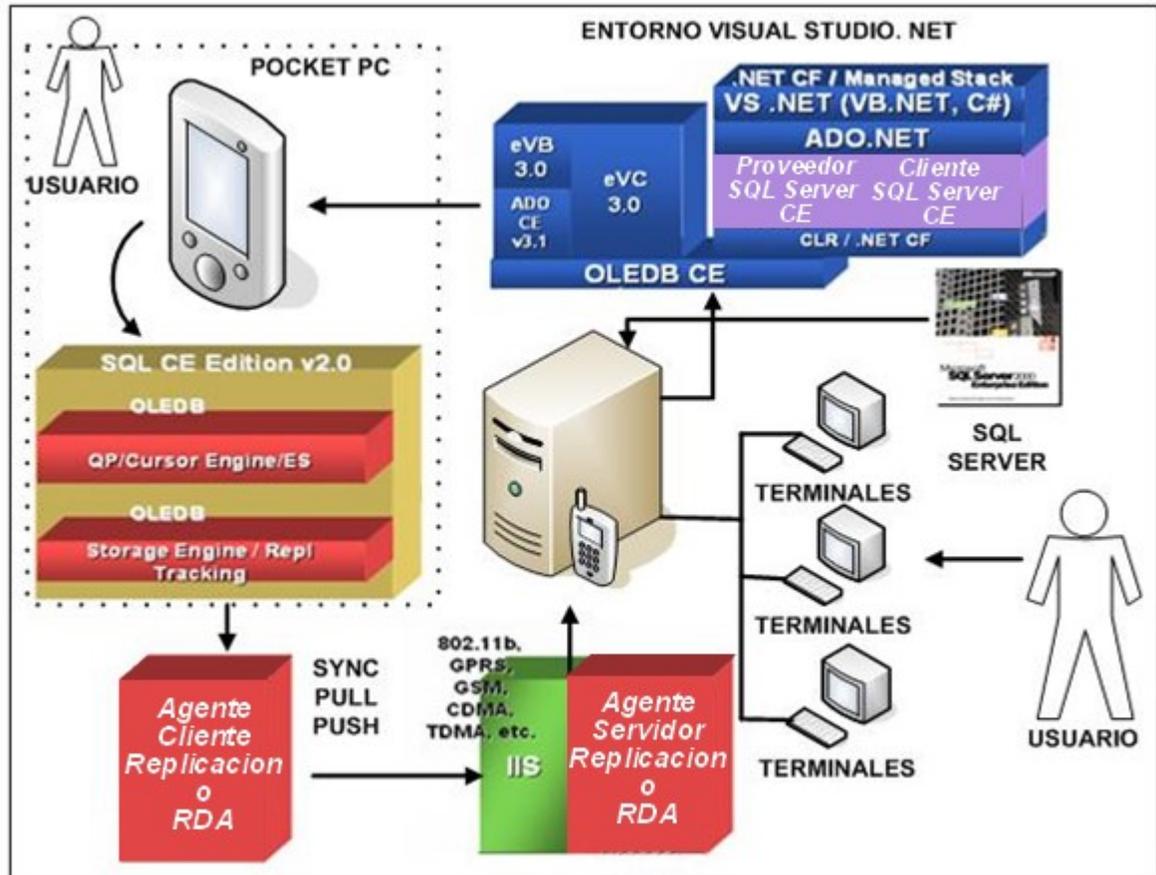
servidor. Si se cuenta con punto de acceso inalámbrico a la red LAN solo hay que sincronizar cuando se detecte el service access point.

**Figura 14 Arquitectura Externa**



Teniendo en cuenta el estudio preliminar de las tecnologías la arquitectura interna hace énfasis en los entornos de desarrollo que se utilizarán para el desarrollo de la aplicación y cuáles componentes dentro de estos entornos son utilizados para la consecución del prototipo de comprendos.

Figura 15 Arquitectura Interna



## **4. DESARROLLO DE LA APLICACIÓN**

### **4.1 MODELAMIENTO DEL SISTEMA**

Durante la investigación realizada, fue de vital importancia seguir paso a paso el diseño metodológico propuesto. En el modelamiento del prototipo software para la gestión de comparendos de tránsito, inicialmente nos centramos en recopilar información tanto del marco teórico como del estudio de las herramientas de desarrollo. También hicimos un proceso de investigación acerca de la forma como funciona la gestión de comparendos actualmente, para poder comprender el problema en contexto, y de ésta manera, buscarle una alternativa de solución académica.

Al haber analizado el problema a solucionar, proseguimos a modelar la información que teníamos a nuestro alcance utilizando el lenguaje de modelado unificado (UML). Con el modelamiento terminado, se continuó con el diseño del prototipo software. Cuando se obtuvo la estructura de la aplicación, empezamos a desarrollarla en Visual C#. Realizamos tres prototipos funcionales, el primero realiza consultas relacionado con el estado de cuenta de un vehículo por medio de una página WAP. El segundo se enfoca en la creación de un comparendo de tránsito desde un dispositivo móvil y en la sincronización de las bases de datos tanto del dispositivo móvil como del servidor. Finalmente, el tercer prototipo se encarga de gestionar la base de datos del servidor, desde una aplicación que opera sobre un sistema operativo Windows 2000 o superior.

### 4.1.1 Requerimientos

4.1.1.1 Casos de uso. El modelo de casos de uso constituye, especialmente en aquellos casos en que el contexto es totalmente ajeno al modelador, una contribución de peso en el análisis del contexto del dominio el cual, es fundamental para el diseño de un sistema. Un modelo de casos de uso plantea las «historias de uso» del sistema y se centran en qué debe hacer el sistema sin decidir cómo lo hará (el diseño). De esta forma, permite declarar los requisitos funcionales que debe contemplar este sistema (recogidos en análisis del dominio), lo cual ayuda a su vez a detectar los elementos (clases e interfaces) que van a formar parte de este diseño.

Los modelos de casos de uso se basan en la redacción –generalmente no exhaustiva – de los requisitos funcionales del sistema. Esta redacción se puede acompañar de un diagrama que muestra un conjunto de casos de uso, sus actores y sus relaciones y contribuye, entre otras cosas, a tener una imagen global del funcionamiento arquitectónicamente significativo del sistema.

Por otro lado, también se puede pensar en un caso de uso como aquél que proporciona un resultado observable de valor para un actor del sistema. La obtención de cada uno de estos elementos de valor para el usuario, implica la implementación de un mayor o menor número de acciones, cuya descripción puede ser recogida en un caso de uso antes de ser detallada en forma de colaboraciones entre clases (diseño).

En consecuencia, para capturar los requisitos del sistema, en vez de preguntarse directamente “¿cuáles son los casos de uso del sistema?”, conviene comenzar preguntándose “¿qué elementos de valor debe proporcionar el sistema?” o “¿cuáles son los objetivos del sistema?”. En este sentido, el nombre del caso de uso para un objetivo de usuario debería reflejar dicho objetivo, para resaltar este punto de vista.

4.1.1.2 Casos de uso prototipo de Sistema de Comparendos. A continuación podemos observar el diagrama de casos de uso correspondiente al prototipo del sistema de comparendos (ver figura 16).

Este diagrama se compone de diez casos de uso y tres actores (ver tabla 3).

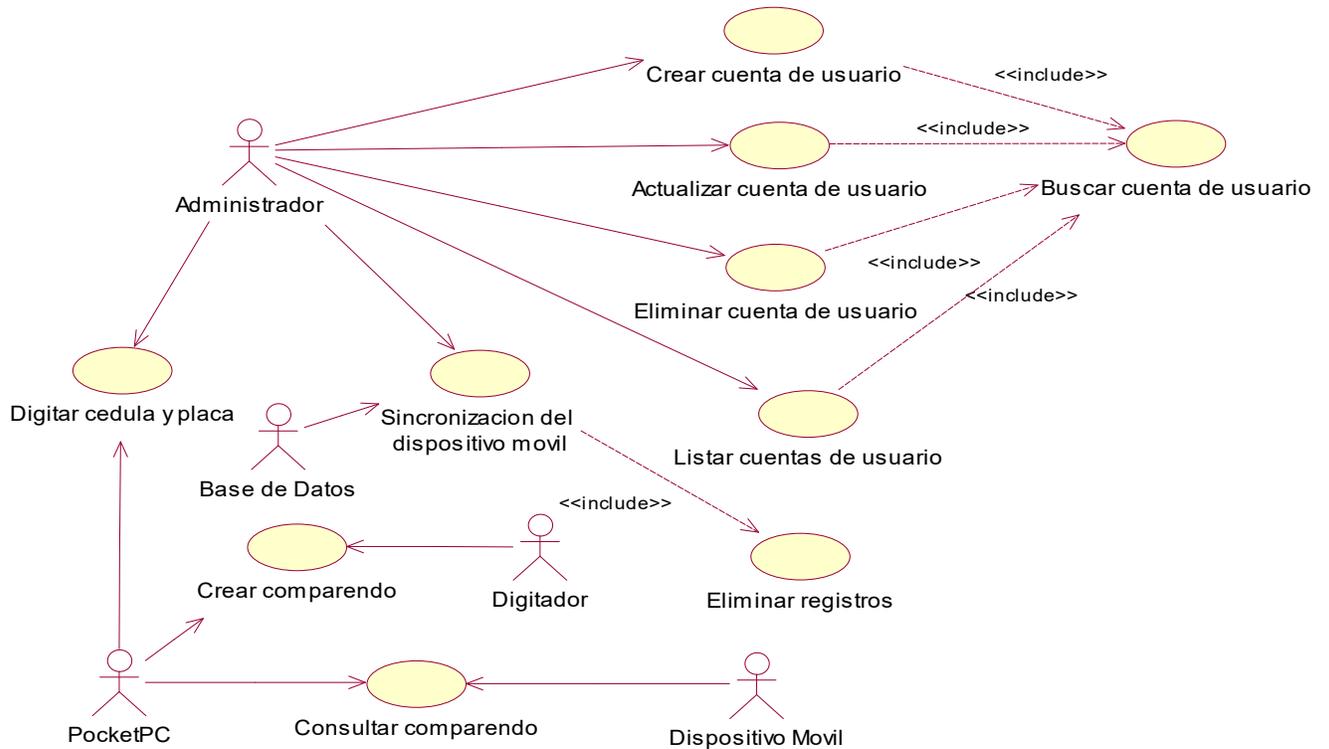
El sistema tiene tres partes importantes: gestión del administrador, gestión del alférez de tránsito y consulta por parte del usuario particular.

La primera, el administrador inicia al sistema en el instante que digita su cedula y placa en el dispositivo para sincronizar; o accede al servidor por medio de una aplicación de Windows que le permite crear, actualizar, eliminar y listar cuentas de usuario.

La segunda parte, el alférez de tránsito inicia al sistema digitando su cedula de ciudadanía y placa de identificación en el dispositivo; y tiene la opción de crear nuevos comparendos o consultarlos desde el dispositivo inteligente.

Por último la tercera parte, el usuario particular ingresa a una página WAP desde un dispositivo inteligente para consultar el estado de cuenta de su vehículo.

**Figura 16. Diagrama de Casos de Uso del Sistema**



**Tabla 3. Casos de Uso y Actores**

Casos de uso	Actores
<ul style="list-style-type: none"> <li>• Crear Cuenta de Usuario</li> <li>• Actualizar Cuenta de Usuario</li> <li>• Eliminar Cuenta de Usuario</li> <li>• Listar Cuentas de Usuario</li> <li>• Buscar Cuenta de Usuario</li> <li>• Eliminar Registros</li> <li>• Digitar Cedula y Placa</li> <li>• Sincronización del Dispositivo Móvil</li> <li>• Crear Comparendo</li> <li>• Consultar Comparendo</li> </ul>	<ul style="list-style-type: none"> <li>• Particular</li> <li>• Alférez de tránsito</li> <li>• Administrador</li> </ul>

### 4.1.2 Análisis

Se hizo un proceso de investigación acerca de la gestión de comparendos de tránsito actualmente en el área metropolitana de Bucaramanga, comprendiendo el problema desde una perspectiva global, y de ésta manera, proponer una alternativa de solución académica.

4.1.2.1 Diagrama de clases. Un diagrama de clases muestra el conjunto de clases y objetos importantes que forman parte de un sistema, junto con las relaciones existentes entre clases y objetos.

Una clase se compone de: nombre de la clase, atributos y métodos.

- **Atributo:** Es la representación de las propiedades o características de una entidad.
- **Método:** Es el conjunto de operaciones que describen el comportamiento de los objetos de una clase.

4.1.2.2 Diagrama de clases prototipo de sistema de comparendos. Podemos observar el diagrama de clases correspondiente al prototipo del sistema de comparendos (ver figura 16).

4.1.2.3 Diagrama de Secuencias. Muestra la interacción de un conjunto de objetos de una aplicación a través del tiempo, esta descripción deja ver a detalle los casos de uso, aclarándolos al nivel de mensajes de los objetos existentes, como también muestra las operaciones realizadas por un actor.

4.1.2.4 Diagramas de Secuencias prototipo de Sistema de Comparendos.

- **Diagrama de Secuencias de Actualizar Cuenta de Usuario.** Se puede observar el diagrama de secuencias de Actualizar Cuenta de Usuario

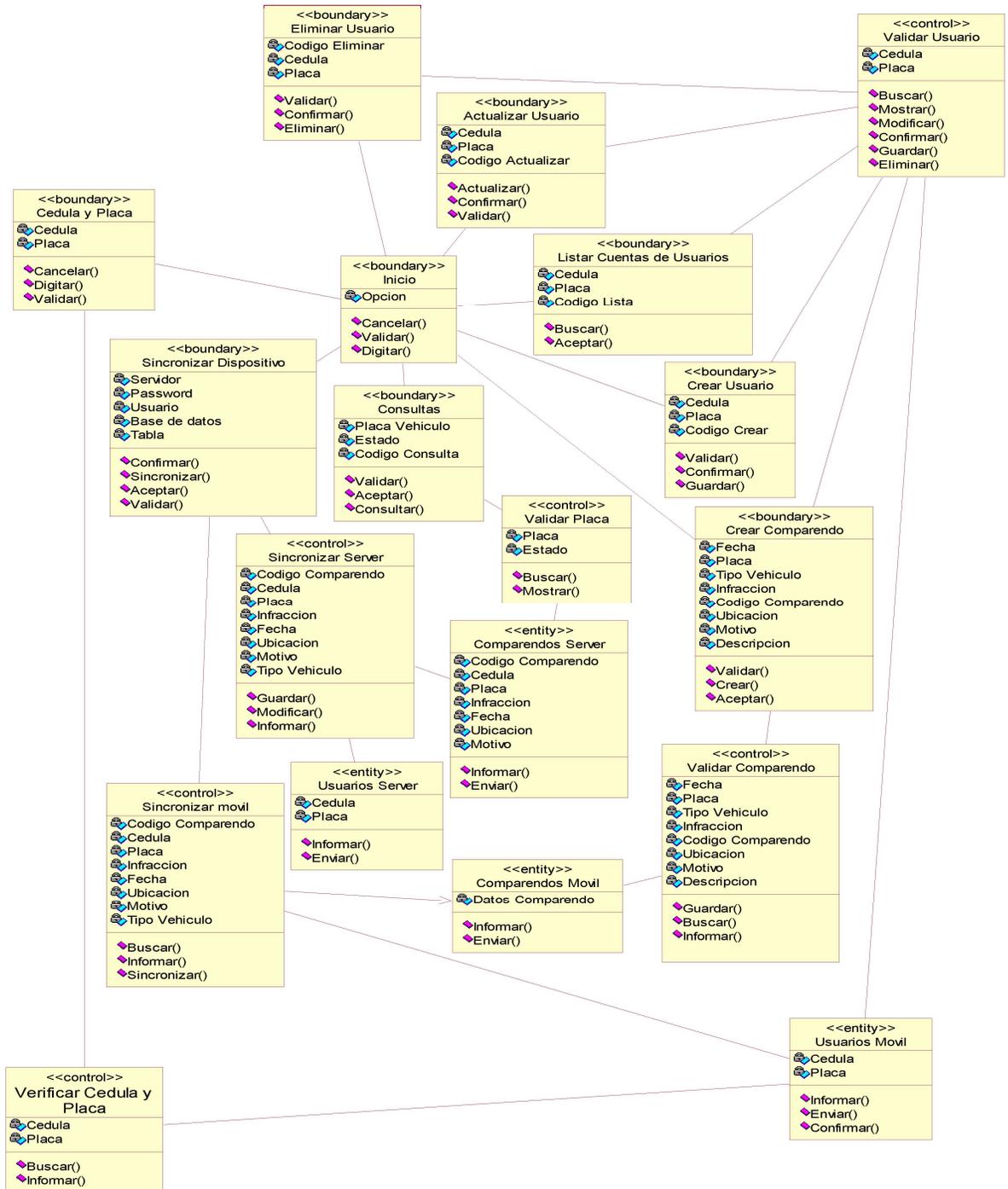
correspondiente al prototipo del sistema de comparendos (ver figura 17). En este diagrama el administrador ingresa la cedula y placa para ingresar al sistema, luego selecciona la opción Actualizar usuario y se escoge el usuario a modificar, se hacen las respectivas modificaciones y después se actualiza. El sistema modifica la información y envía una notificación al administrador.

- **Diagrama de Secuencias de Crear Cuenta de Usuario.** Podemos observar el diagrama de secuencias de Crear Cuenta de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 18). En este diagrama el administrador ingresa la cedula y placa para ingresar al sistema, luego selecciona la opción Crear usuario, se ingresan los datos del nuevo usuario y después se guarda la información.
- **Diagrama de Secuencias de Consultar Comparendo.** Podemos observar el diagrama de secuencias de Consultar Comparendo correspondiente al prototipo del sistema de comparendos (ver figura 19). En este diagrama el usuario particular abre una pagina WAP e ingresa la placa del vehiculo del cual se busca el estado de la cuenta, el sistema da una respuesta a la búsqueda.
- **Diagrama de Secuencias de Crear Comparendo.** Podemos observar el diagrama de secuencias de Crear Comparendo correspondiente al prototipo del sistema de comparendos (ver figura 20). En este diagrama el usuario alférez de transito ingresa la cedula y placa para ingresar al sistema, luego selecciona la opción Crear Comparendo y se ingresan los datos del nuevo comparendo, finalmente se guarda la información. El sistema guarda la información y envía un mensaje de aceptación al usuario alférez.
- **Diagrama de Secuencias de Digital Cedula y Placa.** Podemos observar el diagrama de secuencias de Digital cedula y placa correspondiente al prototipo del sistema de comparendos (ver figura 21). En este diagrama tanto el usuario

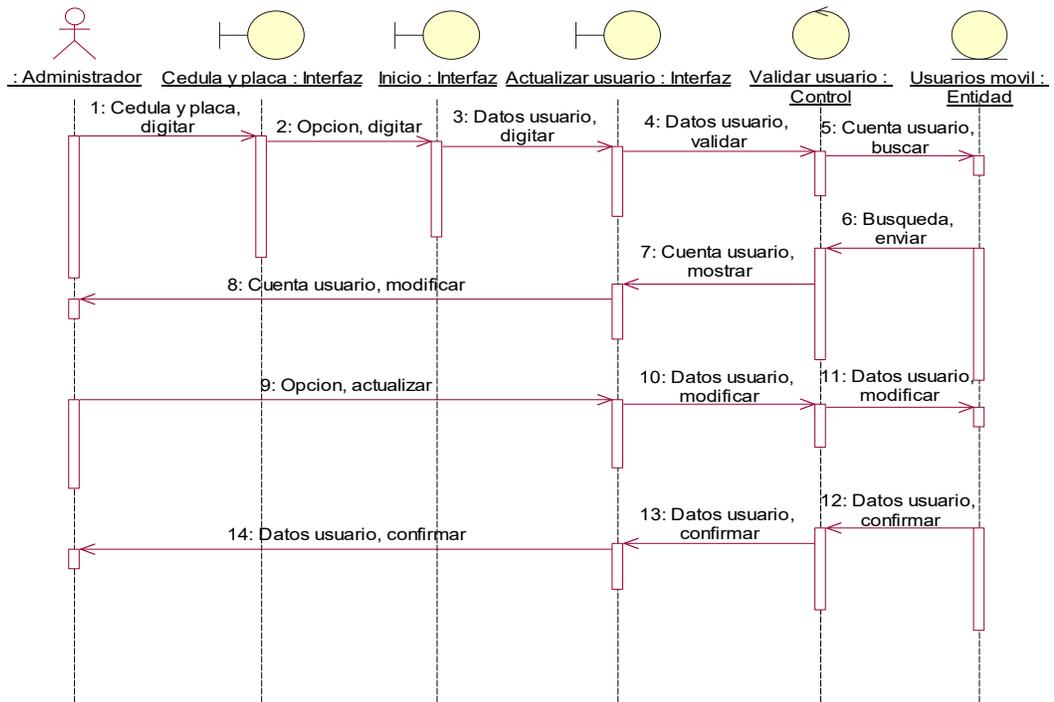
alférez de tránsito como el administrador, ingresan la cédula y placa para ingresar al sistema, el cual los envía a la interfaz de inicio.

- **Diagrama de Secuencias de Eliminar Cuenta de Usuario.** Se puede observar el diagrama de secuencias de Eliminar Cuenta de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 22). En este diagrama el administrador ingresa la cédula y placa para ingresar al sistema, luego selecciona la opción Eliminar usuario, se escoge el usuario a eliminar.

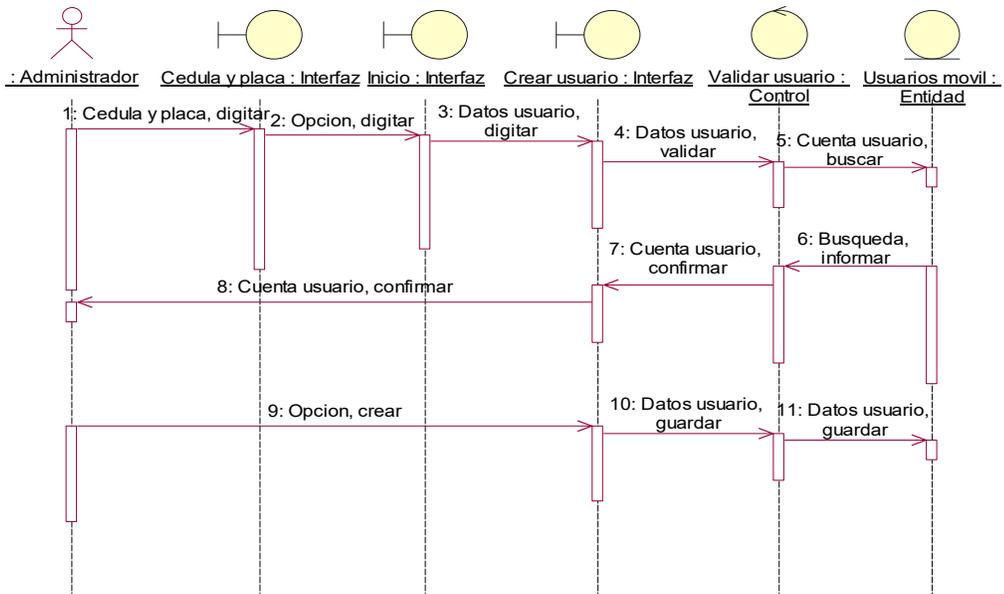
Figura 17. Diagrama de Clases del Sistema



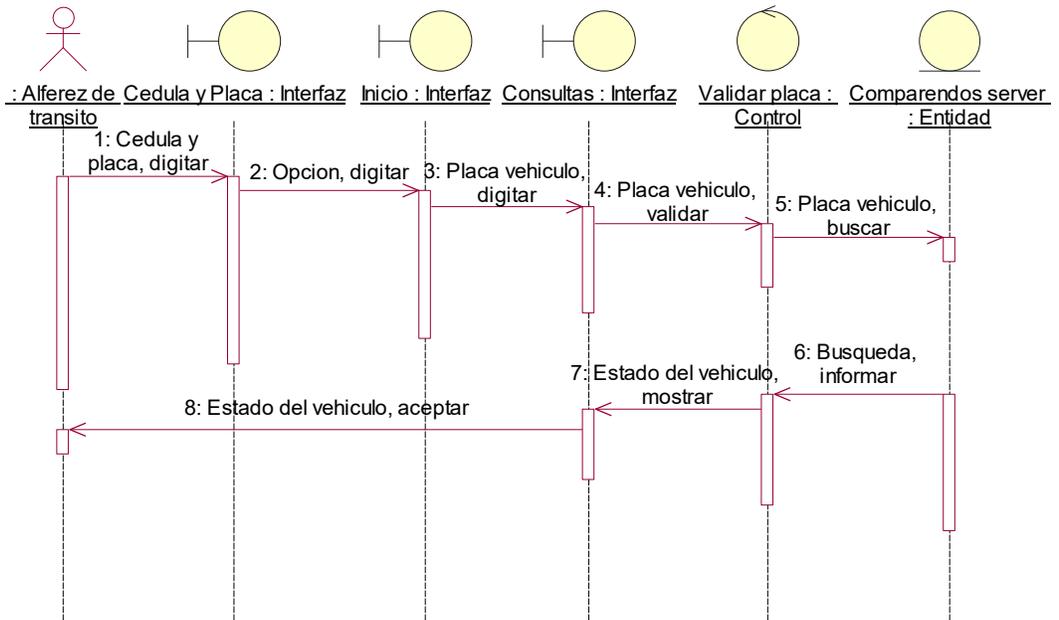
**Figura 18. Diagrama de Secuencia Actualizar Cuenta Usuario**



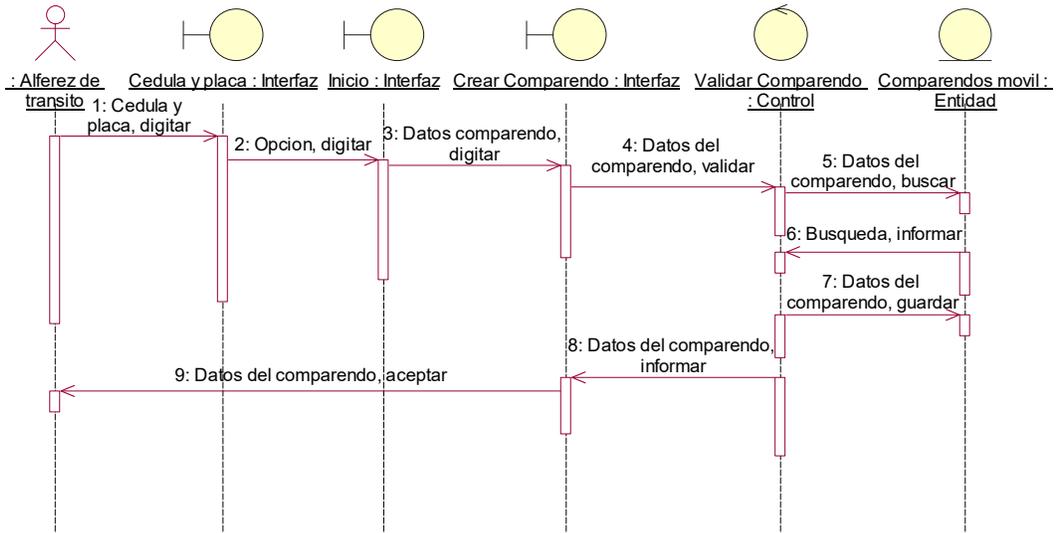
**Figura 19. Diagrama de Secuencia Crear Cuenta Usuario**



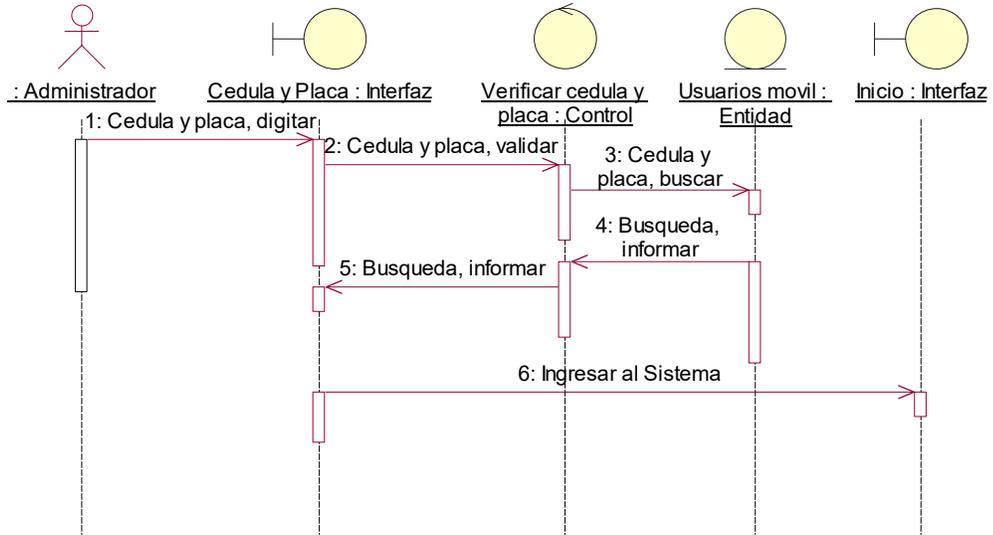
**Figura 20. Diagrama de Secuencia Consultar Comparendo**



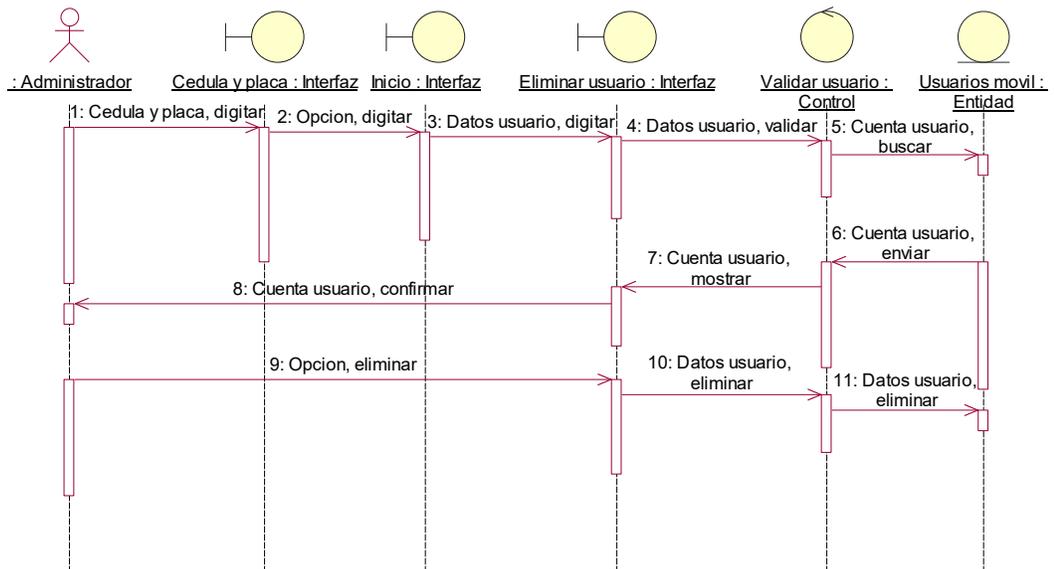
**Figura 21. Diagrama de Secuencia Crear Comparendo**



**Figura 22. Diagrama de Secuencia de Digitar Cedula y Placa**

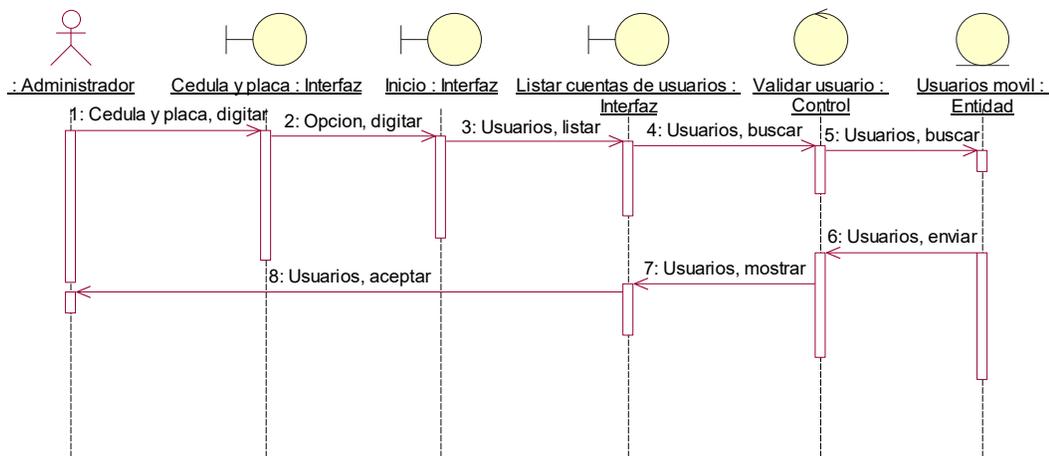


**Figura 23. Diagrama de Secuencia de Eliminar Cuenta de Usuario**



- **Diagrama de Secuencias de Listar Cuentas de Usuario.** Se puede observar el diagrama de secuencias de Listar Cuentas de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 23). En este diagrama el administrador ingresa la cedula y placa para ingresar al sistema, luego selecciona la opción Listar usuarios, el sistema muestra en pantalla los usuarios registrados en la base de datos.

**Figura 24. Diagrama de Secuencia de Listar Cuentas de Usuario**



- **Diagrama de Secuencias de Sincronización del Dispositivo Móvil.** Se puede observar el diagrama de secuencias de Sincronización del Dispositivo Móvil correspondiente al prototipo del sistema de comparendos (ver figura 24). En este diagrama el administrador ingresa la cedula y placa para ingresar al sistema desde el dispositivo, luego selecciona la opción Sincronizar bases de datos.

El sistema busca tanto la base de datos de usuarios como la de comparendos en el dispositivo y los envía a la base de datos del servidor central. Antes de sincronizar el sistema exige una confirmación para empezar con el proceso. Finalmente, el sistema envía un mensaje de aceptación al usuario administrador notificando que el procedimiento ha concluido satisfactoriamente.

4.1.2.5 Diagramas de Colaboración. Presenta una alternativa al diagrama de secuencia para modelar interacciones entre objetos en el sistema. Mientras que el diagrama de secuencia se centra en la secuencia cronológica del escenario que estamos modelando, el diagrama de colaboración se centra en estudiar todos los efectos de un objeto dado durante un escenario.

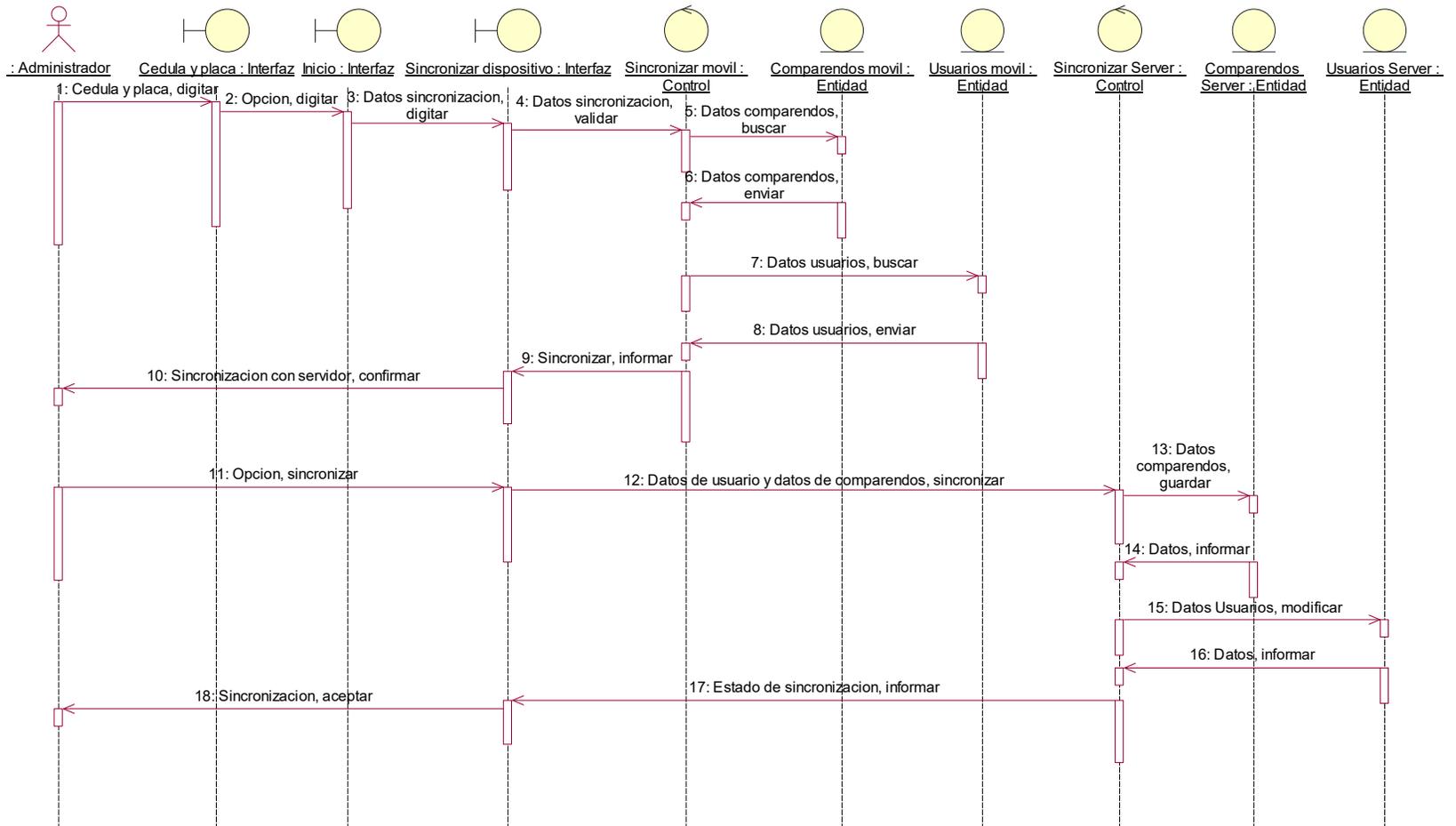
Los objetos se conectan por medio de enlaces, cada enlace representa una instancia de una asociación entre las clases implicadas. El enlace muestra los mensajes enviados entre los objetos, el tipo de mensaje (sincrónico, asincrónico, simple, blanking, y 'time-out'), y la visibilidad de un objeto con respecto a los otros.

4.1.2.6 Diagramas de Colaboración prototipo de Sistema de Comparendos.

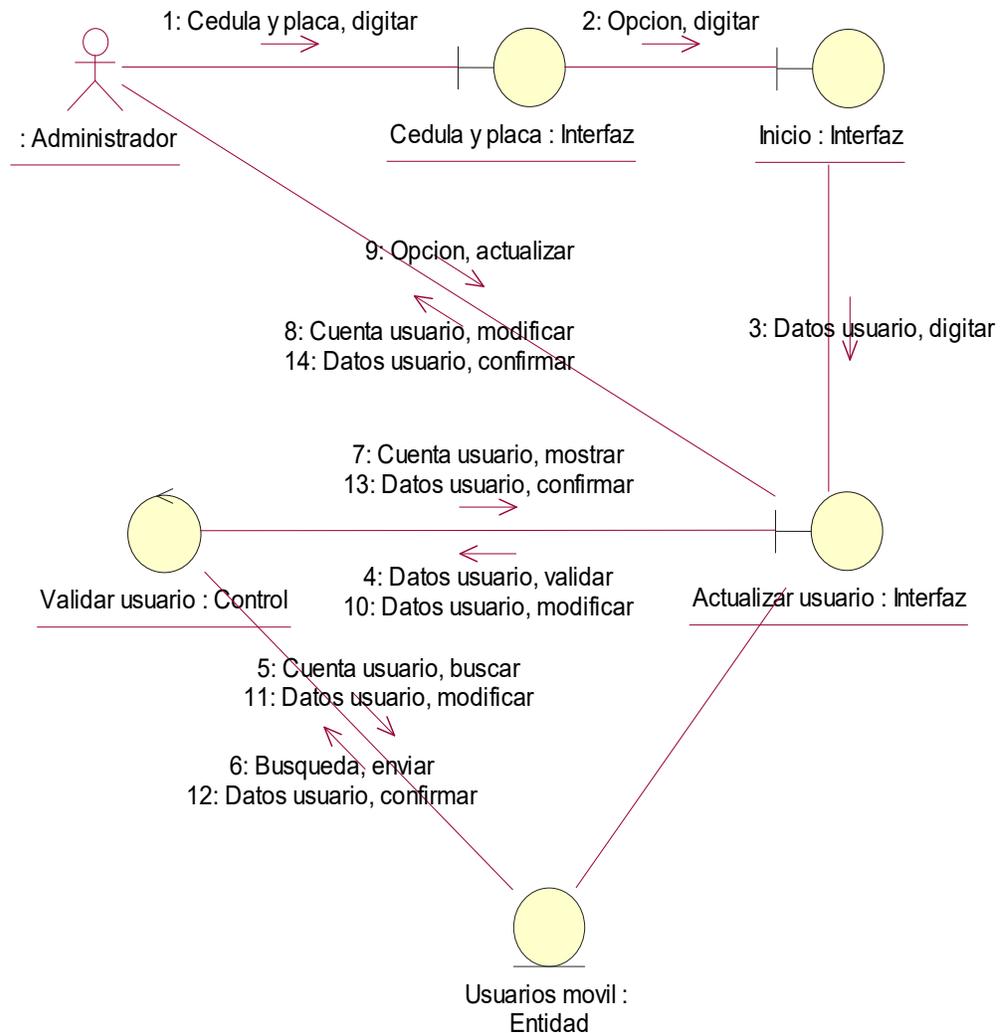
- **Diagrama de Colaboración de Actualizar Cuenta de Usuario.** Se puede observar el diagrama de colaboración de Actualizar Cuenta de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 25).

En este diagrama se puede apreciar paso a paso lo que el administrador debe hacer para poder modificar un usuario.

**Figura 25. Diagrama de Secuencia de Sincronización del Dispositivo Móvil**

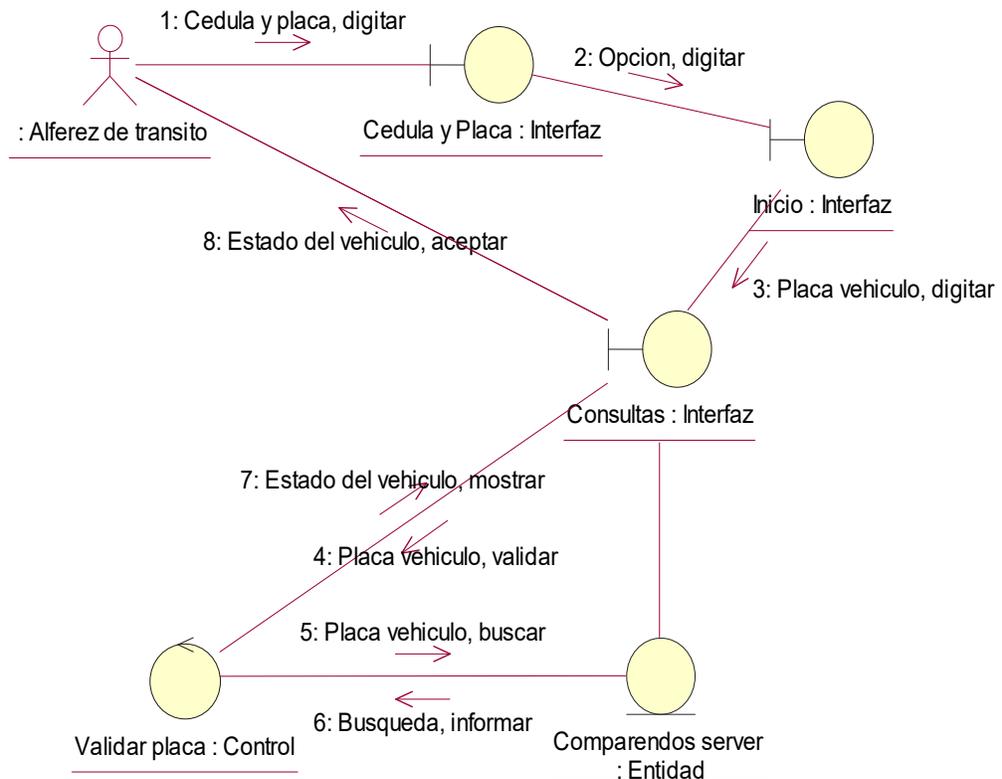


**Figura 26. Diagrama de Colaboración de Actualizar Cuenta de Usuario**



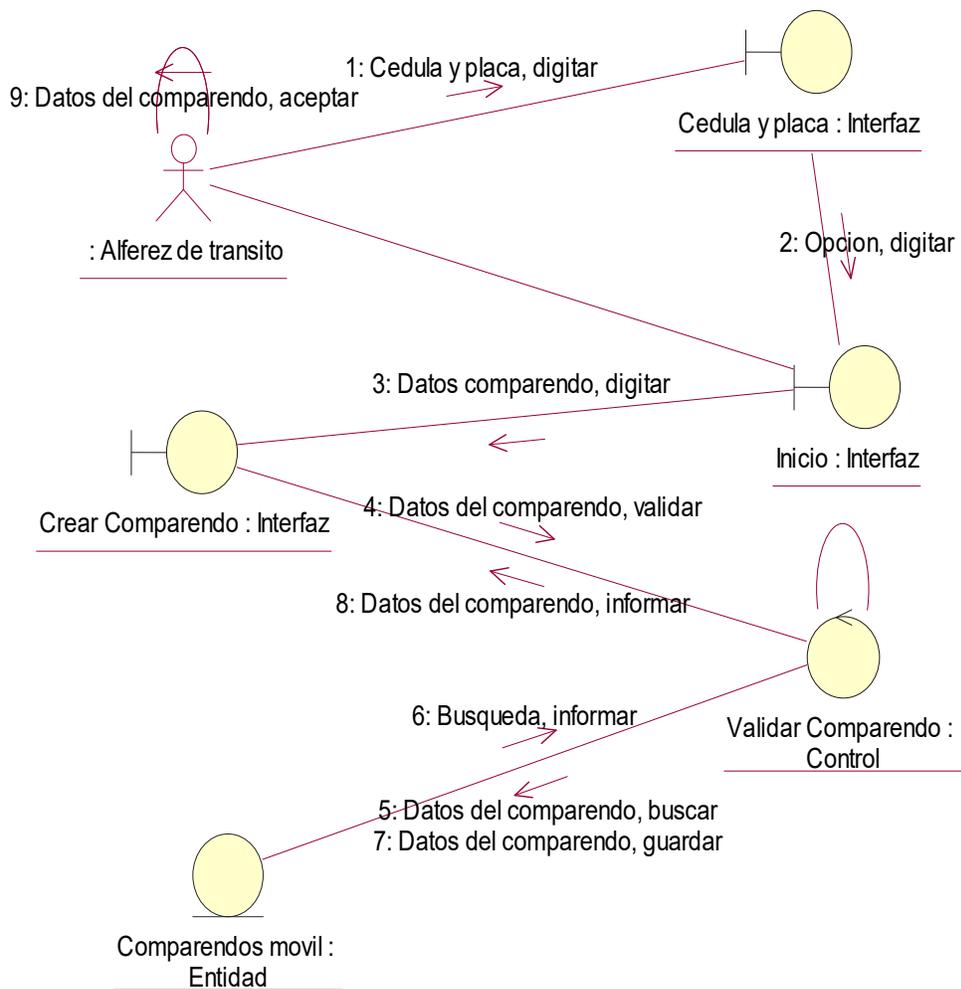
- **Diagrama de Colaboración de Consultar Comparendo.** Se observa el diagrama de colaboración de Consultar Comparendo correspondiente al prototipo del sistema de comparendos (ver figura 26). En este diagrama se puede apreciar paso a paso e lo que el usuario alférez de tránsito o el particular deben hacer para poder consultar el estado de cuenta de un vehículo.

**Figura 27. Diagrama de Colaboración de Consultar Comparando**



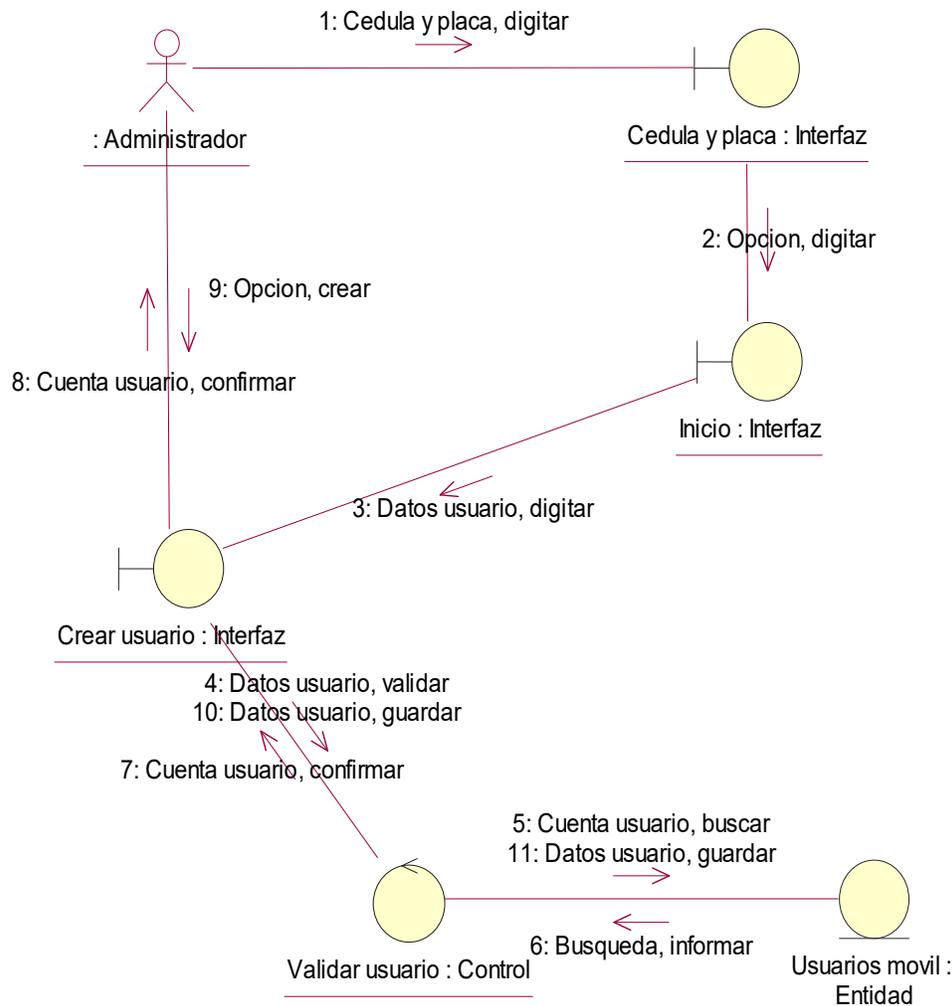
- **Diagrama de Colaboración de Crear Comparendo.** Se observa el diagrama de colaboración de Crear Comparendo correspondiente al prototipo del sistema de comparendos (ver figura 27). En este diagrama se puede observar detalladamente lo que el usuario alférez de tránsito debe hacer para poder registrar un nuevo comparendo de un vehículo.
- **Diagrama de Colaboración de Crear Cuenta de Usuario.** Se observa el diagrama de colaboración de Crear Cuenta de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 28). En este diagrama podemos observar paso a paso lo que el usuario administrador debe hacer para poder Crear un usuario nuevo.

**Figura 28. Diagrama de Colaboración de Crear Comparendo**



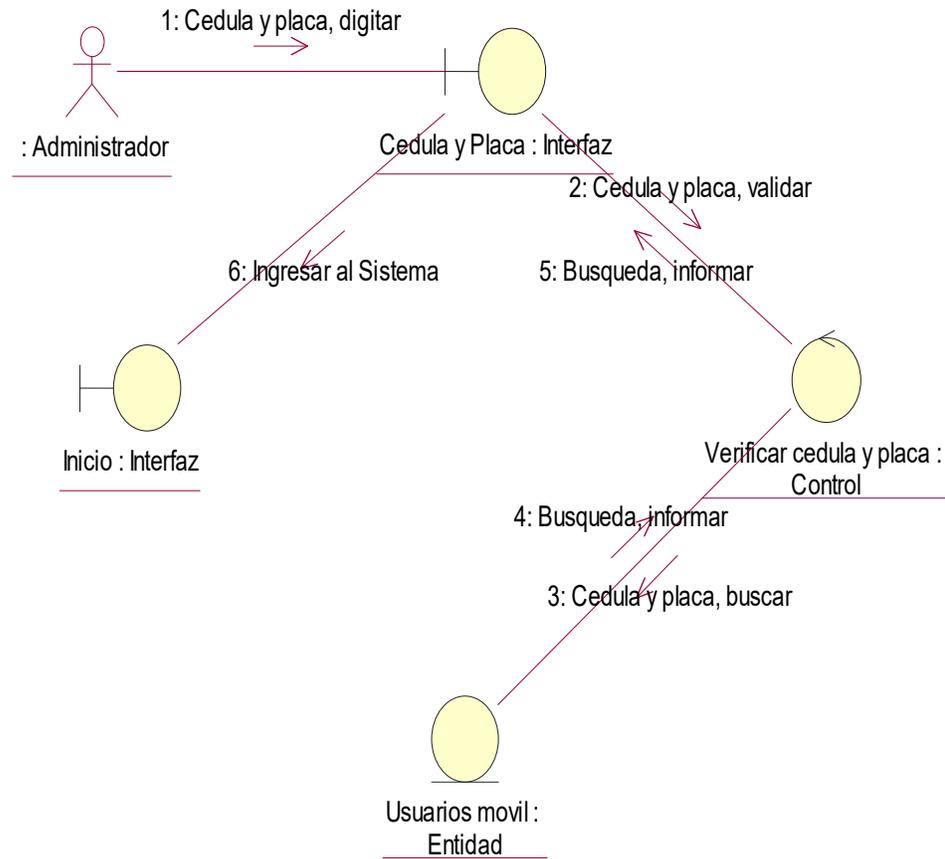
- **Diagrama de Colaboración de Digitar Cedula y Placa.** Se observa el diagrama de colaboración de Digitar Cedula y Placa correspondiente al prototipo del sistema de comparendos (ver figura 29). En este diagrama se puede apreciar secuencialmente lo que tanto el usuario administrador como el alférez de tránsito deben hacer para lograr ingresar al sistema.

**Figura 29. Diagrama de Colaboración de Crear Cuenta de Usuario**



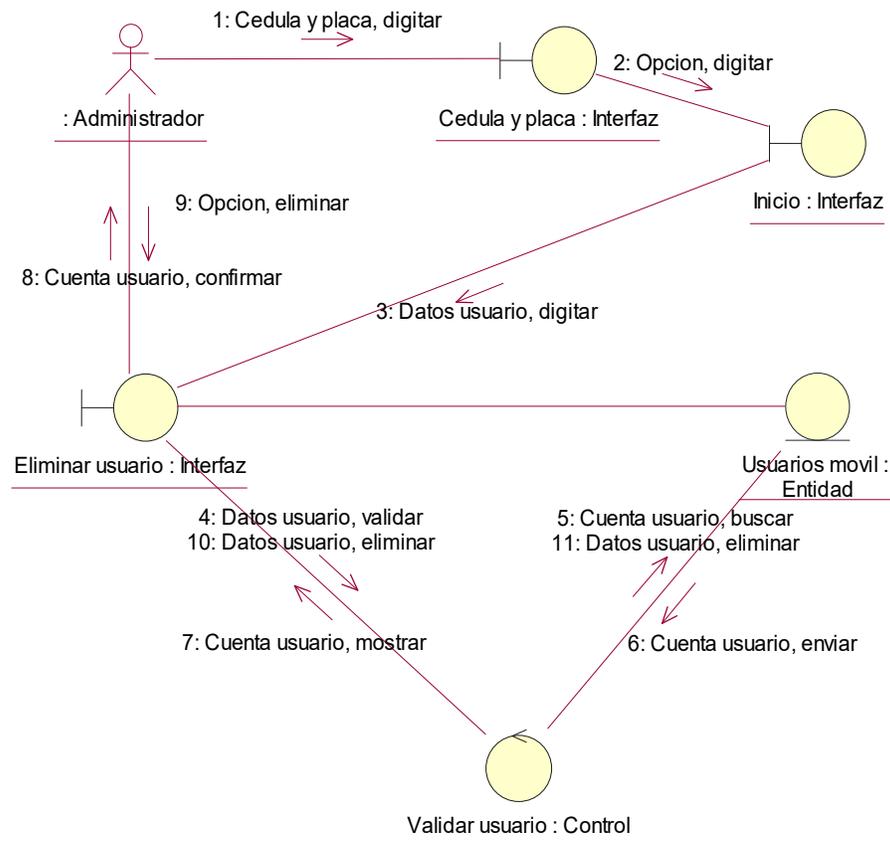
- Diagrama de Colaboración de Eliminar Cuenta de Usuario.** Se observa el diagrama de colaboración de Eliminar Cuenta de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 30). En este diagrama podemos ver secuencialmente lo que el usuario administrador debe hacer para eliminar un usuario del sistema.

**Figura 30. Diagrama de Colaboración de Digitar Cedula y Placa**



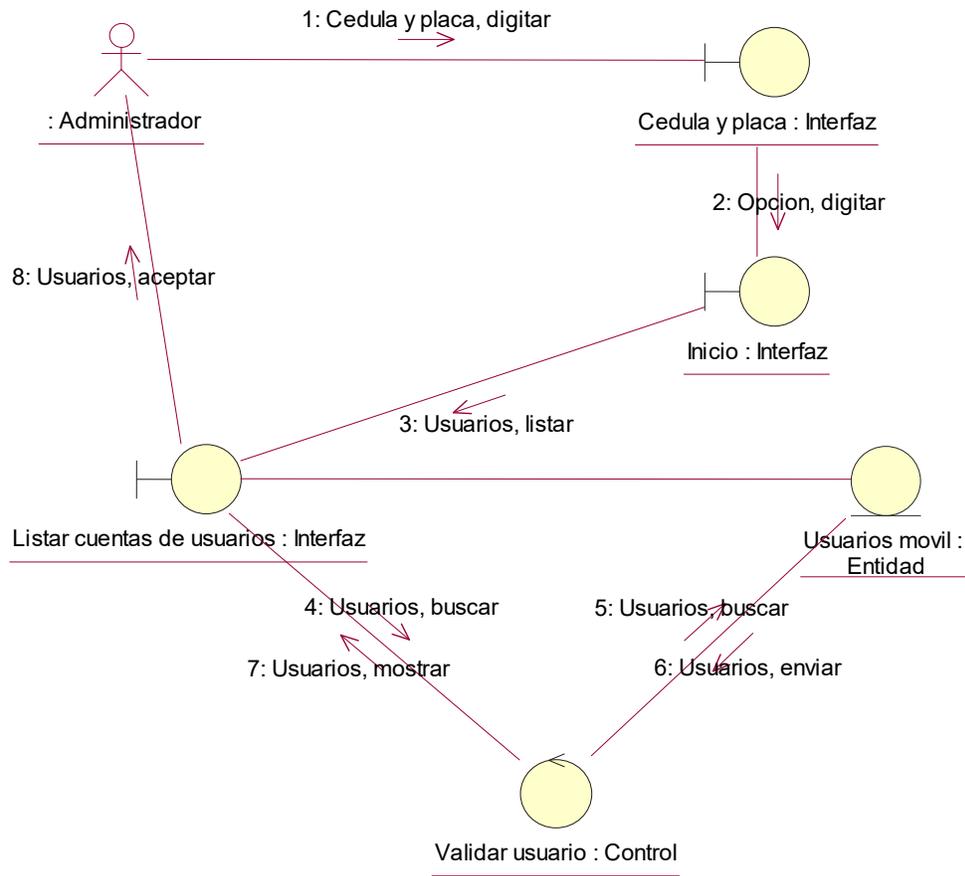
- **Diagrama de Colaboración de Listar Cuentas de Usuario.** Se observa el diagrama de colaboración de Listar Cuentas de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 31). En este diagrama podemos observar el proceso secuencialmente de lo que el usuario administrador debe hacer para listar todos los usuarios disponibles en la base de datos de Usuarios, la cual se encuentra en el servidor y temporalmente en la base de datos del dispositivo móvil. También permite saber si no existen usuarios creados en la base de datos. Solo los usuarios que estén registrados podrán acceder al sistema desde el dispositivo inteligente, el administrador es el encargado de gestionar las cuentas de los usuarios.

**Figura 31. Diagrama de Colaboración de Eliminar Cuenta de Usuario**



- Diagrama de Colaboración de Sincronización del dispositivo móvil.** Se observa el diagrama de colaboración de Sincronización del dispositivo móvil correspondiente al prototipo del sistema de comparendos (ver figura 32). En este diagrama se puede detallar el proceso que el usuario administrador debe hacer para sincronizar las diferentes bases de datos que se encuentran localizadas tanto en el dispositivo como en el servidor principal de la aplicación. Debido a su complejidad de sincronización, éste procedimiento es manejado por el administrador, el cual, es el encargado de asegurarse que los datos hayan sido actualizados en las diferentes bases de datos.

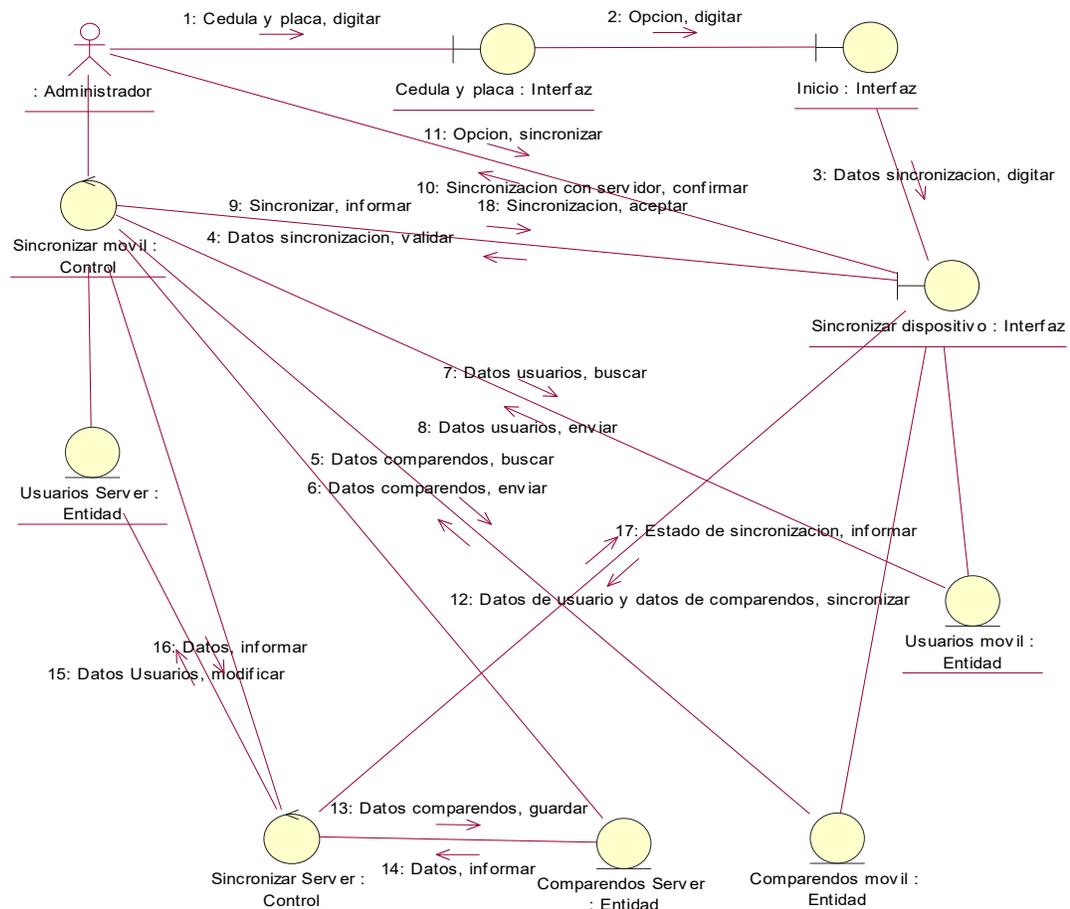
**Figura 32. Diagrama de Colaboración de Listar Cuentas de Usuario**



4.1.2.7 Diagramas de Actividad. Es un diagrama de flujo del proceso multi - propósito que se usa para modelar el comportamiento del sistema. Los diagramas de actividad se pueden usar para modelar un Caso de Uso, o una clase, o un método complicado.

Un diagrama de actividad es parecido a un diagrama de flujo; la diferencia clave es que los diagramas de actividad pueden mostrar procesado en paralelo (parallel processing). Esto es importante cuando se usan diagramas de actividad para modelar procesos 'business' algunos de los cuales pueden actuar en paralelo, y para modelar varios hilos en los programas concurrentes.

**Figura 33. Diagrama de Colaboración de Sincronización de Dispositivos**



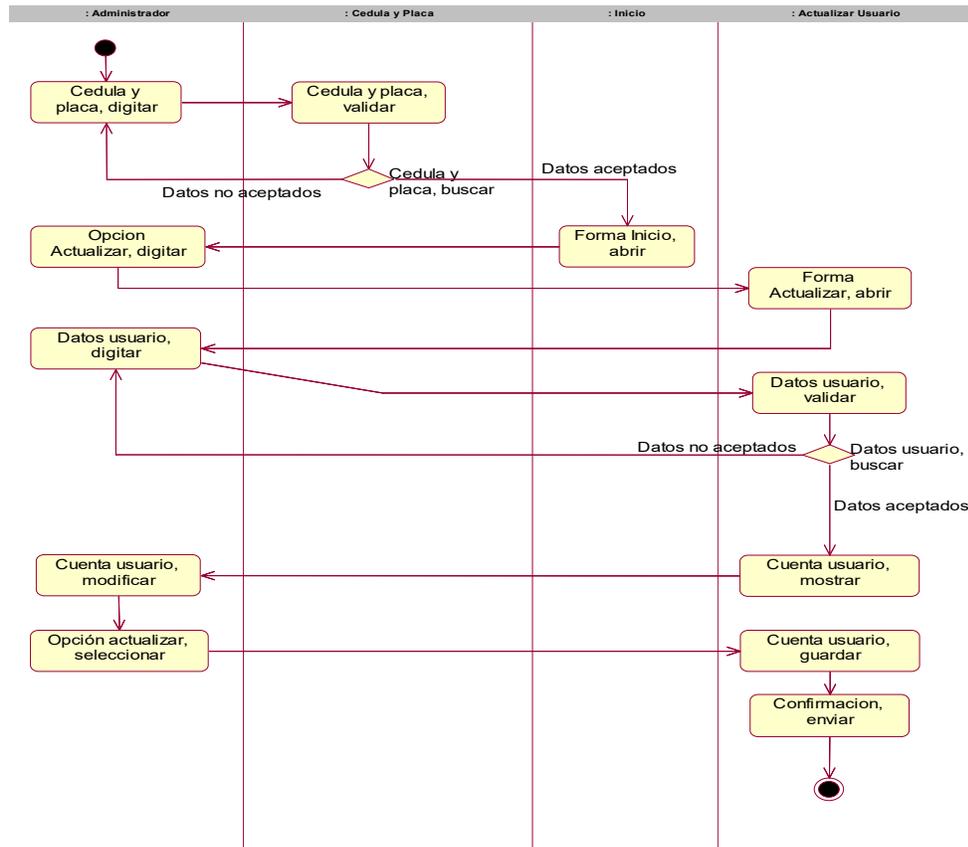
Los Diagramas de Actividad ofrecen una herramienta gráfica para modelar el proceso de un Caso de Uso. Se pueden usar como un añadido a una descripción textual del caso de uso, o para listar los pasos del caso de uso. Una descripción textual, código, u otros diagramas de actividad pueden detallar más la actividad.

#### 4.1.2.8 Diagramas de Actividad prototipo de Sistema de Comparendos

- **Diagrama de Actividad de Actualizar Cuenta de Usuario.** Se observa el diagrama de Actividad de Actualizar Cuenta de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 33). En este diagrama se

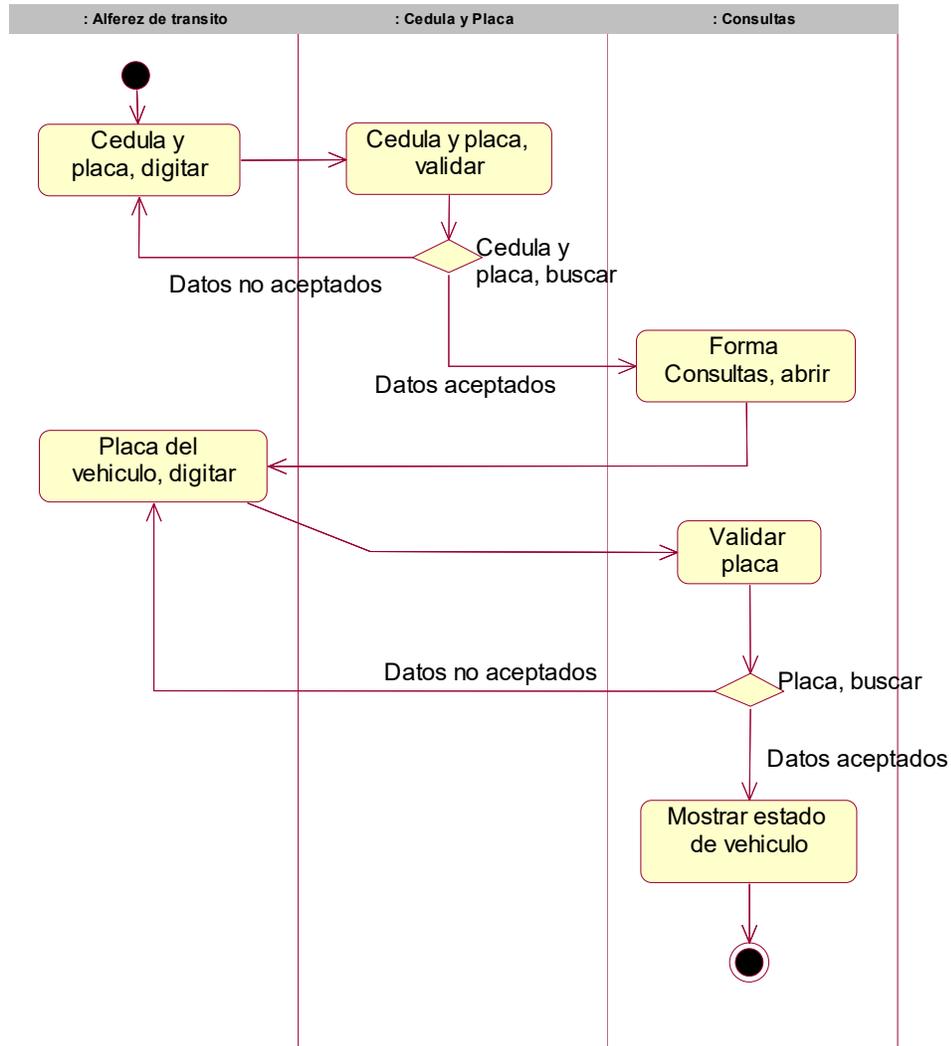
pueden apreciar los diferentes estados del usuario administrador y del sistema para actualizar un usuario.

**Figura 34. Diagrama de Actividad de Actualizar Cuenta de Usuario**



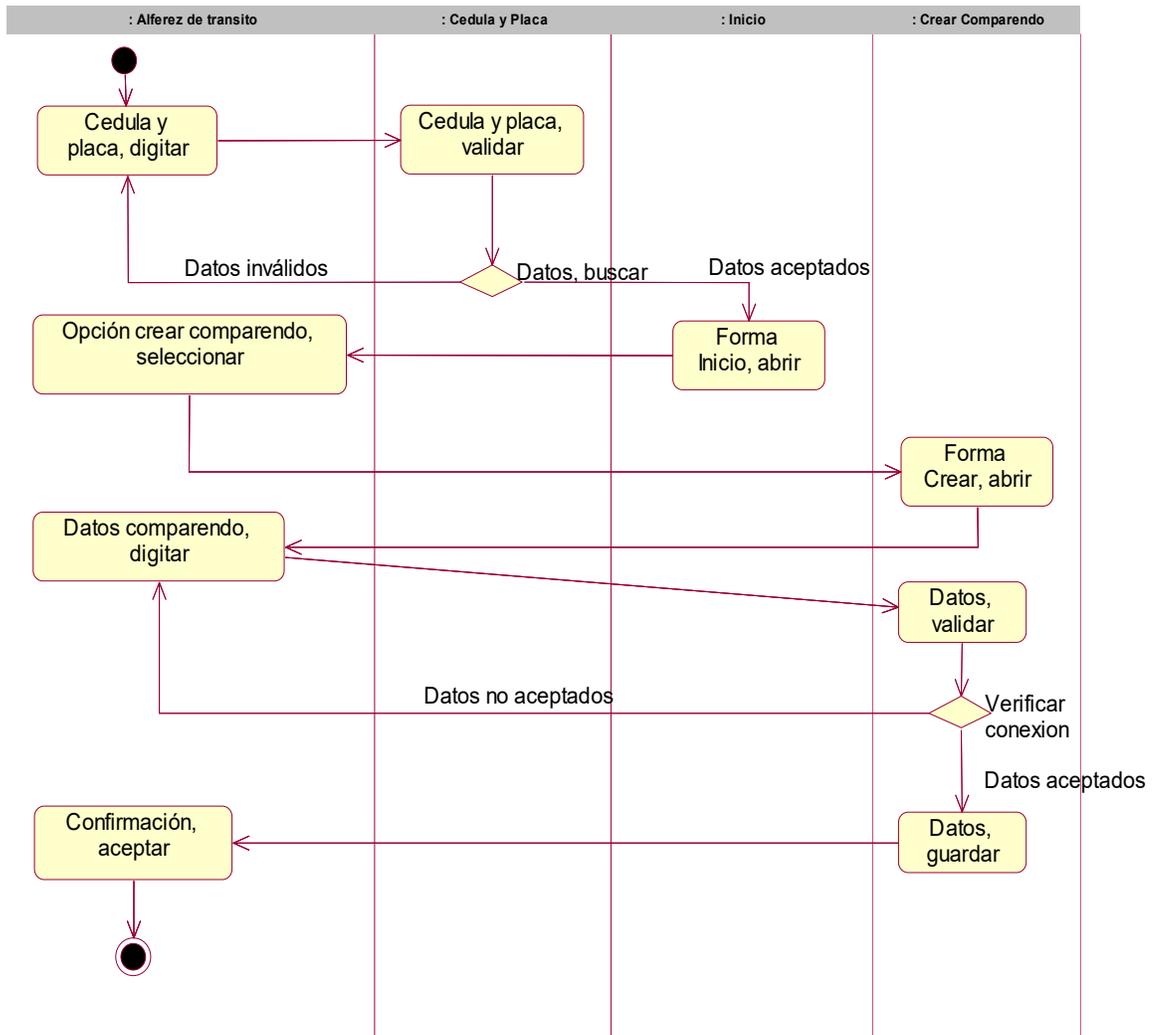
- Diagrama de Actividad de Consultar Comparendo.** Se observa el diagrama de Actividad de Consultar Comparendo correspondiente al prototipo del sistema de comparendos (ver figura 34). En este diagrama se pueden apreciar los diferentes estados tanto del usuario alférez de tránsito como del particular al interactuar con el sistema para consultar un estado de cuenta de un vehículo.

**Figura 35. Diagrama de Actividad de Consultar Comparendo**



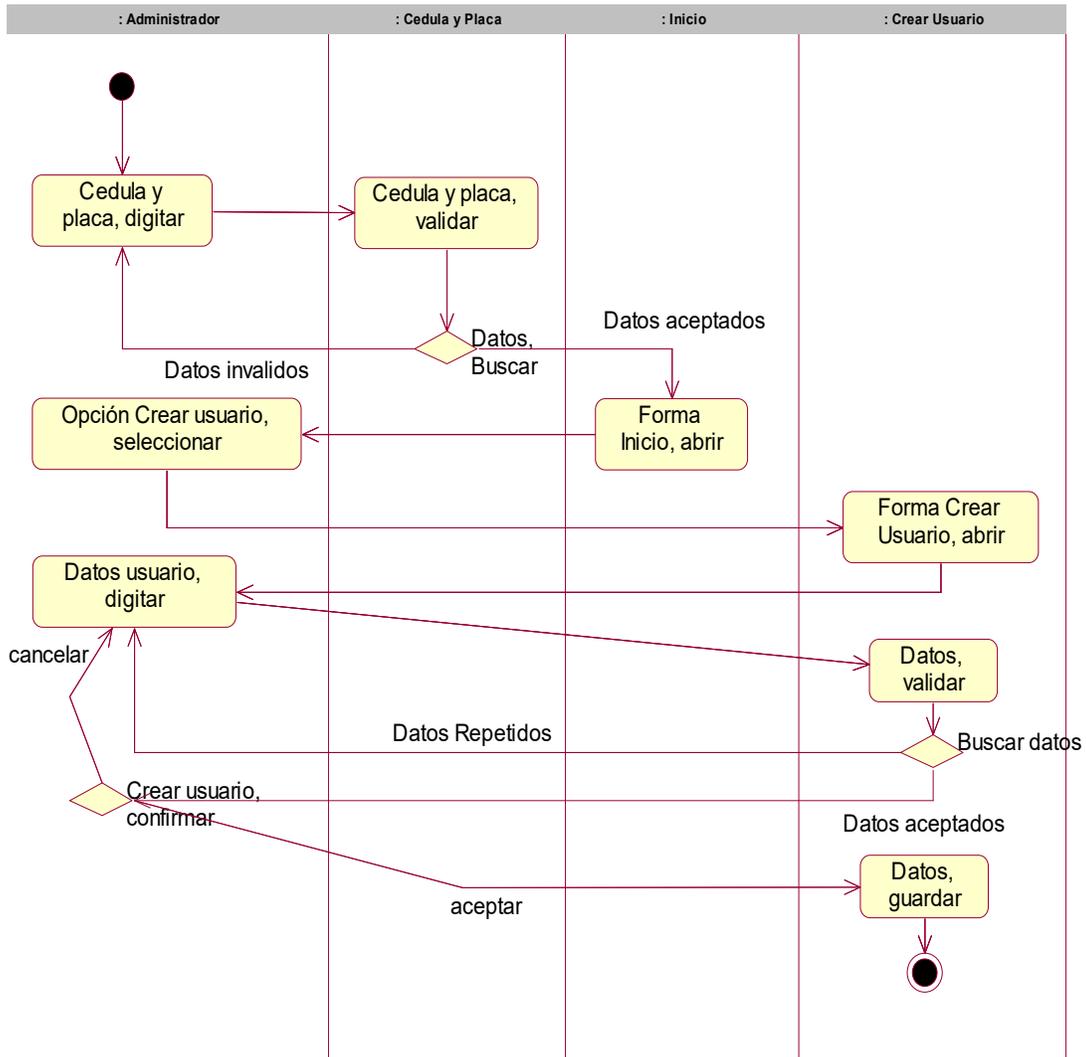
- Diagrama de Actividad de Crear Comparendo.** Se observa el diagrama de Actividad de Crear Comparendo correspondiente al prototipo del sistema de comparendos (ver figura 35). En este diagrama se pueden apreciar los diferentes estados del usuario alferez de tránsito y del sistema para crear un nuevo comparendo de tránsito de un vehículo.

**Figura 36. Diagrama de Actividad de Crear Comparendo**



- Diagrama de Actividad de Crear Cuenta de Usuario.** Se observa el diagrama de Actividad de Crear Cuenta de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 36). En este diagrama se pueden detallar los diferentes estados del usuario administrador al interactuar con el sistema para crear un nuevo usuario en la base de datos.

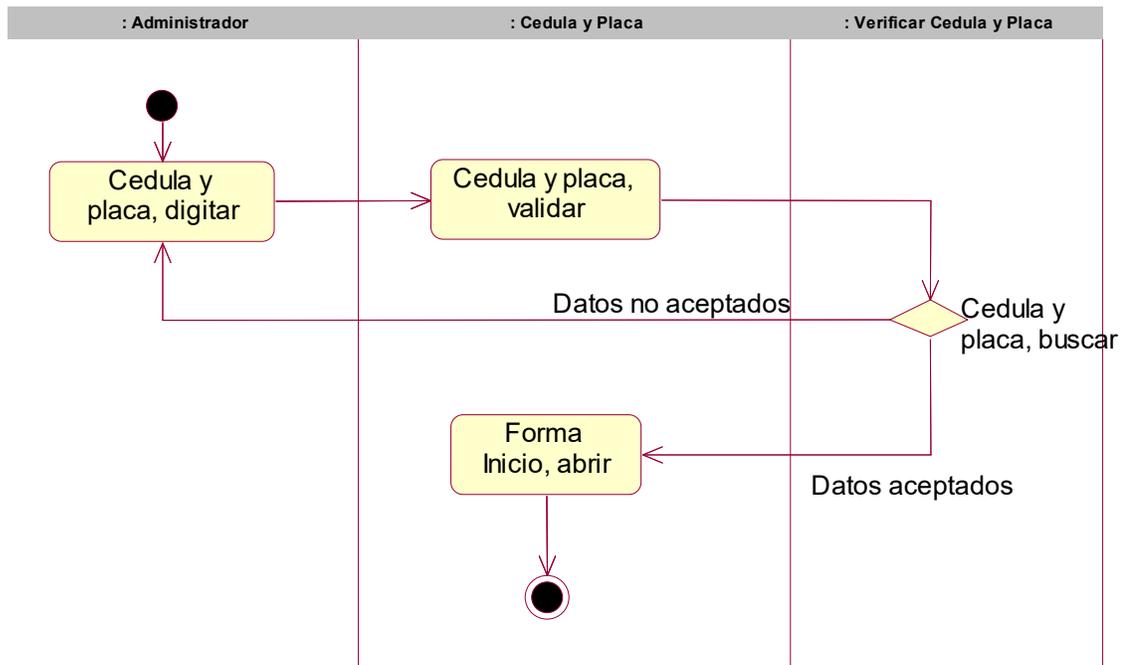
**Figura 37. Diagrama de Actividad de Crear Cuenta de Usuario**



- Diagrama de Actividad de Digitar Cedula y Placa.** Se observa el diagrama de Actividad de Digitar Cedula y Placa correspondiente al prototipo del sistema de comparendos (ver figura 37). En este diagrama se pueden ver los diversos estados tanto del usuario administrador como del alférez de tránsito al interactuar con el sistema para hacer un procedimiento de ingreso hacia las tareas que ofrece el prototipo.

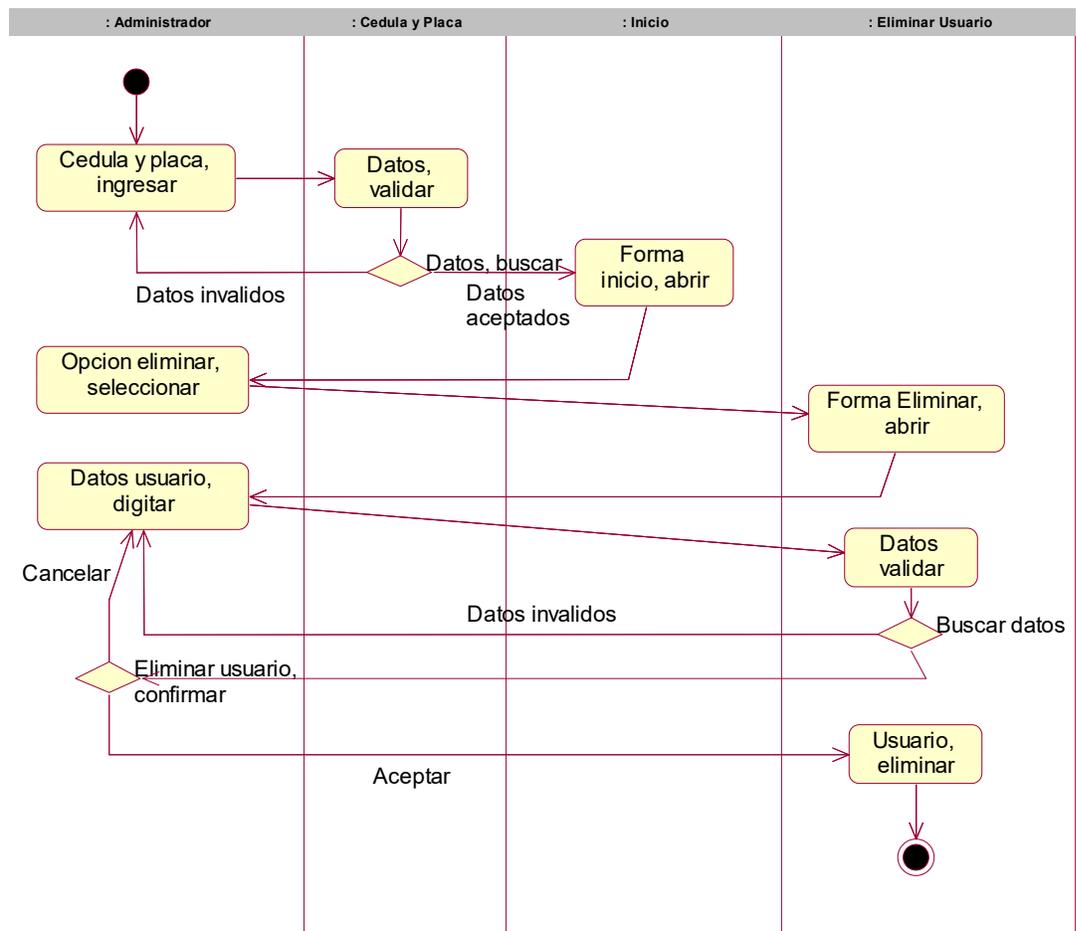
Los usuarios deben haber sido registrados con anterioridad en el sistema y se debe haber hecho una sincronización de las diferentes bases de datos para tener los datos actualizados.

**Figura 38. Diagrama de Actividad de Digitar Cedula y Placa**



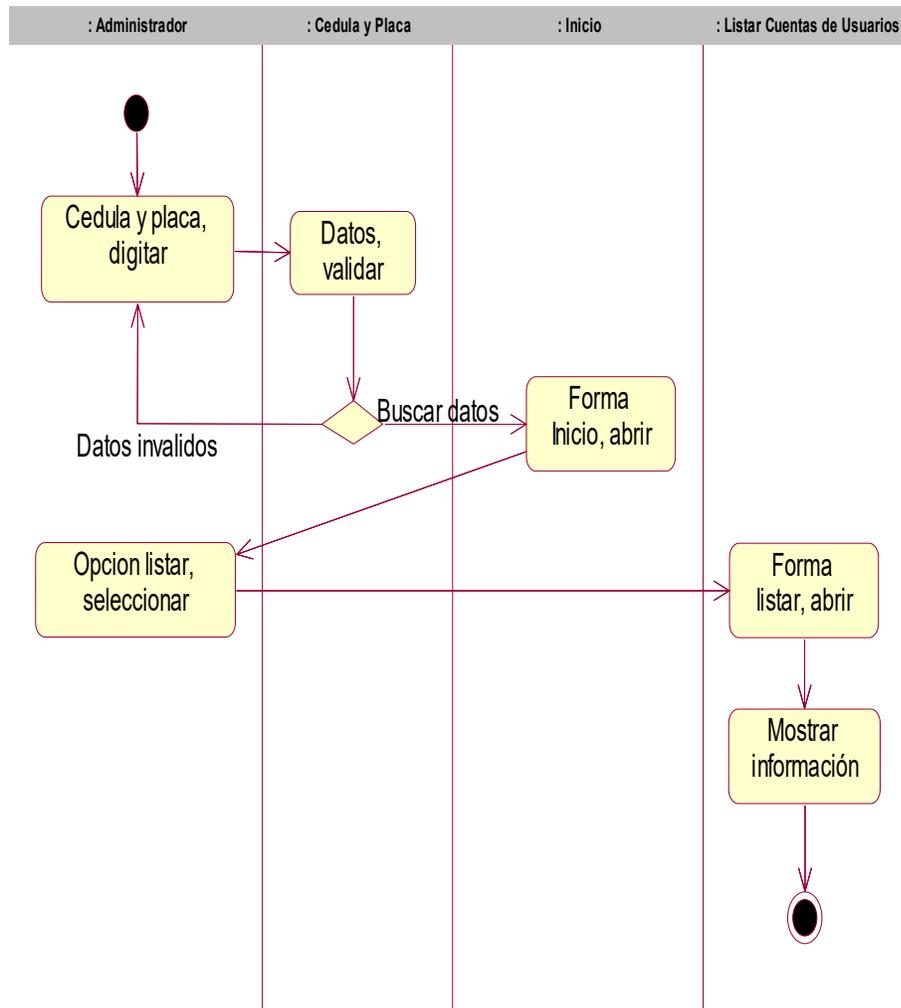
- **Diagrama de Actividad de Eliminar Cuenta de Usuario.** Se observa el diagrama de Actividad de Eliminar Cuenta de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 38). En este diagrama se pueden presenciar los diferentes estados del usuario administrador al interactuar con el sistema para eliminar un usuario de la base de datos.

**Figura 39. Diagrama de Actividad de Eliminar Cuenta de Usuario**



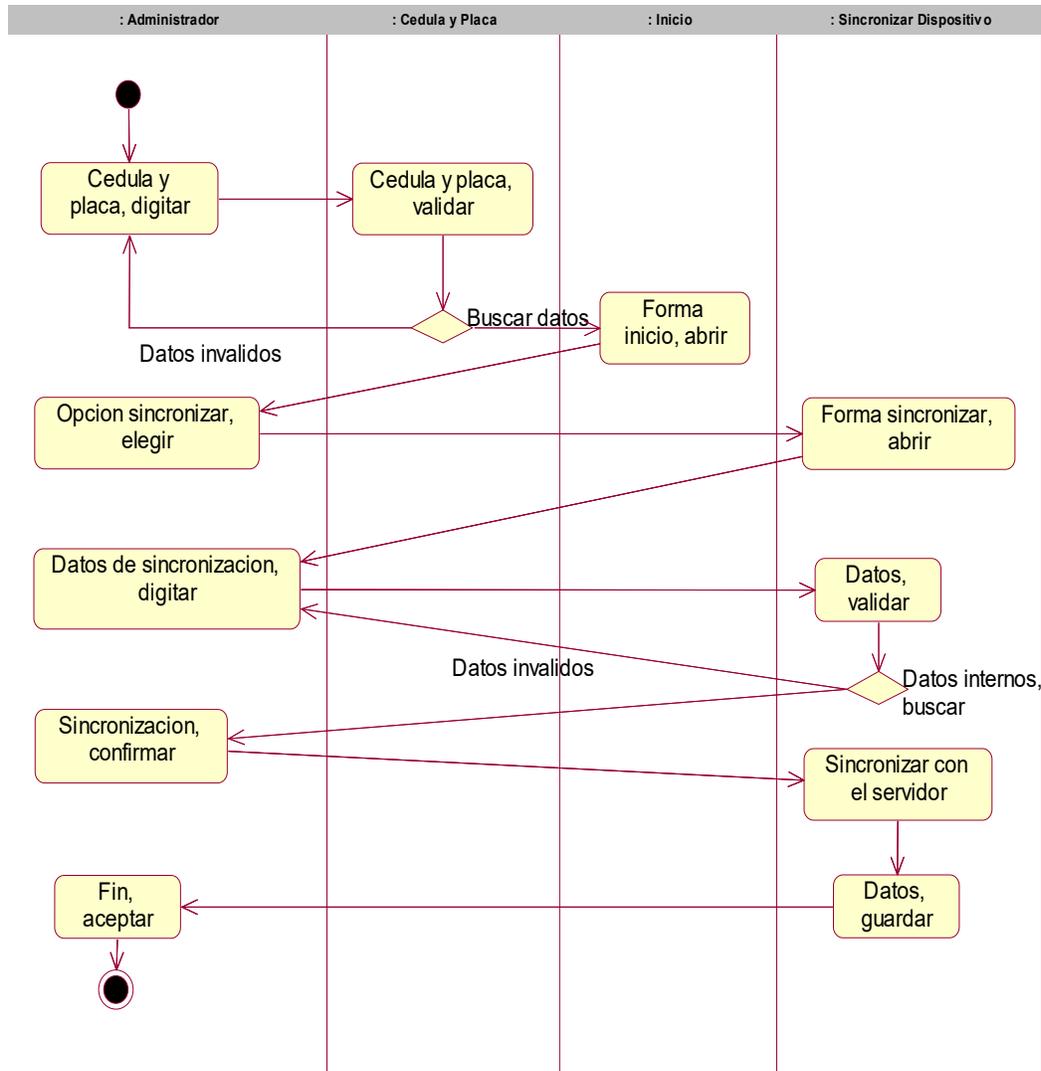
- Diagrama de Actividad de Listar Cuentas de Usuario.** Se observa el diagrama de Actividad de Listar Cuentas de Usuario correspondiente al prototipo del sistema de comparendos (ver figura 39). En este diagrama se pueden visualizar los diferentes estados del usuario administrador al interactuar con el sistema para listar todos los usuarios disponibles.

**Figura 40. Diagrama de Actividad de Listar Cuentas de Usuario**



- Diagrama de Actividad de Sincronización del Dispositivo Móvil.** Se observa el diagrama de Actividad de Sincronización del Dispositivo Móvil correspondiente al prototipo del sistema de comparendos (ver figura 40). En este diagrama se observan los diferentes estados del usuario administrador al sincronizar tanto el servidor como el dispositivo inteligente.

**Figura 41. Diagrama de Actividad de Sincronización del Dispositivo Móvil**

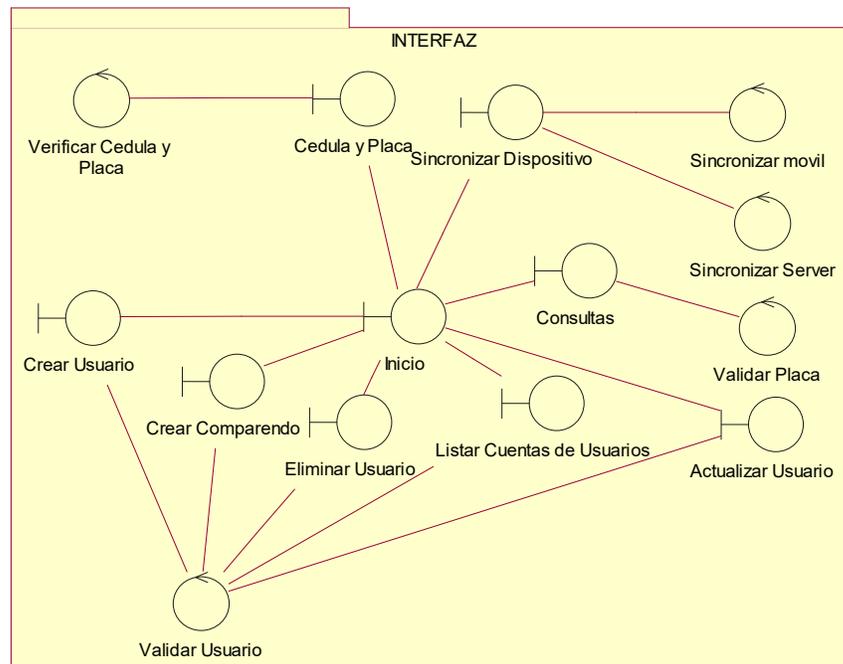


4.1.2.9 Diagrama de Paquetes. Es la forma en que UML agrupa elementos en subsistemas mediante paquetes y los une formando una jerarquía. De hecho un sistema que no tenga necesidad de ser descompuesto en subsistemas se puede considerar como con un único paquete que lo abarca todo.

#### 4.1.2.10 Diagramas de Paquetes prototipo de Sistema de Comparendos

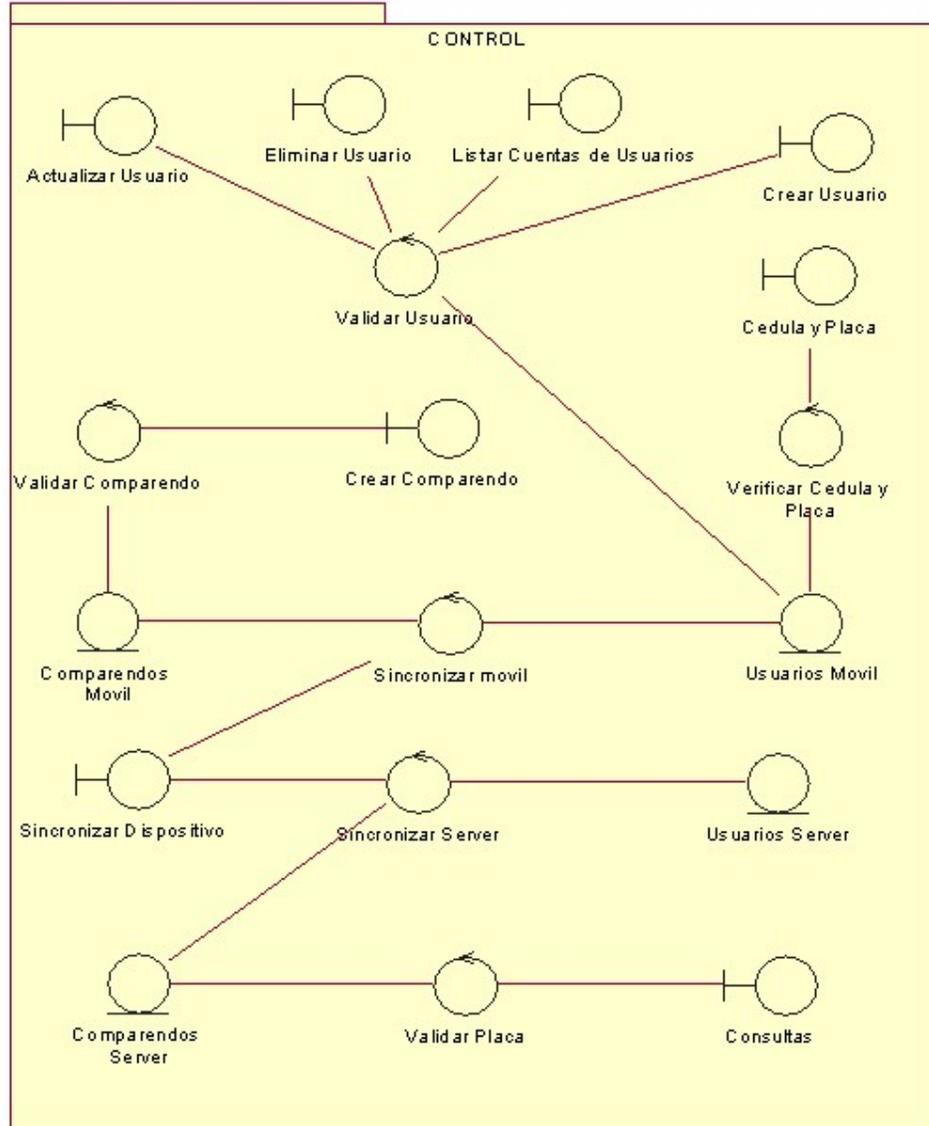
- **Diagrama de Paquetes de Interfaces.** Se puede apreciar el diagrama de Paquetes de Interfaces del Dispositivo Móvil correspondiente al prototipo del sistema de comparendos (ver figura 41). En este diagrama se observan las diferentes conexiones de las interfaces con otros elementos del prototipo software.

**Figura 42. Diagrama de Paquetes de Interfaces**



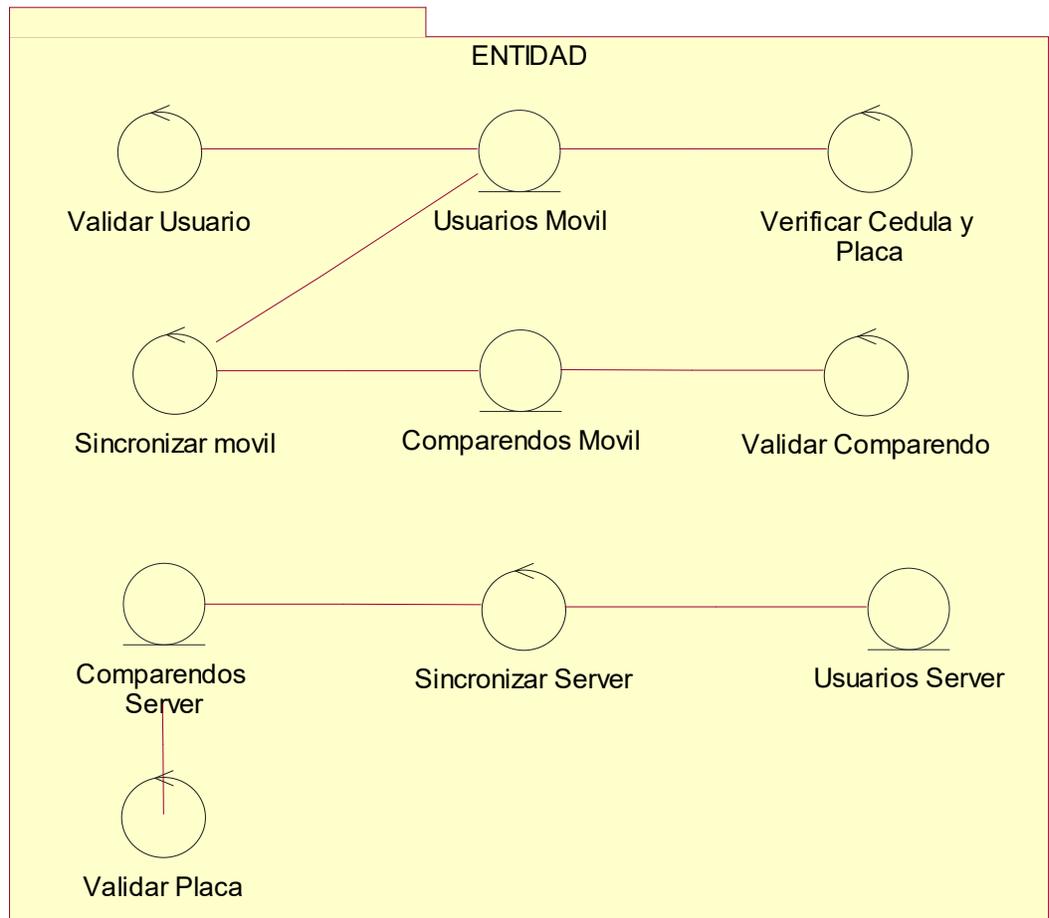
- **Diagrama de Paquetes de Controles.** Se puede apreciar el diagrama de Paquetes de Controles del Dispositivo Móvil correspondiente al prototipo del sistema de comparendos (ver figura 42). En este diagrama se observan las diferentes conexiones de los controles con otros elementos del prototipo software.

**Figura 43. Diagrama de Paquetes de Controles**

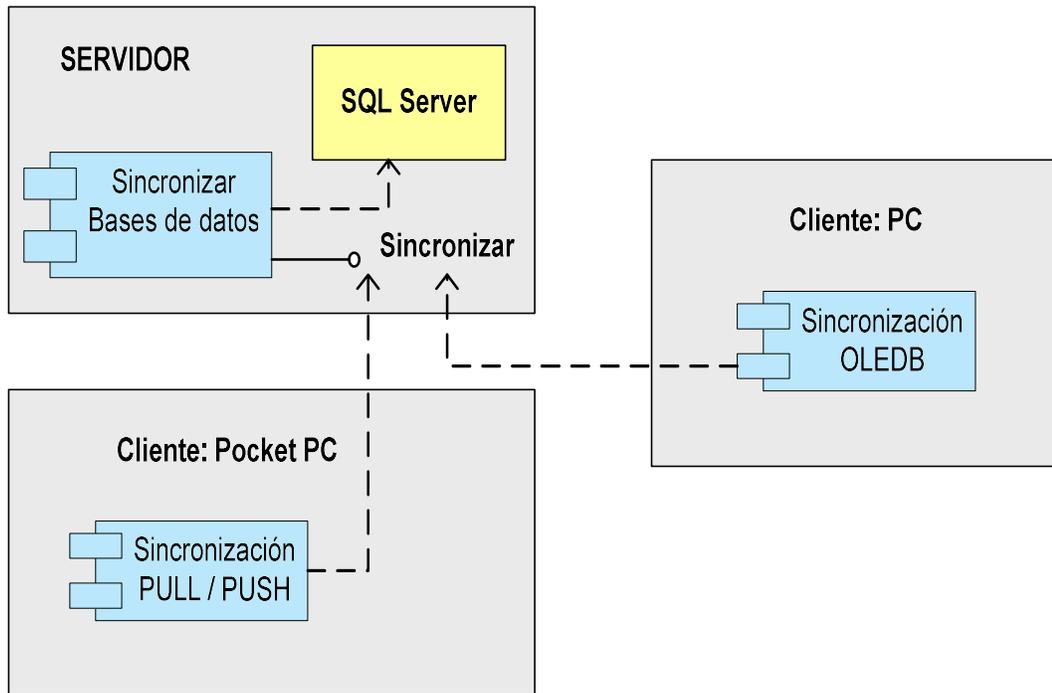


- Diagrama de Paquetes de Entidades.** Se puede apreciar el diagrama de Paquetes de Entidades del Dispositivo Móvil correspondiente al prototipo del sistema de comparendos (ver figura 43). En este diagrama se observan las diferentes conexiones de las entidades con otros elementos del prototipo software.

Figura 44. Diagrama de Paquetes de Entidades



**Figura 45. Diagrama de Implementación**



En éste diagrama podemos observar los principales componentes de sincronización de los diferentes prototipos realizados durante el proyecto de investigación.

La aplicación cliente / servidor utiliza el componente OLEDB para la sincronización. Por otro lado, el prototipo del dispositivo móvil utiliza el componente RDA (PULL / PUSH) para comunicarse con el servidor que contiene la base de datos principal. El sistema de gestión de base de datos SQL SERVER 2000, proporciona un total apoyo a los componentes utilizados en los prototipos, evidenciando la interoperabilidad de las diferentes plataformas.

#### **4.1.3 Diseño**

Para comenzar con el diseño del prototipo software, se tuvo en cuenta el modelamiento del sistema que proyectó la etapa del análisis, creando la estructura de la aplicación.

#### 4.1.3.1 Prototipo de registro de comparendos

- **Interfaz de inicio**

**Descripción:** Esta pantalla permite a los usuarios: tanto administrador y alférez de tránsito; ingresar al sistema mediante el registro de cédula de ciudadanía y placa de identificación correspondientes. Una vez ingresados los datos, la ejecución del botón *Ingresar* los enviará a la forma de inicio, donde podrán tener las opciones disponibles para cada usuario (ver figura 44).

**Figura 46. Interfaz de inicio**



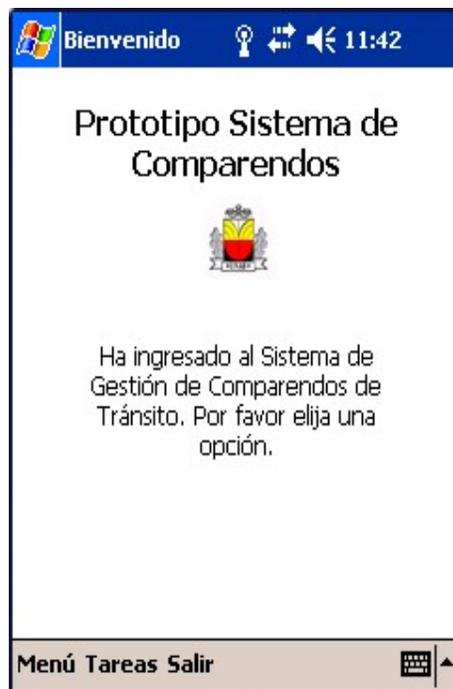
The image shows a screenshot of a mobile application interface. At the top, there is a blue header bar with the text 'Inicio..' on the left, and icons for a key, a double-headed arrow, a speaker, the time '11:37', and an 'ok' button on the right. Below the header, the title 'Prototipo Sistema de Comparendos' is centered. Underneath the title is a small crest logo. The main area contains two input fields: 'Cedula' and 'Placa', each with a corresponding text box. To the right of these fields is a button labeled 'Ingresar'. Below the input fields, there is a block of text: 'Estudio, diseño e implementación de un prototipo software para la gestión de comparendos de tránsito basado en dispositivos inteligentes bajo arquitectura Microsoft .NET'. At the bottom right of the screen, there is a small icon of a keyboard and a cursor arrow.

- **Interfaz de bienvenida**

**Descripción:** Esta forma da una bienvenida a los usuarios que ingresan al sistema y permite tener acceso a las tareas y opciones disponibles para cada usuario (ver figura 47).

Cada usuario tiene tareas específicas en el prototipo sistema de comparendos. El usuario administrador tiene la posibilidad de cargar y sincronizar las bases de datos del dispositivo con el servidor de la aplicación. El usuario alférez de tránsito puede la opción de crear comparendos y consultarlos en el momento que lo desee. Ambos usuarios pueden ver la una forma que contiene la información acerca del prototipo.

**Figura 47. Interfaz de bienvenida**



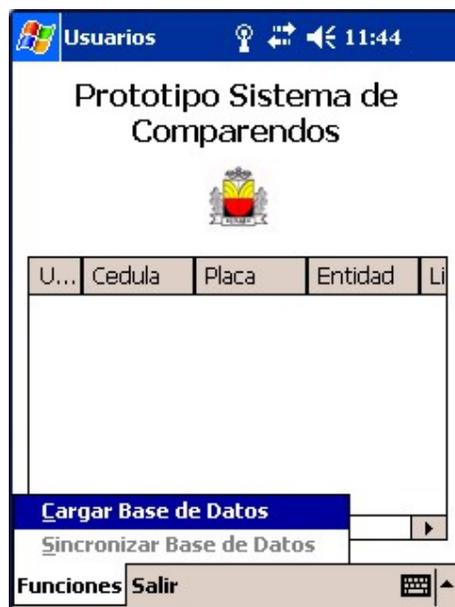
- **Interfaz de sincronización**

**Descripción:** En esta pantalla el usuario administrador tiene la posibilidad de elegir entre cargar la base de datos y sincronizarla (ver figura 48). También en ésta interfaz proporciona al usuario administrador visualizar toda la información de los usuarios disponibles en la base de datos. Las anteriores opciones se encuentran en el menú *Funciones* de la forma en cuestión, y que sirven de gran

utilidad a la hora de sincronizar el dispositivo con el servidor principal donde se encuentran las principales bases de datos.

El usuario administrador del sistema es la única persona encargada de realizar la sincronización y mantener la información actualizada tanto del dispositivo como del servidor de la base de datos.

**Figura 48. Interfaz de sincronización**



- **Interfaz de información del prototipo**

**Descripción:** En esta interfaz los usuarios del prototipo tienen la posibilidad de visualizar una breve información tanto de la aplicación como del sistema operativo (ver figura 49). El sistema operativo para el cual está diseñado la aplicación es el SO Windows CE de Microsoft, que ha sido desarrollado para éste tipo de dispositivos (Pocket PC 2003). En la parte de la información de la aplicación, se puede apreciar que es una primera versión del prototipo software. También podemos encontrar las personas que han desarrollado la aplicación. Se ha colocado el nombre de la Universidad Autónoma de

Bucaramanga UNAB, debido a que es un prototipo software académico, impulsado por un proyecto de grado de dicha universidad.

**Figura 49. Interfaz de Información**



- **Interfaz de registro de comparendo**

**Descripción:** En esta pantalla el usuario alférez de tránsito tienen la posibilidad de seguir una serie de pasos que le permiten generar un nuevo comparendo de tránsito (ver figura 50).

Figura 50. Interfaz de registro de comparendo paso 1

The screenshot shows a mobile application interface titled "REGISTRO" with a status bar at the top displaying the time as 3:39. The main heading is "Prototipo Sistema de Comparendos" with a central emblem. Below this, there are two input fields: "Libreta No." and "Comparendo No.". A section titled "1. FECHA Y HORA" contains two fields: "Fecha" with the value "22/05/2005" and "Hora" with the value "16:39". At the bottom, there is a navigation bar with tabs for "Paso 1", "Paso 2", "Paso 3", "Paso 4", and "Paso 5", along with "Guardar" and "Salir" buttons.

Inicialmente, aparecen los principales datos que deben tener todos los comparendos de tránsito (número de la libreta y número del comparendo); los cuales son tomados directamente del servidor y varían de acuerdo al usuario que desea realizar el comparendo.

Podemos apreciar el título *1. Fecha y hora*, que han sido tomados de la configuración interna del dispositivo.

En la siguiente pestaña se encuentra el título *2. Lugar de la infracción*, el cual lo diligencia el agente de tránsito indicando el lugar donde ocurre la infracción (ver figura 51). En el título (3) se ingresan las letras de la placa del vehículo infractor.

**Figura 51. Interfaz de registro de comparendo paso 2**

The screenshot shows a mobile application interface titled 'REGISTRO'. At the top, there is a blue header bar with the Windows logo, the title 'REGISTRO', and system icons for signal strength, Wi-Fi, and battery, along with the time '3:39'. Below the header, the interface is divided into two main sections. The first section is titled '2. LUGAR DE LA INFRACCION' and contains a text input field with the value 'CRA 31 # 43 - 44'. The second section is titled '3. PLACA (MARQUE LAS LETRAS)' and contains a text input field with the value 'MMC'. At the bottom of the screen, there is a navigation bar with five tabs labeled 'Paso 1', 'Paso 2', 'Paso 3', 'Paso 4', and 'Paso 5'. The 'Paso 2' tab is currently selected. To the right of the tabs are navigation arrows. Below the tabs, there are two buttons labeled 'Guardar' and 'Salir', and a small icon of a keyboard.

En el paso 3 se termina de digitar los datos del vehiculo en el titulo (4) PLACA (Números).

En ésta pestaña se puede apreciar el título 5. *Código de la infracción*; en el cual se ingresa el código de la infracción impuesta (ver figura 52).

Para buscar todos los códigos de infracción disponibles se debe hacer clic en el botón *Ver códigos*.

Cuando se selecciona ese botón, se abrirá un archivo Codes.txt que contiene las infracciones que se pueden aplicar, y están organizados por número, descripción y valor en pesos de la infracción. Éste archivo facilita al agente de tránsito en la realización de comparendos.

Figura 52. Interfaz de registro de comparendo paso 3

REGISTRO 3:39

**4. PLACA (MARQUE LOS NUMEROS)**

665 Letras (motos) (NINGU)

**5. CODIGO DE LA INFRACCION**

46 Ver Códigos

Paso 1 Paso 2 Paso 3 Paso 4 Paso 5

Guardar Salir

Para el siguiente paso se selecciona el tipo de vehículo que hace la infracción (título 6) y la clase de servicio que presta (título 7).

Figura 53. Interfaz de registro de comparendo paso 4

REGISTRO 3:40

**6. TIPO VEHICULO (escoja tipo)**

AUTOMOVIL

**7. CLASE DE SERVICIO (escoja clase)**

DIPLOMATICO

Paso 1 Paso 2 Paso 3 Paso 4 Paso 5

Guardar Salir

Tanto para la clase como para el tipo de vehículo, el dispositivo inteligente proporciona la facilidad de seleccionar lo que se desea por medio de un control de dispositivo de Windows llamado Combo Box, sin necesidad de digitar esa información.

En la siguientes dos pestañas se encuentra el título *8. Datos de infractor*, el cual debe ser llenado con todos los datos de la persona que hace la infracción (ver figura 54).

**Figura 54. Interfaz de registro de comparendo paso 5**

The screenshot shows a Windows-style application window titled 'REGISTRO'. The main content area is titled '8. DATOS DEL INFRACTOR'. It contains the following fields:

- Documento de Identidad: 13454233
- Licencia de Conducción: 45367243
- Categoría: 03 (with up/down arrows)
- Expedida: DÍA (with up/down arrows), MES (with up/down arrows), AÑO (with up/down arrows)

At the bottom of the form is a numeric keypad with the following layout:

123	[	]	{	}	7	8	9	#	%	=	←
^	,	.	<	>	4	5	6	+	-	*	/
±	°	:	\		1	2	3	↓	↑	←	→
\$	¢	€	£	¥	(	0	)	Tab	space	↵	

Below the keypad are the buttons 'Guardar' and 'Salir'.

Para los números se utiliza un teclado digital que el dispositivo trae integrado para ingresar la información.

Al final se completa la información con el nombre y la dirección de la persona infractora (ver figura 55).

**Figura 55. Interfaz de registro de comparendo paso 6**

REGISTRO 3:41

Vence

01 01 2001

Nombres y apellidos: PEDRO PORRAS

Dirección: CRA 25 # 65-63

123 [ ] { } 7 8 9 # % = ←

^ , . < > 4 5 6 + - \* /

± ° : \ | 1 2 3 ↓ ↑ ← →

\$ ¢ € £ ¥ ( 0 ) Tab space ←

Guardar Salir

Luego en la pestaña siguiente, vemos el título. *Tipo de infractor* (ver figura 56). El agente de tránsito debe llenar el tipo de infractor y la licencia de tránsito (título 10), es de la persona que produjo la infracción. En el título 11. *Datos del agente*, no se digitan, debido a que el sistema toma esa información de la tabla usuarios del sistema.

**Figura 56. Interfaz de registro de comparendo paso 7**

REGISTRO 3:42

9. TIPO DE INFRACTOR

CONDUCTOR

10. LICENCIA DE TRANSITO No.

453463-54522

123 [ ] { } 7 8 9 # % = ←

^ , . < > 4 5 6 + - \* /

± ° : \ | 1 2 3 ↓ ↑ ← →

\$ ¢ € £ ¥ ( 0 ) Tab space ←

Guardar Salir

Figura 57. Interfaz de registro de comparendo paso 8

REGISTRO 3:42

**11. DATOS DEL AGENTE**

Apellidos y nombres

Entidad

Placa 33201058

Paso 5 Paso 6 Paso 7 Paso 8 Paso 9

Guardar Salir

Figura 58. Interfaz de registro de comparendo paso 9

REGISTRO 3:42

**12. INMOVILIZACION**

Lugar (NINGUNO)

**13. OBSERVACIONES**

123 ! @ # \$ % ^ & \* ( ) \_ + Del  
Tab Q W E R T Y U I O P { }  
CAP A S D F G H J K L : "  
Shift Z X C V B N M < > ? ←  
Ctl á ü ~ | ↓ ↑ ← →

Guardar Salir

Finalmente, encontramos las últimas pestañas del registro de comparendos (ver figura 58,59). Si el vehículo del infractor es inmovilizado se indica el lugar

donde será llevado (título 12). Si el usuario alférez de tránsito necesita hacer observaciones acerca de algún comparendo impuesto, entonces se utilizará el espacio del título 13. *Observaciones*. En el título 14. *Aceptación del comparendo*, el agente de tránsito indica las personas que presencian la infracción seleccionándolos por medio de un control del dispositivo llamado Check Box.

**Figura 59. Interfaz de registro de comparendo paso 10**

The screenshot shows a mobile application interface titled 'REGISTRO'. The status bar at the top indicates signal strength, Wi-Fi, and a battery level of 3:42. The main content area is titled '14. ACEPTACION DEL COMPARENDO' and contains three checkboxes: 'Agente (Bajo juramento)' (checked), 'Conductor' (checked), and 'Testigo' (unchecked). Below the checkboxes is a horizontal line and a large empty text area. At the bottom, there is a navigation bar with buttons for 'Paso 6', 'Paso 7', 'Paso 8', 'Paso 9', and 'Paso 10'. The 'Paso 10' button is highlighted. To the right of the navigation bar are 'Guardar' and 'Salir' buttons.

La información que ha sido ingresada en el comparendo, se guarda en la base de datos del dispositivo por medio del botón *Guardar*, ubicado en la esquina inferior izquierda. La aplicación valida el ingreso de los datos y que estén diligenciados los campos necesarios para que el comparendo sea guardado. El sistema imprime físicamente un comparendo en el momento de ser guardado.

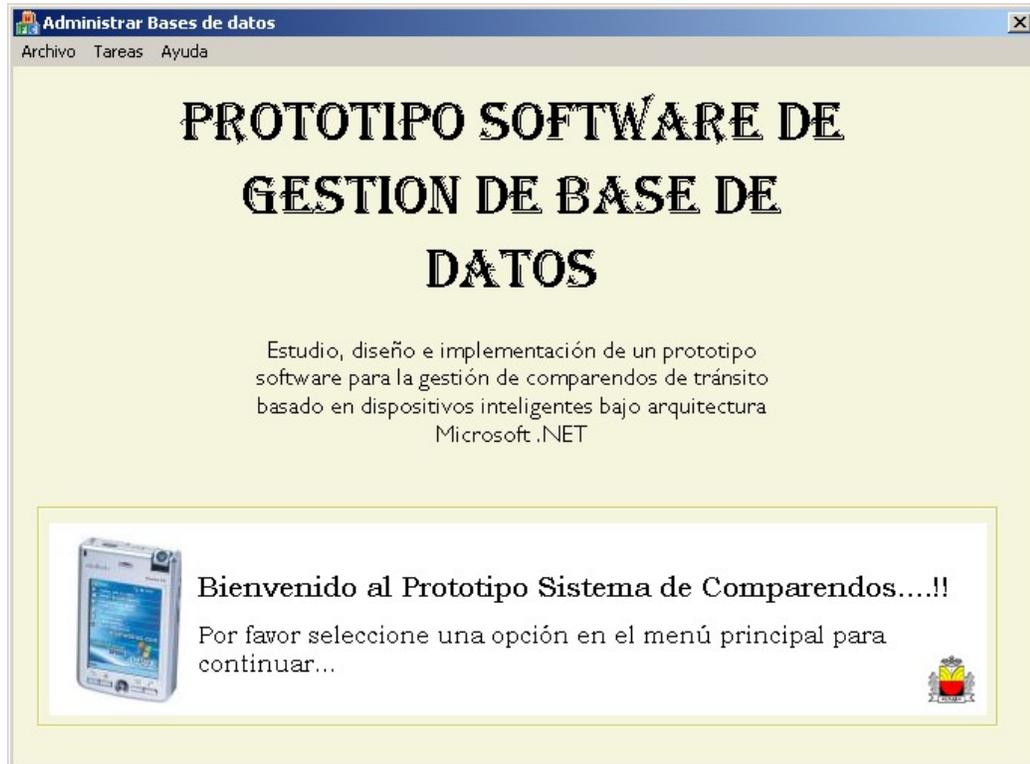
#### 4.1.3.2 Prototipo de gestión de base de datos

- **Interfaz de inicio.**

**Descripción:** Esta pantalla da la bienvenida al ingresar a la aplicación y permite al usuario administrador gestionar las bases de datos del sistema (ver

figura 60). El administrador tiene acceso a tareas relacionadas con las bases de datos.

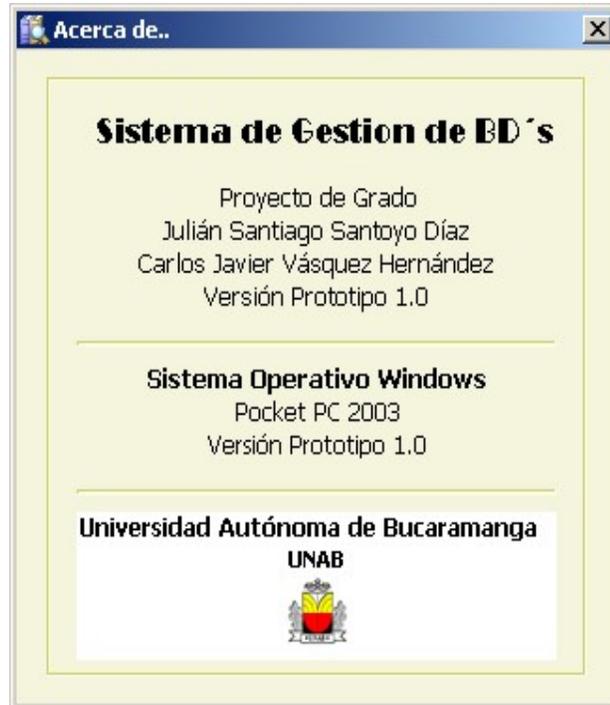
**Figura 60. Interfaz de inicio**



- **Interfaz de Información**

**Descripción:** En esta pantalla el usuario administrador del prototipo puede visualizar una breve información de la aplicación (ver figura 61). En la parte de la información de la aplicación, se puede apreciar que es una primera versión del prototipo software. También podemos encontrar las personas que han desarrollado la aplicación. Se ha colocado el nombre de la Universidad Autónoma de Bucaramanga (UNAB), debido a que es un prototipo software académico, impulsado por un proyecto de grado de dicha universidad.

**Figura 61. Interfaz de Información**



- **Interfaz de gestión de usuarios**

**Descripción:** En esta pantalla el usuario administrador del prototipo puede administrar los usuarios disponibles en la base de datos (ver figura 62). El administrador puede hacer consultas, actualizar, eliminar y crear registros si lo desea.

Figura 62. Interfaz de gestión de usuarios



- **Interfaz de gestión de comparendos**

**Descripción:** En esta interfaz el usuario administrador del prototipo puede administrar los registros de comparendos disponibles en la base de datos (ver figura 63). El administrador puede hacer consultas, actualizar y eliminar registros si es necesario.

Figura 63. Interfaz de gestión de comparendos

LIBRETA No.

COMPARENDO No.

1. FECHA Y HORA  
07/05/2005 15:04

2. LUGAR DE LA INFRACCIÓN  
VIA KILOMETRO O SITIO, DIRECCIÓN Y CIUDAD

3. PLACA (MARQUE LAS LETRAS)

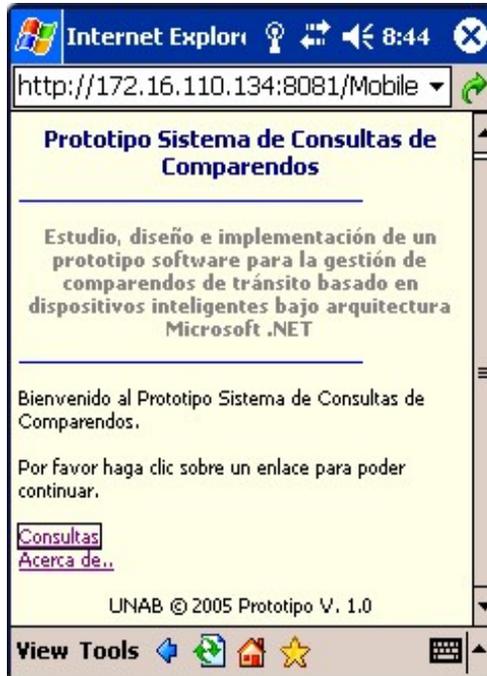
Bienvenido a la Administración de Comparendos....!!  
Por favor seleccione una opción en el menú principal para continuar...

#### 4.1.3.3 Prototipo de consultas de comparendos

- **Interfaz de inicio**

**Descripción:** Esta pantalla da la bienvenida al ingresar a la página y permite tanto a un usuario particular como a un agente de tránsito consulta el estado de cuenta de un vehículo (ver figura 58). El usuario que navegue por ésta página puede ingresar a la interfaz de consultas o a la página de información del prototipo.

Figura 64. Interfaz de inicio



- **Interfaz de consultas de comparendos**

**Descripción:** Esta interfaz brinda a cualquier usuario la posibilidad de consultar el estado de cuenta de un vehículo (ver figura 65). El usuario que navegue por ésta página puede ingresar a la interfaz de consultas o a la página de información del prototipo. El sistema muestra al usuario la información relacionada con el estado de cuenta del vehículo que desee buscar. Se deben ingresar las letras de la placa del vehículo y los números por aparte. El botón *Buscar* es el encargado de realizar la búsqueda en la base de datos y de enviarla al usuario.

Figura 65. Interfaz de consultas de comparendos

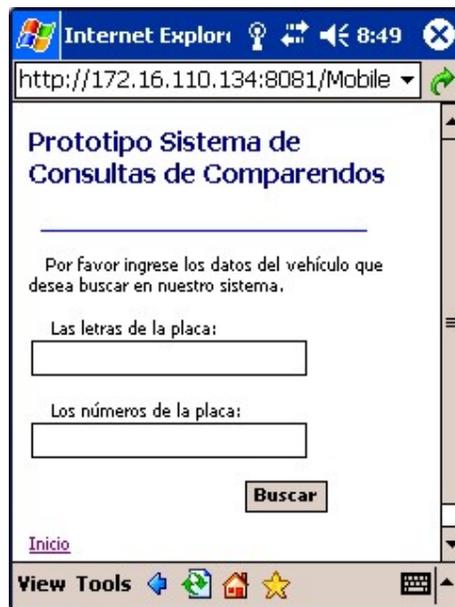
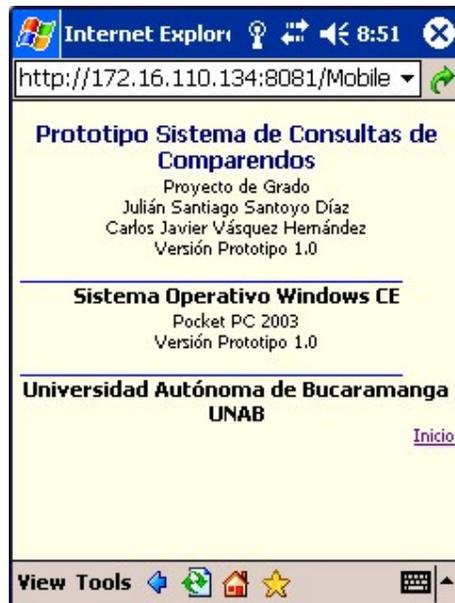


Figura 66. Interfaz de consultas de comparendos



- **Interfaz de información**

**Descripción:** Esta pantalla del prototipo proporciona puede visualizar una breve información de la aplicación (ver figura 60). En la parte de la información de la aplicación, se puede apreciar que es una primera versión del prototipo software. También podemos encontrar las personas que han desarrollado la aplicación. Se ha colocado el nombre de la Universidad Autónoma de Bucaramanga (UNAB), debido a que es un prototipo software académico, impulsado por un proyecto de grado de dicha universidad.

#### **4.1.4 Implementación**

Se realizaron tres prototipos totalmente funcionales, los cuales hacen parte del sistema de gestión de comparendos y todos interactúan entre sí. Se desarrollo una página WAP de consultas estado de cuenta de un vehículo, y un prototipo que permite la creación de un comparendo de tránsito desde un dispositivo móvil, incluyendo la sincronización de las bases de datos tanto de la Pocket PC como del servidor. Finalmente, se realizó una aplicación que opera sobre sistemas Windows 2000 o superior; la cual se encarga de gestionar la base de datos del servidor.

La implementación del sistema de gestión de comparendos se realizó bajo el entorno de desarrollo Visual Studio .NET, el cual ofrece un buen soporte a la hora de desarrollar aplicaciones para dispositivos móviles.

Entre esas ayudas encontramos la información detallada de la librería System.Data.SqlServerCe de la base de datos SQL Server CE (móvil y compacta); la cual nos permite acceso a la base datos local, acceso a los datos mientras se está desconectado y acceder a toda la funcionalidad de SQL Server CE.

SQL Server CE proporciona diferentes modos de trabajo:

- Aplicaciones On – line: Utilizan ADO.NET, la conexión con el servidor es continua y permite desarrollos rápidos.
- Aplicaciones Off – line: Utilizan ADO.NET o ADOCE, SQL CE para acceso a bases de datos locales, no hay ninguna conexión con el servidor.
- Aplicaciones Mixtas (o de sincronización): Ofrece un sistema mixto en donde la aplicación podrá conectar de modo asíncrono con el servidor y a su vez, podrá utilizar datos localmente, uso de protocolo HTTP/HTTPS, métodos de sincronización (RDA – Remote Data Access y Replicación).

Nosotros hemos utilizado el modo de trabajo para aplicaciones mixtas. Nuestro método de sincronización es el de acceso remoto a datos – RDA.

En el código de la aplicación se utilizó la notación húngara, la cual es un estándar de programación que nos permite normalizar el código para lograr entender las tareas que se realizan (ver tabla 4).

**Tabla 4. Convenciones para el nombramiento de controles**

Tipo de Control	Prefijo	Ejemplo
Botón de comando	cmd	cmdSalir
Etiqueta	lbl	lblSistema
Menú	mnu	mnuCarga
Cuadro de texto	txt	txtApellido

En la tabla 5, podemos observar el método mnuCarga\_Click, el cual utiliza el RDA desde el dispositivo móvil para cargar la base de datos SistemaComparendos que se encuentra en el servidor principal de la aplicación.

En la tabla 6, se puede apreciar el método `mnuSync_Click`, encargado de gestionar la sincronización desde el dispositivo móvil al servidor de la base de datos.

**Tabla 5. Código fuente del método `mnuCarga_Click`**

```
// Método mnuCarga_Click
private void mnuCarga_Click(object sender, System.EventArgs e)
{
    SqlCeConnection cn = null;
    SqlCeRemoteDataAccess rda = null;
    SqlCeEngine sqlEngine = null;
    try
    {
        // Crea una nueva BD's si no existe
        if (!File.Exists(@"\My
Documents\SistemaComparendos.sdf"))
        {
            sqlEngine = new SqlCeEngine();
            sqlEngine.LocalConnectionString = "Data
Source=\My Documents\" +
                "SistemaComparendos.sdf;" +
                "Password=siscomp;" +
                "Encrypt Database=True";
            sqlEngine.CreateDatabase();
            sqlEngine.Dispose();
        }
        else
        {
            // Abre una conexion a la BD's
            cn = new SqlCeConnection("Data Source=\My
Documents\" +
                "SistemaComparendos.sdf;Password=siscomp");
            cn.Open();
            SqlCeCommand cmd = cn.CreateCommand();
            // Hace un Drop a la tabla UserTable
            cmd.CommandText = "DROP TABLE USERTABLE";
            cmd.ExecuteNonQuery();
            // Cierra la conexion
            if (cn.State != ConnectionState.Closed)
            {
                cn.Close();
            }
        }
        // Isntancia un objeto RDA
        rda = new SqlCeRemoteDataAccess();
        // String de la conexion SQL Server
        string remoteConnectionString = "Provider=SQLOLEDB;" +
            "Data Source=pocketserver;" +
            "Initial Catalog=SistemaComparendos;" +
            "User Id=sa;" +
            "Password=sa";
        rda.InternetLogin = "administrador";
        rda.InternetPassword = "Qej8h8w54qe94";
    }
}
```

```

        rda.InternetUrl =
"http://172.16.110.134:8081/fieldagentrda/sscesa20.dll";
        rda.LocalConnectionString = "Data Source=\\My
Documents\\" +
            "SistemaComparendos.sdf;" +
            "SSCE:Database Password=siscomp";
        rda.Pull("UserTable",
            "Select * from UserTable",
            remoteConnectionString,
            RdaTrackOption.TrackingOnWithIndexes,
            "UserTableErrorTable");
    }
    catch (SqlCeException sqllex)
    {
        foreach (SqlCeError sqlError in sqllex.Errors)
        {
            MessageBox.Show(sqlError.Message, "Error");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message, "Error");
    }
    finally
    {
        rda.Dispose();
    }
}

```

**Tabla 6. Código fuente del método mnuSync\_Click**

```

// Método mnuSync_Click
private void mnuSync_Click(object sender, System.EventArgs e)
{
    SqlCeRemoteDataAccess rda = null;
    try
    {
        // Isntancia un objeto RDA
        rda = new SqlCeRemoteDataAccess();
        // String de la conexion SQL Server
        string remoteConnectionString =
"Provider=SQLOLEDB;" +
            "Data Source=pocketserver;" +
            "Initial Catalog=SistemaComparendos;"
+
            "User Id=sa;" +
            "Password=sa";
        rda.InternetLogin = "administrador";
        rda.InternetPassword = "Qej8h8w54qe94";
        rda.InternetUrl =
"http://172.16.110.134:8081/fieldagentrda/sscesa20.dll";
        rda.LocalConnectionString = "Data
Source=\\My Documents\\" +
            "SistemaComparendos.sdf;" +

```

```
        "SSCE:Database Password=siscomp";
        rda.Push("UserTable", remoteConnectionString,
RdaBatchOption.BatchingOn);
    }
    catch (SqlCeException sqlEx)
    {
        foreach (SqlCeError sqlError in
sqlEx.Errors)
        {
            MessageBox.Show(sqlError.Message,
"Error");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message, "Error");
    }
    finally
    {
        rda.Dispose();
    }
}
```

## CONCLUSIONES

- El lenguaje de programación C sharp es sin duda mejor si se conoce que la aplicación se hará bajo algún sistema Windows y se cuenta con un buen tiempo para la construcción de la misma; o Java, si se cuenta con un tiempo limitado para el desarrollo y no se conoce bajo que sistema operativo se desea aplicar.
- Debido a la amplia gama de servicios que este tipo de dispositivos ofrecen, se puede llegar a pensar en aplicarlos en problemas de actualidad, aprovechando su facilidad y confiabilidad en el manejo, manipulación y envío de la información sobre las tecnologías que ofrecen (WLAN, Bluetooth, GSM/GPRS/CDMA y IrDA).
- En Colombia, debido a la inseguridad y a la inestabilidad económica, muchos de los proyectos que se plantean con estas nuevas tecnologías no se llevan a cabo, por el temor de invertir en algo que nos traerá muchos problemas o por que no se pueden aplicar en nuestro entorno. Deberían existir empresas encargadas de ofrecer mayor información acerca de éste tipo de soluciones móviles que son muy confiables y fáciles de adaptar a nuestros problemas cotidianos.
- El entorno de desarrollo que ofrece Visual Studio .Net facilita el diseño (GDI+ y Windows Forms); la conectividad (Web services, TCP/IP e IrDA); y el acceso a datos (ADO.NET y XML) de aplicaciones Web móviles para dispositivos inteligentes. Estas herramientas soportadas por el Internet, ayudarían a mejorar procesos y procedimientos, minimizar costos y tiempos, tener integridad en los datos, confiabilidad y calidad de la información.

- Se presentaron algunos inconvenientes en la parte de desarrollo de la aplicación de consultas WAP, ya que no había interoperabilidad con los distintos entornos de desarrollo, inicialmente se hicieron pruebas con diferentes programas y navegadores utilizando lenguajes como PHP y WML sobre WINWAP, NMIT (Nokia Mobile Internet Toolkit) WAPTOR, OpenWave y WebMatrix sin encontrar una forma efectiva para crear la aplicación. La solución se desarrollo con el Visual Studio.NET con una Mobile Web Application donde la capacidad de las conexiones es más fácil de implementar permitiendo que las consultas sean realizadas satisfactoriamente.
- Es muy importante hablar de los problemas que se presentaron a la hora de comunicarnos con la base de datos, los métodos de sincronización y cual se adapta mejor a las características de los requerimientos del sistema, puesto que para el SQL Server 2000 el mejor método es el RDA con sus dos fases el PULL y el PUSH y para otras administradores de bases de datos el método Merge Replication el cual es un mecanismo de sincronización mas sofisticado, el cual requiere de una administración de bases de datos e instalaciones y configuraciones mas complejas.
- Durante el desarrollo del prototipo para un dispositivo inteligente, se hicieron algunas pruebas previas sobre emuladores con sistema operativo Windows CE como el de Smartphone 2003, Pocket PC 2003; y sobre un dispositivo inteligente real con sistema operativo Windows Mobile 2003 Second Edition (Hp IPaq 1710). Dando como resultado la correcta funcionalidad en cada uno de los dispositivos móviles los cuales tengan como base un sistema operativo Windows

## **RECOMENDACIONES Y TRABAJOS FUTUROS.**

Como recomendaciones a trabajos futuros sobre aspectos que se salen del alcance del proyecto, se podrían establecer los mismos métodos de sincronización y consultas de comparendos, pero por medios diferentes como vía wireless para reducir más los procesos y hacer la labor de comparendos mucho mas rápida y optima, como también tratar de complementar con los demás procesos que se manejan en una dirección de transito como determinar en el sistema que usuarios tienen impuestos por pagar, cuales están en proceso de audiencia por inconformidad con el comparendo impuesto, etc. En la ciudad de Bucaramanga a partir del mes de Agosto del 2005 se empezara a implementar una infraestructura tecnológica de antenas Wi-Fi para cubrir toda el área metropolitana con el servicio de redes inalámbricas facilitando esta labor y pues teniendo ya esta infraestructura acoplada en la ciudad se podrían plantear proyectos utilizando mas el GPS con PDA's para control de alférez con el sistema de comparendos, mapas de posicionamiento global para que los alférez detecten el sitio exacto de un accidente o problemas de semaforización. Determinar la posición de los vehículos de transporte publico, para que las compañías sepan en que parte del recorrido van etc. Las aplicaciones que se pueden implementar pueden ser variables de las existentes pero con la versatilidad que maneja la arquitectura de redes inalámbricas.

## BIBLIOGRAFÍA

**FRANCESCO**, Balena; **RICHTER**, Jeffrey. Applied Microsoft .NET Compact Framework. Programming Visual Basic. NET. Editorial Mc Graw Hill, First Edition, 2002

**Grundgeiger**, Dave. Programming Visual Basic .NET, publicaciones: o'Reilly, First Edition January 2002.

**I. SOMMERVILLE**, Ingeniería de Software, Addison Wesley Publishers Limited, Sexta Edición, México DF, 2002.

**INTEL**, DISEÑO DE APLICACIONES INALÁMBRICAS PARA MÓVILES Y SU ESTRUCTURA. Disponible en:

<http://www.intel.com/espanol/update/contents/mo11031.htm>

Sitio Web visitado el día 10 de agosto de 2004.

**KRELL**, Bruce E. Pocket PC Developer's Guide. Editorial Mc Graw Hill, First Edition, 2002.

**MERCER**, Dave. Fundamentos de Programación en XML. Editorial Mc Graw Hill, 2002.

**MICROSOFT CORPORATION**, Microsoft Solution Framework White Paper, Microsoft Corporation, December 1999.

**MICROSOFT CORPORATION**, Microsoft Solutions: The Collaboration in Corporative World, Microsoft Corporation, March 1999.

**MICROSOFT**, SISTEMA OPERATIVO PARA DISPOSITIVOS MÓVILES DE ULTIMA GENERACIÓN Y SUS ASPECTOS GENERALES. Disponible en:  
<http://www.microsoft.com/windowsmobile/default.mspx>  
Sitio Web visitado el día 14 de agosto de 2004.

**O'BRIEN**, J.A. Management Information Systems: Managing Information Technology in the E-Business Enterprise, McGraw-Hill/Irwin, Fifth Edition, New York, 2001.

**PAYNE**, Chris. Teach Yourself ASP.NET in 21 Days, Editorial Prentice Hall, Second edition, 2003.

**PRESSMAN**, Roger S. Ingeniería del Software: Un enfoque practico. Editorial McGraw Hill Quinta Edición, 2002

**STEVE MILROY, KEN COX, DOTTHATCOM.COM, AMIT KALANI, KEN COX, .NET** Mobile Web Developer's Guide, Editorial: Syngress Publishing, Inc. 2003.

**UML**, INGENIERÍA DEL SOFTWARE CICLOS Y DESARROLLO DE SOFTWARE, Disponible en:  
<http://www.uml.org>  
Sitio Web visitado el día 22 de agosto de 2004.

**WIGLEY**, Andy; **SUTTON**, Mark; **MACLOED**, Rory; **BURBIDGE**, Robert; **WHEELWRIGHT**, Stephen. Microsoft .NET Compact Framework (Core Reference), Microsoft Press, Second Edition, 2003.

**WIGLEY**, Andy and **ROXBURGH**, Peter. Building Microsoft® ASP.NET Applications for Mobile Devices, Microsoft Press, Second Edition, 2003.