

ESCUELA POLITECNICA DEL EJERCITO

**DEPARTAMENTO DE ELECTRICA
Y ELECTRONICA**

**CARRERA DE INGENIERIA EN ELECTRONICA,
TELECOMUNICACIONES**

**PROYECTO DE GRADO PARA LA OBTENCION
DEL TITULO EN INGENIERIA**

**Diseño e implementación de un sistema de
vigilancia remota para una residencia utilizando
plataformas GPRS e Internet**

**A. Soledad Zapata Y.
Andrés R. Vallejo P.**

**SANGOLQUI – ECUADOR
2007**

CERTIFICACIÓN

Por medio de la presente se certifica que el Proyecto de Tesis de Grado: “Diseño e implementación de un sistema de vigilancia remota para una residencia utilizando plataformas GPRS e Internet” ha sido elaborado en su totalidad por los señores A. Soledad Zapata Y. y Andrés R. Vallejo P.

Siendo todo el proceso de elaboración de este proyecto dirigida y supervisada por su director y codirector

Ingeniero Rodrigo Silva
DIRECTOR

Ingeniero Darío Duque
CODIRECTOR

Agradezco a mis padres por su gran apoyo y compañía mismos que nunca me faltaron y siempre me ayudaron mucho, a mis amigos por estar siempre junto a mí, a mis compañeros de trabajo por su gran comprensión y apoyo en los momentos más críticos, a Byron Martínez por su ayuda prestada y presencia constante.

Y sobre todo agradezco a mi hija por su gran comprensión y respaldo

A. Soledad Zapata Y.

Agradezco a Dios por su ayuda, a mis padres que lucharon cada día de mi carrera por darme lo que necesitaba por su comprensión y ayuda incondicional como solo ellos saben hacerlo, a mi familia que siempre fue un apoyo moral, a mi compañeros y amigos que siempre supieron apoyarme y darme incentivo por seguir adelante frente a toda adversidad, y a una persona muy especial María José que en este poco mas de un año junto a mi vida me ha dado mucho incentivo y ganas de luchar para seguir adelante.

Andrés R. Vallejo P.

Este trabajo lo dedico a mis padres que gracias a ellos tengo esta maravillosa oportunidad de haber culminado mi carrera y poder ser un profesional, ya que ellos fueron los que siempre estuvieron pendientes en las buenas y en las malas de darme este gran momento en el cual se ve reflejado toda la lucha que tenemos que realizar todos los estudiantes cuando queremos superarnos realmente, a mis amigos y compañeros, y nuevamente a María José una persona que me ha hecho ver el mundo de otra forma “lleno de apoyo”.

Andrés R. Vallejo P.

Este trabajo lo dedico a mis padres por el gran esfuerzo que representó para ellos el darme la educación, a mis amigos pues juntos compartimos muchos momentos muy especiales en nuestra vida universitaria, a Byron Martínez porque sabe lo que representa este trabajo, y a mi hija que es la razón de mi existencia

A. Soledad Zapata Y.

PRÓLOGO

En el Proyecto de Tesis de Grado: “Diseño e implementación de un sistema de vigilancia remota para una residencia utilizando plataformas GPRS e Internet” se buscó el implementar un sistema que, utilizando los conocimientos adquiridos durante la carrera de Ingeniería Electrónica especialidad en Telecomunicaciones, monitoree una vivienda, a través del censo de 6 eventos dentro de ella, para que cuando alguno de estos eventos vigilados presente algún tipo de actividad, esta información sea enviada a través de la red GPRS hacia un servidor de base de datos, para luego ser presentados a través de una interfaz web a la cual podrán acceder tanto el dueño de la vivienda, para conocer el estado de su propiedad, como el supervisor de un centro de vigilancia, para así asegurara al propietario que su vivienda está segura.

INDICE

CERTIFICACIÓN	i
AGRADECIMIENTO	ii
DEDICATORIA	iii
PRÓLOGO	iv
CAPITULO I GENERALIDADES	1
1.1. GENERALIDADES.....	1
1.1.1. INTRODUCCIÓN	1
1.1.2. ANTECEDENTES	2
1.1.3. JUSTIFICACIÓN E IMPORTANCIA.....	2
1.1.4. OBJETIVOS	3
1.1.4.1.OBJETIVO GENERAL	3
1.1.4.2.OBJETIVOS ESPECÍFICOS	3
1.2. EL SISTEMA GPRS	4
1.2.1. EL SISTEMA GSM.....	4
1.2.1.1 CONCEPTO.....	4
1.2.1.2 CARACTERÍSTICAS	5
1.2.1.3.BENEFICIOS QUE OFRECE EL SISTEMA GSM.....	7
Beneficios para el usuario.....	7
Beneficios para el operador	8
1.2.1.4.ARQUITECTURA DEL SISTEMA GSM.....	9
Subsistema de la Radio Base (BSS).....	11
Subsistema de Red y de Conmutación (NSS).....	12
Subsistema de Soporte de Operación	13
1.2.1.5.LA TARJETA SIM.....	14
1.2.1.6.CANALES LÓGICOS DE GSM	16
1.2.2. EL SISTEMA GPRS: CONCEPTO Y CARACTERÍSTICAS PRINCIPALES	17
1.2.3. COMPARACIÓN CON EL SISTEMA GSM	19
1.2.4. BENEFICIOS QUE OFRECE EL SISTEMA GPRS	21
1.2.4.1.VENTAJAS DEL GPRS PARA LA OPERADORA.....	21
1.2.4.2.VENTAJAS DEL GPRS PARA EL USUARIO.....	22
1.2.5. ARQUITECTURA DEL SISTEMA	24
1.2.5.1.INFRAESTRUCTURA DE RADIO.....	24
1.2.5.2.RED INTERNA DE LAS OPERADORAS,	25
1.2.5.3.ESTACIÓN BASE	26
1.2.6. PRINCIPALES FABRICANTES DE INFRAESTRUCTURA PARA REDES GSM – GPRS	26
1.2.7. PROYECCIONES DEL SISTEMA GPRS HACIA EL FUTURO	27
1.3. SERVIDOR DE BASES DE DATOS	28
1.3.1. CARACTERÍSTICAS	28
1.3.1.1.DESARROLLO	29
1.3.1.2.FUNCIONALIDADES EN LA PROGRAMACIÓN.....	29
1.3.1.3.ACCESO Y TRABAJO CON BASES DE DATOS	30
1.3.1.4.MANIPULACIÓN DE LOS DATOS	31
1.3.1.5.SENTENCIAS Y FUNCIONES.....	32
1.3.1.6.CONECTIVIDAD.....	32
1.3.1.7.SEGURIDAD	33
1.3.1.8.CLIENTES Y HERRAMIENTAS	33
1.3.1.9.LOCALIZACIÓN	33
1.3.1.10.ESCALABILIDAD Y LÍMITES	34

1.3.1.11.DIMENSIONES MÁXIMAS DE LAS TABLAS SQL.....	34
1.3.2. BENEFICIOS.....	36
1.3.2.1.VENTAJAS EN EL USO DE BASES DE DATOS.	36
1.3.2.2.VENTAJAS DE SQL	37
1.4. LA PLATAFORMA .NET	37
1.4.1. INTRODUCCIÓN	37
1.4.2. BENEFICIOS QUE PRESENTA .NET.....	38
1.4.3. COMMON LANGUAGE RUNTIME (CLR)	39
1.4.3.1.PRINCIPALES CARACTERÍSTICAS Y VENTAJAS DE CLR.....	40
1.4.4. MICROSOFT INTERMEDIATE LANGUAGE (MSIL)	41
1.4.4.1.PRINCIPALES CARACTERÍSTICAS Y VENTAJAS DE MSIL.....	41
1.4.5. COMPILADOR JUST IN TIME (JIT O JITTER)	42
1.4.6. APLICACIONES Y TECNOLOGÍAS QUE FORMAN PARTE DE LA PLATAFORMA .NET	43
1.4.7. EL .NET FRAMEWORK	43
1.4.7.1.OBJETIVOS DEL .NET FRAMEWORK.....	43
1.4.7.2.COMPONENTES PRINCIPALES DEL .NET FRAMEWORK	44
1.4.8. MEJORAS DE .NET	44
CAPITULO II DISEÑO DEL SISTEMA	47
2.1. DESCRIPCIÓN FUNCIONAL DEL SISTEMA.....	47
2.1.1. ANÁLISIS DE REQUERIMIENTOS.....	47
2.1.2. EVENTOS A CONTROLAR	49
2.1.3. SERVICIOS A PRESTARSE	49
2.1.3.1.MONITOREO REMOTO VÍA INTERNET.....	52
2.1.3.2.COMUNICACIÓN ENTRE EL CENTRO DE MONITOREO Y LA VIVIENDA: GPRS	53
2.1.3.3.ACCESO EN EL ESQUEMA 24-7-365.....	54
2.1.3.4.SEÑAL DE EMERGENCIA	54
2.2. DIAGRAMAS DE BLOQUES FUNCIONALES DEL SISTEMA	55
2.2.1. BLOQUE DE TOMA Y PROCESAMIENTO DE DATOS	56
2.2.1.1.SUB BLOQUE DE MONITOREO Y TRANSFORMACIÓN	57
2.2.1.2.SUB BLOQUE DE PROCESAMIENTO DE LA INFORMACIÓN	57
2.2.1.3.SUB BLOQUE DE CONTROL DE EVENTOS	57
2.2.2. TRANSMISIÓN INALÁMBRICA DE INFORMACIÓN	58
2.2.2.1.SUB BLOQUE EMISOR.....	58
2.2.2.2.SUB BLOQUE ENLACE	59
2.2.2.3.SUB BLOQUE RECEPTOR	59
2.2.3. INTERFAZ DE USUARIO	59
2.2.4. BASE DE DATOS	60
2.3. DISEÑO DE LOS COMPONENTES DEL SISTEMA.....	60
2.3.1. BLOQUE DE TOMA Y PROCESAMIENTO DE DATOS	60
2.3.1.1.SUB BLOQUE DE MONITOREO Y TRANSFORMACIÓN	60
2.3.1.2.SUB BLOQUE DE PROCESAMIENTO DE LA INFORMACIÓN	62
Programación del Microcontrolador	64
2.3.1.3.SUB BLOQUE DE CONTROL DE EVENTOS	64
Activación o desactivación de señal de alarma audible: sirena	65
Ejecución de una llamada telefónica.	65
2.3.1.4.TARJETA ELECTRÓNICA	66
2.3.1.5.DIAGRAMAS ELECTRÓNICOS	68
2.3.2. TRANSMISIÓN INALÁMBRICA DE INFORMACIÓN	68
2.3.2.1.SUB BLOQUE EMISOR.....	69

Generación de datos a enviar al Centro de Monitoreo así como los comandos necesarios para ejecutar dicho envío.....	70
Envío de la información: Trama de datos.....	71
Control y comunicación con el modem: Comandos AT	73
Propósito de los Comandos AT	73
Comunicación con el modem	73
2.3.2.2.SUB BLOQUE ENLACE: DETALLES DE LA CONEXIÓN.....	74
Conexión a través de Sockets	76
2.3.2.3.SUB BLOQUE RECEPTOR	77
Elaboración del Servicio Windows	78
2.3.2.4.DIAGRAMA ELECTRÓNICO.....	79
2.3.3. INTERFAZ DE USUARIO	79
2.3.3.1.CENTRO DE MONITOREO DE VIVIENDAS	80
2.3.3.2.DIAGRAMA DE FLUJO	81
2.3.4. BASE DE DATOS	81
2.3.4.1.CONFIGURACIÓN DE LA BASE DE DATOS	82
2.4. ESPECIFICACIONES TÉCNICAS DE LOS ELEMENTOS DEL SISTEMA.....	82
2.4.1. TARJETA ELECTRÓNICA	82
2.4.2. MÓDEM GPRS	83
2.4.3. INTERFAZ DE USUARIO.....	85
2.4.4. BASE DE DATOS	86
CAPITULO III IMPLEMENTACIÓN DEL SISTEMA	87
3.1. CARACTERÍSTICAS DE LA TARJETA ELECTRÓNICA.....	87
3.1.1. FUENTE DE ALIMENTACIÓN.....	88
3.1.2. CENTRO DE PROCESAMIENTO	89
3.1.3. PERIFÉRICOS PARA INTERACCIÓN CON EL USUARIO	89
3.2. CARACTERÍSTICAS DEL ENLACE GPRS	90
3.2.1. INICIO DE LA COMUNICACIÓN: CONFIGURACIÓN DEL MODEM	91
3.2.1.1.VELOCIDAD DE COMUNICACIÓN	91
3.2.1.2.ACCESO A LA RED DEL PROVEEDOR: CONFIGURACIÓN DEL MODEM.....	92
Tamaño del buffer o tiempo de espera	93
Tipo de Sesión PAD.....	94
Comportamiento: Cliente o Servidor	95
APN a la que pertenece	95
Dirección IP y número del puerto al que se va a enviar los datos.....	96
Modo de auto registro del modem.....	97
Dirección IP y de Puerto de origen	97
Otros comandos	97
3.3. PRESENTACIÓN DEL APLICATIVO WEB	98
3.3.1. VENTANA DE INICIO	100
3.3.2. CREACIÓN EDICIÓN Y BÚSQUEDA DE USUARIOS	100
3.3.3. MONITOREO.....	101
3.3.4. SERVICIO REMOTO.....	102
3.4. CARACTERÍSTICAS DE LA BASE DE DATOS	102

CAPÍTULO IV PRUEBAS DE EVALUACIÓN DEL SISTEMA	103
4.1. BLOQUE DE TOMA Y PROCESAMIENTO DE DATOS	103
4.1.1. SUB BLOQUE DE MONITOREO Y TRANSFORMACIÓN	103
4.1.2. SUB BLOQUE DE PROCESAMIENTO DE LA INFORMACIÓN	104
4.1.3. SUB BLOQUE DE CONTROL DE EVENTOS	104
4.1.4. PRUEBAS DEL BLOQUE COMPLETO	105
4.2. TRANSMISIÓN INALÁMBRICA DE INFORMACIÓN	105
4.2.1. PRUEBAS MODEM – PC	106
4.2.2. PRUEBAS MODEM – TARJETA ELECTRÓNICA	108
4.3. INTERFAZ DE USUARIO Y BASE DE DATOS	110
4.3.1. DESARROLLO Y VISUALIZACIÓN DE LAS PÁGINAS WEB	110
CAPÍTULO V ANÁLISIS ECONÓMICO	134
5.1. ANÁLISIS DE COSTOS	134
5.1.1. COSTOS CENTRALES	134
5.1.1.1. EQUIPOS Y SERVICIOS PARA BASE DE DATOS Y CENTRO DE MONITOREO	135
5.1.1.2. SERVICIOS REQUERIDOS	135
5.1.2. COSTOS POR USUARIO	135
5.1.2.1. MATERIALES DE LA TARJETA ELECTRÓNICA	135
5.1.2.2. MATERIALES Y EQUIPOS PARA ENLACE GPRS	136
5.1.2.3. CONSUMO DE GPRS	137
5.1.2.4. INSTALACIÓN Y CONFIGURACIÓN DEL SISTEMA EN LA VIVIENDA	138
5.1.3. RESUMEN DE VALORES	138
5.2. PERSPECTIVA DE COMERCIALIZACIÓN	142
CONCLUSIONES Y RECOMENDACIONES	144
REFERENCIAS BIBLIOGRÁFICAS	146
ANEXOS	148
INDICE DE FIGURAS	ix
INDICE DE TABLAS	x
GLOSARIO	xi

CAPÍTULO I

GENERALIDADES

1.1. GENERALIDADES

1.1.1. INTRODUCCIÓN

En la actualidad los avances tecnológicos permiten el desarrollo de un sin número de equipos y sistemas para mejorar el modo de vida de las personas, y es en dos de los sistemas más utilizados que nos hemos basado para la elaboración de este proyecto:

- ♦ Sistema de comunicación celular por GPRS
- ♦ Sistema de comunicación de datos por INTERNET

GPRS es la tecnología inalámbrica de transmisión de datos por medio de paquetes más ampliamente soportada en el mundo y se desarrolla a partir de los abonados GSM en más de 210 países y territorios de todo el mundo.

A lo largo de la historia, las telecomunicaciones acompañaron el desarrollo, tanto en la actividad económica como en los aspectos sociales y culturales. Contribuyeron al crecimiento de la producción y a la mejora de las condiciones de vida de la población, y se fueron adecuando a las concepciones políticas de cada época

Una de las bondades de la tecnología de Internet es su flexibilidad para ajustarse rápidamente a necesidades cambiantes. Hoy se manejan todo tipo de equipos en forma remota controlados por Internet, se distribuye información, se almacenan imágenes y se maneja el comercio electrónico, a través de las mismas

tecnologías. Esta flexibilidad y apertura, si se quiere llamar así, hacen también que haya falencias en los sistemas de seguridad de los mismos.

Estar conectados a Internet es ya una necesidad cotidiana, seamos la más grande multinacional, o simplemente un habitante del mundo. Desde las soluciones más sencillas, como el envío de correo electrónico, hasta soluciones complejas de vídeo vigilancia y monitoreo, se efectúan a través de Internet y sus tecnologías.

Si bien se ha avanzado mucho en el uso de la red, también es cierto que la red inició como un proceso colaborativo entre colegas para su utilización en la investigación científica. Nunca se pensó, hace 30 años, que aparecería la Web, el protocolo http y demás elementos que permiten que hoy manejemos ambientes gráficos en la misma, y por lo tanto convirtiéndolo en un excelente elemento de mercadeo.

1.1.2. ANTECEDENTES

El incremento de la delincuencia, que hemos podido observar en los tiempos actuales, ha llevado a la ciudadanía a buscar mecanismos de vigilancia y protección de sus bienes.

En busca de soluciones nos interesamos en desarrollar un sistema de seguridad electrónico teniendo en cuenta la importancia que un sistema como el nuestro ha adquirido en los últimos tiempos.

Se buscó el diseñar un sistema sencillo, práctico pero a la vez lo mas robusto posible, para así obtener un producto cuyo costo sea accesible y su confiabilidad sea elevada.

1.1.3. JUSTIFICACIÓN E IMPORTANCIA

Los sistemas de seguridad desde las más básicos y poco costosos consistentes en personas que cuiden las viviendas, hasta los más sofisticados

constituidos por equipos de alarma con sensores, consolas para el monitoreo de eventos, y personas que supervisen los datos entregados por dichos equipos. Todos ellos son dependientes de un operador y no ofrecen al usuario una forma de conocer el estado de su vivienda en cualquier lugar y momento que este se encuentre.

La dependencia de un operador convierte al sistema de seguridad en una solución muy vulnerable, pues su correcto desempeño se ve directamente influenciado por las cualidades de honradez y capacidad del operador, características que jamás podrán ser respaldadas al cien por ciento por parte de una empresa.

El sistema propuesto se basará en un conjunto de servicios, bibliotecas, aplicaciones y un sistema de comunicación de datos vía GPRS e internet que permite registrar y procesar en el computador del usuario del servicio

Actualmente este tipo de soluciones son utilizadas en proyectos de gran envergadura, por su flexibilidad para adecuarse rápidamente a nuevas necesidades no solo en sistemas de seguridad sino también en otros sistemas de monitoreo tales como los sistemas de control de flota en el área del transporte.

1.1.4. OBJETIVOS

1.1.4.1. OBJETIVO GENERAL

Realizar el diseño e implementación de un sistema de vigilancia remota para una residencia utilizando GPRS e Internet.

1.1.4.2. OBJETIVOS ESPECÍFICOS

- ♦ Implementar una tarjeta electrónica de alarma utilizando un microcontrolador PIC y dispositivos sensores.

- ◆ Acoplar las señales de la tarjeta al modem GPRS para el envío de las señales al centro de gestión del sistema de seguridad.
- ◆ Configurar una base de datos en un servidor para almacenar la información obtenida de la gestión de las viviendas.
- ◆ Desarrollar un software de gestión y administración del sistema utilizando la plataforma .Net.
- ◆ Realizar pruebas y evaluación de funcionamiento del sistema en dos viviendas.
- ◆ Realizar un análisis económico y de mercado para la comercialización del sistema.

1.2. EL SISTEMA GPRS

1.2.1. EL SISTEMA GSM

1.2.1.1. CONCEPTO

Conocido como Sistema Global para Comunicaciones Móviles, GSM es la tecnología digital inalámbrica de segunda generación (2G) más ampliamente utilizada en el mundo, disponible en más de 210 países cuenta con más del 75% del total de clientes inalámbricos por lo que actualmente es la referencia mundial para redes de radio móviles

Sus inicios se remontan a 1982 cuando se crea el Group Spéciale Mobile (GSM) por parte de un consorcio de países europeos cuyo principal objetivo fue el crear una tecnología celular que de al usuario la posibilidad de roaming internacional de modo imperceptible y además soporte servicios avanzados.

Los primeros años de GSM se enfocaron principalmente a la selección de las técnicas de radio para la interfaz aérea. En 1986 se llevaron a cabo, en París, pruebas de campo de diversos sistemas propuestos para la interfaz aérea de GSM. En 1989 este proyecto queda a cargo de El Instituto Europeo de Normas

para Telecomunicaciones (ETSI) quien establece una serie de especificaciones técnicas para esta nueva tecnología, entre las que tenemos:

- ♦ Eficiencia espectral.
- ♦ Calidad de voz subjetiva.
- ♦ Costo del móvil.
- ♦ Viabilidad de la unidad móvil.
- ♦ Costo de la Radio Base o Estación Base.
- ♦ Habilidad de soportar nuevos servicios.
- ♦ Coexistencia con sistemas existentes.
- ♦ Utilización de una banda común, reservada en el ámbito internacional
- ♦ Estructura celular digital
- ♦ Sistema de acceso múltiple AMDT de banda estrecha
- ♦ Algoritmo de codificación de fuente de pequeña velocidad binaria
- ♦ Control de potencia y de transmisión/recepción
- ♦ Arquitectura OSI
- ♦ Señalización avanzada

En el año de 1991 se presentó la primera red GSM y a partir de esta fecha este sistema empezó con un crecimiento muy veloz, siendo adoptada también en países no europeos, con lo cual se hizo evidente que GSM sería una tecnología global y no europea; y así comenzó a ser conocido como "Sistema Global para Comunicaciones Móviles.

1.2.1.2. CARACTERÍSTICAS

Entre las principales características de GSM anotaremos:

- ◆ Provee una evolución fluida y de bajo costo a la tercera generación (3G).
- ◆ Brinda servicios de voz de alta calidad además de servicios de datos conmutados por circuitos.
- ◆ Trabaja en una amplia gama de bandas del espectro, entre ellas las de 850, 900, 1800 y 1900 MHz.
- ◆ Emplea un sistema de multiplexación por división de tiempo (TDM), con el cual se divide a un canal en seis ranuras de tiempo mismas que se asignan a cada usuario mientras transmite, con esto se logra múltiples transmisiones en un mismo canal sin interferirse entre ellas.
- ◆ Por la forma de optimizar el uso de cada canal, GSM provee una capacidad siete veces mayor que los sistemas de primera generación.
- ◆ Utiliza además la técnica de "frequency hopping" (salto de frecuencias) con lo que logra minimizar la interferencia de fuentes externas y dificulta las escuchas no autorizadas.
- ◆ En cuanto a la arquitectura funcional de un sistema de comunicaciones móviles celular, GSM añade una función de autenticación en base a un registro de identificación de equipo (RIE) y la información de la identidad del abonado computadas en el centro de identificación de usuario (CAu)
- ◆ Aunque en un principio estaba previsto que el sistema GSM contara con una transmisión de datos RDSI a 64 kbps, las limitaciones de coste y ancho de banda no lo permitieron. La velocidad máxima de transmisión de datos con GSM es en la actualidad de 9.600 bps lo que limita la cantidad de servicios. Por ejemplo, a 9,6 Kbps no se puede navegar por Internet de una manera satisfactoria.
- ◆ Tiempo de establecimiento de conexión, de 15 a 30 segundos.
- ◆ Pago por tiempo de conexión, lo que ocasiona una sustancial elevación de los costos.

La combinación de estos tres últimos factores negativos hace que GSM sea una tecnología mayoritariamente utilizada para la voz y no para los datos.

1.2.1.3. BENEFICIOS QUE OFRECE EL SISTEMA GSM

Beneficios para el usuario

Si analizamos a los involucrados en este sistema vemos que tanto usuarios como operadores son beneficiados.

Entre los beneficios al usuario tenemos:

- ♦ Cobertura: ya que se encuentra en más de 210 países del mundo. Este beneficio permite al usuario permanecer accesible por medio del mismo dispositivo móvil dentro de su país y en muchos otros más.
- ♦ Selección: debido a la gran cantidad de usuarios que GSM tiene los fabricantes de equipos se han visto en la obligación de presentar en el mercado una gran cantidad y variedad de modelos lo que le da al usuario la posibilidad de escoger.
- ♦ Calidad de voz: GSM presenta un servicio de transmisión de voz a una calidad muy alta.
- ♦ Flexibilidad: La opción de cambiar de equipo sin pasar por las molestias de configurar este nuevo es posible gracias a GSM quien trabaja con una tarjeta conocida como Módulo de Identidad del Abonado (SIM), la cual no solo nos ahorra el trabajo de configuración de nuestro dispositivo nuevo, adicionalmente nos da beneficios como es el no perder servicios de suscripción personalizados tales como mensajería. Además, gracias a esta tarjeta se le facilita al usuario el poder cambiar de operador GSM y mantener el mismo teléfono; la flexibilidad de la tarjeta SIM hace que las redes de datos basadas en GSM, tales como las GPRS, sean atractivas para diversas aplicaciones de datos.

- ♦ Servicios innovadores: gracias a las características de GSM esta ha sido pionera en la presentación de servicios como el de mensajes cortos (SMS) que soportan mensajes de texto y contenidos tales como ringtones. Otro servicio muy importante es el de capacidad de roaming que permite a los usuarios gozar de los servicios dentro y fuera de su área local o país.
- ♦ Movilidad: Hasta hace muy poco tiempo, el concepto de movilidad estaba asociado exclusivamente al terminal telefónico, pero actualmente se relaciona al usuario que utiliza sus servicios.
- ♦ Movilidad del terminal: El usuario dispone de un teléfono asociado a un número de la red y puede utilizarlo en cualquier lugar con cobertura.
- ♦ Movilidad personal: El usuario, y no el terminal, está asociado a un número de teléfono de la red que, de forma 'inteligente', le sigue en sus desplazamientos, pudiendo hacer uso de él desde cualquier teléfono, sea éste fijo o móvil.

Beneficios para el operador

Analizando el sistema desde la perspectiva del operador observamos muchos beneficios, entre los cuales podemos resaltar:

- ♦ Economías de escala: Debido a la gran acogida que ha tenido este sistema en el mundo la demanda de equipos terminales e infraestructura es también presentada en gran escala, lo que hace de este un mercado muy interesante para proveedores y desarrolladores de aplicaciones, y gracias a esta producción en gran volumen, los costos se reducen, permitiendo así a los operadores de GSM el establecer precios para sus servicios a un nivel más competitivo.
- ♦ Cobertura: a causa de la cantidad de países en los que este sistema opera, las empresas proveedoras de este servicio pueden ofrecer a sus clientes la posibilidad de usar su mismo equipo y número celular dentro y fuera del país, opción que es muy atractiva para personas de negocios mismos que

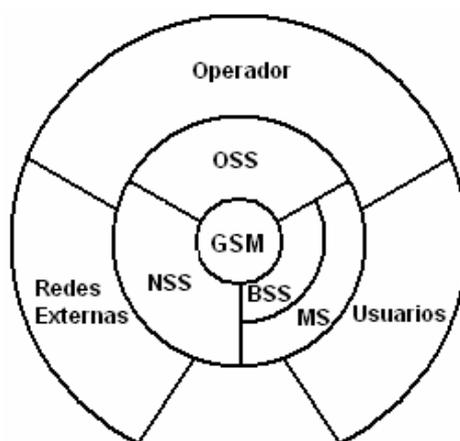
generan una gran parte de los ingresos a las operadoras, y así logran tener una mayor competitividad y aumentan sus ingresos.

- ♦ Flexibilidad: por la gran atención que ha prestado la comunidad GSM a las normas se ha logrado asegurar la interoperabilidad entre dispositivos e infraestructura de diversas marcas, lo que permite que el operador no este esclavizado a un solo proveedor tecnológico, adicionalmente tanto equipos como infraestructura se presentan para las bandas de frecuencia más populares, entre ellas las de 850 y 1900 MHz, permitiendo al operador seleccionar el que mejor se ajuste a sus necesidades de espectro y de mercado.
- ♦ Eficiencia: al dividir los canales en slots de tiempo permite a la operadora trabajar con mas llamadas, lo que representa mas usuarios que genran mas ganancias para la empresa, así se está utilizando los recursos de una forma mucho mejor que los sistemas anteriormente presentados, y si adicionalmente se utiliza optimizaciones tales como el Codec Adaptativo a Múltiples Velocidades (AMR), proveen un incremento adicional de casi tres veces más llamadas de voz simultáneas que la tecnología GSM básica.
- ♦ Capacidad de actualizarse: GSM es estructurado de modo tal que cada paso subsiguiente aprovecha el paso anterior logrando así que un equipo nuevo sea compatible con uno antiguo, lo que preserva tanto las inversiones como los clientes a lo largo de la migración haciéndola de una forma ligera y no traumante. Las normas que rigen la capacidad de actualización y la interoperabilidad de GSM están coordinadas y respaldadas por organizaciones internacionales clave tales como el Proyecto de Asociación para la Tercera Generación (3GPP), 3G Américas entre otros.

1.2.1.4. ARQUITECTURA DEL SISTEMA GSM.

La red GSM está compuesta por varias entidades interconectadas, pero puede reducirse principalmente a tres niveles conocidos también como subsistemas:

- ♦ Subsistema de Soporte de Operación: comprende la Estación Móvil que es en esencia el terminal del usuario;
- ♦ Subsistema de la Radio Base: comprende la Estación Base que controla las comunicaciones de radio del terminal.
- ♦ El Sistema de Red o Subsistema de Red y de Conmutación, que se encarga de realizar las conexiones entre los usuarios.



BSS: Base Station Subsystem

Subsistema de Radio Base

NSS: Network and Switching Subsystem

Subsistema de Red y Conmutación

OSS: Operational Subsystem

Subsistema Operacional

MS: Mobil Station

Unidad Móvil

Figura 1.1. Subsistemas de GSM

La forma de estimar la configuración de la red y tipo de equipos utilizados en la estación base necesarios para manejar todo el tráfico en una zona, es a través de la distribución de usuarios en el área metropolitana y rural y así se logra definir la configuración mas apropiada de radio bases para brindar el servicio GSM a la cantidad de usuarios requerida.

Adicionalmente en GSM se define Interfaces que son los medios por los cuales los Subsistemas se comunican y son tres las más importantes:

- ♦ Interfaz A entre: El MSC y el Controlador de la Radio Base (BSC).

- ♦ Interfaz Abis entre el BSC y el transceptor de la Radio Base (BTS), funciona a la velocidad de 2 Mbps según la recomendación G.703 de la ITU-T.
- ♦ Interfaz Um entre el BTS y la Unidad Móvil (MS).

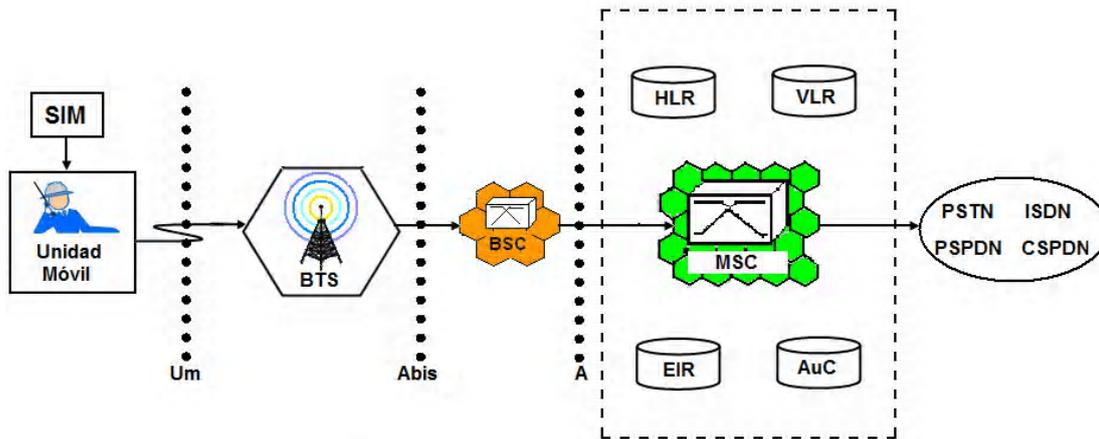


Figura 1.2. Arquitectura de un Sistema GSM

Subsistema de la Radio Base (BSS)

El subsistema de radio base proporciona y maneja las rutas de transmisión entre las unidades móviles y el subsistema de Red y Conmutación. Esto incluye el manejo de la interfaz de radio entre las unidades móviles y el resto del sistema GSM.

La Radio Base, Estación Base o Subsistema de la Estación Base (BSS) es el equipo físico que permite y propicia la comunicación entre el centro de conmutación y el equipo móvil, la comunicación entre la BSS y el móvil se da a través de la interfaz inalámbrica Um.

El sistema BSS se compone, a su vez, de

- ♦ Transceptores de Estación Base (BTS)
- ♦ Controladora de Estación Base (BSC)
- ♦ Transcodificador (TC)

- ♦ Centro de Operación y Mantenimiento (OMC-R). que proporciona los medios adecuados para que el proveedor del servicio controle y maneje el sistema GSM.

La comunicación entre los Tranceptores de Estación Base BTS y la Controladora de Estación Base BSC se realiza a través de la conexión Abis.

Subsistema de Red y de Conmutación (NSS)

Es el centro de procesamiento de la red, su componente principal es el centro de conmutación de servicios móviles. Se encarga de la gestión de una comunicación confiable entre la red GSM y todas las otras redes existentes de entre las cuales la mas importante es la Red Telefónica Pública Conmutada (PSTN), atendiendo los siguientes puntos:

- ♦ Registrar y verificar las comunicaciones y actualizar la localización del usuario.
- ♦ Gestionar los problemas de saturación.
- ♦ Direccionar llamadas.

Todas las llamadas entre subscriptores, sean estas llamadas originadas hacia el PSTN, originadas en el PSTN y terminadas en un subscriptor móvil, o llamadas entre subscriptores móviles, todas son gestionadas y enrutadas por medio del subsistema de conmutación NSS.

Los elementos fundamentales del NSS son:

- ♦ Centro de Conmutación Móvil que contiene integrados el Registro de Ubicación de Visitantes (MSC/VLR), quién se encarga del control de todas las llamadas hacia y desde la Red de Telefonía Pública Conmutada PSTN, así como dentro de la red móvil.
- ♦ Registro de Ubicación Local con el Centro de Autenticación (HLR/AuC) integrado, el registro HLR es el encargado de almacenar y proveer la

identidad y derechos de acceso a servicios de los usuarios. En este registro se encuentra almacenada la ubicación de los suscriptores con lo que se logra establecer el roaming. Por otro lado el registro AuC se encarga de la integridad de los suscriptores, así como de la seguridad de los datos en la red.

- ♦ Registro de Identidad de Equipo (EIR), es un registro opcional que puede ser incorporado en un sistema GSM ofrece la posibilidad opcional de incorporar un Este dispositivo puede ser utilizado para registrar teléfonos móviles robados con el propósito de deshabilitar el acceso a la red.
- ♦ Adicionalmente se puede implementar Servicios de Valor Agregado mediante la adición de equipamientos como Sistema de Mensajería Vocal (VMS) y Centro de Servicio de Mensajes Cortos (SMSC), elementos que hoy en día son tan comunes que se los considera ya como equipamientos estándar.

Subsistema de Soporte de Operación

Comprende las unidades móviles que son el equipo de radiocomunicación que utiliza el usuario para tener acceso a alguno de los servicios de telecomunicaciones que proporciona la red.

Una unidad móvil se puede dividir en dos partes:

- ♦ La primera corresponde al hardware y al software que soporta la interfaz entre el radio y el usuario. Este es en si el equipo móvil y puede ser adquirido en cualquier tienda especializada.
- ♦ La segunda parte incluye datos específicos del usuario y de la terminal en la forma de una tarjeta inteligente (Smart Card), la cual se puede considerar una terminal lógica y a la que se le conoce como tarjeta SIM. Esta tarjeta se la entrega el proveedor del servicio al usuario cuando éste se suscribe a sus servicios.

Un equipo terminal que no esté conectado a una tarjeta SIM, no está asociado con ningún usuario y en consecuencia no puede recibir ni realizar llamadas.

Una unidad móvil tiene los siguientes identificativos:

- ♦ IMEI (Internacional Mobile Equipment Identity).
- ♦ IMSI (Internacional Mobile Subscriber Identity).
- ♦ TMSI (Temporary Mobile Subscriber Identity)
- ♦ Número ISDN.

IMEI: Es un identificador únicamente del equipo, este es un código formado por 15 dígitos asignado por el fabricante del terminal móvil. La información que este contiene es la siguiente:

- ♦ TAC (Type Approval Code), 6 dígitos.
- ♦ FAC (Final Assembly Code), 2 dígitos.
- ♦ SN (Serial Number), 6 dígitos.
- ♦ SP (2 dígitos de reserva).

1.2.1.5. LA TARJETA SIM

La tecnología GSM quiere decir que el equipo terminal viene provisto de un chip, que le permite conservar los datos personalizados en el caso de que el usuario desase cambiar de equipo.

Esta tarjeta SIM (Subscriber Identity Module -Módulo de Identidad del Suscriptor-), al igual que una tarjeta de crédito o una tarjeta inteligente, almacena información segura relativa a la cuenta y los servicios suscritos por el abonado, y puede ser utilizada para soportar servicios tales como e-commerce inalámbrico o "m-commerce" (comercio móvil). La SIM les otorga a los operadores móviles la

flexibilidad de ofrecer servicios complementarios tales como Wi-Fi (LAN inalámbricas).

Una SIM puede insertar dentro de cualquier equipo GSM haciendo que este funcione con el perfil del usuario a quién se le vendió el servicio, esto permite que el usuario no esté esclavizado a un solo equipo y pueda cambiarlo cuando el así lo decida o por el contrario, hace que sea sencillo para el usuario cambiar de operador GSM y conservar el mismo equipo terminal. Esta flexibilidad hace que las redes de datos basadas en GSM, tales como las GPRS, sean atractivas para aplicaciones de datos (por ejemplo, telemetría).

La tarjeta SIM lleva la siguiente información:

- ♦ IMSI: Se almacena en la tarjeta SIM y se asigna al abonado al momento de suscribirse. Identifica a una unidad móvil dada. Se transmite por radio sólo si es necesario, se compone de 15 dígitos e incluye la siguiente información:
- ♦ MCC (Mobile Country Code), 3 dígitos.
- ♦ MNC (Mobile Network Country), 2 dígitos, home GSM PLMN.
- ♦ MSIN (Mobile Subscriber Identification).
- ♦ NMSI (National Mobile Mobile Subscriber Identity).
- ♦ Ki (Authentication Key).
- ♦ Subscriber information.
- ♦ Access Control class.
- ♦ Kc (Cipher Key).
- ♦ TMSI: Es asignado por el Registro de Ubicación de Visitantes (VLR) a los móviles. Permite identificar a un móvil únicamente dentro del área controlada por el VLR, el número máximo de bits que utiliza es de 32.
- ♦ Servicios adicionales de GSM.
- ♦ LAI (Identidad del área donde está ubicado)

- ♦ PLMN perdida.

1.2.1.6. CANALES LÓGICOS DE GSM

El enlace entre los Tranceptores de Estación Base y el equipo Terminal se compone de un enlace de Transmisión (subida o ida) y un enlace de Recepción (bajada o regreso). El canal físico, tanto para la Transmisión como para la Recepción es solamente una portadora para el enlace, aunque, como ya se ha visto, es multiplexada en tiempo y es en cada una de las ranuras de tiempo (time slots) en donde se van acomodando los canales lógicos.

Estos canales lógicos son los siguientes:

- ♦ TCH: Traffic Channel
- ♦ TCH/F: Traffic Channel / Full
- ♦ TCH/H: Traffic Channel / Half
- ♦ CCH: Control Channel
- ♦ BCH: Broadcast Channel
- ♦ CBCH: Cell Broadcast Channel
- ♦ CCCH: Common Control Channel
- ♦ DCCH: Dedicated Control Channel
- ♦ FCCH: Frequency Correction Chanel
- ♦ SCH: Synchronization Channel
- ♦ BCCH: Broadcast Control Channel
- ♦ PCH: Paging Channel
- ♦ AGCH: Access Grant Channel

- ♦ RACH: Random Access Channel
- ♦ SDCCH: Stand – Alone Dedicated Channel
- ♦ ACCH: Associated Control Channel
- ♦ SACCH: Slow Associated Control Channel
- ♦ FACCH: Fast Associated Control Channel

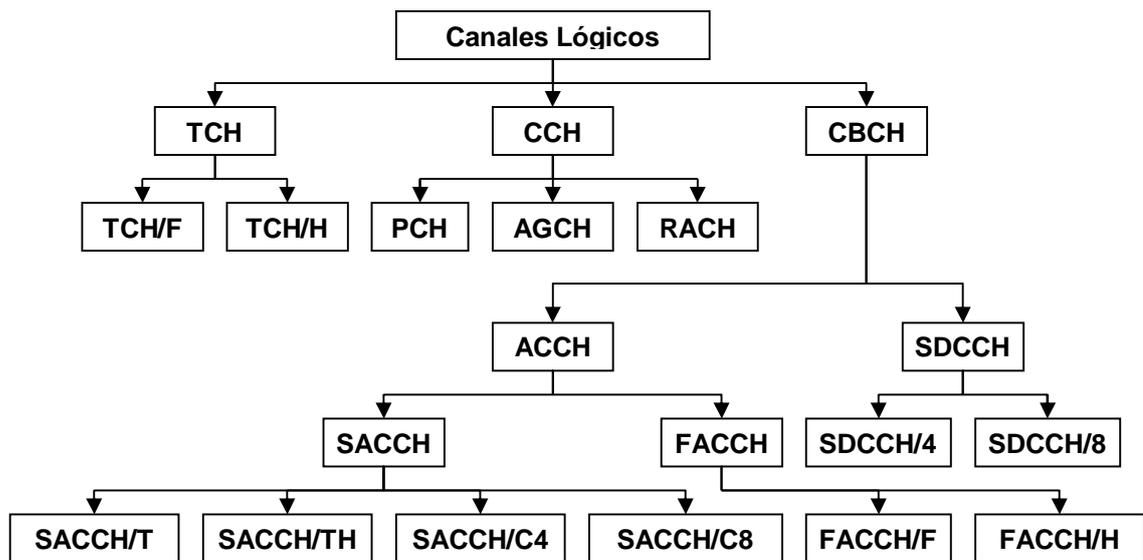


Figura 1.3. Canales Lógicos de un sistema GSM

1.2.2. EL SISTEMA GPRS: CONCEPTO Y CARACTERÍSTICAS PRINCIPALES

GPRS es la tecnología inalámbrica de transmisión de datos por medio de paquetes más ampliamente soportada en el mundo y se desarrolla a partir de los abonados GSM en más de 210 países y territorios de todo el mundo. Al igual que GSM, GPRS soporta roaming imperceptible al usuario, permitiendo que los usuarios tengan acceso a sus servicios de datos mientras se encuentran de viaje.

Conocida como tecnología de 2.5G, constituye el primer paso de un operador GSM hacia la tercera generación (3G); representando una solución para datos móviles pues mientras comparte el rango de frecuencias utilizado en la red GSM,

ofrece eficiencia espectral para nuevos y más veloces servicios de datos cuyo punto mas fuerte es el de ofrecer al usuario un terminal permanentemente conectado, tarifando únicamente por el volumen de datos transferidos y no por el tiempo de conexión.

Como su nombre lo indica, GPRS (General Packet Radio Service –Servicio General de Radio transmisión de Paquetes), es una tecnología basada en paquetes, es decir, a los datos se los divide en paquetes que se transmiten en breves ráfagas sobre una red IP. Este diseño es mucho más eficiente que las redes conmutadas por circuitos, dando lugar a una reducción de los costos operativos de la red. La conmutación de paquetes es un procedimiento más adecuado para transmitir datos, mientras que la transmisión mediante conmutación de circuitos es un procedimiento más adecuado para la transmisión de voz.

Como ya se indicó, GPRS se desarrolla sobre la plataforma GSM y está basado en IP, la norma universal utilizada en Internet, desarrollándose en una norma inalámbrica que no exija exclusivamente equipos propietarios, es decir utiliza una tecnología abierta y totalmente normalizada, razón por la cual es ideal para la provisión de acceso inalámbrico a otras redes basadas en IP, tales como LANs corporativas e ISPs.

Por tratarse de una tecnología de datos inalámbricos, GPRS ofrece velocidades de datos desde 40 kbps hasta máximos de 115 kbps . Siendo realistas podemos hablar de velocidades de transmisión entre 18 y 53 Kbps en sentido descendente (red – terminal) y entre 9 y 13 Kbps en sentido ascendente (terminal-red). Esta velocidad es variable ya que depende de algunos factores como el grado de ocupación de la red, el terminal utilizado, la distancia a la estación base o la velocidad a la que se desplaza el usuario, y un throughput promedio de 30-40 kbps.

Las tasas de transmisión de datos están divididas en categorías dependiendo del número de slots que utilicen (multislots). De esta manera podemos distinguir 4 tipos:

- ♦ CS-1 : 9,05 kbps

- ♦ CS-2 : 13,4 kbps
- ♦ CS-3 : 15,6 kbps
- ♦ CS-4 : 21,4 kbps

Con estas velocidades GPRS es más veloz que anteriores tecnologías como Datos Celulares Digitales en Paquetes (CDPD) con 19.2 kbps y Datos Conmutados por Circuitos (CSD) cuya velocidad es de 9.6 kbps, pudiendo así complementar o reemplazar a estas tecnologías.

Por otro lado, GPRS transporta una carga efectiva de datos mucho mayor que el Servicio de Mensajes Cortos (SMS), donde el número de caracteres se limita a 160.

Esta combinación de velocidad y capacidad convierte a GPRS en el medio o "portador" ideal de servicios tales como Protocolo de Aplicaciones Inalámbricas (WAP) y Mensajería Multimedia (MMS). El resultado final es que con GPRS un operador puede ofrecer una variedad mucho mayor de servicios innovadores y generadores de facturación.

Entre los servicios que este sistema ofrece se destacan los siguientes:

- ♦ Servicios basados en el envío de mensajes cortos
- ♦ Servicios generales de Internet
- ♦ Conexiones a Intranets
- ♦ Servicios específicos para GPRS
- ♦ Servicios basados en la localización
- ♦ Aplicaciones WAP: en este punto es necesario aclarar que el acceso a Internet de un móvil está limitado por la tecnología de visualización del móvil.

1.2.3. COMPARACIÓN CON EL SISTEMA GSM

GPRS es una evolución no traumática de la actual red GSM, por esta razón, el implementar esta red no conlleva grandes inversiones pues reutiliza parte de las infraestructuras actuales de GSM. Debido a esto, GPRS contará, desde sus inicios, con la misma cobertura que la actual red GSM.

Por otro lado GPRS (Global Packet Radio Service) es una tecnología que subsana las deficiencias de GSM, empezando por su velocidad de transmisión que, como ya se indicó, alcanza hasta los 115 kbps, valor mucho mayor que el ofrecido por la tecnología GSM. Una segunda gran ventaja de este sistema sobre GSM es el modo de tarificación, pues en GPRS se factura por el volumen de datos de la transferencia, a diferencia del sistema GSM en el que se cobraba por el tiempo de conexión.

Otras diferencias entre estos dos sistemas se presentan a continuación:

- ◆ Tiempo de Conexión

En GPRS el tiempo promedio de conexión es elevado, generalmente medido en horas; por otro lado, en GSM la llamada promedio es de dos minutos con una sola llamada por hora, es decir, mientras en GSM el usuario está conectado un 33% del tiempo por hora, en GPRS se puede tener hasta un 100% de conexión.

- ◆ Modo de Transmisión

Los datos en GPRS se transmiten a ráfagas, además, se tiene los enlaces ascendente y descendente independientes. En el sistema GSM se tiene un flujo continuo de datos en ambas direcciones.

- ◆ Activación de Servicios

GPRS permite al usuario activar servicios de forma independiente, pues este sistema soporta el principio de conectividad específica por servicio; mientras que en GSM se activan todos los servicios cuando se accede a la red.

- ◆ Acceso al Registro de ubicación Local (HLR)

Recordemos que el HLR es el encargado de almacenar y proveer la identidad y derechos de acceso a servicios de los usuarios.

En el sistema GPRS donde cada paquete es tratado como una entidad independiente, no se requiere acceder al HLR cada vez que se desea transmitir, pero en GSM es obligatorio el acceder al HLR cada vez que se activa una llamada.

- ♦ Tamaño de los paquetes transmitidos

Debido a que GPRS realiza las transmisiones por paquetes, estos son muy pequeños, tienen entre 500 y 1500 octetos, en GSM la transmisión se la realiza en un solo paquete, por esta razón su tamaño puede llegar a ser sumamente grande.

- ♦ Uso de los canales de comunicación

En GPRS los canales de comunicación se comparten entre los distintos usuarios dinámicamente, de modo que un usuario sólo tiene asignado un canal cuando se está realmente transmitiendo datos. En GSM, cuando se realiza una llamada se asigna un canal de comunicación al usuario, que permanecerá asignado aunque no se envíen datos.

- ♦ Velocidad y uso de los recursos

Como ya se ha indicado anteriormente, GPRS presenta una mayor velocidad de transmisión, además del hecho de asignar a un usuario un canal únicamente mientras este está transmitiendo, permitiendo así hacer una optimización de los recursos, característica que en GSM no se logra.

1.2.4. BENEFICIOS QUE OFRECE EL SISTEMA GPRS

GPRS marca un gran cambio en la telefonía móvil: de emplearse el terminal sólo para hablar, pasa a convertirse en un verdadero instrumento de comunicación al permitir la transmisión de todo tipo de datos.

1.2.4.1. VENTAJAS DEL GPRS PARA LA OPERADORA.

La operadora se beneficia por:

- ♦ A operadoras que trabajen con el sistema GSM les presenta la opción de una migración poco costosa: debido a que GPRS trabaja en la misma plataforma que GSM, resulta a los ya proveedores de GSM un cambio muy económico. Si una infraestructura de radio GSM tienen menos de cinco años de antigüedad, generalmente puede actualizarse a GPRS mediante software y tarjetas de canales adicionales. Muchas redes GSM de América tienen cuanto mucho unos pocos años de existencia, lo que significa que el tiempo y el costo de añadir GPRS son mucho menores que si debiera reemplazarse la infraestructura de radio.
- ♦ Atracción al cliente: debido a la gran variedad de opciones que ofrece GPRS la operadora puede ofrecer al usuario un servicio polifacético con un costo relativamente bajo, lo que lo convierte en algo muy atractivo.
- ♦ Ofrecer al cliente un alto nivel de calidad de servicio, QoS de la red existente al reducir las llamadas bloqueadas y pérdidas., factor que en la actualidad ha tomado mucha importancia.
- ♦ Diversas formas de facturación: Debido a que en GPRS lo que cuesta no es el tiempo conectado sino el volumen de transmisión, la operadora puede ofrecer diversas formas de facturación al cliente:
 - ♦ pago por bit
 - ♦ pago por sesión
 - ♦ tarifa plana....etc.
- ♦ Uso eficiente de los recursos de la red: los usuarios sólo ocupan los recursos de la red en el momento en que están transmitiendo o recibiendo datos, y además se pueden compartir los canales de comunicación entre distintos usuarios y no dedicados como en el modelo GSM.

1.2.4.2. VENTAJAS DEL GPRS PARA EL USUARIO.

Las diversas características del sistema GPRS se reflejan en beneficios al usuario, entre los cuales enlistaremos los siguientes:

- ♦ Siempre Conectado: debido a que solo se utiliza la red el momento de transmitir o recibir información, un usuario puede estar conectado todo el tiempo que desee y se le cobrará solo cuando haga uso de los recursos del sistema, es decir no paga mientras no esté recibiendo ni transmitiendo datos.
- ♦ Pago por volumen: solo se pagará la cantidad de datos transferidos, no por el tiempo que el usuario estuviese conectado.
- ♦ Costo nulo de establecimiento de conexión a la red GPRS, frente a los quantum de conexión existentes actualmente en GSM.
- ♦ Mayor velocidad de transmisión: gracias a la posibilidad de tener mas de un canal asignado (time slot) tanto en el sentido de subida –uplink- como de bajada –downlink-, la velocidad de transmisión aumentará en función a dicho número de canales asignados. Además, GPRS permite el uso de esquemas de codificación de datos que permiten una velocidad de transferencia de datos mayor que en GSM.
- ♦ Multitarea: muchos equipos GPRS permiten al usuario el realizar/recibir llamadas de voz mientras está conectado a la red GPRS utilizando cualquiera de los servicios disponibles con esta tecnología (transferencia de datos).
- ♦ Modo de transmisión asimétrico, más adaptado al tipo de tráfico de navegación html o wml en el cual el usuario recibe mucha mas información que la que envía. Así por ejemplo un terminal GPRS 4+1 (4 slots downlink y 1 uplink) tendrá cuatro veces mayor capacidad de transmisión de bajada que de subida).
- ♦ Soporta IP (Internet Protocol - Protocolo Internet) y X.25
- ♦ Por basarse en tecnologías IP y gracias a la amplia disponibilidad de "know-how IP" y equipos ofrecidos en versiones estándar, GPRS permite el

desarrollo de avanzados servicios de datos de manera más rápida y menos costosa.

- ◆ GPRS puede combinar hasta 8 canales para transferir datos.
- ◆ Creación de redes móviles: pues gracias a este sistema el usuario puede crear una red móvil, este tipo de redes es muy útil para aplicaciones de monitoreo, y está teniendo mucho éxito en especial en sistemas de monitoreo vehicular y en sistemas bancarios.
- ◆ Permite el acceso a intranets IP.
- ◆ Uso de GPS: tecnología mediante la cual podemos saber la ubicación de dispositivos móviles (y personas) basándonos en la información entregada por un arreglo de satélites, este servicio es muy utilizado para aplicaciones vehiculares: diagnóstico remoto de vehículos, seguimiento de vehículos robados y tarificación de autos alquilados.

1.2.5. ARQUITECTURA DEL SISTEMA

Todas las redes inalámbricas de próxima generación, independientemente de la tecnología que empleen, tienen tres componentes principales:

- ◆ La infraestructura de radio
- ◆ La red interna de las operadoras
- ◆ La estación base

1.2.5.1. INFRAESTRUCTURA DE RADIO

Incluye las celdas y el equipo terminal, en GPRS existen básicamente tres tipos de terminales lo cuales se enlistan a continuación:

- ◆ **Clase A:** Permite el uso simultáneo de los sistemas GSM y GPRS, en otras palabras, con este tipo de terminales es posible realizar una llamada y

al mismo tiempo acceder a Internet. Se utilizará un Time-slot para GSM y otro Time-slot para GPRS sin degradarse ninguno de los dos servicios.

- ♦ **Clase B:** Permite el uso de los dos sistemas pero no al mismo tiempo, es decir, mientras uno de los servicios está activo el otro permanece en suspenso la prioridad de uso la tiene GSM, es decir, las llamadas, por esta razón GPRS puede sufrir degradación de calidad de servicio (QoS).
- ♦ **Clase C:** No permite el uso simultáneo de los dos sistemas, se debe escoger entre uno de estos, es decir, el terminal puede funcionar solo como GSM o solo como GPRS, esta elección, de GPRS o GSM, se realiza de forma manual.

1.2.5.2. RED INTERNA DE LAS OPERADORAS,

Es la infraestructura del núcleo de paquetes y maneja sólo el tráfico de datos en paquetes.

Este núcleo de paquetes constituye la parte más importante del despliegue de GPRS, pues, como ya se indicó, manejar el tráfico de datos además de provee servicios tales como Redes Privadas Virtuales (VPNs).

Como GPRS es una red superpuesta a GSM, comparte con ella la red de acceso. Sin embargo GPRS, con la finalidad de manejar el tráfico de datos, introduce dos nuevos nodos:

- ♦ Gateway GPRS Support Node (GGSN) quién actúa como un interfaz lógico hacia las redes de paquetes de datos externas
- ♦ Serving GPRS Support Node (SGSN). es responsable de la entrega de paquetes al terminal móvil en su área de servicio

La infraestructura de conmutación por circuitos continúa manejando llamadas de voz únicamente, es decir que no se necesitan cambios al actualizar a GPRS.

1.2.5.3. ESTACIÓN BASE

Es la estación encargada de recoger las señales emitidas por el terminal y transferirlas a la red GPRS. En este nivel el sistema GPRS introduce el denominado Packet Control Unit (PCU).

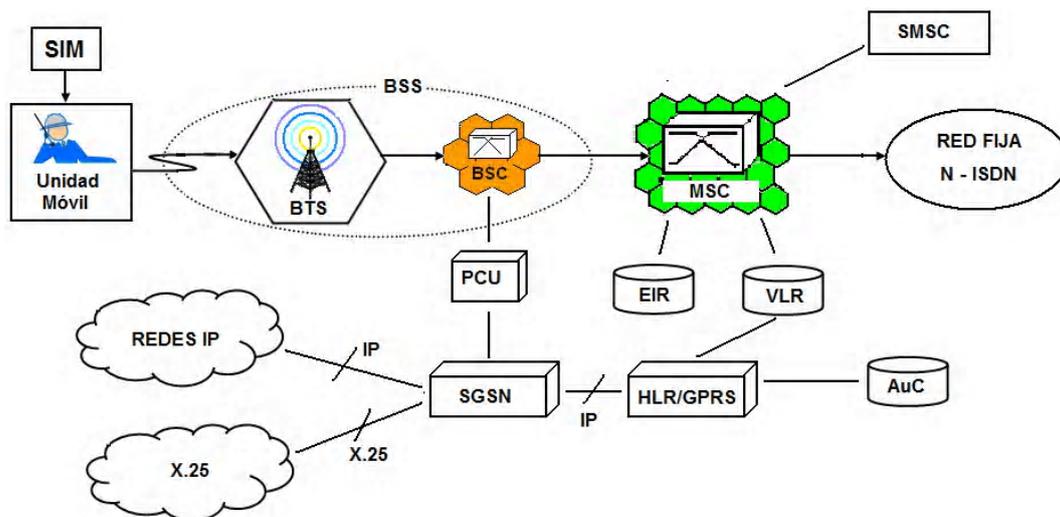


Figura 1.4. Esquema de una red GPRS

1.2.6. PRINCIPALES FABRICANTES DE CANALES E INFRAESTRUCTURA PARA REDES GSM – GPRS

El hecho de que GSM es una norma abierta permite que cualquier proveedor este habilitado para fabricar equipos con esta tecnología, razón por la cual existen en el mercado una gran gama de proveedores y equipos, permitiendo, tanto a usuarios como a operadores, la posibilidad de escoger un que se ajuste a todas sus necesidades, haciendo de GSM una tecnología singular.

La presencia en el mercado de tantos proveedores ocasiona que estos deban generar productos llamativos al cliente para así poder ganar mercado, lo que

desemboca en una gran variedad de equipos, cada uno con sus características y aplicativos mas atractivos que otros.

Los principales proveedores de equipos para esta tecnología son: HP, Motorola, Nokia, Siemens, Sony Ericsson y Research in Motion (RIM's Blackberry

La infraestructura para redes GSM está disponible a través de cientos de proveedores, entre ellos Ericsson, Nokia, Nortel Networks, Motorola y Siemens.

Por otro lado, GPRS soporta la más amplia gama de dispositivos para satisfacer las necesidades de una gran diversidad de consumidores y empresas. Actualmente, hay más de 590 modelos de teléfonos y tarjetas módem para PC GPRS que pueden adquirirse a importantes fabricantes entre los que se encuentran Motorola, Nokia, Siemens y Sony Ericsson.

La infraestructura de red GPRS se encuentra disponible a través de proveedores líderes como Nokia, Nortel Networks y Siemens.

1.2.7. PROYECCIONES DEL SISTEMA GPRS HACIA EL FUTURO

La evolución natural de GPRS es UMTS (Universal Mobile Telephony System). UMTS pero el cambio a esta nueva tecnología no es tan simple pues se requiere muchas modificaciones en el sistema actual:

- ♦ Una nueva tecnología de radio lo que representa grandes inversiones en infraestructuras
- ♦ Una red de mayor capacidad ya que las velocidades de transferencia varían de 384 Kbps a 2 Mbps.
- ♦ Nuevos terminales.

Esta nueva tecnología puede aportarnos muchos mas beneficios, entre los cuales anotaremos:

- ♦ Anchos de banda de hasta 2 Mb en cualquier lugar y momento.
- ♦ Conexión permanente a internet mediante un dispositivo personal, esta conexión será instantánea y el pago será en base a la información transmitida.
- ♦ Dispositivos que intercambiarán información con el sólo hecho de acercarlos (Bluetooth).
- ♦ Cualquier dispositivo digital y móvil podrá ser conectado a Internet.

1.3. SERVIDOR DE BASES DE DATOS

Para el desarrollo de este proyecto hemos decidido utilizar el sistema de bases de datos SQL básicamente por tres razones:

- ♦ Es un sistema lo suficientemente robusto como para soportar la cantidad de datos que se planea almacenar
- ♦ Se considera el tema costos, pues este es un software de distribución libre y se lo puede encontrar fácilmente en Internet
- ♦ Consultoría, ya que es un sistema de libre distribución, su popularidad es muy alta, lo que conlleva a la existencia de mucha información sobre dicho software, y resulta rápido y fácil el encontrar en Internet documentos de consulta, tutoriales, foros y otros lugares donde alguien que esta aprendiendo a desarrollar sistemas con esta herramienta puede respaldarse y responder sus dudas

1.3.1. CARACTERÍSTICAS

Es un sistema de gestión de base de datos (SGBD) con soporte SQL, orientado a trabajar con múltiples accesos: multihilo y multiusuario, con una gran popularidad, en la actualidad cuenta con más de seis millones de instalaciones.

Es desarrollado por la empresa SQL AB, una de las desarrolladoras de software libre más grandes del mundo, se distribuye en formato de software libre

bajo licencia GNU que permite que un programa sea ejecutado, copiado, distribuido y modificado libremente. Adicionalmente presenta la posibilidad de ser distribuido como software propietario en casos de problemas de licencias de software.

Se trata de un sistema sumamente popular, hasta el 2005 se tenía más de seis millones de copias de SQL, índice que supera la base instalada de cualquier otra herramienta de base de datos. En el 2004 el sitio web de SQL AB superó la cantidad de tráfico existente en el sitio de IBM.

1.3.1.1. DESARROLLO

- ◆ Está desarrollado en su mayor parte en lenguaje ANSI C y en C++.
- ◆ Probado con un amplio rango de compiladores diferentes
- ◆ Trabaja bajo licencia GPL (GNU Public License) y es en base de ésta que se definen el alcance y las restricciones del uso del software en diversas situaciones.
- ◆ Usa tablas en disco B-tree (ISAM) muy rápidas con compresión de índice.
- ◆ El código SQL se prueba con Purify (un detector de memoria perdida comercial) así como con Valgrind, una herramienta GPL. Con la finalidad de detectar cualquier fuga de memoria.
- ◆ Tablas hash en memoria, que son usadas como tablas temporales.
- ◆ Las funciones SQL están implementadas usando una librería altamente optimizada y deben ser tan rápidas como sea posible. Normalmente no hay reserva de memoria tras toda la inicialización para consultas.

1.3.1.2. FUNCIONALIDADES EN LA PROGRAMACIÓN

- ◆ Permite el incluir código SQL en aplicaciones comerciales a través de la compra de una versión comercial licenciada.
- ◆ Amplio subconjunto del lenguaje SQL.
- ◆ Funciona en diferentes plataformas, debido a que se ha utilizado GNU Autoconfig, es posible portar SQL a todos los sistemas modernos que tengan un compilador de C++ y una implementación funcional de subprocesos (threads) POSIX, siendo este último necesario para aplicaciones de servidor, para compilar un código de cliente solo se requiere el compilador de C++.
- ◆ Usa GNU Automake, Autoconf y Libtool (herramientas de compilación y generación de archivos) para portabilidad.
- ◆ APIs disponibles para C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, y Tcl.
- ◆ Relativamente sencillo de añadir otro sistema de almacenamiento. Esto es útil si desea añadir una interfase SQL para una base de datos propia.
- ◆ El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor. También está disponible como biblioteca y puede ser incrustado (linkado) en aplicaciones autónomas. Dichas aplicaciones pueden usarse por sí mismas o en entornos donde no hay red disponible.

1.3.1.3. ACCESO Y TRABAJO CON BASES DE DATOS

- ◆ Conectividad segura.
- ◆ Replicación.
- ◆ Transacciones y claves foráneas que nos permiten el asignar referencias a un campo de una tabla hacia un registro de otra tabla.
- ◆ Un sistema de reserva de memoria muy rápido basado en threads.

- ♦ Proporciona sistemas de almacenamiento transaccionales y no transaccionales.

1.3.1.4. MANIPULACIÓN DE LOS DATOS

- ♦ Es un sistema de administración relacional de bases de datos, es decir, no guarda los datos en un solo archivo, sino más bien, los archiva en tablas separadas.
- ♦ Joins (unión de tablas por medio de una codición) muy rápidos usando un multi-join de un paso optimizado.
- ♦ Cada tabla se relaciona o conecta con las demás a través de relaciones definidas, con lo cual es posible el relacionar datos de diferentes tablas.
- ♦ Puede mezclar tablas de distintas bases de datos en la misma consulta
- ♦ Presenta diversas opciones de almacenamiento permitiendo al usuario almacenar en función de la velocidad en las operaciones o del mayor número de operaciones disponibles.
- ♦ Permite la búsqueda e indexación de campos de texto.
- ♦ Uso completo de multi-threaded mediante threads del kernel, este proceso permite el multiprocesamiento, es decir que un programa realice más de una tarea.
- ♦ Diversos tipos de columnas de datos: enteros con/sin signo de 1, 2, 3, 4, y 8 bytes de longitud, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET, ENUM, y tipos espaciales OpenGIS.
- ♦ Registros de longitud fija y longitud variable.

1.3.1.5. SENTENCIAS Y FUNCIONES

- ◆ Soporte completo para operadores y funciones en las cláusulas de consultas SELECT y WHERE.
- ◆ Soporte completo para las cláusulas SQL GROUP BY y ORDER BY. Soporte de funciones de agrupación (COUNT(), COUNT(DISTINCT ...), AVG(), STD(), SUM(), MAX(), MIN(), y GROUP_CONCAT()).
- ◆ Soporte para LEFT OUTER JOIN y RIGHT OUTER JOIN cumpliendo estándares de sintaxis SQL y ODBC.
- ◆ Soporte para alias en tablas y columnas como lo requiere el estándar SQL.
- ◆ DELETE, INSERT, REPLACE, y UPDATE devuelven el número de filas que han cambiado (han sido afectadas). Es posible devolver el número de filas que serían afectadas usando un flag al conectar con el servidor.
- ◆ El comando específico de SQL SHOW puede usarse para obtener información acerca de la base de datos, el motor de base de datos, tablas e índices. El comando EXPLAIN puede usarse para determinar cómo el optimizador resuelve una consulta.

1.3.1.6. CONECTIVIDAD

- ◆ Los clientes pueden conectar con el servidor SQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows de la familia NT (NT,2000,XP, o 2003), los clientes pueden usar named pipes para la conexión. En sistemas Unix, los clientes pueden conectar usando ficheros socket Unix.
- ◆ En SQL 5.0, los servidores Windows soportan conexiones con memoria compartida si se inicializan con la opción --shared-memory. Los clientes pueden conectar a través de memoria compartida usando la opción --protocol=memory.

- ♦ La interfaz para el conector ODBC (ODBC) proporciona a SQL soporte para programas clientes que usen conexiones ODBC (Open Database Connectivity). Por ejemplo, puede usar MS Access para conectar al servidor SQL. Los clientes pueden ejecutarse en Windows o Unix. El código fuente de ODBC está disponible. Todas las funciones para ODBC 2.5 están soportadas, así como muchas otras.
- ♦ La interfaz para el conector J SQL proporciona soporte para clientes Java que usen conexiones JDBC. Estos clientes pueden ejecutarse en Windows o Unix. El código fuente para el conector J está disponible.

1.3.1.7. SEGURIDAD

- ♦ Un sistema de privilegios y contraseñas que es muy flexible y seguro, y que permite verificación basada en el host.
- ♦ Las contraseñas son seguras porque todo el tráfico de contraseñas está encriptado cuando se conecta con un servidor.

1.3.1.8. CLIENTES Y HERRAMIENTAS

- ♦ SQL server tiene soporte para comandos SQL para chequear, optimizar, y reparar tablas. Estos comandos están disponibles a través de la línea de comandos y el cliente SQLcheck. SQL también incluye ISAMchk, una utilidad de línea de comandos muy rápida para efectuar estas operaciones en tablas ISAM.
- ♦ Todos los programas SQL pueden invocarse con las opciones --help o -? para obtener asistencia en línea.

1.3.1.9. LOCALIZACIÓN

- ♦ El servidor puede proporcionar mensajes de error a los clientes en muchos idiomas.
- ♦ Soporte completo para distintos conjuntos de caracteres, incluyendo latin1 (ISO-8859-1), german, big5, ujis, y más. Por ejemplo, los caracteres escandinavos 'â', 'ä' y 'ö' están permitidos en nombres de tablas y columnas. El soporte para Unicode está disponible
- ♦ Todos los datos se guardan en el conjunto de caracteres elegido. Todas las comparaciones para columnas normales de cadenas de caracteres son case-insensitive.
- ♦ La ordenación se realiza acorde al conjunto de caracteres elegido (usando colación Sueca por defecto). Es posible cambiarla cuando arranca el servidor SQL. Para ver un ejemplo de ordenación muy avanzada, consulte el código Checo de ordenación. SQL Server soporta diferentes conjuntos de caracteres que deben ser especificados en tiempo de compilación y de ejecución.

1.3.1.10. ESCALABILIDAD Y LÍMITES

- ♦ Soporte a grandes bases de datos. Se tiene casos de aplicación donde SQL Server trabaja con 60.000 tablas y acerca de 5.000.000 de registros.
- ♦ Se permiten hasta 64 índices por tabla (32 antes de SQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de SQL 4.1.2).Un índice puede usar prefijos de una columna para los tipos de columna CHAR, VARCHAR, BLOB, o TEXT.

1.3.1.11. DIMENSIONES MÁXIMAS DE LAS TABLAS SQL

En SQL 2005 gracias a la aplicación del un or de almacenamiento, se tiene una gran capacidad de almacenamiento, así, el máximo tamaño de las tablas es de

65536 terabytes ($256^7 - 1$ bytes). Por lo tanto, el tamaño efectivo máximo para las bases de datos en SQL usualmente los determinan los límites de tamaño de ficheros del sistema operativo, y no por límites internos de SQL.

Aproximadamente se tiene los siguientes valores para tamaño máximo de ficheros en los diversos sistemas operativos:

Sistema Operativo	Fichero
Linux 2.2 – Intel 32 -bit	2Gb (LFS: 4Gb)
Linux 2.4 con un sistema de ficheros ext3	4 Tb
Solaris 9/10	16 Tb
Sistema de ficheros NetWare w/NSS	8 Tb
Win32 w/ FAT	2 Gb
Win 32 /FAT32	4 Gb
Win32 w/ NTFS	2 Tb
MacOS X w/ HFS+	2 Tb

Tabla 1.1. Capacidad de Fichero en diversos sistemas operativos

Por otro lado, gracias al motor de almacenamiento InnoDB que mantiene las tablas en un espacio que puede ser creado a partir de varios ficheros, es posible que una tabla supere el tamaño máximo individual de un fichero. Este espacio puede incluir particiones de disco, lo que permite tablas extremadamente grandes. El tamaño máximo del espacio de tablas es 64TB.

En SQL se crea por defecto tablas ISAM cuya estructura interna permite un máximo de 4 Gb,

Gracias al comando SHOW TABLE STATUS o con ISAMchk -dv tbl_name es posible revisar el tamaño máximo para una tabla.

En casos en que se requiera trabajar con tablas de mayor tamaño, y que el sistema operativo soporte este tipo de ficheros, se puede acudir al comando CREATE TABLE el cual presenta las opciones AVG_ROW_LENGTH y MAX_ROWS.. Asi mismo, es posible aumentar el tamaño máximo de una tabla ya creada gracias al comando ALTER TABLE.

Otros métodos para cambiar los límites de tamaño de ficheros para tablas ISAM son:

- ♦ Si una tabla es de sólo lectura, puede usar ISAMpack para comprimirla. ISAMpack normalmente comprime una tabla al menos un 50%, lo que permite, a efectos prácticos, tablas mucho mayores. isampack puede mezclar múltiples tablas en una misma tabla
- ♦ SQL incluye la biblioteca MERGE que permite tratar una colección de tablas ISAM con una estructura idéntica en una tabla MERGE.

1.3.2. BENEFICIOS

1.3.2.1. VENTAJAS EN EL USO DE BASES DE DATOS.

- ♦ Globalización de la información: los datos se convierten en un recurso corporativo dejando de pertenecer a un dueño específico.
- ♦ Sin información redundante: pues al ingresar nuevos datos, se verifica siempre que no se trate de información duplicada
- ♦ Sin información inconsistente: pues si un elemento de la tabla se altera, este dato se actualiza en todas las tablas relacionadas a éste.
- ♦ Información Compartida: pues se puede dar acceso a la información a varios usuario
- ♦ Integridad de la Información: pues solo se almacena información correcta.
- ♦ Independencia de datos y tratamiento: pues es permitido el realizar cambios en las bases de datos sin que se requiera un cambio en las aplicaciones que trabajan con ellas, así también permite el acceso a los datos a través de más de una aplicación.
- ♦ Seguridad y restricción en el acceso: pues es posible el crear niveles de acceso a una base de datos, así como las operaciones sobre dicha información.

1.3.2.2. VENTAJAS DE SQL

- ◆ Disponibilidad en gran cantidad de plataformas y sistemas.
- ◆ Diferentes opciones de almacenamiento según si se desea velocidad en las operaciones o el mayor número de operaciones disponibles.
- ◆ Transacciones y claves foráneas.
- ◆ Conectividad segura.
- ◆ Replicación.
- ◆ Búsqueda e indexación de campos de texto.
- ◆ Por ser un sistema de administración relacional de bases de datos nos permite mayor velocidad y flexibilidad.
- ◆ Es un software de código abierto, lo cual permite al usuario modificar, utilizar, copiar y distribuir libremente y de forma gratuita el código fuente para SQL.
- ◆ Como la distribución modificación y uso del código fuente es libre, facilita al usuario el obtener ejemplos, estudiarlos, modificarlos a su necesidad así también el distribuirlos.
- ◆ Existe una gran cantidad de herramientas para trabajar con el sistema.
- ◆ Las contraseñas son seguras porque todo el tráfico de contraseñas está encriptado cuando se conecta con un servidor.

1.4. LA PLATAFORMA .NET

1.4.1. INTRODUCCIÓN

Se trata de un conjunto de nuevas tecnologías que desde hace algunos años Microsoft se encuentra desarrollado.

El objetivo es obtener una plataforma sencilla y potente para distribuir el software en forma de servicios los cuales puedan ser suministrados remotamente y además puedan comunicarse y combinarse unos con otros de manera totalmente independiente de la plataforma, lenguaje de programación y modelo de componentes con los que hayan sido desarrollados.

Esta es la Llamada plataforma .NET, y a los servicios antes comentados se les denomina servicios Web. Con la ayuda de esta plataforma se busca transformar Internet en una plataforma informática distribuida a escala completa. Proporcionando nuevas formas de desarrollar aplicaciones a partir de colecciones de Servicios Web, para ello, la plataforma .NET soporta totalmente la infraestructura existente de Internet, incluyendo HTTP, XML y SOAP.

Con la finalidad de permitir la creación de aplicaciones que trabajen sobre esta plataforma, Microsoft ha publicado un kit de desarrollo de software con las herramientas necesarias para el desarrollo, distribución y ejecución de las aplicaciones, este kit es conocido como .NET Framework SDK, y uno de sus componentes mas populares es justamente Visual Studio .Net, herramienta que permite la elaboración de aplicaciones desde un interfaz visual típico de Windows. Es esta aplicación quien se utilizó para la elaboración de la aplicación parte de este trabajo.

1.4.2. BENEFICIOS QUE PRESENTA .NET

En resumen podemos decir que la plataforma .NET proporciona:

- ♦ Un modelo de programación coherente e independiente del lenguaje para todas las capas o niveles de una aplicación.
- ♦ Una interoperabilidad transparente entre tecnologías.
- ♦ Una fácil migración desde tecnologías existentes.
- ♦ Un completo soporte de tecnologías de Internet independientes de la plataforma y basadas en estándares, incluyendo Hypertext Transfer Protocol

(HTTP), Extensible Markup Language (XML) y Simple Object Access Protocol (SOAP).

1.4.3. COMMON LANGUAGE RUNTIME (CLR)

Es una máquina virtual y constituye uno de los pilares de la tecnología .NET, siendo considerado su núcleo. Es el motor encargado de gestionar la ejecución de las aplicaciones para ella desarrolladas y a las que ofrece numerosos servicios que simplifican su desarrollo y favorecen su fiabilidad y seguridad.

Hasta la aparición de la tecnología .NET existían dos posibles formas de compilar un lenguaje:

- ♦ Interpretación: como es el caso de VScript y JScript
- ♦ Compilados a binario: tal es el caso de los lenguajes de programación C++, COBOL, FORTRAN

Con JAVA aparece el concepto de Máquina Virtual (VM) y gracias a éste, se presenta una nueva forma de compilación del lenguaje a un nivel intermedio que puede ser ejecutado en toda máquina que tenga una máquina virtual.

CLR es una adaptación de la idea de JAVA a .NET, realizada por Microsoft, con la diferencia de que .NET admite una gran variedad de perfiles tecnológicos, así un desarrollador que programaba en C, adoptará C#, si trabajaba con Visual Basic, adoptará Visual Basic .NET, si viene de ASP, puede optar por Visual Basic.NET o Jscript.NET. Así también es independiente del sistema operativo, razón por la cual, en un futuro, CLR puede implantarse en otros sistemas que no pertenezcan a la serie Windows y ser utilizado como base para el desarrollo de aplicaciones de escritorio, característica que actualmente ASP no posee.

El CLR gestiona la ejecución de las aplicaciones diseñadas para la plataforma .NET. Por esta razón, al código de estas aplicaciones se le suele llamar código gestionado, y al código no escrito para ser ejecutado directamente en la plataforma .NET se le suele llamar código no gestionado.

1.4.3.1. PRINCIPALES CARACTERÍSTICAS Y VENTAJAS DE CLR

- ◆ Transforma el código intermedio al código nativo de la máquina sobre la que se esté ejecutando el programa
- ◆ Modelo de programación consistente: este modelo es el orientado a objetos.
- ◆ Modelo de programación sencillo: Se eliminan conceptos complejos existentes en sistemas operativos anteriores (registro de Windows, GUIDs, HRESULTS, IUnknown, etc.)
- ◆ Ejecución multiplataforma: Una aplicación .NET puede ser ejecutada en cualquier plataforma para la que se haya desarrollado una versión del CLR.
- ◆ Integración de lenguajes: se admiten aplicaciones desarrolladas en cualquier lenguaje que, al ser compilado, pueda generar código para la plataforma .NET.
- ◆ Gestión de memoria: Permite liberar espacio en la memoria dinámica asociada al programa a través del borrado de elementos no utilizados permitiendo la creación de nuevos. Gracias a este recolector se evitan errores de programación muy comunes como intentos de borrado de objetos ya borrados, agotamiento de memoria por olvido de eliminación de objetos inútiles o solicitud de acceso a miembros de objetos ya destruidos.
- ◆ Seguridad de tipos: El CLR verifica que toda conversión de tipos sea compatible, con lo cual se eliminan errores difíciles de detectar.
- ◆ Aislamiento de procesos: Se asegura que no se pueda acceder desde el código de un proceso al código de otro, así se evita que un proceso “ataque” a otro, invada su espacio de memoria, etc.
- ◆ Tratamiento de excepciones: en sistemas Windows anteriores la propagación de un error podía ser por códigos de error en formato Win32, HRESULTs y

excepciones, con CLR todo error que se presente se propaga mediante excepciones.

- ♦ Soporte multihilo: permite trabajar con aplicaciones divididas en múltiples hilos de ejecución cuya evolución puede ir por separado en paralelo o intercaladamente.
- ♦ Distribución transparente: el acceso a objetos remotos es completamente transparente.
- ♦ Seguridad avanzada: A través de la restricción de ejecución de ciertos códigos el otorgar permisos a los mismos según la procedencia o el usuario que los ejecute, así se puede asegurar que el código que se esté ejecutando no pueda poner en peligro la integridad de archivos, del registro de Windows, etc.
- ♦ Interoperabilidad con código antiguo.

1.4.4. MICROSOFT INTERMEDIATE LANGUAGE (MSIL)

Como ya se dijo, una de las características de la plataforma .NET es que todas sus aplicaciones se compilan a un lenguaje intermedio que es interpretado por una máquina virtual (CLR), pues bien, este lenguaje es precisamente el MSIL (Microsoft Intermediate Language) y es el único lenguaje que puede ser interpretado por el CLR, por ello cuando se dice que un compilador genera código para la plataforma .NET lo que se está diciendo es que genera MSIL.

1.4.4.1. PRINCIPALES CARACTERÍSTICAS Y VENTAJAS DE MSIL

- ♦ Se trata de un lenguaje creado por Microsoft
- ♦ Tiene un nivel de abstracción mucho más alto que el de la mayoría de los códigos máquina de las CPUs existentes, que incluye instrucciones para trabajar directamente con objetos (crearlos, destruirlos, inicializarlos, llamar a

métodos virtuales, etc.), tablas y excepciones (lanzarlas, capturarlas y tratarlas)

- ♦ Por ser independiente del CPU, facilita la ejecución multiplataforma y la integración entre lenguajes.
- ♦ Entrega un formato común para todo código máquina generado por compiladores para.NET.
- ♦ No puede ser ejecutado directamente por una máquina, es necesario primero convertirlo al código nativo de la misma esto lo logramos gracias a un componente del CLR conocido como compilador JIT (Just-In-Time) o jitter mismo que convierte dinámicamente el código MSIL a ejecutar en código nativo según sea necesario.

1.4.5. COMPILADOR JUST IN TIME (JIT O JITTER)

Componente del CLR, es el encargado de convertir el código intermedio MSIL al lenguaje nativo del computador donde se ejecute la aplicación.

Se distribuye en tres versiones:

- ♦ Jitter normal: Usado por defecto, compila únicamente código MSIL a código nativo conforme se lo va requiriendo así logra ahorrar tiempo y memoria pues al código no ejecutado no se lo compila.
- ♦ Jitter económico: Su trabajo es similar al jitter normal, con la diferencia de que este no realiza optimización alguna, simplemente traduce cada instrucción a código de máquina, si bien con este procedimiento se logra código menos eficiente, el tiempo y memoria utilizados para esta traducción es mucho menor, por ello es muy útil en máquinas de poca memoria.

Es gracias a la facilidad de implementación de esta versión de jitter que se realiza la adaptación de la plataforma .NET a nuevos sistemas, a través del rápido desarrollo de una versión de CLR que ejecute de forma poco eficiente

aplicaciones gestionadas, para luego centrar el desarrollo en la optimización al pasar al Jitter Normal.

- ♦ Prejitter: Distribuido como una aplicación en línea de comandos conocida como “ngen.exe”, convierte completamente un código gestionado al código nativo de la máquina, eliminando la posible pérdida de tiempo con compilaciones dinámicas.

1.4.6. APLICACIONES Y TECNOLOGÍAS QUE FORMAN PARTE DE LA PLATAFORMA .NET

Dentro de la plataforma .Net se incluye a una amplia gama de nuevas aplicaciones que día a día se van perfeccionando por Microsoft o terceros, con el objetivo de dar mas fuerza a esta gran herramienta de trabajo, entre estas aplicaciones tenemos: Windows.NET, Hailstorm, Visual Studio.NET, MSN.NET, Office.NET. Y los nuevos servidores para empresas de Microsoft: SQL Server.NET, Exchange.NET, entre otros.

1.4.7. EL .NET FRAMEWORK

Es el responsable de proporcionar todos los servicios comunes requeridos para ejecutar una aplicación desarrollada en Visual Basic .NET, es capaz de interpretar todo en cuanto a programación orientada a objetos se refiera.

Soporta clases, herencia, métodos, propiedades, eventos, polimorfismo, constructores y otras construcciones orientadas a objetos.

1.4.7.1. OBJETIVOS DEL .NET FRAMEWORK

- ♦ Basado en estándares y prácticas Web: promueve la creación de servicios Web débilmente conectados y sin estado y es capaz de soportar totalmente las tecnologías existentes de Internet, incluyendo Hypertext Markup

Language (HTML), XML, SOAP, Extensible Stylesheet Language for Transformations (XSLT), Xpath y otros estándares Web.

- ♦ Extensible: La jerarquía del .NET Framework no queda oculta al desarrollador. Podemos acceder y extender clases .NET (a menos que estén selladas) utilizando herencia. También podemos implementar herencia multi-lenguaje.
- ♦ Fácil de usar por los desarrolladores: pues se ha organizado al código en espacios de nombres jerárquicos y clases.
- ♦ Sistema de tipo unificado: es un sistema de tipos común utilizado por todos los lenguajes compatibles con .NET. En este sistema, todo es un objeto. No existen tipos variant, sólo hay un tipo de cadena (string), y todos los datos de la cadena son Unicode.
- ♦ Diseñado utilizando modelos de aplicaciones unificados: La funcionalidad de una clase .NET está disponible desde cualquier modelo de programación o lenguaje compatible con .NET.

1.4.7.2. COMPONENTES PRINCIPALES DEL .NET FRAMEWORK

- ♦ Common Language Runtime - CLR
- ♦ Biblioteca de clases del .NET Framework
- ♦ ADO.NET: datos y XML
- ♦ ASP.NET: Formularios Web y Servicios Web
- ♦ Interfaz de usuario

1.4.8. MEJORAS DE .NET

Visual Basic .NET proporciona importantes mejoras del lenguaje, un acceso total al .NET Framework y un desarrollo Web mejorado.

En Visual Basic .NET, Visual Basic ha experimentado algunos cambios importantes para permitir a los desarrolladores de Visual Basic crear aplicaciones empresariales potentes, robustas y escalables.

- ◆ Soporte orientado a objetos mejorado: Permite a los desarrolladores de Visual Basic .NET utilizar herencia de clases, constructores, destructores, sobrecarga, interfaces y polimorfismo. Esto otorga a Visual Basic .NET tanto poder orientado a objetos como cualquier otro lenguaje .NET, como C# o Visual C++ con extensiones gestionadas.
- ◆ Gestión de excepciones estructurada: Simplifica el control de excepciones y permite utilizar potentes características como las excepciones anidadas. Esto mejora enormemente el anterior gestor de errores de Visual Basic.
- ◆ Nuevas opciones de manejo de hilos: Permite crear aplicaciones que utilizan capacidades con múltiples hilos de ejecución. Por primera vez, los desarrolladores de Visual Basic podrán crear aplicaciones capaces de competir con aplicaciones basadas en Visual C++ en esta área.
- ◆ Desarrollo Web mejorado: Visual Basic .NET permite a los desarrolladores crear rápidamente potentes aplicaciones Web.
- ◆ Crear Formularios Web fácilmente: Es tan fácil como crear formularios Windows utilizando la técnica de escribir código tras el evento de un control. Los desarrolladores ya no necesitan utilizar un entorno de desarrollo integrado (IDE, Integrated Development Environment) para el desarrollo de páginas Web (como Microsoft Visual InterDev®) y otro entorno para el desarrollo de componentes. Ahora, todos los aspectos de una aplicación Web se crean simplemente en un único entorno para los desarrolladores de Visual Basic y de un modo fácil de entender.
- ◆ Crear Servicios Web rápidamente: Es posible crear Servicios Web rápidamente de modo similar a cómo se creaban componentes en las versiones anteriores de Visual Basic. Los Servicios Web son una potente tecnología que le permite acceder a sus componentes (o a los componentes de otras personas) a través de Internet mediante HTTP. Cualquier

desarrollador de Visual Basic que haya creado componentes podrá realizar fácilmente la transición a los Servicios Web.

CAPITULO II

DISEÑO DEL SISTEMA

2.1. DESCRIPCIÓN FUNCIONAL DEL SISTEMA

2.1.1. ANÁLISIS DE REQUERIMIENTOS

Al observar los actuales sistemas de seguridad utilizados en la ciudad de Quito concluimos que hoy en día son pocos los sectores residenciales que cuentan con un sistema de seguridad con características similares a las que deseamos implementar. En una gran mayoría de los sistemas de monitoreo actuales se presenta el siguiente cuadro:

En cada vivienda se ha instalado un sistema de alarma mismo que al momento de presentarse algún evento, tal como robo, genera una señal de alarma tipo audible, la cuál deberá ser atendida por el guardia de seguridad contratado para vigilar todo el conjunto residencial.

Otro sistema de seguridad comúnmente utilizado que es un poco mas completo consiste básicamente en una serie de sensores conectados con un Centro de Monitoreo misma que es controlada por un operador quién es el único que tiene acceso a esta información dejando al dueño de la residencia sin capacidad de acción sobre dicho monitoreo.

Bajo este esquema de operación si el usuario desea conocer el estado de su vivienda mientras está fuera de ella debe obligatoriamente comunicarse con el operador del sistema de seguridad, convirtiendo muchas veces a esta situación en algo incómodo para dicho usuario quién opta por no realizar esta consulta.

Mediante un análisis social hemos observado que las personas buscan quien esté en capacidad de ofrecerles una solución confiable y sencilla a este problema, presentando básicamente los siguientes requerimientos para este sistema:

- ♦ Monitoreo remoto Sin importar que tan lejos se encuentre el propietario de la vivienda, poder tener una idea del estado de esta, poder tener la seguridad de que todo se encuentra en orden sin tener la necesidad de confiar e la palabra de alguna otra persona.
- ♦ Cobertura en zona residencial que sin importar donde este su vivienda, poder tener la garantía de que será monitoreada, es decir que se encuentre dentro del área de cobertura del sistema.
- ♦ Acceso 24 horas Sin importar que hora sea, el usuario desea poder tener acceso a la información de su vivienda, así también como tener la seguridad de que sin importar horarios el sistema de vigilancia estará funcionando a la perfección sin depender de la eficiencia de algún operador.
- ♦ Fácil implementación de Centro de Monitoreo Se busca un sistema que no requiera de mayor arquitectura para establecer un Centro de Monitoreo, pues mientras mas complejo es un sistema mas costoso este se vuelve, y es mucho mas difícil el reubicarlo por algún evento que ocurriese.
- ♦ Sistema simple y poco costoso por ello se desea algo sencillo y efectivo con la capacidad de permitir al proveedor ofertar al cliente una solución eficiente y no costosa.
- ♦ Aplicable a múltiples ambientes se requiere que el sistema no esté preparado para un medio específico, es decir que se adapte a las diversas configuraciones que se pueden presentar en una vivienda, esto es, número de habitaciones, eventos a controlar e cada habitación así como a los requerimientos que cada cliente puede tener.
- ♦ Versátil y flexible que le de al proveedor la posibilidad de “personalizar” su servicio para cada uno de los clientes, considerando eventos a monitorear, lugares en los que se realizará el control, entre otros.

- ♦ Permita crecimiento de manera sencilla debido a que en la actualidad un proveedor no puede limitarse a un número fijo de clientes, se requiere un sistema que permita una fácil y no costosa ampliación, para así reducir los gastos de la empresa.

2.1.2. EVENTOS A CONTROLAR

Considerando que todos los eventos posibles que se pueden suscitar dentro de una vivienda son muy variados, hemos decidido generalizar al máximo todos estos y entre ellos seleccionar los más importantes.

Luego de un análisis general de la situación llegamos a la conclusión de que existe un evento que preocupa mucho a los propietarios, pues es el más frecuente y nocivo a la propiedad privada, este es el robo.

Por este motivo decidimos dedicar nuestro monitoreo a señales provenientes de movimiento dentro de una vivienda y de apertura tanto de puertas como de ventanas.

2.1.3. SERVICIOS A PRESTARSE

Este sistema fue diseñado para la gestión y monitoreo en tiempo real de grupos de viviendas, permitiendo a las empresas constructoras y de seguridad un planeamiento y control operacional eficiente, optimizando los métodos actuales de vigilancia domiciliaria.

- ♦ En general, los sistemas de monitoreo de viviendas con información en “tiempo real” permiten:
- ♦ Obtener información sobre el estado de la vivienda en cualquier momento.
- ♦ Respuesta inmediata en caso de suscitarse un inconveniente dentro del domicilio.
- ♦ Monitorear situaciones de riesgo como robos.

- ♦ Aumentar la velocidad operacional promedio de un sistema de seguridad.

La solución propuesta se basa en un conjunto de servicios, bibliotecas y aplicaciones destinados a proveer comunicación de datos, procesamiento y distribución de informes de gestión, con un excelente desempeño en el manejo de numerosos condominios y conexión con un gran número de usuarios distribuidos local o remotamente.

Basándonos en los requerimientos que detectamos tiene los posibles usuarios de nuestro sistema lo hemos diseñado bajo el siguiente formato:

1. Un sistema de sensores de: Apertura de puertas o ventanas, y presencia, monitorea un máximo de 6 zonas dentro de la vivienda.
2. La distribución de estas zonas es a libre elección del usuario, siempre y cuando se ajuste a los eventos que nosotros controlamos así por ejemplo se podría tener:
 - ♦ Zona 1: Puerta de Entrada (detección de apertura)
 - ♦ Zona 2: Detección de presencia en sala comedor
 - ♦ Zona 3: Detector de humo en cocina
 - ♦ Zona 4: Detector de apertura de ventanas
 - ♦ Zona 5: Detector de ruptura de ventanas
 - ♦ Zona 6: Detector de presencia en dormitorios
3. Gracias a la división en zonas, es posible realizar el monitoreo de la vivienda en forma parcial, así por ejemplo por la noche –cuando todos duermen- se puede activar el monitoreo sobre aquellas zonas que se sabe no se va a activar, por ejemplo la puerta de entrada misma que durante la noche no se la abrirá –excepto en casos eventuales-.
4. Las señales de estos sensores son recogidas y monitoreadas por un microcontrolador, mismo que al detectar algún evento dentro de la vivienda

entra en un estado de alerta y envía una señal a través de un módem por el canal de GPRS.

5. El proveedor del servicio de GPRS toma las señales enviadas y las entrega en una dirección IP pública misma que configuramos previamente en el módem.
6. En esta dirección IP estará conectado nuestro servidor, el cual que se encarga de registrar, toda información recibida, en el repositorio de datos.
7. Toda señal recibida y almacenada en nuestro servidor es validada y en función de esto se genera una señal de alarma que informará al encargado del monitoreo sobre un determinado evento.
8. Adicionalmente El usuario podrá acceder a un control 24 horas del estado de las zonas que se monitorean en su vivienda a través de Internet, así como a un reporte sobre el historial de los eventos presentados en la misma.
9. Se presenta también la opción de que, una vez activado alguno de los eventos monitoreados, el equipo a más de enviar la señal al servidor realice una llamada telefónica, con el limitante de que el audio de esta llamada será una señal de pulsos audible, pues el gravar voz nos resultaría más costos.

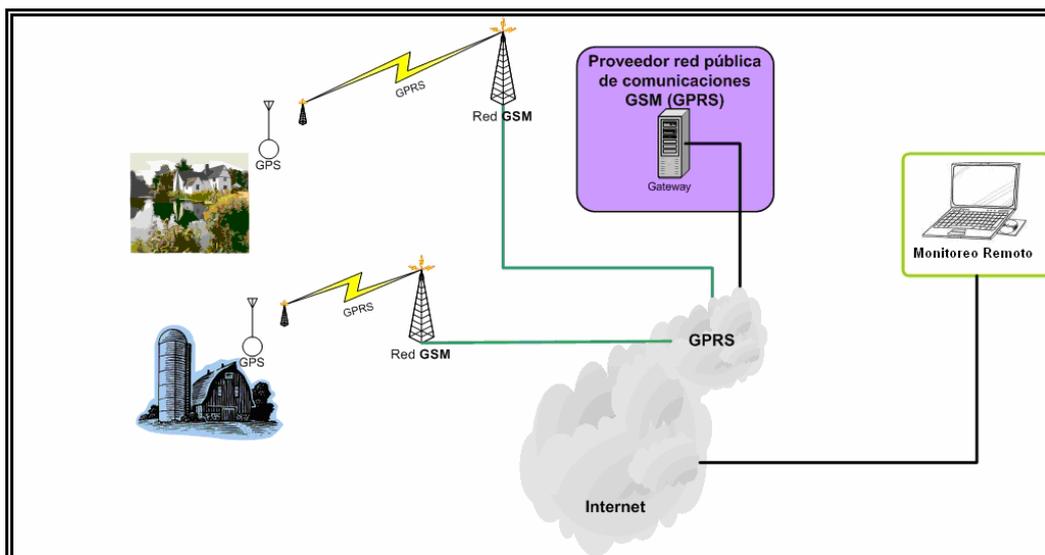


Figura 2.1. Esquema de la solución presentada

Como se puede observar, se trata de un sistema sumamente versátil y de un costo relativamente bajo en función del alcance que el servicio ofrecido presenta,

Por sus características técnicas y de diseño, este sistema permite registrar y procesar simultáneamente en más de un Centro de cómputo, además de poder acceder remotamente mediante Internet, y monitorear un gran número de viviendas sin requerir de una gran infraestructura de manera local

Actualmente esquemas de este tipo son utilizados en proyectos de gran envergadura, por su flexibilidad para adecuarse rápidamente a nuevas necesidades.

2.1.3.1. Monitoreo remoto vía Internet

Internet es la red más grande de computadoras en el mundo, la cual facilita la globalización de los mercados. Las ventajas de usar Internet son muchísimas entre las cuales mencionaremos

- ♦ Accesibilidad.- pues en casi todas las partes del mundo se puede encontrar una forma de enlazarse a internet.
- ♦ Costos.- la masificación del internet ha desembocado en la reducción sustancial de los costos de navegación, razón por la cual resulta mucho mas económico el realizar una consulta por internet que por algún otro medio como telefónico y otro.
- ♦ Es una sola red para todos los usuarios del mundo por lo que el usuario puede tener acceso ,desde cualquier lugar, a nuestro aplicativo,
- ♦ Es un servicio abierto las 24 horas del día sin rentas altas, pagos de servicios, etc.
- ♦ No existe discriminación de marcas o modelos de computadoras para tener acceso a este servicio.

En función de lo aquí expuesto podemos definir un conjunto de beneficios que obtendríamos al trabajar con un monitoreo por Internet ya que muchas de las deficiencias presentadas en un sistema convencional se eliminarían, entre ellas podemos anotar:

- ♦ Se consigue un sistema más efectivo y de menor costo.
- ♦ Permite al proveedor montar su infraestructura de monitoreo sobre un sistema muy conocido, sencillo y con un costo relativamente bajo.
- ♦ Se puede acceder al sistema desde cualquier parte del mundo y a cualquier hora sin pasar por la incomodidad de tener que molestar a alguna persona.
- ♦ Involucra al usuario en la vigilancia de su propiedad de forma tal que el tiene la certeza de que sus bienes se encuentran seguros.
- ♦ No requiere del montaje de un sofisticado Centro de Monitoreo, basta con un computador con acceso a Internet.
- ♦ El factor de que el Centro de Monitoreo no requiere de mayor infraestructura lo hace de fácil y rápida instalación y reubicación.
- ♦ Por el fácil montaje de un Centro de Monitoreo, resulta también muy simple y económico el crecimiento de dicho centro, a través de la implementación de nuevas sucursales.

2.1.3.2. Comunicación entre el Centro de Monitoreo y la Vivienda utilizando GPRS

La aplicación de la tecnología GPRS está implicando un nuevo cambio de filosofía en los entornos donde los recursos móviles suponen un activo principal: empresas de transportes, empresas de seguros, fuerzas de seguridad, etc.”

Su bajo costo, las economías de escala aplicadas a los terminales móviles, su nivel de cobertura a nivel nacional e internacional, y su adecuación en la velocidad de transmisión y protocolos para el acceso transparente a Internet,

están permitiendo que los recursos móviles se encuentren prácticamente integrados en las redes de área local de las mismas.

Además, la tecnología GPRS simplifica el acceso a la información proporcionada desde los terminales. Ello supone una novedad importante respecto a la forma tradicional de hacer comunicaciones.

El implementar un sistema de comunicación a través de la red GPRS nos da un gran beneficio en el alcance en cuanto a sectores se refiere, pues mientras la vivienda se encuentre dentro del área de cobertura de nuestro proveedor de servicio GPRS, podremos ofrecer nuestro sistema. Llegando incluso al punto en que, teniendo el Centro de Monitoreo en la ciudad de Quito, podemos tener en nuestro sistema viviendas ubicadas en otras ciudades, provincias o regiones.

Así mismo, gracias a la amplia cobertura actual de la red GPRS en Ecuador, en el caso en que el usuario, es decir el dueño de la vivienda, decidiera cambiar de vivienda, nos resulta sumamente fácil el reubicar nuestro sistema.

2.1.3.3. Acceso en el esquema 24-7-365

Gracias a que tanto el acceso a Internet como el servicio de GPRS están presentes todo el día todos los días del año, es factible el implementar un sistema de monitoreo completamente permanente.

2.1.3.4. Señal de Emergencia

Cada vez que se genere un evento en alguna de las zonas monitoreadas dentro de la vivienda se generará dos tipos de señales: una señal de alarma local –sirena- que alertará sobre dicho percance, y al mismo tiempo se envía una señal al Centro de Monitoreo generándose en el mismo un mensaje de alerta y se presenta la opción de la realización de una llamada telefónica a un número designado por el dueño de la vivienda de modo tal que este se encuentre informado del percance.

Una vez detectada la activación de una zona, se dispone de un corto tiempo, en el orden de segundos, para interrumpir el proceso de generación de la señal de alarma, esto se lo realizó con la finalidad de evitar alarmas falsas, que pueden generarse por los mismos residentes de la vivienda.

Una vez generada la señal de alarma, y activada la alarma local – sirena – el sistema local contará 5 minutos para luego proceder a la desactivación de dicha señal audible. Más en el Centro de Monitoreo no se dará por cerrado este evento hasta que sea desactivado tras uno de dos eventos: desactivación a través del ingreso de una clave en el equipo instalado en la vivienda o por desactivación de un administrador del Centro de Monitoreo a través de su ingreso al sistema.

2.2. DIAGRAMAS DE BLOQUES FUNCIONALES DEL SISTEMA

La solución se divide en cuatro componentes principales, uno de ellos, encargado de recibir y procesar la información obtenida de los diferentes sensores utilizados.

El segundo componente, está constituido por el equipo transmisor de la información a través del canal de GPRS, el cual va a acceder a la red del operador local para poder realizar el envío de los mismos.

El tercer componente es la interfaz con el usuario, que comprende los servicios de procesamiento, operación y explotación. Este accede a las bases de datos y es la puerta de entrada a los clientes, mismo que por sus características técnicas y funcionales, permiten un esquema equilibrado entre procesamiento y facilidad de manejo por parte del usuario.

Finalmente tenemos un elemento muy importante en cuanto a registros se trata, pues es en este bloque donde se va a almacenar todos los datos de nuestro sistema y por ende va a ser una base muy importante para el bloque de interfaz con el usuario, se trata de la Base de datos.

A continuación se hace una descripción funcional para cada uno de los bloques operativos antes mencionados:

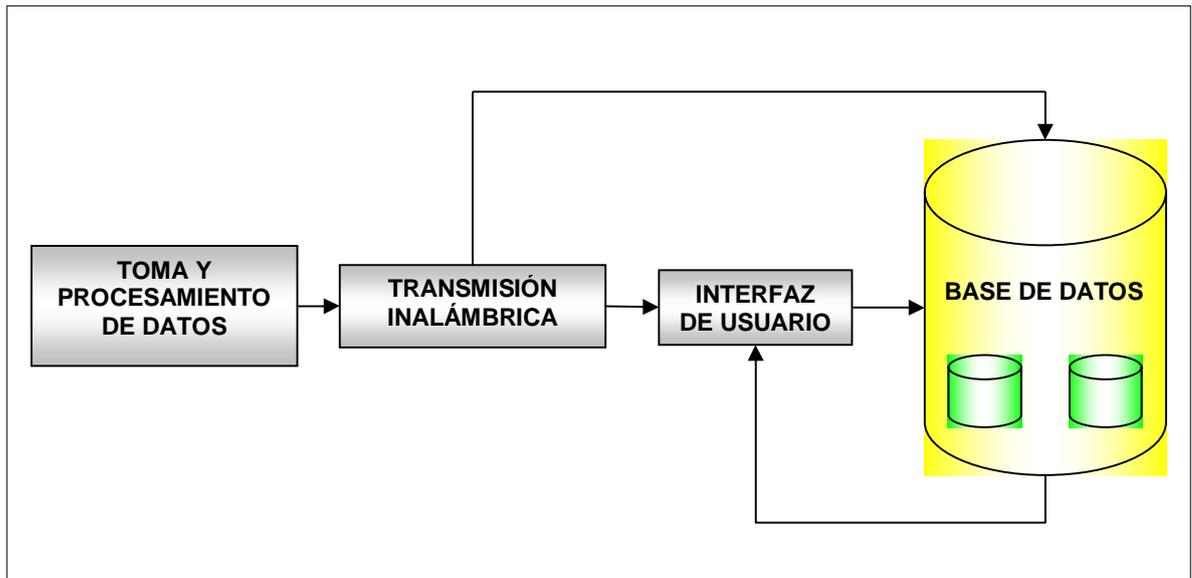


Figura 2.2 Diagrama de bloques del sistema presentado

2.2.1. BLOQUE DE TOMA Y PROCESAMIENTO DE DATOS

Es el bloque es el encargado de tomar la información de los eventos que se están controlando para en base a esta información tomar la decisión de enviar o no señales de alerta al Centro de Monitoreo.

También es el responsable de procesar la información que será transmitida por el siguiente bloque, así también es el encargado de generar las instrucciones necesarias para lograr que el siguiente bloque envíe dicha información, es decir, es el encargado de controlar el bloque de Transmisión.

En base a las funciones que este bloque debe cumplir, podemos decir que se compone básicamente de tres sub-bloques:

- ♦ Monitoreo de eventos y conversión de estos en información.
- ♦ Toma de decisiones, procesamiento de la información.
- ♦ Control de eventos para generación de acciones locales.

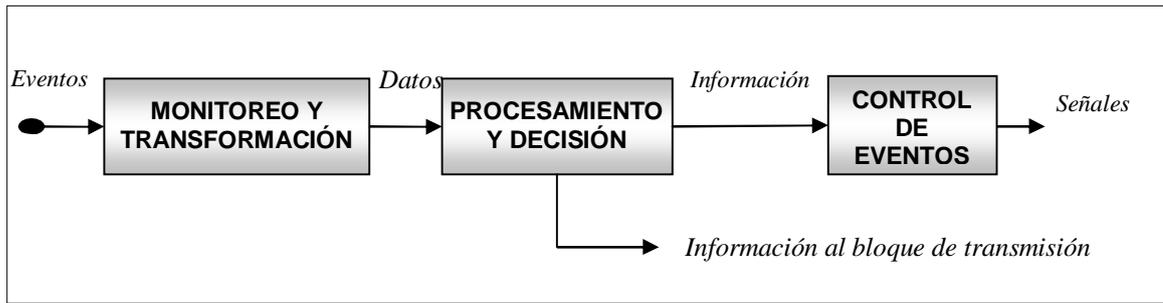


Figura 2.3. Sub-bloques del bloque de Toma y Procesamiento de Información

2.2.1.1. Sub bloque de Monitoreo y Transformación

Es el encargado de la recolección de la información, a este bloque no le interesa el estado de los eventos monitoreados, pues él simplemente censa los eventos y transforma sus resultados en señales eléctricas que serán enviadas al siguiente bloque.

2.2.1.2. Sub Bloque de Procesamiento de la Información

Recibe las señales eléctricas que contiene los datos censados para luego procesarlos y transformarlos en información que permitirá el control de eventos por parte de los siguientes bloques.

2.2.1.3. Sub bloque de Control de Eventos

Este bloque tiene básicamente la función de Activar y Desactivar eventos locales. Básicamente existen dos eventos locales que se controlan:

- ♦ Activación o desactivación de señal de alarma audible: sirena
- ♦ Ejecución de una llamada telefónica.

2.2.2. TRANSMISIÓN INALÁMBRICA DE INFORMACIÓN

Es el encargado del envío de los datos desde la vivienda hasta el Centro de Monitoreo, incluye todos los elementos y procesos que hacen posible una comunicación inalámbrica que en nuestro caso será por GPRS.

En este bloque intervienen los elementos básicos de una comunicación es decir: emisor → canal → receptor, que se representamos en el siguiente diagrama

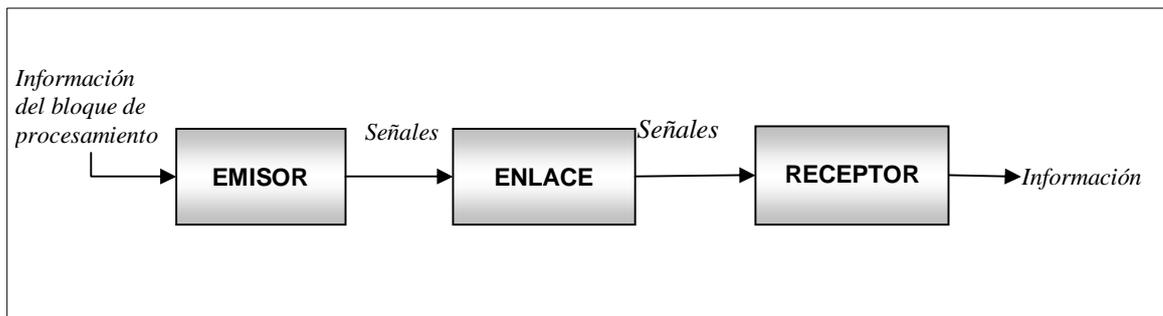


Figura 2.4. Sub-bloques del bloque de Transmisión inalámbrica de Información

2.2.2.1. Sub bloque Emisor

Este bloque recibe la información generada luego del procesamiento y análisis de los datos generados por el monitoreo, para luego transmitir esta información por el canal GPRS.

- ♦ La función principal de este sub bloque es la generación de información a enviar al Centro de Monitoreo así como los comandos necesarios para ejecutar dicho envío

En otras palabras, es el encargado de generar los comandos adecuados y oportunos para garantizar un envío correcto de la información.

2.2.2.2. Sub bloque Enlace

Este es un bloque muy particular pues si bien su importancia es muy grande dentro de nuestro sistema, en su mayoría no está controlado ni diseñado por nosotros ya que el canal de comunicaciones pertenece a nuestro proveedor de servicios GPRS.

Este bloque se encarga de tomar la información que se transmite desde el bloque Emisor y la entrega al bloque Receptor garantizando su fiabilidad, es decir que el dato que recibe del Emisor sea exactamente el mismo dato que entregue.

El correcto funcionamiento de este canal depende mucho de la forma y los parámetros de comunicación que se hayan establecido tanto en el Emisor como en el Receptor, ya que si estos no trabajan bajo el mismo esquema y con los mismos protocolos de comunicación, de nada sirve que este bloque funcione correctamente.

2.2.2.3. Sub bloque Receptor

Este es el bloque encargado de la recepción, procesamiento y almacenamiento de las señales enviadas desde la vivienda. Toma las señales del canal para luego procesarlas y almacenarlas en la base de datos.

2.2.3. INTERFAZ DE USUARIO

Este bloque es la cara que nuestro sistema da al usuario, pues es a través de éste que el usuario puede realizar todas las transacciones que está permitido a realizar.

Este bloque se encarga de conectarse a la base de datos para buscar la información que el usuario busca, organizarla y presentarla en una forma clara y práctica.

Así mismo, en el caso de usuarios con permisos suficientes, este bloque se encargará de tomar los datos que estos usuarios le entreguen y almacenarlos en la base de datos.

Este bloque se preocupa de ordenar la información existente en la base de datos en función de cada uno de los usuarios y clasificarla de manera tal que a cada usuario se le presente únicamente la información que le corresponde.

Está e la capacidad de generar reportes que permitan al usuario tener un conocimiento total de todos los eventos que han sucedido en una vivienda dada o un grupo de ellas.

Adicionalmente es el encargado de informar al usuario sobre la ocurrencia de alguna señal de alarma a través de mensajes y también de permitir a los administradores del Centro de Monitoreo el desactivar una alarma que se ha presentado.

2.2.4. BASE DE DATOS

Se encarga de almacenar toda la información, es la memoria de nuestro sistema. Aquí se almacenan tanto información de cada uno de los clientes y administradores como de las alarmas que se generen.

2.3. DISEÑO DE LOS COMPONENTES DEL SISTEMA

2.3.1. BLOQUE DE TOMA Y PROCESAMIENTO DE DATOS

2.3.1.1. Sub bloque de Monitoreo y Transformación

Este bloque está compuesto únicamente por hardware consistiendo este en sensores de movimiento y apertura o cierre de puertas y ventanas, es decir este bloque básicamente consiste en sensores, y como ellos, su funcionamiento es el traducir una situación en una señal eléctrica.

En el caso de la elaboración de este proyecto se ha utilizados sensores de respuesta digital, es decir unos '1' o ceros '0' para indicar que el evento que se está controlado esta activo o no. Decidimos utilizar este tipo de sensores debido a dos factores:

- ♦ El sistema en si es digital, es decir trabaja a base de Si o No, no considera un evento intermedio.
- ♦ Los eventos que se controlan solo pueden dar dos posibles resultados ya que, por ejemplo, una ventana se la considera en estado de abierta o cerrada, en una habitación existe al menos una persona o está vacía. No se considera casi intermedios.

Considerando que en un concepto global, el monitoreo de los dos eventos a controlar, presencia y apertura/cierre, el sensor utilizado trabaja de la misma forma, explicaremos su funcionamiento refiriéndonos a ambos como un Sensor Modelo.

El funcionamiento del Sensor Modelo es sumamente elemental pues debido a la forma en que se lo conecta tiene el comportamiento de un switch de dos posiciones, el cual permanecerá en un estado inicial mientras las condiciones normales estén presentes y por ende estará enviando un nivel de voltaje al que llamaremos Voltaje de Condiciones Normales.

Al presentarse algún evento dentro de la vivienda que ocasione la alteración de alguno de las condiciones monitoreadas nuestro Sensor Modelo, gracias a su comportamiento de switch, pasará de su estado inicial a un segundo estado, y al mismo tiempo cambiará el nivel de voltaje en su salida, mismo al que llamaremos Voltaje de Condición de Alarma.

Cabe anotar que ambos voltajes: de Condiciones Normales y de Condiciones de Alarma son valores lógicos y opuestos, es decir si el uno toma el valor de uno '1' lógico el otro obligadamente deberá tener el valor de cero '0' lógico. Resumiendo el comportamiento de este Sensor Modelo tendremos

SITUACIÓN	SALIDA
Normal	Voltaje de Condiciones Normales
En alarma	Voltaje de Condiciones de Alarma

Tabla 2.1. Posibles salidas de Sensor Modelo

Como ya se dijo, se lo puede representar como un switch de dos posiciones, cuyo cambio depende de la señal recibida del medio monitoreado:

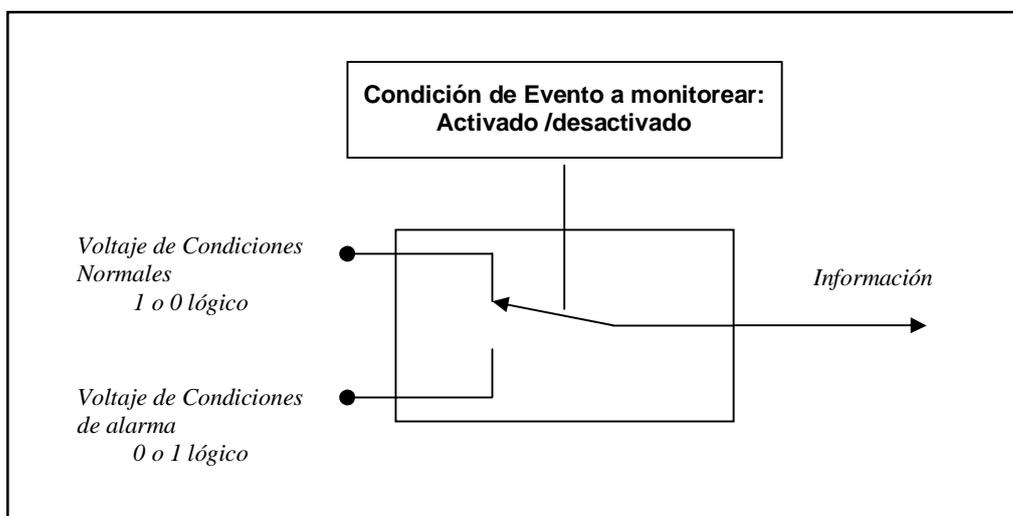


Figura 2.5. Esquema de comportamiento del Sensor Modelo

2.3.1.2. Sub Bloque de Procesamiento de la Información

Este bloque es mas complejo que el anterior, pues se compone tanto de hardware como de software.

Desde el punto de vista de hardware para poder tomar las señales y luego analizarlas debemos tener muy en cuenta que necesitamos utilizar un elemento que sea muy robusto que no sea susceptible a daños o fallas de lecturas, para poder llegar a tener un sistema eficiente, luego de un análisis técnico decidimos trabajar con un microcontrolador PIC, por la facilidad de programación que estos

elementos presentan y la gran gama de lenguajes de programación que existen, cada uno de ellos más fácil que el anterior.

La utilidad primordial del microcontrolador es permitir la ejecución de un programa que, al recibir la información entregada por el bloque anterior la analice y sepa distinguir si los eventos controlados se encuentran en su condición normal o en estado de alarma y si esta segunda opción se encuentra presente en alguno de los eventos, se generen los datos y las instrucciones necesarias para la ejecución de las señales de alarma tanto locales –señal auditiva– como remotas –mensajes en el Centro de Monitoreo–.

Para analizar estas señales las dividimos en zonas para de este modo poder tener un control tanto global como individual de las mismas para lograr una versatilidad del sistema que se verá reflejada cuando necesitemos actuar sobre cada una por separado.

Gracias a que en la programación del microcontrolador es posible trabajar con cada uno de los puertos del mismo como si se tratase de una entrada digital diferente, se ha conectado todos los sensores, que forman parte del sub bloque de Monitoreo y Transformación y que están ubicados en la vivienda, directamente a un pin del microcontrolador, siguiendo un estándar de conexión en cada una de las placas que se implemente, así, en el programa se asigna a un determinado pin un número de zona, facilitando así nuestro proceso de monitoreo, mantenimiento e implementación del equipo.

Cabe recalcar que en esta etapa el sistema no distingue a que tipo de evento pertenece cada zona, simplemente lo conoce como zona 1 , 2 , etc. Sin saber que tipo evento –presencia o apertura/cierre– se está vigilando.

Al analizar las señales vamos a compararlas con su estado normal para así poder obtener todos los posibles casos de combinaciones de eventos a presentarse, en función de los cuales se configurará la acción a tomar.

Programación del Microcontrolador

Por motivos de facilidad y disponibilidad, hemos decidido utilizar un lenguaje de programación Visual, ya que este tiene mucha semejanza con lenguajes antes utilizados en nuestra vida universitaria.

En este medio de programación vamos a encontrar sentencias con estructuras altamente similares a los lenguajes ya conocidos, como pueden ser: el condicional if, el secuencial do, entre otros, cuya aplicación y forma de usarlos son para un estudiante de Ingeniería electrónica conocidos muy ampliamente.

El lenguaje que consideramos apropiado para nuestra aplicación es, como ya dijimos, lenguaje Visual muy similar al Visual Basic en algunas sentencias, debido a que el lenguaje Basic es mucho más fácil de leer y escribir que el lenguaje ensamblador Microchip, y para la creación y edición del código fuente se utilizó el programa Micro Code Studio.

El compilador utilizado es PicBasic Pro (PBP) es un lenguaje de programación de nueva generación que hace mas fácil y rápido el programar micro controladores Pic micro de Microchip Technology.

El PICBasic Pro produce código que puede ser programado para una variedad de micro controladores PIC que tengan de 8 a 68 pines y varias opciones en el chip incluyendo convertidores A/D, temporizadores y puertos seriales.

2.3.1.3. Sub bloque de Control de Eventos

Este sub bloque trabaja dentro del mismo software que el bloque anterior, de Procesamiento de la Información, y por ende se ejecuta sobre el mismo hardware, el microcontrolador, con la adición de elementos de transformación de las señales ya que, por motivos de niveles de voltajes y/o corrientes, resultaría imposible el conectar los dispositivos a controlar de manera directa al microcontrolador.

Este bloque tiene básicamente la función de Activar y Desactivar eventos locales. Básicamente existen dos eventos locales que se controlan:

- ♦ Activación o desactivación de señal de alarma audible: sirena
- ♦ Ejecución de una llamada telefónica.

Activación o desactivación de señal de alarma audible: sirena

En el caso de la señal audible, es decir la sirena realizaremos un circuito en base a un transistor de potencia en su configuración de conmutador, es decir llevando al transistor a sus zonas de corte y saturación la idea es lograr que con la señal enviada por el microcontrolador se controle la activación o desactivación de la sirena.

Ejecución de una llamada telefónica.

En este caso se trabajará con un relé, lo que se busca es que a través de la señal enviada por el microcontrolador se pueda conectar o desconectar nuestro sistema de la línea telefónica, para así dar inicio o fin a una llamada telefónica.

En este caso, utilizaremos una marcación por tonos también llamado sistema multifrecuencial o DTMF (Dual-Tone Multi-Frequency), que consiste en el envío de dos tonos de diferentes frecuencias, estos tonos son decodificados por la central telefónica con la ayuda de filitos y así se detecta el número marcado.

Las frecuencias correspondientes a cada tecla se muestran en la siguiente tabla:

	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

Tabla 2.2. Frecuencias generadas para tonos DTMF

Un esquema básico de la forma en que se conectará esta parte del sistema se muestra a continuación.

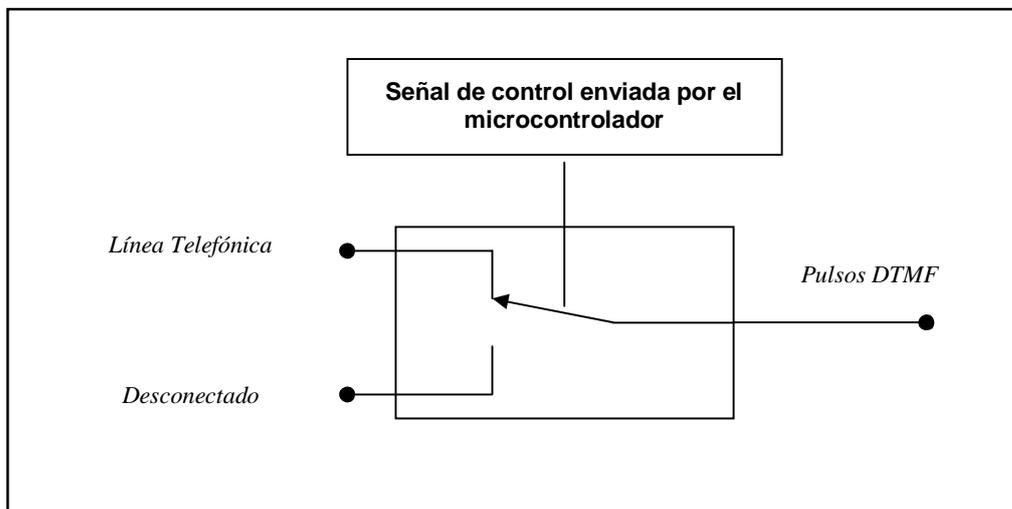


Figura 2.6. Esquema de conexión para ejecutar llamada telefónica

Inicialmente el sistema está desconectado, y al recibir una señal de cambio este conmuta y conecta la salida del microcontrolador con la línea telefónica permitiendo que a través del envío de pulsos DTMF se inicie una nueva llamada telefónica.

2.3.1.4. Tarjeta electrónica

En esta se encuentran todos los bloques antes indicados, mas para empezar con el diseño mismo del sistema es necesario el diseño de una fuente de voltaje capaz de entregar la suficiente corriente y garantizar la estabilidad en los niveles de voltaje, es decir que estos no caigan cuando uno de los elementos que están conectados a nuestra tarjeta entren en actividad.

Analizando todos los componentes que componen nuestra tarjeta vimos que con una fuente que entregue dos niveles de voltaje -5 y 12 voltios- y 1 amperio, funciona sin problema, mas al momento de realizar la conexión serial hacia el modem GPRS observamos un gran problema por los niveles de voltaje, pues por tantos elementos conectados se producía una caída de voltaje que se volvía

crítica para esta comunicación, es por ello que nos vimos obligados a realizar un cambio en la fuente de voltaje que originalmente entregaba 1 amperio por una que entregue 3 amperio, así logramos que las caídas del nivel de voltaje, originadas por el consumo de nuestros dispositivos, resultan imperceptibles a nuestro sistema.

Esta fuente trabaja en función a un conversor de voltaje LM338 que tiene las siguientes características:

- ♦ Voltaje de Salida: Entre 1,2 y 40 Voltios
- ♦ Corriente entregada: hasta 5 amperios con un voltaje entre 1,2 y 32 voltios.
- ♦ Esquema de conexión: el esquema de conexión de este conversor se muestra en la figura siguiente

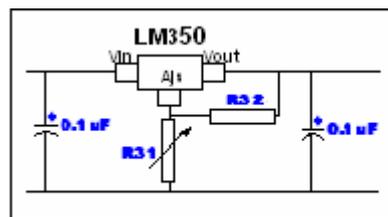


Figura 2.7. Esquema de conexión del conversor LM338

Resuelto el problema de la alimentación, seguiremos con los elementos de potencia, pues, como se indicó anteriormente tanto para la ejecución de la llamada telefónica como para la activación o desactivación de la sirena, se utilizan etapas de potencia ya sea para controlar directamente un dispositivo, como es el caso de la sirena, o para controlar un actuador como en el caso del relé con el que se conecta nuestro sistema a la línea telefónica.

En toda etapa de potencia en esta tarjeta se utilizaron transistores de potencia configurados de forma tal que con la señal de control se lo lleve de su zona de corte a la de saturación, obteniéndose así un circuito de conmutación.

Debido a que los elementos que van después de la etapa de potencia contienen bobinas, como es el caso del relé y las sirenas, es necesario proteger

nuestro circuito contra las corrientes parásitas que se generan por inducción, esto lo logramos con la colocación de un diodo en anti paralelo a la carga, tal como se muestra en el esquema siguiente:

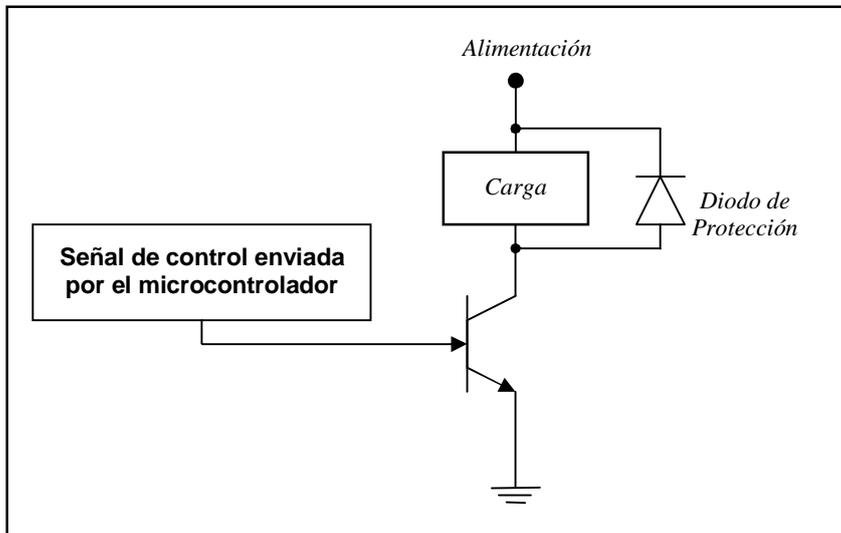


Figura 2.8. Esquema de conexión para una etapa de potencia

2.3.1.5. Diagramas electrónicos

Todos los diagramas electrónicos tanto del circuito como de la placa realizada se encuentran en el Anexo I

2.3.2. TRANSMISIÓN INALÁMBRICA DE INFORMACIÓN

En este bloque se incluyen todos aquellos elementos involucrados en la transmisión de la información desde la vivienda hasta el centro de monitoreo incluyendo tanto el proceso de envío como el de recepción de esta información.

Si nos referimos a los componentes de este bloque veremos que este resulta ser el mas complejo y peculiar, pues se compone de un sinnúmero de elementos entre los cuales se incluyen algunos en los cuales nosotros no tenemos nada que ver y no podemos alterarlos, mas cada uno de estos elementos los explicaremos a su momento.

Básicamente en este bloque intervienen los elementos básicos de una comunicación es decir: emisor → canal → receptor mismos que los representamos en el siguiente diagrama

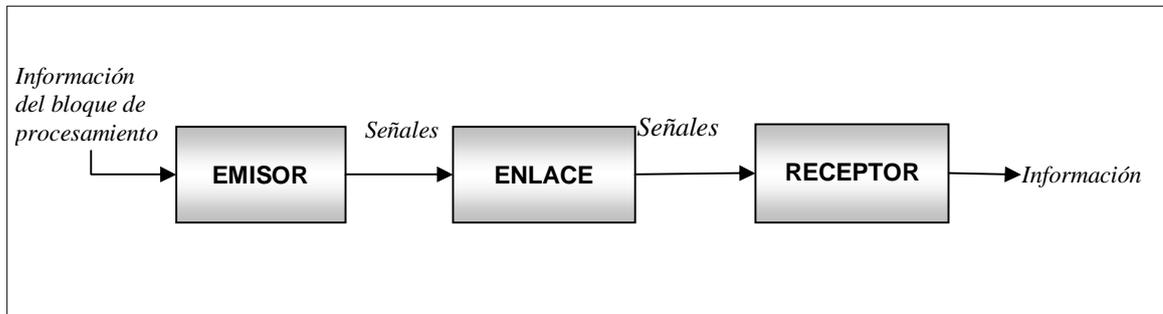


Figura 2.9. Sub-bloques del bloque de Transmisión inalámbrica de Información

2.3.2.1. Sub bloque Emisor

Es el bloque que recibe la información generada en el bloque de toma y procesamiento de datos.

A nivel de Software este está incluido en dos partes:

- ♦ La primera es el programa que se ejecuta sobre el microcontrolador, sobre esta parte del software nosotros tenemos total control, pues el programa lo realizamos en su totalidad.
- ♦ La segunda es el programa que se ejecuta en el modem mismo que viene grabado de fábrica, por ello este no fue escrito por nosotros mas a través de comandos e instrucciones podemos controlarlo.

A nivel de hardware este bloque se ejecuta básicamente sobre tres elementos:

- ♦ Microcontrolador: sobre el cual también se ejecuta el bloque anterior.
- ♦ Modem: que nos permite el envío de datos
- ♦ Conversor TTL a RS232 que establece los niveles de voltaje tales que permiten la comunicación entre el modem y el microcontrolador.

Este bloque recibe la información generada luego del procesamiento y análisis de los datos generados por el monitoreo, para luego transmitir esta información por el canal GPRS.

- ♦ La función principal de este sub bloque es la generación de información a enviar al Centro de Monitoreo así como los comandos necesarios para ejecutar dicho envío

Generación de datos a enviar al Centro de Monitoreo así como los comandos necesarios para ejecutar dicho envío

El envío de ambos tipos de información – instrucciones y datos – se lo realiza a través de comunicación serial, así que aquí es donde encontramos el conversor TTL RS232: ya que el nivel de voltaje del microcontrolador es TTL, es decir un uno lógico equivale a 5 Voltios y en comunicación serial se trabaja con el protocolo RS232 que requiere niveles de voltaje de +12 Voltios, es necesario el colocar un conversor de niveles de voltaje que en este caso es el dispositivo MAX232, su conexión es sencilla y fácil, se alimenta, igual que el microcontrolador, con señales TTL y entrega niveles RS232.

Para establecer esta comunicación se han definido algunos puntos:

- ♦ Debido a que el módem interpreta todo lo recibido como si estuviese en código ASCII, es precisamente en este formato que vamos a enviar tanto las instrucciones como los datos
- ♦ La comunicación entre el microcontrolador y el módem será de forma asíncrona
- ♦ No se ha planificado el envío de señales desde el Centro de Monitoreo hacia la vivienda, pues únicamente se realizará una comunicación unidireccional desde la vivienda hacia el Centro.

- ♦ Por el modelo de comunicación aquí establecido, solo se utilizará el pin de transmisión del microcontrolador, y el de tierra en nuestra comunicación serial.

Una vez definida la forma en que se va a realizar la conexión para la comunicación entre el microcontrolador y el módem, debemos establecer dos parámetros:

- ♦ En que manera se va a enviar la información
- ♦ Los comandos necesarios para realizar el control del modem,

Envío de la información: Trama de datos

Como ya se explicó anteriormente, el envío de los datos es por la red GPRS, y en si este sistema ya tiene una trama de datos, y debido a que del sistema GPRS ya hablamos en el primer capítulo, en este punto no lo incluiremos, mas hablaremos únicamente de la trama que elaboramos nosotros.

Debido a que tanto en transmisión como en recepción se trabaja en formato texto, definiremos el tamaño de la trama en función de cuantos caracteres lo conforman, Esta trama contiene los siguientes campos:

- ♦ Código del cliente: que es un valor generado por medio de software al momento en que se ingresa el cliente al sistema. Este campo se compone de 4 caracteres.
- ♦ Zonas: este campo se compone de dos dígitos por cada zona, estos dígitos pueden tener uno de dos valores según los dos posibles estados de la zona monitoreada:
- ♦ Zona en condiciones normales (no existe alarma): se enviará el valor de XX en toda zona que se encuentre en este estado.

- ♦ Zona en condiciones de alerta (alerta activada): se enviará el dígito 'Z' mas uno que represente la zona en cuestión, así por ejemplo si la zona 4 esta en este estado, se enviará el código 'Z4' .

Debido a que tenemos seis zonas este campo tendrá una longitud total de doce (12) caracteres estructurados de la siguiente forma:

Zona 1	Zona 2	Zona 3	Zona 4	Zona 5	Zona 6
Z/X 1/X	Z/X 2/X	Z/X 3/X	Z/X 4/X	Z/X 5/X	Z/X 6/X

Tabla 2.3. Trama de Zonas

Como se ve en el esquema, cada carácter puede tomar uno de dos valores, para hacer un poco mas entendible este tema, veamos un ejemplo:

Supongamos que en una vivienda se encuentran en estado de alarma las zonas 1, 3 y 6, así, la trama correspondiente a zonas será la siguiente: Z1XXZ3XXXXZ6.

- ♦ Fin de Trama: se enviará dos caracteres que nos permitirán identificar en que momento finalizó la trama, esto se lo realiza para evitar posibles montajes de datos.

Sumando los doce caracteres de las zonas a los cuatro que enviaremos inicialmente con el código del cliente y los dos caracteres de fin de trama, tendremos en total una trama de dieciocho (18) caracteres.

Así, completando el ejemplo anterior, la trama enviada sería:

Código del cliente: H45T (puede componerse de letras o números indistintamente)

Código de zonas: Z1XXZ3XXXXZ6 (activas las zonas 1, 3 y 6)

Fin de Trama: AA

Trama enviada: H45TZ1XXZ3XXXXZ6AA

Control y comunicación con el modem: Comandos AT

Los comandos AT (attention command) son, en pocas palabras, un tipo de comandos para llamar la atención al modem e indicarle que hacer. En otras palabras, son un conjunto de comandos necesarios para el control del modem.

Propósito de los Comandos AT

Los Comandos AT representan el único lenguaje que un módem entiende, por ello nos permiten básicamente tres acciones:

- ♦ Establecer parámetros iniciales de configuración de un modem, tal como la tasa de transmisión, la red a la que pertenece, entre otros.
- ♦ Iniciar una llamada de datos
- ♦ Enviar comandos de acción mientras el modem está trabajado, tal como el inicio o fin de una llamada.

Comunicación con el modem

Los módem siempre funcionan en uno de dos modos:

- ♦ El modo de comando: Este modo se usa para la configuración del módem o para iniciar la comunicación. Una vez recibido un comando el modem lo ejecuta para luego regresar un código de resultado de confirmación.

En este estado el modem reconoce todo lo recibido como comandos e intenta ejecutarlos, para pasar de este estado al modo en línea basta con iniciar una llamada que en nuestro caso será de datos.

En este estado, si ocurre un error durante la ejecución de una línea de comando, el procesamiento se detiene y todo aquello que sigue al comando incorrecto se ignora.

- ♦ El modo en línea: Este modo se establece una vez iniciada una conexión, es decir una llamada que en nuestro caso es una llamada de datos, en este estado el modem convierte todo tipo de datos que reciba por el puerto serial para luego proceder al envío de estas señales a través del canal de comunicación.

En este estado el modem reconoce todo lo recibido como datos, los convierte y lo transmite, inclusive si lo recibido es un comando AT, a excepción de ciertos eventos especiales:

- ♦ Un punto y coma (;) ocurre al fin de la secuencia de marcado
- ♦ El módem recibe una secuencia de escape definida o una señal de interrupción mientras está en el modo en línea, esto suele ocurrir con el envío de un comando especial utilizado para finalizar la llamada que generalmente suele ser 'ctr+c'.
- ♦ Se desconecta una llamada por cualquier motivo ajeno tal como la desconexión del módem remoto o caída del servicio.

2.3.2.2. Sub bloque Enlace: Detalles de la conexión

Este es un bloque muy particular pues el enlace en si entre el modem y el Centro de Monitoreo no depende en lo absoluto de nosotros. Consiste en un canal GPRS completamente transparente cuya calidad es garantizada por nuestro proveedor CONECEL quién actualmente es el único que ofrece una comunicación GPRS probada, existe otra operadora TELEFÓNICA que también está incursionando en este campo, pero debido a que está apenas empezando, decidimos trabajar con CONECEL.

Este servicio de GPRS funciona a través de la creación de una VPN por parte del proveedor, misma que nos asignará. En esta VPN o Red Privada Virtual se contrata un número máximo de clientes, este número se valida a través del set de direcciones IP que tenemos disponibles.

Cada IP pertenecerá a un módem mismo que accederá a la VPN a través de su identificación en la red del proveedor a través de los datos almacenados dentro de una Sim Card. En lo referente a la asignación de direcciones IP existen dos posibilidades: el tener una asignación dinámica en la cual cada vez que un equipo se conecte a la red adquirirá una dirección diferente, o la asignación de una IP fija.

En este punto es importante analizar una situación particular y es que si un equipo tiene una IP fija, y por alguna razón pierde el enlace a la red, el volver a conectarse le toma un largo tiempo, la razón de este evento lo explicaremos a través de un consecutivo de eventos:

1. El modem pierde la conexión y sale de la red
2. Inmediatamente el modem intenta volver a conectarse
3. Al llegar al firewall del proveedor este compara la dirección IP con la que el modem intenta ingresar con las tablas de direcciones que este reconoce como dentro de la red.
4. Si aún no se han actualizado estas tablas, evento que suele ocurrir muy frecuentemente, el firewall reconoce a este intento de acceso como una agresión y niega el ingreso a la red a nuestro equipo.

Si bien las tablas de direcciones IP dentro de una red son actualizadas en un período corto de tiempo por los elementos de conectividad de esta red, comprobamos que el equipo llegaba inclusive a pasar horas sin lograr acceder a la red.

Por esta razón se decidió utilizar una VPN con asignación de la dirección IP de forma dinámica. Pues con ello cada vez que un equipo trate de ingresar a la red lo hará con una dirección diferente y se elimina prácticamente en su totalidad el problema de desconexiones accidentales.

Aquí encontramos nuestra primera necesidad de Servicios Contratados: Servicio GPRS con creación de una VPN, asignación dinámica de direcciones IP.

En este bloque también se puede incluir un pequeño pero muy importante proceso del diseño y es el método de transmisión – recepción de los datos, es decir, el definir de que manera se va a transmitir los datos para que lleguen al Centro de Monitoreo.

Como ya se indicó, el canal de GPRS que nuestro proveedor ofrece es completamente transparente, mas, adicionalmente nos presenta la opción de un canal que luego de recibir la información que enviemos éste la coloque en una dirección IP que nosotros definamos, esta IP la configuramos directamente en el modem. Es importante el recalcar que esta IP debe obligatoriamente ser pública, y a través de ésta, el equipo en el que se encuentra nuestra base de datos deberá estar conectado 24 horas a Internet, de aquí vemos nuestra segunda necesidad de Servicios Contratados: Acceso a Internet que para asegurar la calidad del servicio debe ser de Banda Ancha y con IP fija. Este tipo de cualidades lo tienen los servicios conocidos como Acceso Corporativo y en el mercado existe un sinnúmero de proveedores.

Hasta aquí hemos dado la dirección a donde deben llegar los datos, mas, si hacemos una homología, podríamos comparar esta dirección con un edificio de oficinas en el cual existen un sinnúmero de departamentos cuyo acceso tiene su propia numeración, y es el número de este “departamento” el que debemos asignar.

Para esta asignación vamos a incluir en nuestro trabajo un concepto adicional y es el de “sockets”.

Conexión a través de Sockets

A través del término sockets se designa un concepto abstracto a través del cual dos programas que no necesariamente corren sobre el mismo equipo logran conectarse y comunicarse intercambiando datos de una forma ordenada y confiable.

Para definir un socket es necesario el establecer previamente tres parámetros:

- ♦ Dirección IP en la que se encuentra el programa al que se va a acceder.
- ♦ Protocolo: que no es mas que un conjunto de reglas que controlan la secuencia de mensajes que ocurren durante una comunicación entre entidades que forman una red. En nuestro caso utilizaremos el protocolo de comunicación TCP.
- ♦ Puerto: que no es más que una dirección exacta, de una entrada por la cual ingresarán todos los datos, regresando a la homología antes realizada, este puerto se lo puede comparar con el número de la oficina a la que se desea ingresar.
- ♦ Un número de puerto, que identifica a un programa dentro de una computadora.

2.3.2.3. Sub bloque Receptor

Este es el bloque encargado de la recepción, procesamiento y almacenamiento de las señales enviadas desde la vivienda.

Físicamente este bloque se encuentra en un computador –servidor- que se encuentra conectado permanentemente a Internet.

Considerando que el hardware de este sub bloque consiste únicamente en el computador, diremos que este bloque es básicamente software.

Previo al diseño de este software se han planteado las siguientes funciones que este debe desempeñar:

- ♦ Debido a que será el encargado de recibir y almacenar toda señal de alarma que se genere en una vivienda, debe estar funcionando todo el día y todos los días del año.
- ♦ Debe monitorear todo el tiempo el puerto por el cual se reciben los Consiste en un servicio Windows configurado de tal modo que cada vez que el

computador se encienda o reinicie por cualquiera sea el motivo, este servicio se iniciará automáticamente.

- ♦ Cada dato que sea recibido será validado, procesado y esta información se almacenará en la base de datos.

En función a lo aquí expresado llegamos a la conclusión de que lo que requerimos es la elaboración de un Servicio Windows.

Elaboración del Servicio Windows

Un Servicio Windows no es más que un tipo de aplicaciones que ejecuten tareas de larga duración y que se inician automáticamente ante determinados eventos como al encender la computadora, pueden ser pausadas, reiniciadas, carecen de interfaz y no interfieran con los otros trabajos que realizan los usuarios en la misma computadora. Tales aplicaciones son idóneas cuando se desea implementar agentes que trabajen en background y realicen tareas de rutina de forma periódica o que traten con carga de trabajo considerable.

Este servicio será elaborado en la plataforma .NET, en Visual Basic . NET, pues este presenta ya una plantilla para la elaboración de un servicio, misma que ya trae incorporado los elemento y referencias básicas para dicha creación.

Básicamente nuestro servicio Windows lo que hace es leer el puerto por el cuál recibirá información del estado de las viviendas monitoreadas, esta información no es más que una cadena de caracteres formada, como ya se indicó anteriormente, por el código de usuario –4 caracteres– y descripción del estado de las zonas –12 caracteres– .

Una vez recibida esta cadena, nuestro programa, revisa su longitud para luego proceder a almacenar los datos. Un diagrama de flujo de este bloque se muestra en el Anexo II.

2.3.2.4. Diagrama Electrónico

Debido a que es únicamente la parte del Emisor en la que interviene hardware diseñado por nosotros, es esta la única parte de la que se puede mostrar un diagrama electrónico, este se encuentra en el Anexo I.

2.3.3. INTERFAZ DE USUARIO

Como se definió desde un comienzo en este trabajo, se busca el crear una interfaz que permita al usuario acceder a la información de su vivienda desde cualquier parte del mundo donde se tenga un computador con Internet.

Es por ello que elaboramos un aplicativo web basándonos nuevamente en las ventajas que la plataforma .NET presenta, y escribimos el código en el programa Visual Basic .NET, este programa presenta ya una plantilla para aplicativos web, en esta plantilla se encuentran incluidos todas la referencias básicas necesarias para la elaboración de este tipo de aplicativos.

Es el bloque encargado de acceder a la base de datos y presentar toda la información en esta base disponible en una forma visual y amigable que permita al usuario pueda ver el estado de su domicilio, así como la administración del sistema por parte de las personas que se encuentran a cargo del Centro de Monitoreo.

Esta interfaz no es más que un programa ejecutable compuesto por un conjunto de páginas web cada una con una función específica, cuyo acceso se permite según el nivel que tenga el usuario, para lo cual se ha establecido dos tipos de usuarios:

- ◆ Administrador: Este tiene acceso a la información de todos los clientes y está en posibilidades de:
- ◆ Crear, editar o borrar usuarios de menor jerarquía: clientes.
- ◆ Editar nombres de las zonas del domicilio de un cliente

- ♦ Desactivar una señal de alarma
- ♦ Cliente: tiene acceso únicamente a su información y la de su vivienda, no puede editar ni eliminar elemento alguno, únicamente puede visualizar.

En la elaboración de cada página se puso especial atención en el elaborarlas de manera tal que resulten sumamente amigables y fáciles de utilizar, es por ello que En cada una de ellas se muestra una breve descripción de las actividades que se puede realizar.

Una vez explicada la interfaz de Usuario, podemos definir los requisitos básicos para que una persona puede acceder a nuestro sistema y estos son tres:

- ♦ Un computador con acceso a Internet.
- ♦ El ejecutable de nuestro programa –interfaz de usuario-
- ♦ Clave de acceso.

2.3.3.1. Centro de Monitoreo de Viviendas

Si bien la implementación de un centro de monitoreo no es parte de este proyecto creemos necesario, por la perspectiva con que se implementa el mismo, el mencionar este punto.

Es importante recalcar que dada la arquitectura del sistema, no es necesario instalar aplicaciones, equipamiento o servicios del lado cliente es decir en el Centro de Monitoreo, sino que este puede operar en línea disponiendo de un terminal (PC) y una adecuada conexión a nuestro servidor a través de Internet.

Esta posibilidad genera ahorros de infraestructura y mantenimiento y aumenta la confiabilidad del sistema, dado que los servicios se brindan de manera centralizada mediante una plataforma de alta disponibilidad.

Gracias a este particular es posible a través de nuestra base de datos y de asignación de claves el poder tener mas de un grupo de clientes, así por ejemplo se puede tener los clientes divididos por condominios, zonas, etc.

Si buscamos un similar podríamos pensar en un seguro médico que es ofrecido al personal de varias empresas, donde se atiende a cada uno de los empleados, pero en realidad el cliente es la empresa, así cada uno de los individuos pertenece a un grupo y el administrador de este grupo solo tiene acceso a la información de los miembros del mismo desconociendo por completo el estado de los demás grupos.

Así es posible con nuestro servidor proveer la información y seguridades necesarias para abastecer a mas de un grupo de viviendas montando en cada una de ellas un pequeño centro de monitoreo cuyo costo y complejidad es bajo lo que resulta en una fácil expansión del servicio.

2.3.3.2. Diagrama de Flujo

Este programa define los alcances y limitaciones del sistema mediante el reconocimiento del perfil de usuario que tiene acceso al sistema.

Cada usuario del sistema tiene su NICK (Identificación de Usuario) y Contraseña (Password), además se le definen perfiles con niveles de accesos y roles definidos.

Para poder ingresar al sistema, siempre es necesario que el usuario haga login en el sistema. Una vez que haya ingresado se le habilitarán las pantallas y opciones que su perfil le permita.

El diagrama de flujo de este bloque se encuentra en el Anexo II.

2.3.4. BASE DE DATOS

Como ya se dijo, este bloque es la memoria de nuestro sistema, la elaboramos en SQL2005, la razón de por qué utilizar esta herramienta es la gran capacidad que SQL posee para almacenamiento de datos. Así como todas las herramientas necesarias para su desarrollo,

2.3.4.1. Configuración de la Base de Datos

Consiste en un conjunto de seis (6) tablas interconectadas entre si, estas tablas son:

- ◆ Cliente: Contiene toda la información personal referente a cada uno de los usuarios del sistema, tal como: Nombre, Nick, Clave, Tipo de usuario (administrador o cliente), entre otras.
- ◆ Alertas: Esta tabla tendrá almacenado los datos que se recibirán de los clientes si tuviesen una violación en las zonas protegidas.
- ◆ Alertazonas: esta tabla contendrá la cantidad de alertas por zona de un cliente
- ◆ Clientezona: En esta tabla se almacenará la descripción de la zona por cliente, esto quiere decir por ejemplo si la zona 1 es la sala se almacenará sala.
- ◆ Nivel: Esta tabla contendrá la información de los niveles de cada usuario para saber si son administradores o usuarios.
- ◆ Zonas: Aquí le daremos el nombre a cada zona por ejemplo a la ZONA 1 le llamaremos Z1, así para cada una de las demás.

Un esquema explicativo de la configuración de esta base de datos y de cómo se interconectan las diferentes tablas se muestra en el Anexo III.

2.4. ESPECIFICACIONES TÉCNICAS DE LOS ELEMENTOS DEL SISTEMA

2.4.1. TARJETA ELECTRÓNICA

Fabricante:

- ◆ Fabricación Particular

Características:

- ◆ Voltaje de alimentación para la fuente: 110 Voltios
- ◆ Voltaje de alimentación para placa madre: 12 Voltios (Relé, chicharra) – 5 Voltios resto de circuitos incluido microcontrolador.
- ◆ 6 Entradas digitales para conexión de sensores
- ◆ Velocidad de procesamiento 4 MHz
- ◆ 1 Salida serial para conexión con módem
- ◆ 1 Salida analógica para conexión con línea telefónica
- ◆ 1 Salida digital para conexión con sirena

2.4.2. MÓDEM GPRS**Fabricante:**

- ◆ Multitech

Modelo:

- ◆ ENFORA MTCBA-G-F1

Características:

- ◆ Interfaz RS232- DB-9
- ◆ Dimensiones :3 2.5 x 2.5 x 0.94 pulgadas
- ◆ Caja de protección: Aluminio
- ◆ Conector para antena SMA
- ◆ Radio Frecuencia: 850/900/1800/1900
- ◆ Sensibilidad: -106 dB (Típico)

- ◆ Potencia de transmisión:
- ◆ Clase 4 : 2W a 850/900 MHz
- ◆ Clase 1 1W a 1800/1900 MHz
- ◆ Indicador de estatus: Encendido / Conexión a la red
- ◆ Paquete de datos GPRS
- ◆ Modo: Clase B, Multislot 10 Protocolo: GPRS Rel 97 y 99, SMG 31
- ◆ Esquemas de codificación: CS1-CS4
- ◆ Packet Channel: PBCCH/PCCCH
- ◆ Funcionalidades GSM
- ◆ Voz: FR, EFR, HR & AMR
- ◆ CS Data: Asynchronous, Transparente y No-Transparente sobre los 14.4 KB
- ◆ GSM SMS: Text, PDU, MO/MT
- ◆ Cell broadcast
- ◆ Interfáz de aplicación
- ◆ Protocolos de enlace: AT Commands, UDP/API
- ◆ Protocolos Internos: PPP, UDP/API, UDP/PAD, CMUX, TCP/PAD
- ◆ API Control / Estatus: AT o UDP
- ◆ Comportamiento en red IP: Software de Auto-Registro al encenderse
- ◆ Ingreso de SIM externo a – 3V con mecanismo de anclaje
- ◆ Temperatura de trabajo: -20°C a 60°C
- ◆ Temperatura de almacenamiento: -40°C a 85°C

- ◆ Humedad: Hasta 95% no-condensing
- ◆ Voltaje DC: 5V – 9V DC (2 Pin Molex Conn)
- ◆ Certificaciones:
- ◆ FCC: Part 15, 22 & 24
- ◆ GCF: Version 3.11
- ◆ PTCRB: Version 2.9.1
- ◆ Industry Canada and CE Mark
- ◆ Número de parte: MTCBA-G-F1

2.4.3. INTERFAZ DE USUARIO

Fabricante:

- ◆ Fabricación Particular

Características:

- ◆ Desarrollado en lenguaje Visual Basic.NET
- ◆ Compatible con bases de datos MySQL, PostGresSQL y Oracle.
- ◆ Acceso a usuarios para explotación vía Internet
- ◆ Flexibilidad en el manejo:
- ◆ Pantalla adaptable en forma dinámica.
- ◆ Fácil e intuitivo
- ◆ Perfiles usuario y control de privilegios

2.4.4. BASE DE DATOS

Fabricante:

- ♦ Microsoft

Características:

- ♦ Almacenamiento en tablas relacionadas entre si
- ♦ Separación de la información por categorías y almacenamiento de cada una en tablas diferentes.
- ♦ Amplio espacio de almacenamiento

CAPITULO III

IMPLEMENTACIÓN DEL SISTEMA

3.1. CARACTERÍSTICAS DE LA TARJETA ELECTRÓNICA

Nuestra tarjeta electrónica se divide básicamente en tres bloques:

- ◆ Fuente de alimentación
- ◆ Centro de procesamiento
- ◆ Periféricos para interacción con el usuario

Por motivos de seguridad del sistema que se está diseñando, se ha decidido el implementar los dos primeros bloques en uno solo que incluye la tarjeta electrónica, elementos externos: como el transformador requerido para la fuente y su caja de protección, y el tercer bloque se implementará por separado en su propia tarjeta y con su propia caja de protección, uniéndolas a través de un bus de dato.

La explicación de el por qué de esta separación es muy sencilla y obedece a la finalidad de este proyecto: seguridad, pues, considerando que los elementos para la interacción con el usuario deben ubicarse en un lugar visible y accesible dentro de la vivienda, es muy vulnerable, pues en el caso en que un delincuente o un extraño ingrese al domicilio y el sistema de alarma se active, lo primero que éste individuo va a intentar es el anular la alarma atacando a este bloque a través del corte de cables o golpeándolo.

Adicionalmente a este peligro, existen otros posibles ataques accidentales que este bloque puede sufrir como golpes, líquidos, y otros posibles que los mimos residentes del domicilio pueden realizar de manera no intencional.

Y si el bloque de procesamiento o el de alimentación se encuentran físicamente a la vista será muy fácil el anular nuestro sistema, mas si los ubicamos en otro lugar que físicamente sea mas difícil de encontrar tenemos una mayor protección ante cualquier ataque.

3.1.1. FUENTE DE ALIMENTACIÓN

Tenemos dos fuentes de voltaje: una de 12 Voltios y otra de 5 Voltios, mismas que están conectadas en cascada, y pueden suministrar hasta 5 amperios para el total del sistema; las justificaciones y causas para esta configuración de fuente son:

1. Fuente que entregue hasta 5 amperios: pues se tiene equipos que requieren altas corrientes, estos son:
 - ♦ Modem: consume hasta 0,42 amperios
 - ♦ Sirenas: su consumo llega hasta 0,3 amperios
 - ♦ Pantalla del LCD: requiere hasta 0,4 amperios
 - ♦ Relé: consume hasta 0,2 amperios
2. Conexión en cascada: en nuestra tarjeta las fuentes están conectadas de manera tal que la fuente de 5 voltios se alimenta de la de 12 voltios, bajo el esquema siguiente:

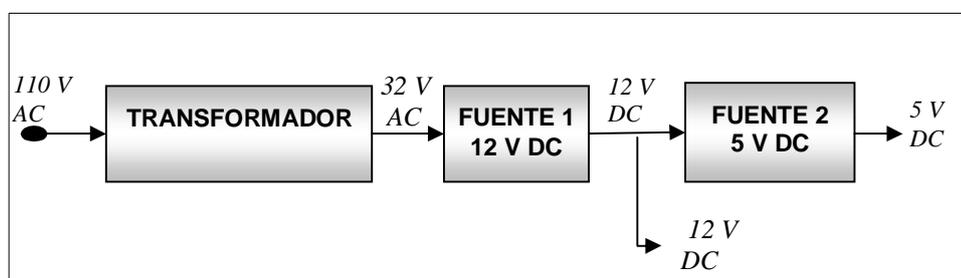


Figura 3.1. Interconexión entre los bloques de la fuente de alimentación

El motivo de esta conexión es muy sencillo, pues dentro de la fuente de 12 Voltios se ha incorporado una batería de respaldo que nos permitirá mantener el sistema funcionando durante un tiempo aproximado de mínimo 2 horas en el caso en que el suministro de corriente alterna se suspenda, siendo esta batería de respaldo de 12 Voltios DC, es por ello que, para poder alimentar a los elementos TTL con esta batería es necesario reducir primero el voltaje.

3.1.2. CENTRO DE PROCESAMIENTO

Consiste de todos los elementos necesarios para que el sistema procese la información y genere las acciones necesarias, es decir es el cerebro de nuestra tarjeta. Que, en general, se compone de dos tipos de componentes, según el nivel de voltaje que estos requieren:

- ♦ Elementos de procesamiento: que en general son todos aquellos elementos encargados de tareas de procesamiento: microcontrolador, conversor RS232 a TTL, y elementos menores como resistencias, capacitores, cristal. Estos elementos se alimentan, todos, con un voltaje de 5 Voltios (TTL)
- ♦ Elementos de control y potencia: son los elementos que utilizamos para activar o desactivar un evento: relé y transistores, estos últimos si bien son controlados por una señal TTL, no es este el nivel de voltaje que utiliza para alimentar a su carga. Estos elementos trabajan con niveles de 12 Voltios.

3.1.3. PERIFÉRICOS PARA INTERACCIÓN CON EL USUARIO

Básicamente tenemos tres elementos, dos son para entregar información al usuario y uno para permitirle el activar o desactivar las zonas a monitorear.

Todos estos elementos trabajan a un nivel de voltaje de 5 voltios DC (TTL) y están, como ya se indicó, físicamente separados de los dos bloques antes mencionados, pues se encuentran en una caja de protección diferente, misma que va instalada en una parte visible y accesible dentro de la vivienda.

Por su ubicación es un bloque sumamente vulnerable pues está al alcance de muchos, más no es un bloque fundamental para el sistema pues el centro de procesamiento no se encuentra aquí.

Los elementos que componen este bloque son:

- ♦ LCD: que nos permite presentar mensajes explicativos al usuario para facilitarle el trabajo con el sistema.
- ♦ LEDS: a través de los cuales mostramos dos cosas: si la alarma está activada o no, y en el caso de estarlo: que zonas son las que se está monitoreando.
- ♦ TECLADO: a través del cual el cliente puede:
 - ♦ Activar o desactivar el sistema
 - ♦ Desactivar el sistema cuando se encuentra en estado de alarma, es decir, cuando una de las zonas monitoreadas ha presentado algún problema.
 - ♦ Escoger que zonas desea monitorear
 - ♦ Cambio de clave

3.2. CARACTERÍSTICAS DEL ENLACE GPRS

Básicamente diremos que el enlace se caracteriza por:

- ♦ Canal Transparente
- ♦ Trabajo bajo la creación de una VPN
- ♦ Asignación de dirección IP dinámicamente

3.2.1. INICIO DE LA COMUNICACIÓN: CONFIGURACIÓN DEL MODEM

Para la implementación de la comunicación por GPRS inicialmente se debe configurar al modem con los parámetros básicos que este requiere.

Esta configuración inicial se la realizó a través del aplicativo Hyperterminal de Windows, conectando el modem al puerto serial de la computadora, para ello se sigue las pautas que indica el manual de este equipo, estas son:

Bits por segundo: 115200

Bits de datos: 8

Paridad: Ninguna

Bits de parada: 1

Control de flujo: Ninguno

Una vez conectados es necesario configurar los siguientes puntos:

3.2.1.1. Velocidad de comunicación

Al revisar los manuales de un modem vemos que generalmente estos equipos se comunican a una velocidad de 115200 Bps, mas esta velocidad de transmisión resulta demasiado elevada para el microcontrolador, es por ello que es necesario configurarlo a la velocidad de 9600 Bps, velocidad comúnmente utilizada en una comunicación serial con el microcontrolador.

Para lograr este cambio empleamos el comando AT: AT+IPR, mismo que nos permite establecer la velocidad de comunicación, este comando presenta, tres formatos:

- ♦ AT+IPR=? : Nos presenta todas las posibles velocidades que se puede establecer, en el caso del modem Enfora son: 75,150,300,600,1200,2400,4800,7200,9600,14400,19200,28800,33900,38400, 57600 y 115200.

- ♦ AT+IPR? Nos presenta la velocidad actual del equipo.
- ♦ AT+IPR= <velocidad>: Nos permite establecer la velocidad del puerto.

Una vez establecida la velocidad en 9600 es necesario reiniciar la comunicación con el modem a través de Hyperterminal estableciendo en éste la nueva velocidad.

Es importante recalcar que este punto no necesariamente debe ser el primero a seguir en el proceso de configuración, pero es muy importante para que exista comunicación entre el módem y el microcontrolador.

3.2.1.2. Acceso a la red del Proveedor: Configuración del Modem

Inicialmente debemos indicar que si bién existe muchos dispositivos seriales que puede soportar aplicaciones que trabajan en redes de empaquetamiento de datos como es el caso de GPRS, muchos de estos dispositivos no tienen la suficiente capacidad o nivel de procesamiento para soportar el stack IP necesario para que el dispositivo pueda comunicarse con la red, este es el caso de el microcontrolador que estamos utilizando. Es decir, el implementar el stack IP en el microcotrolador resultaría una tarea muy pesada para nuestro sistema.

Para solventar este problema, existen protocolos como el protocolo PAD (Packet Assembly and Disassembly) que provee una arquitectura tal que puede ser configurada para conectarse con dispositivos a través de una red IP sin la necesidad de implementar un stack IP.

Una vez configurado y conectado, el protocolo PAD acepta transmisión de datos sobre líneas seriales, encapsulando los datos para enviarlos sobre una red IP.

El modem se encarga de establecer la conexión a la red IP así como de encapsular los datos dentro de paquetes UDP o TCP. Así se elimina la necesidad de tener un stack IP corriendo sobre el microcontrolador.

Así, la red que vamos implementar quedará bajo el esquema mostrado en la figura:

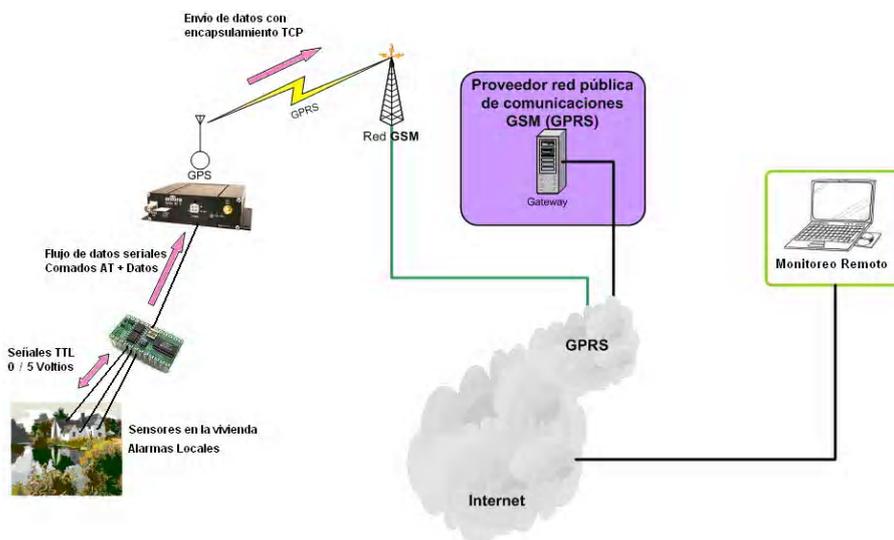


Figura 3.2. Esquema de conexión a la red GPRS

Bajo el esquema de comunicación aquí presentado, existen varios parámetros que se deben considerar para permitir que un modem tenga acceso a una red GPRS, entre los cuales tendremos:

Tamaño del buffer o tiempo de espera

Se puede configurar el tamaño de la cadena de datos que se va a enviar, así como el tiempo que el modem esperará a que se llene este buffer para enviar los datos, es decir el modem enviará los datos cuando:

- ♦ El tamaño de la información a enviarse llene el buffer definido.
- ♦ Cuando los datos no llenan el buffer y el tiempo de espera por datos para llenar el buffer se haya superado.

En este caso, debido a que se conoce el tamaño de la trama que se va a enviar, definiremos al tamaño del buffer en función de la trama, recordando la definición de la trama enviada (Capítulo II, punto 2.3.2.1) enviaremos una trama

compuesta por dieciocho caracteres, considerando que cada carácter se envía como código ASCII el cuál se compone de dos bytes, tendremos una trama de 36 bytes, por ello definiremos el tamaño del buffer en treinta y ser (36) bytes.

Para la definición de este parámetro utilizamos el comando AT\$PADBLK, este comando tiene tres formatos:

- ◆ AT\$PADBLK=? : Nos presenta el rango de valores que se puede asignar al buffer, medido en bytes, en el caso del modem Enfora este rango va de 3 a 512 bytes.
- ◆ AT\$PADBLK? Nos presenta el tamaño actual del buffer.
- ◆ AT\$PADBLK = <tamaño del buffer en bytes>: Nos permite establecer el tamaño del buffer.

Tipo de Sesión PAD

En este caso se establecerá una sesión TCP, para la configuración de este parámetro utilizamos el comando AT: AT\$HOSTIF, este comando presenta los siguientes formatos;

- ◆ AT\$HOSTIF=? : Nos presenta el rango de valores que se puede asignar a este parámetro, en este caso la respuesta sera:

\$HOSTIF: (0-3) siendo:

0: Se establece una conexión del modem a una red externa a través de dial up.

1: Se establece una sesión UDP.

2: Se establece una sesión TCP.

- ◆ AT\$HOSTIF?: Nos presenta el tipo de sesión configurado actualmente.
- ◆ AT\$HOSTIF = <valor>: Nos permite establecer el tipo de sesión deseado.

Comportamiento: Cliente o Servidor

Nos permite establecer si el modem trabajará como cliente o servidor, en el caso en que se comporte como servidor estará a espera de recibir un mensaje, mientras que cuando está en el formato cliente, el módem estará listo para enviar los datos que reciba por el puerto serial. Para establecer este parámetro utilizamos el comando AT: AT\$ACTIVE, mismo que tiene estas tres formas:

- ♦ AT\$ACTIVE=? : Nos presenta el rango de valores que se puede asignar a este parámetro, en este caso la respuesta será:

\$ACTIVE: (0-1) siendo:

0: Se establece un comportamiento de Servidor.

1: Se establece un comportamiento de Cliente.

- ♦ AT\$ACTIVE?: Nos presenta el tipo de comportamiento establecido actualmente.
- ♦ AT\$ACTIVE = <valor>: Nos permite establecer el tipo de comportamiento deseado.

APN a la que pertenece

Debido a que el servicio SIM contratado trabaja bajo la creación de una APN para nuestro sistema, es necesario indicar a cada uno de los miembros de esta red virtual el nombre de esta APN, esto lo logramos gracias al comando AT: AT+CGDCONT, el cual tiene los siguientes formatos:

- ♦ AT+CGDCONT =?: Nos presenta el rango de valores que se puede asignar a este parámetro, así como el formato para establecer los parámetros a través de este comando; en este caso la respuesta será:

+CGDCONT: (1-2),"IP",<APN>,(0,1),(0,1) dónde:

(1-2) es el identificador de Contexto PDP (PDP Context Identifier), es un parámetro numérico que especifica una definición particular de PDP, en nuestro caso se le asignará el valor de 1.

“IP”: es el tipo de PDP, en nuestro caso es IP

<APN>: es el nombre del Access Point Network, este es una cadena de texto en formato de dirección electrónica, tal como cliente.porta.com.ec

- ♦ AT+CGDCONT?: Nos presenta el APN actual.
- ♦ AT+CGDCONT = 1, "IP", <APNr>: Nos permite establecer el APN deseado.

Dirección IP y número del puerto al que se va a enviar los datos

Se trata de la IP pública con la cual nuestro servidor accede a Internet y el puerto sobre el cual se va a realizar la comunicación desde el equipo instalado en la vivienda hacia el programa que recibe y procesa los datos, mismo que se encuentra en nuestro servidor.

Este parámetro lo configuramos con el comando AT: AT\$PADDST, mismo que presenta, al igual que los casos anteriores, tres formatos:

- ♦ AT\$PADDST=?: Nos presenta el rango de valores que se puede asignar y el formato en que se los deberá ingresar, en este caso, como se trata de una dirección IP y un puerto, los cuatro primeros valores pertenecerán a la dirección IP y el último valor al puerto, por ello la respuesta a este comando será:

```
$PADDST: "(0-255),(0-255),(0-255),(0-255)",(0-65535)
```

- ♦ AT\$PADDST? Nos presenta la dirección IP y el puerto al que actualmente se encuentra apuntando el modem.
- ♦ AT\$PADDST = <"dirección IP">, <puerto>: Nos permite establecer la dirección IP y el puerto al que se van a enviar todos los datos recibidos por el modem.

Modo de autoregistro del modem

En este caso deseamos que al encenderse el modem se enlace a la red GPRS, esto lo logramos a través del comando AT: AT\$AREG, el cual tiene los siguientes formatos:

- ♦ AT\$AREG=? : Nos presenta el rango de valores que se puede asignar y el formato en que se los deberá ingresar, la respuesta a este comando será:

\$ AREG: (0-2) donde;

0: Autoregistro apagado

1: Autoregistro encendido

2: Activación automática de GPRS al encender el modem.

- ♦ AT\$AREG? Nos presenta la configuración actual de este parámetro.
- ♦ AT\$AREG = <valor>: Nos permite establecer el modo en que deseamos que este parámetro quede establecido.

Dirección IP y de Puerto de origen

Nos permite establecer la dirección IP y el puerto por el cuál el módem enviará y recibirá información, debido a que la asignación de IP la obtenemos de nuestro proveedor de servicio GPRS de manera dinámica, no establecemos ningún valor en este parámetro, así también, en el caso del puerto, no necesitamos asignar ningún valor pues no estamos recibiendo información en el modem, solo la enviamos.

Otros comandos

Adicionalmente existen otros comandos que, si bien, su utilidad no es el establecer o configurar un parámetro fijo, son necesarios para la configuración y trabajo con el modem, en nuestro caso estos comandos son:

- ♦ **AT:** este comando se lo envía para verificar la comunicación entre el módem y el equipo, en nuestro caso lo utilizamos al conectar el modem a la computadora pues esta es comunicación bidireccional y podemos verificar la respuesta. Si la comunicación esta establecida el módem resonde **OK**.
- ♦ **AT&F:** permite regresar al modem a su configuración original de fábrica, se elimina cualquier configuración realizada durante su uso.
- ♦ **AT&W:** graba en la memoria toda configuración realizada, este parámetro es muy importante pues si se realizan cambios en la configuración y no son grabados, al apagar el modem se perderán.
- ♦ **AT\$RESET:** permite reiniciar el modem,
- ♦ **AT\$NETIP?:** Verifica la activación de GPRS, si la respuesta del módem es diferente a cero significa que todo está trabajando normalmete.

3.3. PRESENTACIÓN DEL APLICATIVO WEB

Como se ha venido indicando, el objetivo al elaborar este aplicativo fue el crear una interfaz lo mas amigable posible para facilitar así el trabajo del usuario al realizar sus consultas a nuestro sistema. Este aplicativo no es mas que un juego de diferentes ventanas mismas que se van abriendo en secuencia según lo que el usuario desee, un diagrama de flujo en el que se presenta la forma en que estas ventanas se van mostrando se presenta en el siguiente gráfico:

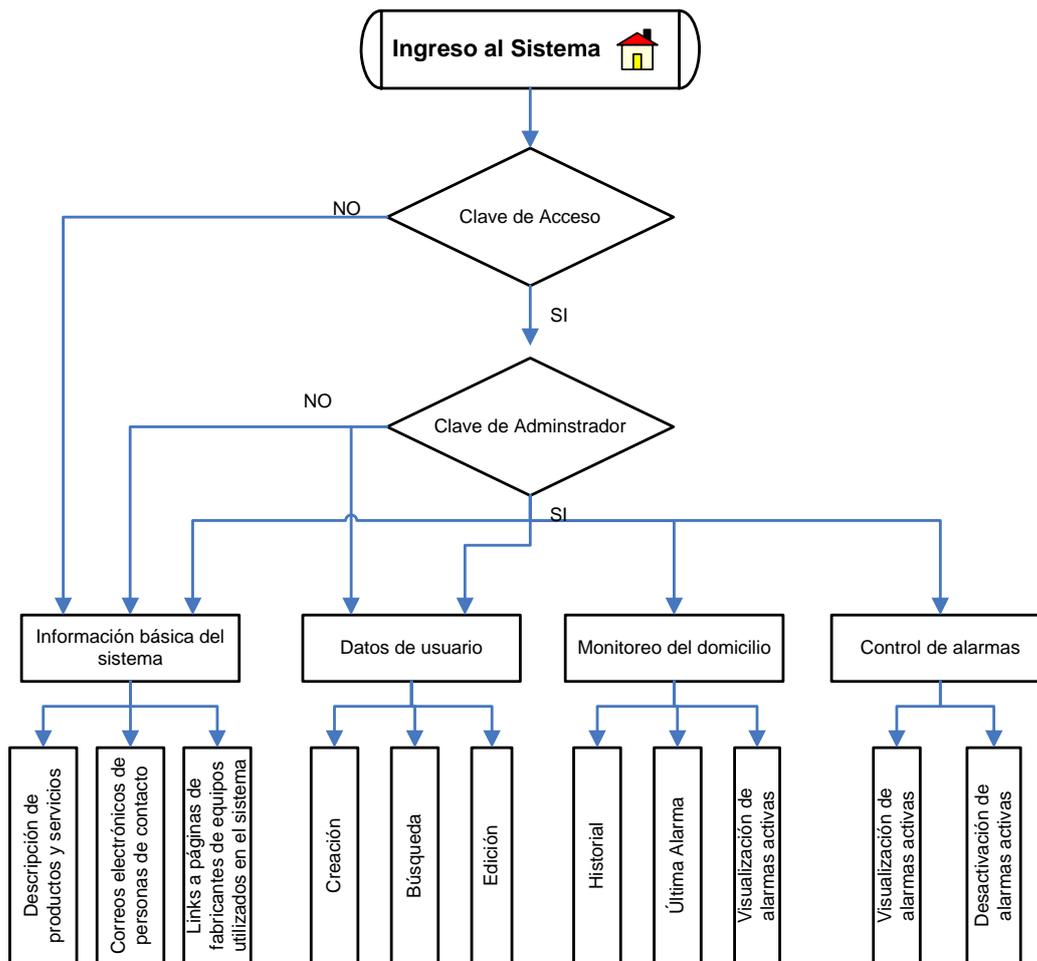


Figura 3.3. Diagrama de flujo dentro del aplicativo web

Inicialmente debemos indicar que cada usuario, ya sea este un cliente o un administrador tiene una clave y una contraseña mediante la cual se validará los permisos que éste tendrá al ingresar la sistema.

Creemos importante el recalcar que cada una de las ventanas presenta su respectiva explicación con la cual buscamos el facilitar el trabajo a usuarios nuevos. A continuación presentaremos un ejemplo de las ventanas que se presentan y su descripción de uso y objetivo.

3.3.1. VENTANA DE INICIO

En esta ventana tenemos la opción de ingresar una clave y una contraseña mediante la cual tendremos acceso al sistema de monitoreo, mas, para aquellos que no tengan esta clave, es decir, que no estén dentro de nuestro sistema esta ventana presenta varias opciones tales como:

- ♦ Productos: en donde se muestra una descripción del hardware que forma parte de nuestra solución.
- ♦ Servicios: Se describe el servicio en si con todos sus alcances y aplicaciones.
- ♦ Acerca de: Detalles propios de nuestro sistema, tal como fecha de creación, tiempo, motivo, entre otros.
- ♦ Contactos: direcciones de correo a las que se puede contactar cualquier persona que tenga dudas o sugerencias.
- ♦ Links: enlaces a datasheets, páginas web y demás, de los equipos que estamos utilizando.

Dependiendo de que clave se ingrese se tendrá acceso a las ventanas que se describen a continuación.

3.3.2. CREACIÓN EDICIÓN Y BÚSQUEDA DE USUARIOS

Estas son tres ventanas diferentes que tienen mucho en común, pues la plantilla es básicamente la misma, con diferencias lógicas según el objetivo de cada una de ellas, en general se presenta una tabla con los datos principales de cada uno de los usuarios del sistema siendo estos:

- ♦ Tipo de Usuario: pudiendo ser Usuario o Administrador
- ♦ Nombres del Usuario
- ♦ Apellidos del usuario

- ◆ Dirección del domicilio
- ◆ Sector
- ◆ Teléfonos fijos y móviles: siendo hasta un máximo de dos números telefónicos por cada caso.
- ◆ Dirección de correo electrónico del usuario
- ◆ Ciudad, Provincia y País en que se encuentra la vivienda
- ◆ Código del equipo del cliente – número serial
- ◆ Lugar de trabajo del Cliente
- ◆ Dirección y teléfono del Trabajo del Cliente
- ◆ Nombre de un familiar al que se puede comunicar en caso de una emergencia
- ◆ Teléfono fijo y móvil del familiar
- ◆ Descripción de la asignación de cada una de las seis zonas.

3.3.3. MONITOREO

En esta opción, cabe recalcar que es únicamente para administradores del sistema, se presenta un listado de todas las viviendas registradas con una descripción de cada una de las zonas indicando además cuál de estas zonas se encuentra en estado de alerta.

Es posible, además, el desactivar cualquier zona que se encuentre en alerta, misma que deberá ser desactivada siempre y cuando se verifique que el problema que generó la señal ya fue atendido.

Por otro lado, informa de la generación de nuevas señales de alerta presentadas en cualquier vivienda del sistema.

3.3.4. SERVICIO REMOTO

Permite a los clientes el tener acceso a un historial de sus viviendas, así como a conocer el estado actual de cada una de las zonas siendo esta opción dividida de la siguiente manera:

- ♦ **Historial de Alarmas:** Muestra un historial de todas las alarmas presentadas en un periodo de tiempo determinado, describiendo la fecha y hora de ocurrencia y en que zona se presentó, así también indica si esta alarma está registrada como pendiente o como ya tratada, presentado en este último caso en que fecha y hora fue desactivada desde el centro de monitoreo
- ♦ **Última Alarma:** muestra la última alarma presentada en la vivienda hincando la fecha y hora de ocurrencia así como si esta ya fue tratada o no.
- ♦ **Revisión en línea:** Muestra un listado de las zonas de la vivienda con una descripción de cada una de las zonas indicando si existe alguna zona con una señal de alarma no tratada, así también indica si se presentara alguna alerta en ese momento mediante el despliegue de una ventana de alerta.

3.4. CARACTERÍSTICAS DE LA BASE DE DATOS

La base de datos se compone de seis tablas en las cuales se almacenan los diversos datos que requerimos para el funcionamiento de nuestro sistema.

El almacenamiento de cada elemento nuevo se lo realiza en forma tal que los datos se guarden en la tabla respectiva y, debido a que en cada tabla se incluye un elemento de código, el encontrar los elementos buscados resulta sumamente fácil.

CAPÍTULO IV

PRUEBAS DE EVALUACIÓN DEL SISTEMA

Con la finalidad de certificar el funcionamiento de cada uno de los componentes del sistema, realizamos inicialmente pruebas individuales con cada bloque del sistema para luego realizar pruebas en conjunto; estas pruebas las describimos a continuación.

4.1. BLOQUE DE TOMA Y PROCESAMIENTO DE DATOS

Se empezó por pruebas individuales de cada uno de los sub bloques, estas pruebas fueron:

4.1.1. SUB BLOQUE DE MONITOREO Y TRANSFORMACIÓN

Básicamente fue una prueba del buen desempeño de los sensores, sin instalarlos se simulo la situación que se desea monitorear conectado la salida de dicho sensor a un multímetro, así se verificó los niveles de voltaje que cada sensor entrega.

Se verificó los niveles de voltaje que los sensores entregan quedando establecidos de la siguiente manera:

- ♦ Sensor de apertura cierre:
 - Evento monitoreado activo: 0 Voltios
 - Evento monitoreado desactivado: 5 Voltios
- ♦ Sensor de presencia:
 - Evento monitoreado activo: 0 Voltios
 - Evento monitoreado desactivado: 5 Voltios

4.1.2. SUB BLOQUE DE PROCESAMIENTO DE LA INFORMACIÓN

Para probar este bloque simplemente se simuló, en las entradas al microcontrolador destinadas para las señales recibidas por los sensores, los diversos niveles de voltaje que entregarán los sensores, y en las salidas del mismo se colocó diodos led de modo tal que este se activen o desactiven en función de las condiciones generadas en el programa grabado en el microcontrolador.

Se verificó el correcto funcionamiento del proceso de análisis de las señales y generación de las condiciones dentro del programa grabado en el microcontrolador.

4.1.3. SUB BLOQUE DE CONTROL DE EVENTOS

La prueba realizada en este bloque consistió en la conexión de las salidas del microcontrolador con todo el código de programación necesario- a los elementos utilizados para la ejecución de los eventos deseados: llamada telefónica, señal audible.

Debido a que durante la etapa de diseño se consideró los elementos necesarios de manera tal que no existieran problemas por los niveles de corriente consumidos, se verificó que no existen problemas con este bloque y que los elementos funcionan correctamente.

4.1.4. PRUEBAS DEL BLOQUE COMPLETO

Debido a que aún nos faltaba probar la comunicación con el modem. Las pruebas las realizamos instalando todo el sistema de alarma en una vivienda, mas la placa no la implementamos aún, y todos los elementos del circuito electrónico se encontraban montados sobre un proto board.

Se encontraron varios detalles en la instalación y montaje del sistema, siendo en muchos casos todo un trabajo manual como la forma y ubicación de sensores, mas en el mismo sistema eléctrico no tuvimos mayores inconvenientes, verificamos:

- ♦ La carga de la batería de respaldo, y el funcionamiento de todo el sistema conectado únicamente a ésta.
- ♦ Activación de las señales de alarma locales al presentarse un evento dentro de las zonas controladas.
- ♦ Desactivación del sistema a través del ingreso local de una clave.
- ♦ Activación y desactivación del sistema: selección de zonas a controlar.
- ♦ Caída de los niveles de voltaje: comprobándose que esta no supere 1 voltio en el caso de la fuente de 12 voltios y 0,5 voltios para la fuente de 5 voltios.

4.2. TRANSMISIÓN INALÁMBRICA DE INFORMACIÓN

Las pruebas con este bloque las realizamos en dos etapas:

- ♦ Conectado el modem a un computador y a través del programa Hyper Terminal probando comunicación y aceptación de comandos
- ♦ Conectando el modem al sistema para probar el sistema.

Debido a que el sistema consiste en la transmisión de datos a través de GPRS e Internet hacia un computador ubicado remotamente, fue necesario, para estas pruebas, el elaborar un pequeño programa en visual Basic que esté siempre monitoreando el puerto por el que enviaremos información y nos la presente en una pantalla de texto. Este programa fue la base para la implementación de nuestro Servicio Windows.

4.2.1. PRUEBAS MODEM – PC

Estas pruebas se las realizó con la finalidad de verificar que los comandos enviados desde el computador sean aceptados por el modem para así, al momento de conectar el modem a nuestra tarjeta podamos descartar la posibilidad de un comando incorrecto.

Empezamos por enviar comandos de consulta al modem para así verificar que exista comunicación y comprensión entre las partes, el proceso de pruebas para este bloque siguió el siguiente proceso:

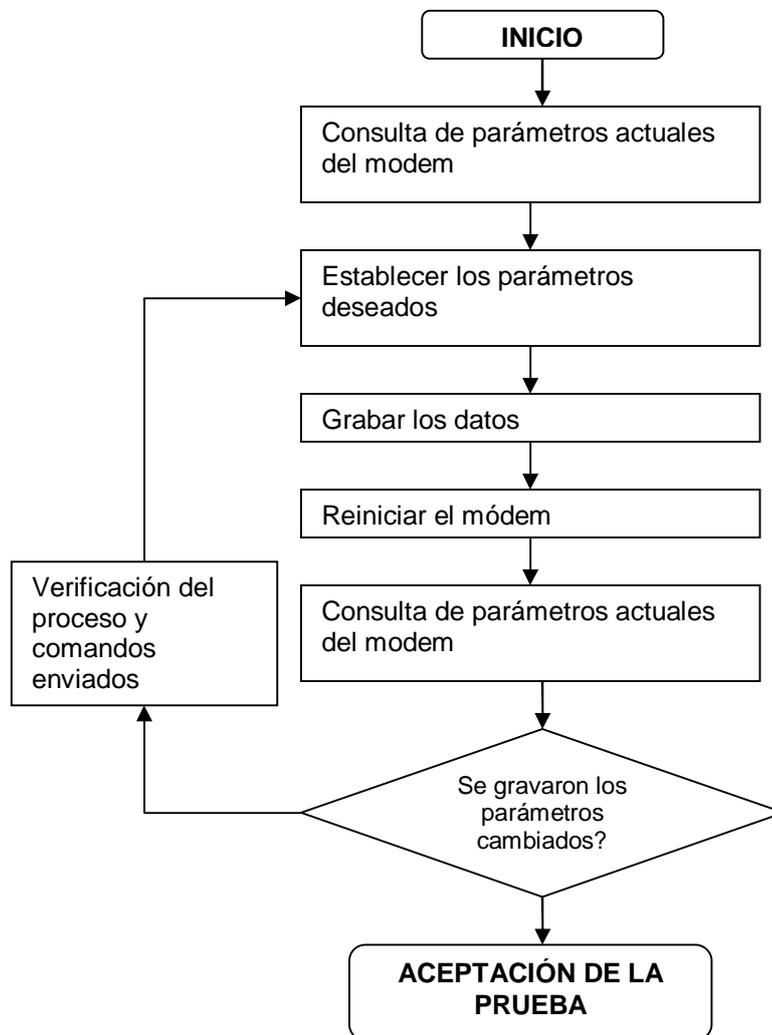


Figura 4.1. Proceso de pruebas para comunicación PC - modem

Para estas pruebas utilizamos básicamente los siguientes comandos:

- ◆ AT+IPR: para conocer y establecer la velocidad del puerto de comunicación serial
- ◆ AT\$PADBLK: nos permite conocer y establecer el tamaño del buffer
- ◆ AT\$HOSTIF: consultamos el tipo de sesión PAD para luego establecerla en TCP.
- ◆ AT\$ACTIVE: nos indica el comportamiento del módem y lo establecemos en tipo Cliente, es decir que esté listo para transmitir información
- ◆ AT+CGDCONT: para establecer el nombre de la APN con la que trabajamos, misma que nos fue asignada por nuestro proveedor del sistema GPRS
- ◆ AT\$PADDST: Para consultar la dirección IP y el puerto al que se va a enviar la información, para luego fijar los valores correspondientes a nuestro servidor.
- ◆ AT&W: para grabar en la memoria toda configuración realizada
- ◆ AT\$RESET: permite reiniciar el modem.
- ◆ AT\$NETIP?: Verifica la activación de GPRS

Una vez terminado este proceso de comandos procedimos al envío de datos y verificando la recepción de lo enviado en nuestro servidor, aquí pudimos verificar básicamente el trabajo del tamaño del buffer y que el módem una vez configurado funciona “por si solo”, pues si se apaga o por alguna otra razón pierde el enlace con el servidor, la sesión se reinicia automáticamente luego de que el problema que ocasionó la desconexión se supere.

Las pruebas realizadas fueron satisfactorias pues se verificó la correcta comunicación con el módem y el envío de datos. Fue muy importante para nosotros el verificar la auto conexión del modem al servidor pues esto nos representa una menor cantidad de líneas de programación en el programa que trabaja sobre el microcontrolador.

4.2.2. PRUEBAS MODEM – TARJETA ELECTRÓNICA

En este proceso de pruebas tuvimos muchos problemas al inicio, pues la comunicación entre el módem y la tarjeta electrónica no se la lograba y es aquí donde observamos las siguientes necesidades para nuestro sistema, mismas que nos obligaron a realizar cambios:

- ♦ Inicialmente la velocidad del puerto serial del módem estuvo establecida en 115200 bps, y del microcontrolador en 9600 bps, razón por la cual bajamos la velocidad de transmisión serial del modem.
- ♦ Observamos que al conectar el computador al modem el puerto tenía un nivel de voltaje de alrededor de 9 voltios mientras que al conectar a la tarjeta electrónica el nivel fue inferior a 7 voltios, y luego de un análisis completo del sistema observamos que el problema se debía a la fuente de alimentación, por ello nos vimos en la necesidad de cambiar nuestra fuente de alimentación original que se basaba en un conversor LM317 con una capacidad máxima de 1 amperio por la fuente final basada en el regulador LM350K con una capacidad de hasta 3 amperios.

Una vez cambiados estos dos parámetros logramos establecer la comunicación de nuestra tarjeta electrónica con el módem y así verificar el correcto funcionamiento de esta parte de nuestro sistema al recibir en nuestro servidor los códigos correspondientes a la activación de todas las posibles señales de alarma dentro de la vivienda.

El microcontrolador va a ser el encargado de monitorear los eventos y si es el caso transmitir los datos al modem, como la alarma va a realizar esto tiene que grabarse el código del cliente que será generado por el aplicativo Web que veremos más adelante, para que al transmitir los datos se reconozca de que cliente fue, el código que se pone para la transmisión de datos desde el microcontrolador es el siguiente:

```
SEROUT PORTA.5,T9600,["ARVPT"]  
IF SEN1=0 THEN  
    SEROUT PORTA.5,T9600,["Z1"]
```

```
ELSE
    SEROUT PORTA.5,T9600,["XX"]
ENDIF
IF SEN2=0 THEN
    SEROUT PORTA.5,T9600,["Z2"]
ELSE
    SEROUT PORTA.5,T9600,["XX"]
ENDIF
IF SEN3=0 THEN
    SEROUT PORTA.5,T9600,["Z3"]
ELSE
    SEROUT PORTA.5,T9600,["XX"]
ENDIF
IF SEN4=0 THEN
    SEROUT PORTA.5,T9600,["Z4"]
ELSE
    SEROUT PORTA.5,T9600,["XX"]
ENDIF
IF SEN5=0 THEN
    SEROUT PORTA.5,T9600,["Z5"]
ELSE
    SEROUT PORTA.5,T9600,["XX"]
ENDIF
IF SEN6=0 THEN
    SEROUT PORTA.5,T9600,["Z6"]
ELSE
    SEROUT PORTA.5,T9600,["XX"]
ENDIF
```

Las variables que están marcadas como (SEN1, SEN2, SEN3, SEN4, SEN5, SEN6) serán aquellas que van a ver la variación de voltaje en los sensores para saber que zona fue violada, entonces en el código que vemos anteriormente verificamos esto y según los que recibamos transmitimos error de zona o sin problemas.

4.3. INTERFAZ DE USUARIO Y BASE DE DATOS

Debido a que el interfaz de usuario trabaja siempre en función de los datos grabados en la base de dato fue necesario el realizar las pruebas de estos dos bloques en conjunto, y en muchos casos conectando ya el sistema completo.

Las pruebas realizadas consistieron en:

- ♦ Ingreso al sistema con perfil de usuario y administrador para verificar los privilegios y restricciones de estos.
- ♦ Creación visualización y modificación de usuarios
- ♦ Verificación de datos de una vivienda: nombres de las zonas, historial de alarmas.
- ♦ Verificación del acceso al sistema de manera remota a través de Internet.
- ♦ Visualización de mensajes de alarma al activarse una señal de alarma en una zona de la vivienda de un cliente.
- ♦ Desactivación en el sistema de alarmas presentadas.

4.3.1. DESARROLLO Y VISUALIZACIÓN DE LAS PÁGINAS WEB

Antes de iniciar con la explicación de la programación de las páginas vamos a decir que para el desarrollo del portal Web se utilizo lo que se llama programación en capas, dentro de esto existen tres capas de programación que son las siguientes:

- ♦ Capa de datos: Es aquella en donde van todas las conexiones a la base de datos, es la encargada de que el aplicativo Web interactúe con la base de datos que en nuestro caso fue desarrollada en SQL 2005, a continuación vemos el esquema de esta dentro de la programación en VB.NET:

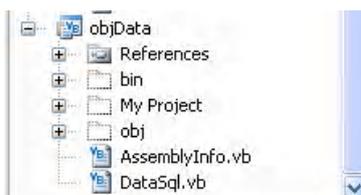


Figura 4.2. Desarrollo de las Capas de Datos

La Capa de datos en este caso tiene como nombre **objData** como vemos en la (Fig. 1) dentro de esta capa tenemos una carpeta con nombre References, en esta va la referencia o podríamos decir que es un link hacia las otras capas ya que todas estas deben estar interconectadas entre si porque ninguna de estas puede trabajar sola sino que trabajan juntas, en bin, MyProject y obj se encuentran los dll que el programa al ser compilado crea para interconectar a las mismas, ya que para que las páginas sean publicadas en la internet no se puede copiar todo el código directamente sino que estos dll son fruto de todo el código escrito y compilado, lo más importante de esta capa se encuentra en la clase con el nombre **DataSql.vb** ya que es dentro de esta que se escribe todo el código necesario para que esta capa realice su función, como dijimos la de conectarse a la base de datos, el código dentro de estas clase es la siguiente:

```
Imports System.Data.SqlClient
Imports System.Configuration
Public Class DataSql
    Public Shared con As New SqlConnection

    Public Sub New()
    End Sub
    Public Shared Function AbreConexion(ByVal strCon As String) As Boolean
        Dim valida As Boolean = False

        Try
            con.ConnectionString = strCon
            con.Open()
            valida = True
        Catch ex As Exception

            valida = False
        End Try
        Return valida
    End Function
    Public Shared Function CierraConexion() As Boolean
        Try
            If con.State = ConnectionState.Open Then
                con.Close()
            End If
        Catch ex As Exception

        End Try
    End Function
    Public Shared Function TraeReader(ByVal sql As String) As
    IDataReader
        Dim cmd As New SqlCommand
        Dim dr As SqlDataReader
        cmd.Connection = con
        cmd.CommandText = sql
        dr = cmd.ExecuteReader()
```

```
Return dr
End Function

Public Shared Function TraeTabla(ByVal sql As String) As DataTable
    Dim dt As New DataTable
    Dim cmd As New SqlCommand(sql, con)
    Dim da As New SqlDataAdapter
    Dim ds As New DataSet

    da.SelectCommand = cmd
    da.Fill(dt)
    Return dt
End Function

Public Shared Function EjecutaSQL(ByVal sql As String) As Boolean
    Dim filas As Integer
    Dim valido As Boolean = False
    Try
        Dim cmd As New SqlCommand
        cmd.Connection = con
        cmd.CommandText = sql
        filas = cmd.ExecuteNonQuery()
        valido = True
    Catch ex As Exception
        valido = False
    End Try
    Return valido
End Function

Public Shared Function EjecutaEscalar(ByVal sql As String) As
Decimal
    Dim cmd As New SqlCommand
    Dim valor As Decimal

    cmd.Connection = con
    cmd.CommandText = sql
    Try
        valor = cmd.ExecuteScalar
    Catch ex As Exception
        valor = 0
    End Try
    Return valor
End Function

Public Shared Function EjecutaEscalarProcedure(ByVal nombreSP As
String, Optional ByVal param() As SqlParameter = Nothing) As Integer
    Dim cmd As New SqlCommand
    Dim valor As Integer
    Dim i As Integer
    cmd.Connection = con
    cmd.CommandType = CommandType.StoredProcedure
    cmd.CommandText = nombreSP

    If param Is Nothing Then
        valor = cmd.ExecuteScalar
    Else
        For i = LBound(param) To UBound(param)
            cmd.Parameters.Add(param(i))
        Next
    End If
    Return cmd.ExecuteNonQuery()
End Function
```

```
        Next
        valor = cmd.ExecuteNonQuery
    End If
    Return valor

End Function
Public Shared Sub EjecutaSP(ByVal nombre As String, Optional ByVal
param() As SqlParameter = Nothing)
    Dim cmd As New SqlCommand
    Dim i As Integer
    Dim filas As Integer

    cmd.Connection = con
    cmd.CommandType = CommandType.StoredProcedure
    cmd.CommandText = nombre

    If param Is Nothing Then

        filas = cmd.ExecuteNonQuery()
    Else
        For i = LBound(param) To UBound(param)
            cmd.Parameters.Add(param(i))
        Next
        filas = cmd.ExecuteNonQuery
    End If
End Sub
End Class
```

Todos estos comando que utilizamos se encuentran dentro de la clase System.Data.SqlClient que se debe importar a nuestra clase ya que si no hacemos esto vamos a tener errores al escribir el código, como podemos ver el código es código de programación orientada a objetos.

- ♦ Capa de negocios: Es aquella en donde van los que podríamos decir “reflejos” de las tablas de la base de datos, el número de clases dentro de esta capa que tiene el nombre de **objAlarma** es el mismo que las tablas dentro de la base de datos, a continuación se muestra las tablas que existen dentro de la base de datos y la capa de negocios dentro del VB.NET:



Figura 4.3. Tablas dentro de la Base de Datos Alarma

Como podemos ver existen seis clases con los mismos nombres de las tablas, así como una clase con el nombre **Aplicación.vb** que es la encargada de referenciarse a la capa de datos y su clase.

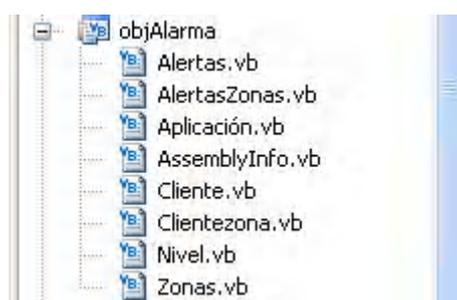


Figura 4.4. Desarrollo de la Capa de Datos

Con fines explicativos se va a presentar una clase con su código interno, el resto de clases tienen el mismo formato, lo único que cambia son los nombres de los campos dentro de cada tabla, vamos a ver la clase con el nombre **Cliente.vb** esta es la clase que hace referencia a la tabla Cliente (Fig. 4) dentro de la base de datos:

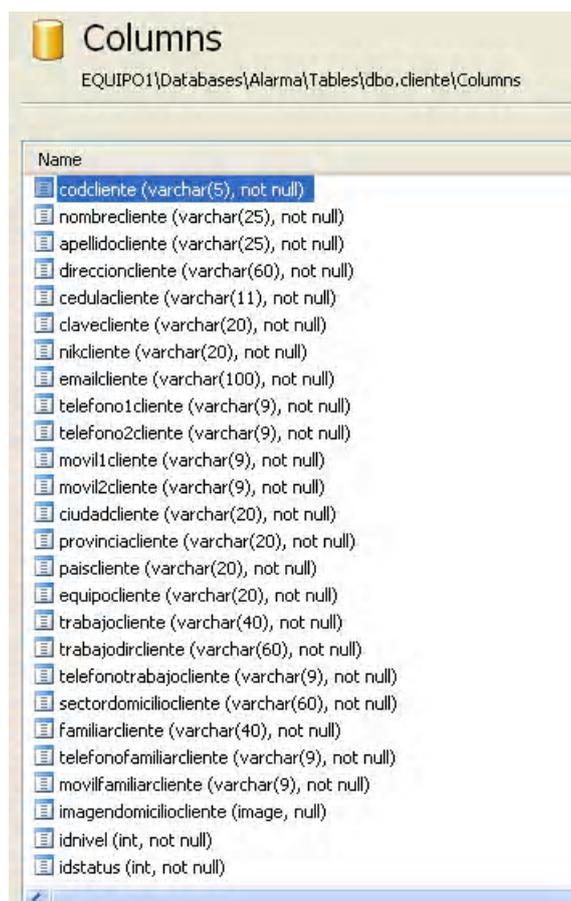


Figura 4.5. Campos de la capa Cliente dentro de la Base de Datos

Como vimos en la figura anterior todos estos son los campos dentro de la tabla Cliente de la base de datos, cada campo tiene un tipo de dato diferente, por ejemplo, el campo `codcliente` tiene como dato el tipo `varchar`, que es un tipo de datos string (cadena de caracteres), como podemos ver hay otro tipo de datos que es `int` (entero), cada uno de estos campos debe ser declarado dentro de la clase **Alertas.vb** con su respectivo tipo de dato como se dijo los `varchar` como (string) y los `int` como (integer), dentro de esta clase también va el código SQL que se utilizará para guardar, actualizar, leer y borrar campos de la tabla Cliente, a continuación vemos el código de esta clase:

```
Imports System.Text
Imports objData.DataSql
Public Class Alertas
#Region "VARIABLES"
    Private codalerta As Integer
    Private codcliente As String
    Private mensajealerta As String
```

```
Private estadoalerta As Integer
Private fechaalerta As String
Private horaalerta As String
#End Region
#Region "PROPIEDADES"
Public Property Codigo_alerta() As Integer
Get
    Return Me.codalerta
End Get
Set(ByVal Value As Integer)
    Me.codalerta = Value
End Set
End Property

Public Property Codigocliente() As String
Get
    Return Me.codcliente
End Get
Set(ByVal Value As String)
    Me.codcliente = Value
End Set
End Property

Public Property Mensaje() As String
Get
    Return Me.mensajealerta
End Get
Set(ByVal Value As String)
    Me.mensajealerta = Value
End Set
End Property

Public Property Estado() As Integer
Get
    Return Me.estadoalerta
End Get
Set(ByVal Value As Integer)
    Me.estadoalerta = Value
End Set
End Property

Public Property Fecha() As String
Get
    Return Me.fechaalerta
End Get
Set(ByVal Value As String)
    Me.fechaalerta = Value
End Set
End Property

Public Property Hora() As String
Get
    Return Me.horaalerta
End Get
Set(ByVal Value As String)
    Me.horaalerta = Value
End Set
End Property

#End Region
#Region "Metodos"
```

```

Public Sub Guardar()
    Dim sql As New StringBuilder
    If Me.codalerta = Nothing Then
        'Guardar
    Else
        'Actualizar
    End If

    EjecutaSQL(sql.ToString)
End Sub

#End Region
#Region "Funciones"
    Public Shared Function GetAlertas(ByVal codcliente As String) As
    ArrayList
        Dim sql As String
        Dim A As Alertas
        Dim li As New ArrayList
        Dim dt As DataTable

        sql = String.Format("Select *from Alertas where
codcliente='{0}'", codcliente)

        dt = TraeTabla(sql)

        For Each dr As DataRow In dt.Rows
            A = New Alertas
            A.codalerta = dr("codalerta")
            A.codcliente = dr("codcliente")
            A.estadoalerta = dr("estadoalerta")
            A.fechaalerta = dr("fechaalerta")
            A.mensajealerta = dr("mensajealerta")
            A.Hora = dr("horaalerta")
            li.Add(A)
        Next
        Return li
    End Function

    Public Shared Function GetAlertas(ByVal codcliente As String, ByVal
desde As Date, ByVal hasta As Date) As ArrayList

        Dim sql As String
        Dim A As Alertas
        Dim li As New ArrayList
        Dim dt As DataTable
        Dim d As String
        Dim h As String

        d = Format(desde, "yyyy/MM/dd")
        h = Format(hasta, "yyyy/MM/dd")

        sql = String.Format("Select *from Alertas where codcliente='{0}'
and fechaalerta>='{1}' and fechaalerta<='{2}'", codcliente, d, h)

        dt = TraeTabla(sql)

        For Each dr As DataRow In dt.Rows
            A = New Alertas
            A.codalerta = dr("codalerta")
            A.codcliente = dr("codcliente")

```

```

        A.estadoalerta = dr("estadoalerta")
        A.fechaalerta = dr("fechaalerta")
        A.mensajealerta = dr("mensajealerta")
        A.Hora = dr("horaalerta")
        li.Add(A)
    Next
    Return li
End Function

Public Shared Function GetAlertas(ByVal codcliente As String, ByVal
max As Integer) As ArrayList
    Dim sql As String
    Dim A As Alertas
    Dim li As New ArrayList
    Dim dt As DataTable

    sql = String.Format("select top {0} * from alertas where
codcliente='{1}' order by fechaalerta DESC", max, codcliente)

    dt = TraeTabla(sql)
    For Each dr As DataRow In dt.Rows
        A = New Alertas
        A.codalerta = dr("codalerta")
        A.codcliente = dr("codcliente")
        A.estadoalerta = dr("estadoalerta")
        A.fechaalerta = dr("fechaalerta")
        A.mensajealerta = dr("mensajealerta")
        A.Hora = dr("horaalerta")
        li.Add(A)
    Next
    Return li
End Function
#End Region
End Class

```

Como se ve declaramos cada uno de los campos como variables privadas por seguridad y performance de programación, luego de esto nos creamos funciones públicas para cada una de estas variables declaradas en función a los campos de la base de datos, ponemos un nombre a la función que represente a lo que va a representar por ejemplo la función con nombre Codigocliente que va a ser la que contendrá el momento de que se la utilice el dato código del cliente, luego declaramos las regiones cada una de las regiones por orden en la programación va a tener cierto código para las diferentes opciones, por ejemplo: la región "VARIABLES" contendrá la declaración de las variables que representarán a los campos de la base de datos, la región "PROPIEDADES" se va a definir todas las funciones que representarán a las variables definidas, la región "METODOS", van a estar todos los comandos SQL que tienen que ver con guardar, borrar o actualizar la base de datos, si nos damos cuenta todos estos comando van a

modificar la base de datos, y por último la región “FUNCIONES” contendrá todos los comandos SQL que tienen que ver con la lectura de los datos y en este caso esto no modificará la base de datos.

- ♦ Capa de presentación: Es aquella en donde van todas las páginas que el usuario que utilice el sistema va a verlas, este es el entorno gráfico que se publicará en la Internet la distribución de esta capa se la puede ver en la figura siguiente:

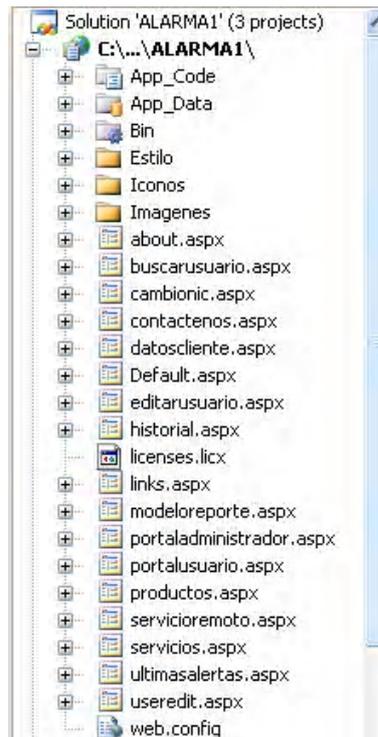


Figura 4.6. Desarrollo de la Capa de Presentación

En esta capa podemos ver todas las páginas Web, son las que llevan la extensión .aspx y el archivo web.config es aquel en donde se pone la conexión a la base de datos y este archivo es el que una vez publicado en la internet el aplicativo este archivo va a ser el que se lea para ejecutar las páginas.

Todas estas capas en conjunto forman en si el sistema Web a continuación vemos como están unidas en el programa (Fig. 6):

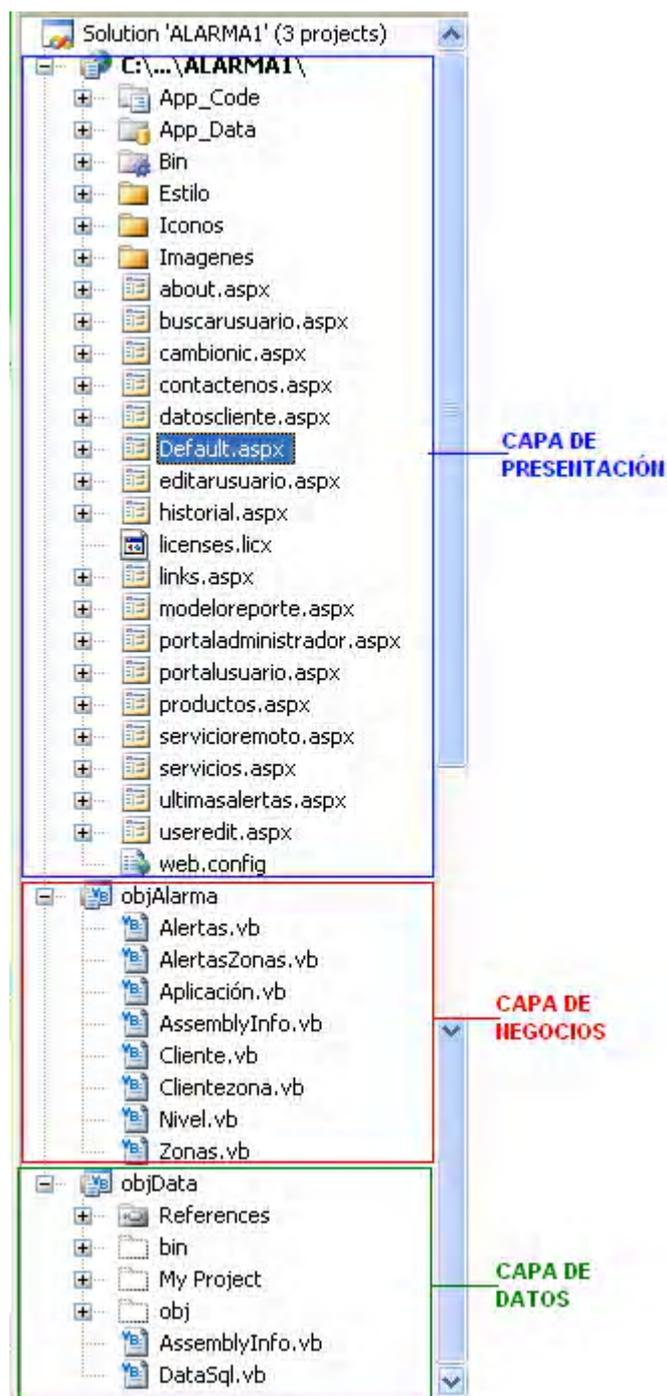


Figura 4.7. Modelo de las tres capas vistas en el entorno de programación

Como vimos el programa fue desarrollado en Visual Basic.NET, para el desarrollo de una página Web sabemos que se utilizaba códigos HTML, los cuales se les escribía dentro de un editor de texto como por ejemplo el Bloc de notas de Windows y una vez seguros de este código lo guardábamos con la extensión .html y luego todos lo que escribimos era representado en un entorno

que lo podíamos ver ya en la página Web, en la aplicación que utilizamos para el desarrollo de nuestras páginas ya no escribimos códigos HTML para programar sino código propio del visual Basic, y al compilar este código se genera automáticamente los códigos HTML, por este motivo la programación de páginas Web se vuelve más amistosa para el usuario y mucho más amplia en lo que respecta a aplicaciones que se pueden presentar al usuario, a continuación se va a presentar todo lo que respecta a la programación y presentación de las páginas:

La primera página que va a ser visualizada en el momento de ingresar al aplicativo Web va a ser la ventana de la (Fig. 8) en la cual vamos a ingresar el nombre de usuario y contraseña para según esto dependiendo del nivel del usuario se ingrese a las ventanas para administrador o para usuario.

Para el desarrollo de esta página tenemos que ir agregando a la ventana de programación controles y comandos que los encontramos en la barra de herramientas del VB.Net como se ve en la (Fig. 7)

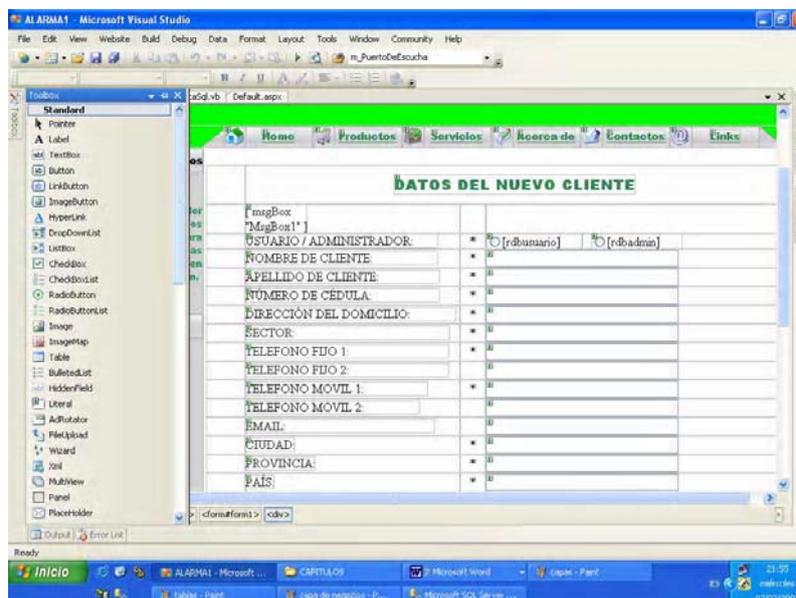


Figura 4.8. Ingreso de Controles a la Ventana de Programación

Así se va creando la presentación gráfica agregando tablas cuadros de texto, campos de imágenes, etc. una vez hecho esto y luego de haber dado un nombre a cada control procederemos a iniciar con la programación:

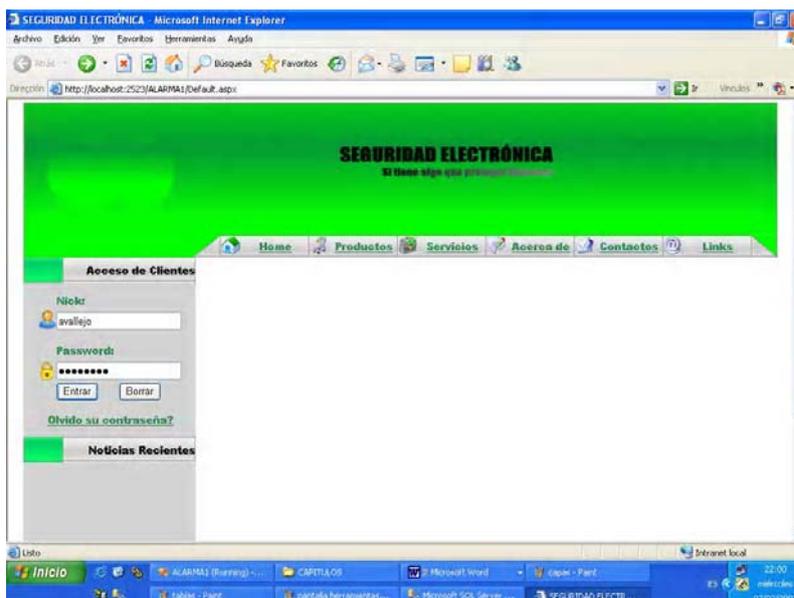


Figura 4.9. Ventana Principal del Aplicativo Web

A continuación vemos el código de Visual Basic que representa las acciones que esta ventana va a realizar:

```

Partial Class Default2
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.Load
        Dim str As String =
ConfigurationManager.ConnectionStrings("data").ToString
        If Me.IsPostBack = False Then
            objAlarma.Aplicación.OpenDB(str)
            Me.txtnic.Focus()
        End If
    End Sub

    Protected Sub cmdborrar_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles cmdborrar.Click
        If txtnic.Text = "" And txtclave.Text = "" Then
            MsgBox1.alert("Debe ingresar la clave o contraseña para
borrar")
        End If
        txtnic.Text = ""
        txtclave.Text = ""
    End Sub

    Protected Sub cmdlogin_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles cmdlogin.Click
        Dim c As objAlarma.Cliente
        'Caja de mensaje para indicar que no ha sido ingresado ni clave
ni contraseña
        If txtnic.Text = "" And txtclave.Text = "" Then
            MsgBox1.alert("Debe ingresar la calve y la contraseña")
        End If
    End Sub
End Class

```

```
End If

    c = objAlarma.Cliente.GetCliente(Me.txtnic.Text,
Me.txtclave.Text)

    Session.Add("Usuario", c)
    Select Case c.Nivel
        Case 1
            Me.Response.Redirect("portaladministrador.aspx")
        Case 2
            Me.Response.Redirect("portalusuario.aspx")
    End Select
End Sub

Protected Sub cmdhome_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles cmdhome.Click
    Response.Redirect("default.aspx")
End Sub

Protected Sub cmdproductos_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles cmdproductos.Click
    Response.Redirect("productos.aspx")
End Sub

Protected Sub cmdservicios_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles cmdservicios.Click
    Response.Redirect("servicios.aspx")
End Sub

Protected Sub cmdacercade_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles cmdacercade.Click
    Response.Redirect("about.aspx")
End Sub

Protected Sub cmdcontactos_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles cmdcontactos.Click
    Response.Redirect("contactenos.aspx")
End Sub

Protected Sub cmdlinks_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles cmdlinks.Click
    Response.Redirect("links.aspx")
End Sub
End Class
```

Como podemos ver aquí llamamos y utilizamos las clases de la capa de negocios la sentencia. **Dim c As objAlarma.Cliente** sirve para declara una variable de nombre c que va a referenciarse a la capa de negocios que vimos con nombre **objAlarma.vb** y el **.Cliente** quiere decir que va a ocupar de esta capa la clase Cliente, aquí vemos la programación orientada a objetos, luego de esto observamos la sentencia:


```

        <td background="Imágenes/nav_button.png"
width="113">
        <table align="right" border="0"
cellpadding="0" cellspacing="0" height="29" width="113">
        <tr>
        <td align="center" width="27">
        <asp:Image ID="Image8"
runat="server" ImageUrl="~/Iconos/home.gif" /></td>
        <td align="center" valign="middle">
        <asp:LinkButton ID="cmdhome"
runat="server" Font-Names="Arial Black" Font-Size="Smaller"
ForeColor="SeaGreen">Home</asp:LinkButton></td>
        </tr>
        </table>
        </td>
        <td background="Imágenes/nav_button.png"
width="113"><table align="right" border="0" cellpadding="0"
cellspacing="0" height="29" width="113">
        <tr>
        <td align="center" width="27">
        <asp:Image ID="Image1"
runat="server" ImageUrl="~/Iconos/productos.gif" /></td>
        <td align="center" valign="middle">
        <asp:LinkButton ID="cmdproductos"
runat="server" Font-Names="Arial Black" Font-Size="Smaller"
ForeColor="SeaGreen">Productos</asp:LinkButton></td>
        </tr>
        </table>
        </td>
        <td background="Imágenes/nav_button.png"
width="113">
        <table border="0" cellpadding="0" cellspacing="0" height="29"
width="113" align="right">
        <tr>
        <td width="27" align="center" style="height: 29px">
        <asp:Image ID="Image2" runat="server"
ImageUrl="~/Iconos/servicios.gif" /></td>
        <td align="center" style="height: 29px" valign="middle">
        <asp:LinkButton ID="cmdservicios" runat="server"
Font-Names="Arial Black" Font-Size="Smaller"
ForeColor="SeaGreen">Servicios</asp:LinkButton></td>
        </tr>
        </table>
        </td>
        <td background="Imágenes/nav_button.png"
width="113">
        <table id="TABLE1" border="0" cellpadding="0" cellspacing="0"
height="29" width="113" language="javascript" onclick="return
TABLE1_onclick()" align="right">
        <tr>
        <td width="27" align="center">
        <asp:Image ID="Image3" runat="server"
ImageUrl="~/Iconos/acerca_de.gif" /></td>
        <td align="center" valign="middle">
        <asp:LinkButton ID="cmdacercade" runat="server"
Font-Names="Arial Black" Font-Size="Smaller"
ForeColor="SeaGreen">Acerca
de</asp:LinkButton></td>

```

```

        </tr>
    </table>
        </td>
        <td background="Imágenes/nav_button.png"
width="113">
            <table align="right" border="0"
cellpadding="0" cellspacing="0" height="29" width="113">
                <tr>
                    <td width="27">
                        <asp:Image ID="Image4"
runat="server" ImageUrl="~/Iconos/contactos.gif" /></td>
                    <td align="center" valign="middle">
                        <asp:LinkButton
ID="cmdcontactos" runat="server" Font-Names="Arial Black" Font-
Size="Smaller"
ForeColor="SeaGreen">Contactos</asp:LinkButton></td>
                </tr>
            </table>
        </td>
        <td background="Imágenes/nav_button.png"
width="113">
            <table align="right" border="0"
cellpadding="0" cellspacing="0" height="29" width="113">
                <tr>
                    <td width="27">
                        <asp:Image ID="Image5"
runat="server" ImageUrl="~/Iconos/links.gif" /></td>
                    <td align="center" valign="middle">
                        <asp:LinkButton ID="cmdlinks"
runat="server" Font-Names="Arial Black" Font-Size="Smaller"
ForeColor="SeaGreen">Links</asp:LinkButton></td>
                </tr>
            </table>
        </td>
        <td width="33"
background="Imágenes/nav_right.gif">
        </td>
    </tr>
</table>
    </table>
        <table align="left" border="0" cellpadding="0"
cellspacing="0" height="33" width="220">
            <tr>
                <td background="Imágenes/arrow_left.png"
height="33" width="50">
                </td>
                <td background="Imágenes/bar_left.png"
height="33" style="width: 170px" align="right" valign="middle">
                    <span style="font-size: 10pt; font-
family: Arial Black">Acceso de Clientes</span></td>
            </tr>
            <tr>
                <td colspan="2" width="220"
style="background-color: lightgrey; height: 150px;" align="center">
                <table border="0" cellpadding="0" cellspacing="0" width="170"
height="25" style="background-color: lightgrey;">
                    <tr>

```

```

        <td align="left" colspan="1" style="width: 12px; height:
11px" valign="middle">
        </td>
        <td align="left" colspan="1" style="width: 12px; height:
11px" valign="middle">
        </td>
        <td align="left" colspan="2" style="height: 11px"
valign="middle">
        </td>
        <td align="left" colspan="1" style="height: 11px; width:
9px;" valign="middle">
        </td>
    </tr>
    <tr>
        <td align="left" colspan="1" style="width: 12px; height:
11px" valign="middle">
        </td>
        <td align="left" colspan="1" style="width: 12px; height:
11px" valign="middle">
        </td>
        <td style="height: 11px;" align="left" colspan="2"
valign="middle">
            <asp:Label ID="Label1" runat="server" Font-
Names="Arial Black" Font-Size="Smaller"
                ForeColor="SeaGreen"
Text="Nick:"></asp:Label></td>
        <td align="left" colspan="1" style="height: 11px; width:
9px;" valign="middle">
            <span style="color: darkgray"></span></td>
    </tr>
    <tr>
        <td align="right" colspan="1" height="30" style="width:
12px" valign="middle">
        </td>
        <td align="right" colspan="1" style="width: 12px;"
height="30" valign="middle">
            <a href="historial.aspx"></a>
            </td>
        <td align="left" colspan="2" height="30">
            <asp:TextBox ID="txtnic" runat="server" TabIndex="1"
Width="154px"></asp:TextBox></td>
        <td align="left" colspan="1" style="width: 9px;"
height="30">
        </td>
    </tr>
    <tr>
        <td align="left" colspan="1" style="width: 12px; height:
11px">
        </td>
        <td align="left" colspan="1" style="width: 12px; height:
11px; line-height: 3mm;">
        </td>
        <td align="left" colspan="2" style="height: 11px">
            <span style="color: darkgray; line-height:
3mm;"></span></td>
        <td align="left" colspan="1" style="width: 9px; height:
11px;">
        </td>

```

```

        </tr>
        <tr>
            <td align="left" colspan="1" style="width: 12px; height:
11px" valign="middle">
                </td>
            <td align="left" colspan="1" style="width: 12px; height:
11px" valign="middle">
                </td>
            <td style="height: 11px;" align="left" colspan="2"
valign="middle">
                <asp:Label ID="Label2" runat="server" Font-
Names="Arial Black" Font-Size="Smaller"
                    ForeColor="SeaGreen" Text="Password:"
Width="89px"></asp:Label></td>
            <td align="left" colspan="1" style="height: 11px; width:
9px;" valign="middle">
                </td>
        </tr>
        <tr>
            <td align="right" colspan="1" height="30" style="width:
12px" valign="middle">
                </td>
            <td align="right" colspan="1" style="width: 12px"
height="30" valign="middle">
                <asp:Image ID="Image7" runat="server"
ImageUrl="~/Iconos/contraseña.gif" /></td>
            <td align="left" colspan="2" height="30">
                <asp:TextBox ID="txtclave" runat="server"
TabIndex="2" Width="154px" TextMode="Password"></asp:TextBox></td>
            <td align="left" colspan="1" style="width: 9px"
height="30">
                </td>
        </tr>
        <tr>
            <td style="width: 12px; height: 11px">
                </td>
            <td style="width: 12px; height: 11px">
                </td>
            <td style="width: 1px; height: 11px;" align="left">
                <asp:Button ID="cmdlogin" runat="server"
Text="Entrar" TabIndex="3" Width="54px" /></td>
            <td style="width: 3px; height: 11px;" align="left">
                <asp:Button ID="cmborrar" runat="server"
Text="Borrar" TabIndex="4" Width="54px" /></td>
            <td align="left" style="width: 9px; height: 11px">
                </td>
        </tr>
        <tr>
            <td align="left" style="width: 12px; height: 35px">
                </td>
            <td align="center" style="height: 35px;" colspan="3"
valign="bottom">
                <span style="color:
darkgray"></span>&nbsp;<asp:LinkButton ID="cmdrecuperarpass"
                    runat="server" Font-Names="Arial Black" Font-
Size="Smaller" ForeColor="SeaGreen">Olvido su
contraseña?</asp:LinkButton></td>
            <td align="left" style="width: 9px; height: 35px;">
                </td>
        </tr>
        <tr>

```

```

        <td align="left" style="width: 12px; height: 11px">
        </td>
        <td align="center" colspan="3" style="height: 11px"
valign="bottom">
            <ccl:msgbox id="MsgBox1"
runat="server"></ccl:msgbox>
        </td>
        <td align="left" style="width: 9px; height: 11px">
        </td>
    </tr>
</table>

                </td>
                </tr><tr>
                <td height="33" width="50" style="background-image:
url(Imagenes/arrow_left.png)">
                </td>
                <td height="33" style="width: 170px;
background-image: url(Imagenes/bar_left.png);" align="right"
valign="middle">
                    <span style="font-size: 10pt; font-
family: Arial Black">Noticias Recientes</span></td>
                </tr><tr><td height="300" width="220" colspan="2" style="background-
color: lightgrey;" align="center" valign="middle">
                    </td></tr>
                </table>
                <br /><br /><br /><br />&nbsp;<br />
                <br /><br /><br /><br /><br />
                <br /><br /></td></tr>
    </table></div>
</form></body>
</html>

```

Una vez que el sistema haya reconocido si el código ingresado pertenece a un administrador de sistema o a un usuario nos presentara la ventana de administrador (Fig. 9) o de usuario (Fig. 10)

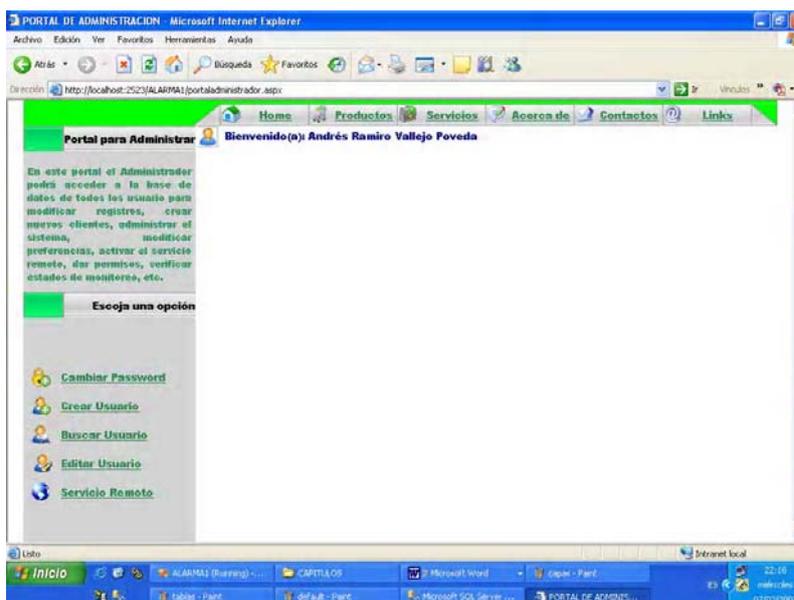


Figura 4.10. Ventana para Administradores

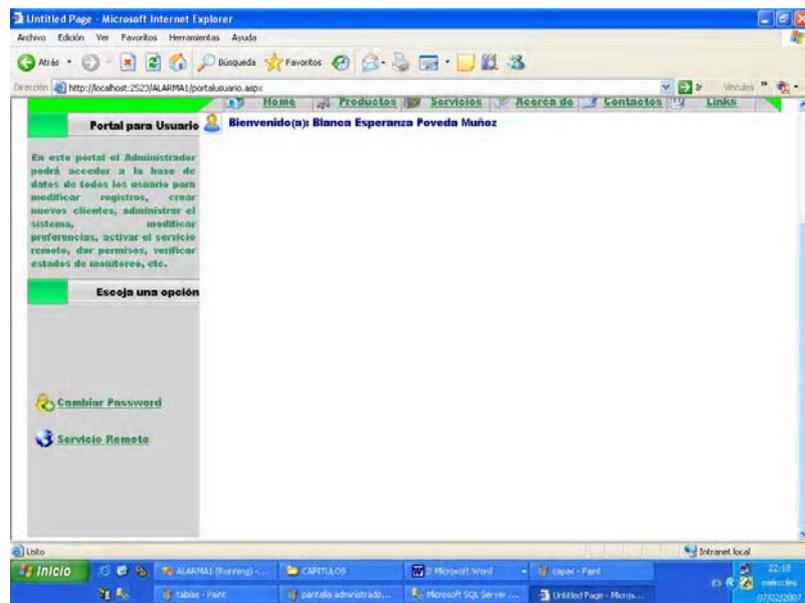


Figura 4.11. Ventana para Usuarios

Igual que en la ventana que se explico anteriormente a estas también se la agrega controles y código para volverlas funcionales en la figura 4.10. vemos que por se administrador tenemos más opciones como Crear nuevo usuario, buscar usuarios y modificarlos, si queremos crear un nuevo usuario tendremos la pantalla de la Figura 4.12, en la cual vamos a ingresar todos los datos que vamos a necesitar de el usuario y vamos a guardarlo, aquí para identificar al usuario trabajamos con la creación aleatoria de un código de cinco dígitos entre letras y número, el cual servirá para reconocer al usuario dentro del sistema, a parte de que a este código se lo va a grabar dentro del microcontrolador que es el encargado de procesar las señales dentro del sistema de monitoreo como tal (La Alarma).

INGRESO DE DATOS NUEVO CLIENTE - Microsoft Internet Explorer

En este portal el Administrador del sistema podrá crear nuevos Administradores e Usuarios para que puedan acceder a las diversas aplicaciones que en nuestro portal Web se presenten.

DATOS DEL NUEVO CLIENTE

USUARIO / ADMINISTRADOR: *

NOMBRE DE CLIENTE: *

APELLIDO DE CLIENTE: *

NÚMERO DE CÉDULA: *

DIRECCIÓN DEL DOMICILIO: *

SECTOR: *

TELÉFONO FJO 1: *

TELÉFONO FJO 2: *

TELÉFONO MOVIL 1: *

TELÉFONO MOVIL 2: *

EMAIL: *

CIUDAD: *

PROVINCIA: *

PAÍS: *

EQUIPO CLIENTE: *

LUGAR DE TRABAJO: *

DIRECCIÓN: *

TELÉFONO: *

NOMBRE FAMILIAR: *

TELÉFONO FJO: *

TELÉFONO MÓVIL: *

ZONA 1: *

Figura 4.12. Ventana para la creación de Nuevo Cliente

La ventana para buscar y modificar cliente es la misma con la diferencia de que en buscar solo vamos a ver los datos y en modificar los podemos cambiar, esta ventana se verá como la Figura 4.13:

Editar Usuario - Microsoft Internet Explorer

Bienvenido(a): Andrés Ramiro Valjejo Poveda

En este portal el Administrador podrá editar a cualquier usuarios ejecutando una búsqueda la cual se la realizará mediante el ingreso del número cédula para la búsqueda ya que esta no se repite es única para cada persona.

Elegir una opción

Regresar

NÚMERO DE CEDULA: Buscar

CODIGO DEL CLIENTE:

NOMBRE DE CLIENTE:

APELLIDO DE CLIENTE:

NÚMERO DE CÉDULA:

DIRECCIÓN DEL DOMICILIO:

SECTOR:

TELÉFONO FJO 1:

TELÉFONO FJO 2:

TELÉFONO MOVIL 1:

TELÉFONO MOVIL 2:

DIRECCIÓN DE CORREO ELECTRÓNICO:

CIUDAD:

PROVINCIA:

PAÍS:

EQUIPO CLIENTE:

LUGAR DE TRABAJO:

DIRECCIÓN:

Figura 4.13. Ventana para visualizar y actualizar Clientes

La funcionalidad principal de nuestro sistema sería la posibilidad de que la persona que sea participe del sistema pueda ver un registro del estado de su casa si ha tenido o no alertas que presentar para esto vamos a acceder a la opción que

dice **Servicio Remoto**, una vez que accedamos a esta vamos a ver la pantalla de la Figura siguiente

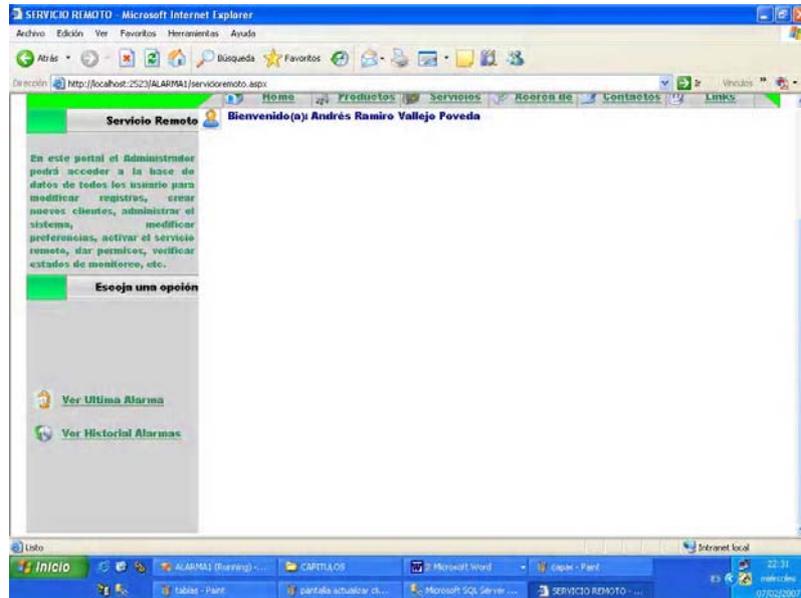


Figura 4.14. Ventana para menú de servicio remoto

Aquí podemos observar dos opciones, una que dice ver última alarma, si accedemos a esta opción veremos la pantalla (Fig. 14) o si escogemos la opción Historial Alarmas tendremos la pantalla (fig. 15):

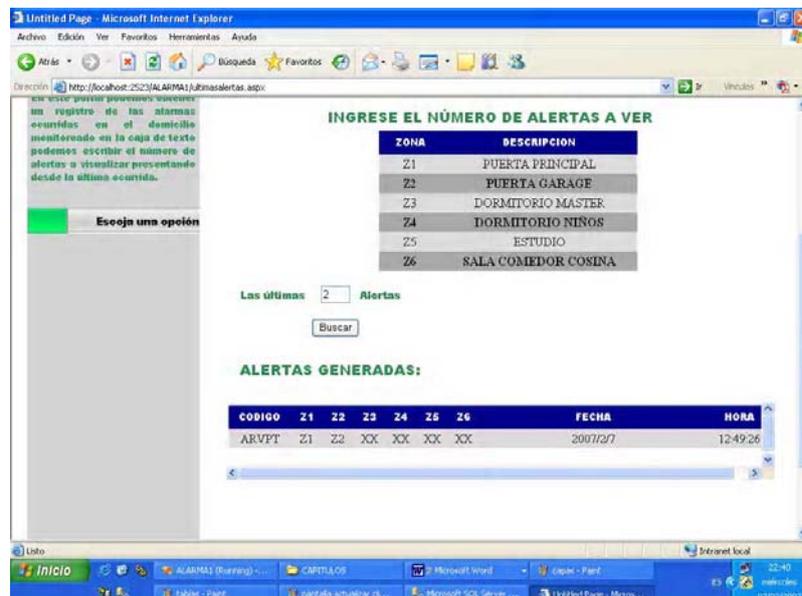


Figura 4.15. Ventana Ultima Alarma

Aquí en el campo de texto para el número de alertas ingresamos las alertas a partir de la última que queremos visualizar si ingresamos 1 se presentara la última si es 2 las dos últimas y así sucesivamente, vemos que se visualiza una tabla con la descripción de las zonas, y otra con las alertas, las zonas que tienen el identificativo de zona (Z1, Z2, Z3, Z4, Z5, Z6) quiere decir que han tenido problemas, y las que tiene XX quiere decir que no han tenido problemas.

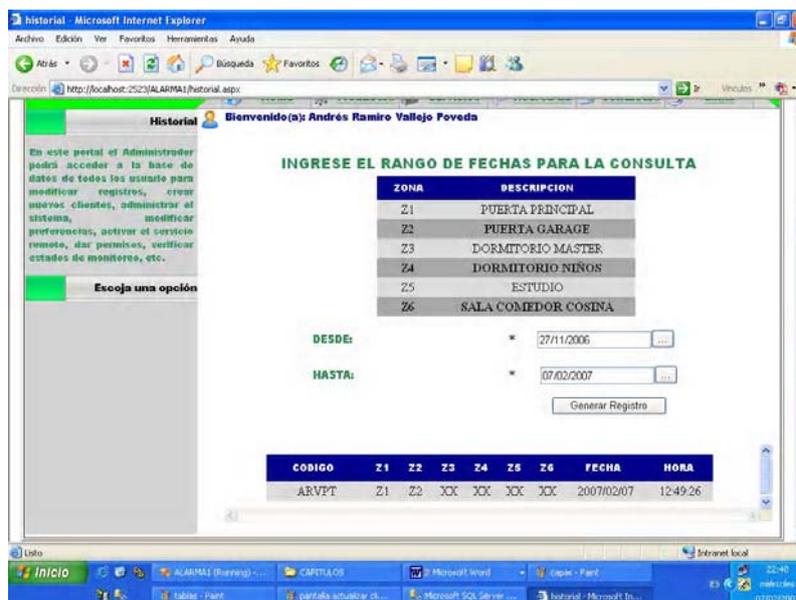


Figura 4.16. Ventana Ultima Alarma

En la ventana que acabamos de ver vamos a ingresar un límite de fechas dentro de las cuales queremos ver si ha existido o no alertas y se nos va a generar el registro junto con la descripción de las zonas.

CAPITULO V

ANÁLISIS ECONÓMICO

5.1. ANÁLISIS DE COSTOS

Para la implementación de nuestro sistema vamos a utilizar diversos materiales, y servicios, los costos que estos nos generan los dividiremos en dos tipos de acuerdo a la forma en que estos se reparten:

- ♦ Costos Centrales: Son valores únicos que se dividirán para el número de usuarios.
- ♦ Costos por Usuario: Son valores que se generan por cada uno de los usuarios que se tenga, es decir, si tenemos 10 usuarios el monto al que ascienda este costo será 10 veces menor que si tuviéramos 100 usuarios.

5.1.1. COSTOS CENTRALES

Los dividiremos en dos grupos:

- ♦ Gastos generados por los equipos y servicio para la base de datos y Centro de monitoreo: que son gasto que se generan una sola vez.
- ♦ Gastos generados por servicios contratados para centro de monitoreo, estos son gastos mensuales.

5.1.1.1. Equipos y servicios para base de datos y Centro de Monitoreo

CANTIDAD	DESCRIPCIÓN	VALOR UNITARIO	TOTAL
1	Computador – Servidor	\$ 1000	\$ 1000
1	Computador – Centro de Monitoreo	\$ 400	\$ 400
1	Instalación Acceso a Internet	\$ 300	\$ 300

TOTAL:	\$ 1700,00
--------	------------

Tabla 5.1. Costo de Equipos que forman parte del Centro de Monitoreo y Base de Datos

5.1.1.2. Servicios Requeridos

CANTIDAD	DESCRIPCIÓN	VALOR UNITARIO	TOTAL
1	Acceso a Internet – Banda Ancha e IP fija	\$ 200	\$ 200
1	Personal en centro de monitoreo (24 horas / 7 días)	\$ 600	\$ 600
1	Gastos varios: luz, agua, teléfono	\$ 50	\$ 50

TOTAL:	\$ 850
--------	--------

Tabla 5.2. Costo de Servicios consumidos para el Sistema

5.1.2. COSTOS POR USUARIO

5.1.2.1. Materiales de la Tarjeta electrónica

CANTIDAD	DESCRIPCIÓN	VALOR UNITARIO	TOTAL
15	Diodos 1N4007	\$ 0,10	\$ 1,50
1	Transformador	\$ 4,00	\$ 4,00
1	Buzzer	\$ 4,90	\$ 4,90
1	Puente de Diodos	\$ 1,00	\$ 1,00
1	Zócalo 16 pines	\$ 0,85	\$ 0,85
1	Zócalo 40 pines	\$ 1,20	\$ 1,20
1	Capacitor 2200 uF	\$ 0,70	\$ 0,70
2	Capacitor 0,1uF	\$ 0,06	\$ 0,12
2	LM350	\$ 4,80	\$ 9,60

3	Capacitor 1uF	\$ 0,08	\$ 0,24
1	Capacitor 4700 uF	\$ 2,40	\$ 2,40
1	Resistencia 220 Ohm 1W	\$ 0,15	\$ 0,15
2	2N3904	\$ 0,10	\$ 0,20
2	TIP110	\$ 0,70	\$ 1,40
4	Leds	\$ 0,10	\$ 0,40
25	Resistencias 1/4 W	\$ 0,02	\$ 0,50
8	Resistencias 1/2 W	\$ 0,04	\$ 0,32
1	PIC 16F877A	\$ 5,00	\$ 5,00
2	Capacitores 22pF	\$ 0,08	\$ 0,16
1	Conectores 3 entradas	\$ 0,60	\$ 0,60
9	Conectores 2 entradas	\$ 0,40	\$ 3,60
1	Disipador de calor	6,5	\$ 6,50
1	Ventilador	6,5	\$ 6,50
1	Batería de respaldo	18,5	\$ 18,50
2	Aislante para transistor	0,5	\$ 1,00
1	LCD	15,8	\$ 15,80
1	Elaboración de caja de protección y placa impresa	\$ 50	\$ 50,00

TOTAL:	\$ 137,14
--------	-----------

Tabla 5. 3. Costo de Materiales para elaboración de Tarjeta Electrónica

Cabe recalcar que los costos aquí presentados son valores de compra al por menor, más si la compra se realiza al por mayor, recibimos una reducción en un 30% de los costos, teniendo un valor final de noventa y seis dólares (USD\$ 96,00)

5.1.2.2. Materiales y equipos para enlace GPRS

CANTIDAD	DESCRIPCIÓN	VALOR UNITARIO	TOTAL
1	Convertor RS232 – TTL	\$ 4,0	\$ 4,00
1	Capacitor uF	\$ 0,10	\$ 0,50
1	Modem GPRS	\$ 200,00	\$ 200,00
1	SIM CARD	\$ 2,50	\$ 2,50

TOTAL:	\$ 207,00
--------	-----------

Tabla 5.4. Costo de Materiales y Equipos para Enlace GPRS

El módem aquí presentado es un equipo ya listo para trabajar, es decir para el usuario final, mas existen en el mercado otro tipo de equipos mucho mas sencillos, con lo cual podríamos hacer que estos costos desciendan a sesenta y siete dólares (USD \$67).

5.1.2.3. Consumo de GPRS

Para el cálculo de este valor debemos tener en cuenta que el proveedor de este servicio cobra dos valores:

Un valor fijo por el servicio: que es igual a cinco dólares (US\$ 5,00) mensuales por cada APN que se adquiere.

Un valor variable que depende de la cantidad de bytes que se hayan transmitido, para el cálculo de este valor tenemos como referencia la siguiente tabla de valores que nos fue entregada por el proveedor del servicio:

CONSUMO en MB		COSTO por MB
Desde	Hasta	
0	1024	\$ 1.20
1025	10240	\$ 1.10
10241	102400	\$ 1.00
1024001	En adelante	\$ 0.90

Tabla 5.5. Valor de servicio GPRS según la cantidad de MB consumidos

Si asumimos un promedio de 3 transmisiones diarias por cada vivienda, tendremos que al mes se realizarán 90 transmisiones, ahora, si por cada transmisión se envían en promedio 50 bytes al sumar cabeceras a la trama, tendremos que cada vivienda envía 4500 bytes por mes, redondeando este valor tendremos 5 Kb por vivienda por mes, como se ve es un valor sumamente mínimo.

El valor antes calculado se traduce en un valor mensual de seis milésimas de dólar (USD \$ 0.006), valor sumamente bajo, por ello asumimos un consumo mensual de un centavo de dólar.

Sumando el valor del consumo mensual al valor fijo a pagarse por el servicio tenemos que cada usuario nos representa un gasto de 5,01 dólares, si a este valor le añadimos los impuestos tenemos:

Valor de consumo mensual: 0,01 dólares

Valor a pagar por el servicio: 5,00 dólares

Impuestos (IVA 12%) 0,6012 dólares

5.1.2.4. Instalación y configuración del sistema en la vivienda

Considerando que para la instalación y configuración de los equipos se requiera una persona que trabaje un día completo tendremos:

CANTIDAD	DESCRIPCIÓN	VALOR UNITARIO	TOTAL
3	Hora de trabajo	\$ 15,00	\$ 45,00
1	Viáticos por transporte	\$ 10,00	\$ 10,00

TOTAL:	\$ 55,00
---------------	-----------------

Tabla 5.6. Costo de Servicios de Montaje y configuración en vivienda

5.1.3. RESUMEN DE VALORES

En total tendremos los siguientes valores:

GASTOS A REALIZARSE UNA SOLA VEZ

DESCRIPCIÓN	VALOR
Gastos centrales	\$ 1700
Gastos por usuario	\$ 218

Tabla 5.7. Resumen de Costos iniciales

GASTOS A REALIZARSE CADA MES

DESCRIPCIÓN	VALOR
Gastos centrales	\$ 855
Gastos por usuario	\$ 0,62

Tabla 5.8. Resumen de Costos Mensuales

En función de los datos antes indicados sabemos que existen dos tipos de costos:

- ♦ Costos iniciales: que se generan una sola vez
- ♦ Costos mensuales

Ahora bien, existen dentro de cada uno de los costos antes mencionados dos tipos de valores:

- ♦ Valores Fijos: que no dependen del número de usuarios
- ♦ Valores Variables: que dependen del número de usuarios

Así, en función de esto, y considerando que:

$n \rightarrow$ número de usuarios

$C_i \rightarrow$ Costos iniciales

$C_m \rightarrow$ Costos mensuales

Tenemos que

$$C_i = 218 + \frac{1700}{n} \quad (\text{Ec. 5.1})$$

Y

$$C_m = 0,62 + \frac{855}{n} \quad (\text{Ec. 5.2})$$

Un análisis de estas dos ecuaciones nos indica que el valor mínimo de estas cantidades será de 218 dólares para el costo inicial y 064 dólares para el costo mensual.

Entonces, debido a que se trata de un sistema de monitoreo constante, con una facturación mensual al cliente, consideramos que el valor inicial se lo puede diferir a m número de meses, así que por ello, considerando que:

$m \rightarrow$ número de meses

$P_{ccm} \rightarrow$ Valor que cada cliente deberá pagar cada mes

Tenemos

$$P_{ccm} = C_m + \frac{C_i}{m} \quad (\text{Ec. 5.3})$$

De dónde

$$P_{ccm} = 0,62 + \frac{855}{n} + \frac{218 + \frac{1700}{n}}{m} \quad (\text{Ec. 5.4})$$

Que es igual a

$$P_{ccm} = \frac{0,64nm + 855m + 218n + 1700}{nm} \quad (\text{Ec. 5.5})$$

Cabe resaltar que este valor indicado por P_{ccm} , es inicial ya que faltaría añadirle el porcentaje de ganancia.

Ahora el objetivo es encontrar un valor mensual a facturarse a cada usuario de manera que nos permita presentar este producto como algo sumamente atractivo y así podamos obtener la mayor cantidad de clientes.

Para esto realizamos un estudio de mercado en el cual investigamos:

1. Valor promedio que la mayoría de las personas estarían dispuestas a pagar cada mes por un servicio como el que les ofrecemos

2. Valor promedio mensual que se paga normalmente por un servicio de vigilancia privada
3. Tiempo promedio que las personas estarían dispuestas a pagar por este servicio, esto es, para diferir los costos iniciales.

Por medio de una consulta telefónica presentamos a las personas un pequeño set de preguntas mismo que detallamos en el Anexo V, y gracias a esta consulta obtuvimos los siguientes resultados:

DESCRIPCIÓN	CANTIDAD
Personas que conocen de empresas que brindan un servicio como el nuestro	20%
Personas que tienen un sistema de alarma en su casa	20%
Personas que de alguna forma pagan un sistema de guardianía	45%
Lapso promedio en el que estarían interesados en adquirir un sistema de este tipo	3 años
Valor mensual que cree costaría un servicio como el ofrecido	50 dólares

Tabla 5.9. Resultado de consulta efectuada al público

En función a estos resultados vemos que un tiempo prudencial es de 3 años, es decir 36 meses, con este valor analizaremos nuestra fórmula:

$$P_{ccm} = \frac{0,64nm + 855m + 218n + 1700}{nm} \quad (\text{Ec. 5.6})$$

Reemplazando el valor de m por 36 que equivale a tres años que el usuario contratará el servicio tendremos:

$$P_{ccm} = \frac{36 * 0,64n + 855 * 36 + 218n + 1700}{36n} \quad (\text{Ec. 5.7})$$

De dónde

$$P_{ccm} = \frac{241,04n + 32480}{36n} \quad (\text{Ec. 5.8})$$

$$P_{ccm} = 6,7 + \frac{902,22}{n} \quad (\text{Ec. 5.9})$$

Mediante un análisis de esta ecuación, vemos que el valor mínimo que el cliente pagará mensualmente es de 12,00 dólares, pero este valor es considerando una cantidad infinita de usuarios, así que analizando una tabla de valores elaborada en función a la ecuación antes presentada tenemos que una cantidad prudente de clientes estaría entre los 40 y 70 usuarios, y considerando que se fije el precio al público mensual en cincuenta dólares (USD\$ 45,00) tendremos una ganancia como la indicada en la tabla siguiente:

Usuarios	Costo	Ganancia		
		Porcentaje	Por Usuario	Mensual
10	98,29	-53,29	-118,43	-1184,27
20	52,18	-7,18	-15,96	-319,16
30	36,81	8,19	18,20	545,96
40	29,13	15,87	35,28	1411,07
50	24,51	20,49	45,52	2276,18
60	21,44	23,56	52,35	3141,29
70	19,24	25,76	57,23	4006,40
80	17,60	27,40	60,89	4871,51
90	16,32	28,68	63,74	5736,62
100	15,29	29,71	66,02	6601,73
110	14,45	30,55	67,88	7466,84
120	13,76	31,24	69,43	8331,96

} Valor aceptable

Tabla 5.10. Valor a ganar según el número de usuarios

5.2. PERSPECTIVA DE COMERCIALIZACIÓN

Como se mostró anteriormente, este sistema es enfocado al manejo de grandes volúmenes de clientes, pues es la única forma de conseguir costos reducidos que generen altas ganancias.

Basándonos en los resultados obtenidos de la encuesta telefónica cuyas preguntas se presentan en el Anexo V, vemos que un sistema como el nuestro resulta más atractivo para las personas que no viven en condominios ni conjuntos residenciales, mas, en estos casos, observamos que los administradores de

dichos conjuntos residenciales si estarían interesados en adquirir un sistema como este.

Por otro lado, observamos que resulta muy útil y atractivo para monitoreo de otro tipo de edificaciones, tales como fábricas, bodegas, pues le permite al dueño o administrador tener un control certero de dicha instalación.

CONCLUSIONES Y RECOMIENDACIONES

Una vez terminado nuestro por Proyecto de Tesis de Grado: “Diseño e implementación de un sistema de vigilancia remota para una residencia utilizando plataformas GPRS e Internet” hemos obtenido las siguientes conclusiones y recomendaciones

- Se realizó el diseño e implementación de un sistema de vigilancia remota para una residencia utilizando GPRS e Internet mismo que luego de ser probado cumplió con nuestros objetivos y nos permitió entender un poco mas sobre sistemas que actualmente están siendo muy utilizados, como son las plataformas GPRS e Internet.
- Se implementó una tarjeta electrónica de alarma utilizando un microcontrolador PIC y dispositivos sensores en la cual se aplicó todos los conocimientos adquiridos durante nuestros años de estudio, así como algoritmos en este período ya utilizados y probados
- Se logró acoplar las señales de la tarjeta al modem GPRS para el envío de las señales al centro de gestión del sistema de seguridad a través del envío de dichas señales por puerto serial.
- Se configuró una base de datos en un servidor para almacenar la información obtenida de la gestión de las viviendas gracias a la aplicación de SQL Server.
- Se desarrolló un software de gestión y administración del sistema utilizando la plataforma .Net obteniéndose un set de páginas web que conforman un entorno amigable y fácil de utilizar.
- Se realizó pruebas de funcionamiento, verificándose que todos nuestros requerimientos iniciales sean cubiertos.

- A través un análisis económico y de mercado realizado para entender que tan factible es la comercialización del sistema observamos que se trata de un proyecto innovador y que podría tener mucha apertura en el mercado.
- Gracias al estudio económico observamos que un sistema como el nuestro es mucho mas rentable mientras mayor sea la cantidad de abonados con que el sistema cuente.

REFERENCIAS BIBLIOGRÁFICAS

- www.newlc.com/How-to-retrieve-the-IMSI-number.html, How-to-retrieve-the-IMSI-number
- <http://www.pt.com/products/gsmintro.html>, Introducción a GSM
- <http://www.dlh.lahora.com.ec/paginas/chasqui/paginas/chasqui273.htm>, CYBERTERRORISMO - Historia de nunca acabar, José Camilo Daccach T
- <http://dev.mysql.com/doc/refman/5.0/es/>, Manual de referencia de MySQL 5.0
- http://www.aulaclic.es/sql/b_6_1_1.htm, Conceptos básicos de integridad referencial.
- <http://aspn.activestate.com/ASPN/Mail/Message/php-spanish/1789630>, Re: [PHP-ES] Media con mysql 5 dias , 2004 ActiveState, a division of Sophos
- http://www.suse.de/~aj/linux_lfs.html, File Support in Linux
- <http://www.mysql-hispano.org/page.php?id=31&pag=7>, Extrayendo información de una base de datos
- <http://www.mysql-hispano.org/page.php?id=33>, Conectando Visual Studio.NET con MySQL 5.0
- <http://www.mysql-hispano.org/page.php?id=43>, Comparativa MySQL vs Oracle
- http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art29.asp#to pic1, c_sharp.net
- <http://guille.costasol.net/NET/cursoVB.NET/tutorVBNET01.htm>, Curso de iniciación a la programación con Visual Basic .NET Guillermo Som “el Guille”
- <http://www.mycodebase.com/newmsg.php?ID=319&CL=08>, FORO PUBLICO DE MySQL
- <http://valgrind.org>, Valgrind Developers

- <http://es.tldp.org/Manuales-LuCAS/lfs-es/lfs-es-5.0/appendixa/automake.html>, Linux From Scratch: Versión 5.0 Apéndice A. Descripción de los paquetes y sus dependencias
- <http://www.gnu.org/software/autoconf/autoconf.html>, Free Software Foundation
- <http://www.desarrolloweb.com/articulos/956.php>, Common Lenguaje Runtime es el famoso -Escribe una vez, ejecuta donde quieras-. Descripción de cómo se trata en .NET.
- http://www.alvarezperea.com/alberto/descargas/tutorial_argosoft_mailserver_v2.pdf, Configuración Argosoft Mail Server Freeware
- <http://support.microsoft.com/default.aspx?scid=kb;es;298805>, CÓMO: Habilitar SSL para todos los clientes que interactúan con su sitio Web.
- http://www.devjoker.com/asp/ver_contenidos.aspx?co_contenido=25, Introducción a .NET.
- http://gbtcr.chileforge.cl/info_web/node112.html 12.3 Ejecutando Comandos AT
- http://www.wikilearning.com/como_usar_los_comandos_at-wkccp-3477-18.htm, Cómo Usar los Comandos AT en Curso de Qué son y como funcionan los modems

ANEXO I: DIAGRAMAS ELECTRÓNICOS

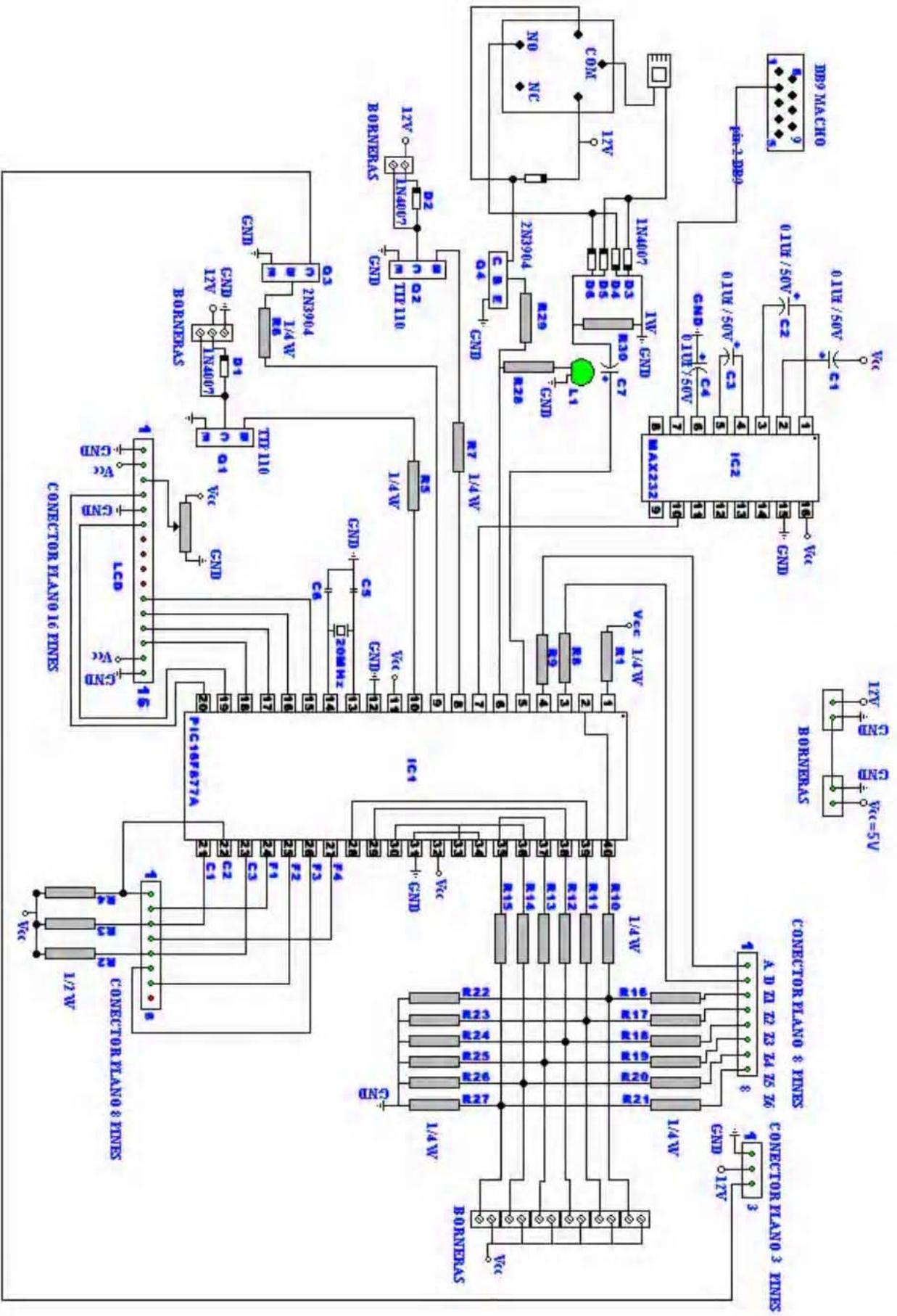


Diagrama electrónico - Placa Principal

Anexo

1/A

ESCALA

1:1

Proyecto
Sistema de Monitoreo Remoto

HOJA

1 DE 8

REV.

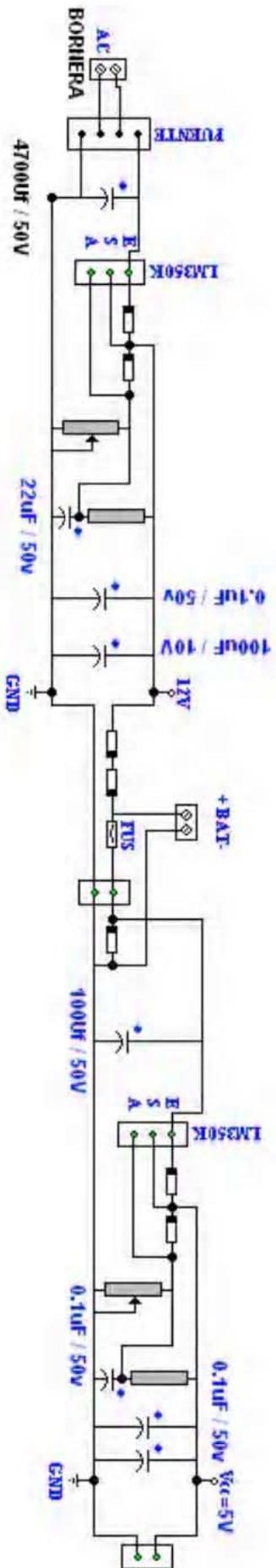


Diagrama electrónico - Placa de Alimentación de Voltaje

Anexo	IB
-------	----

ESCALA	1:1	Proyecto	Sistema de Monitoreo Remoto	REV.
		HOJA	1 DE 1	

ANEXO II: DIAGRAMAS DE FLUJO DEL SOFTWARE

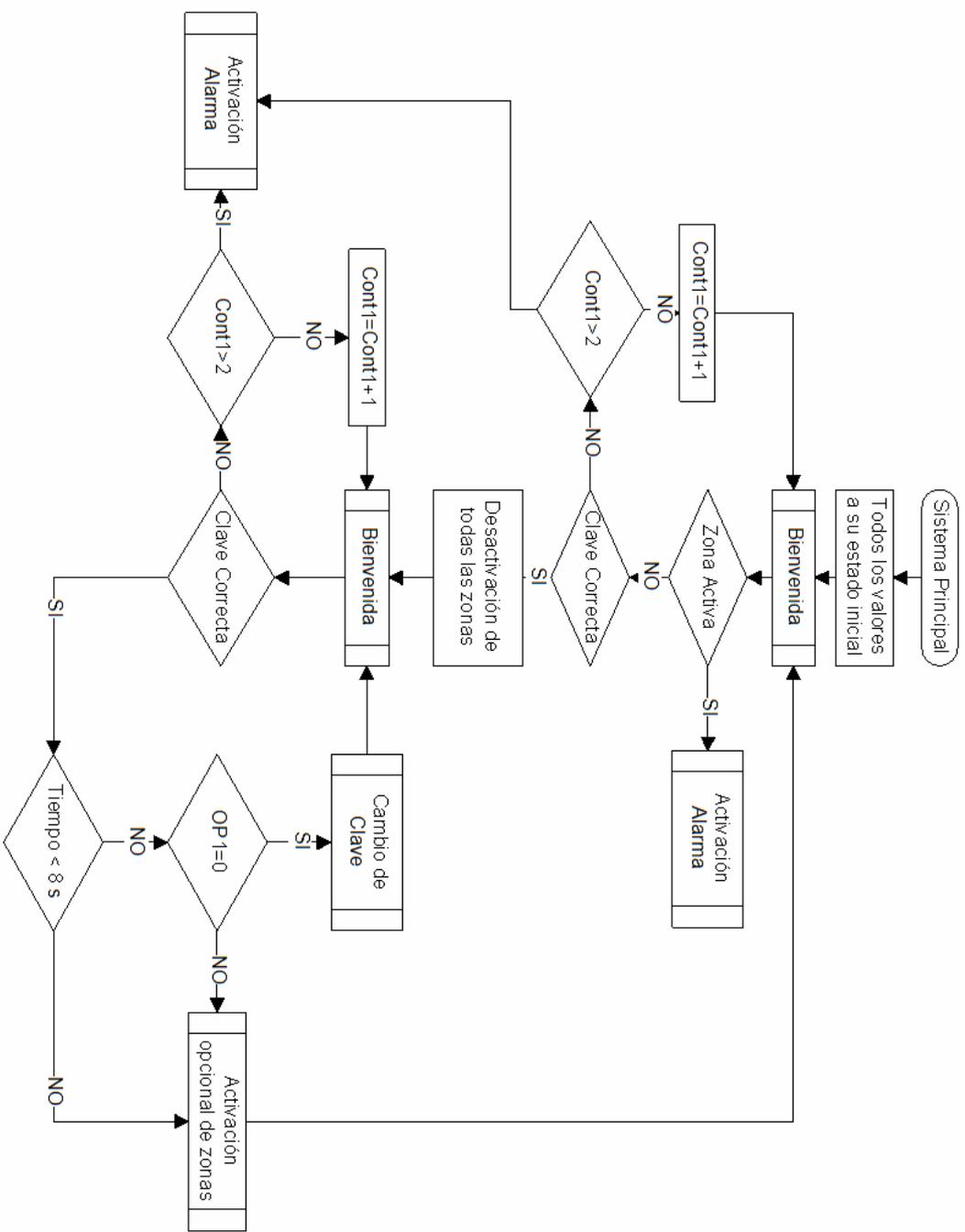


Diagrama de Flujo: Rutina Principal en el PIC

Anexo	II A
-------	------

ESCALA	1:1	Proyecto	Sistema de Monitoreo Remoto	REV.
		HOJA	1 DE 3	

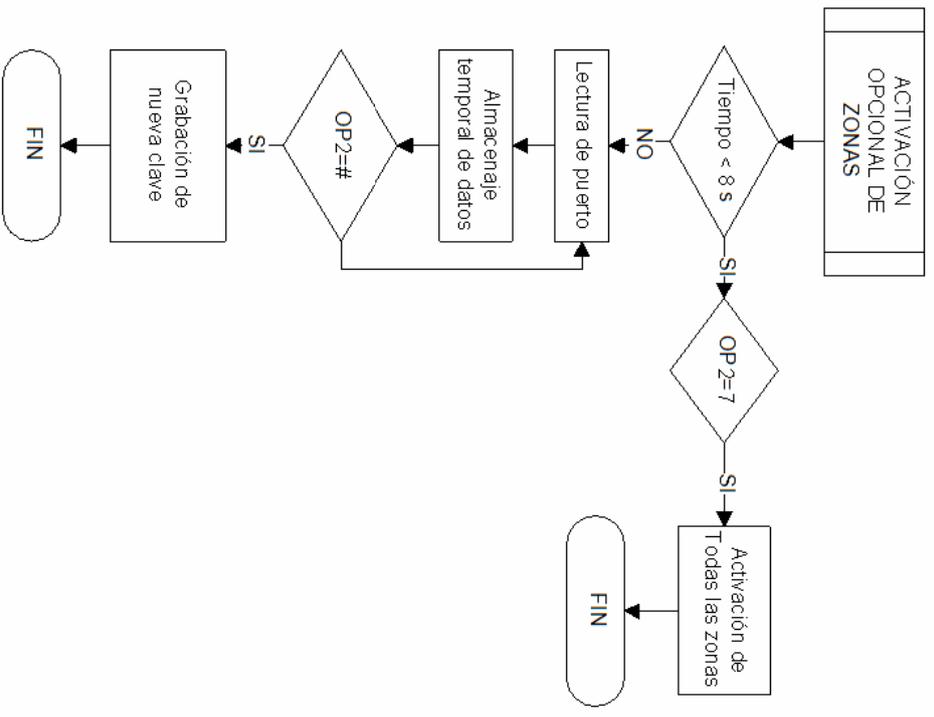
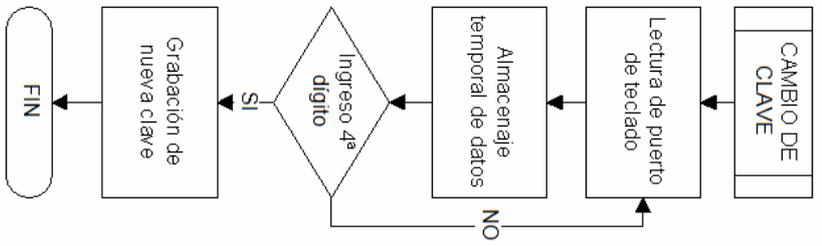
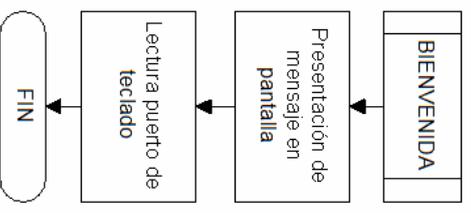


Diagrama de Flujo: Subrutinas en el PLC
Anexo II B

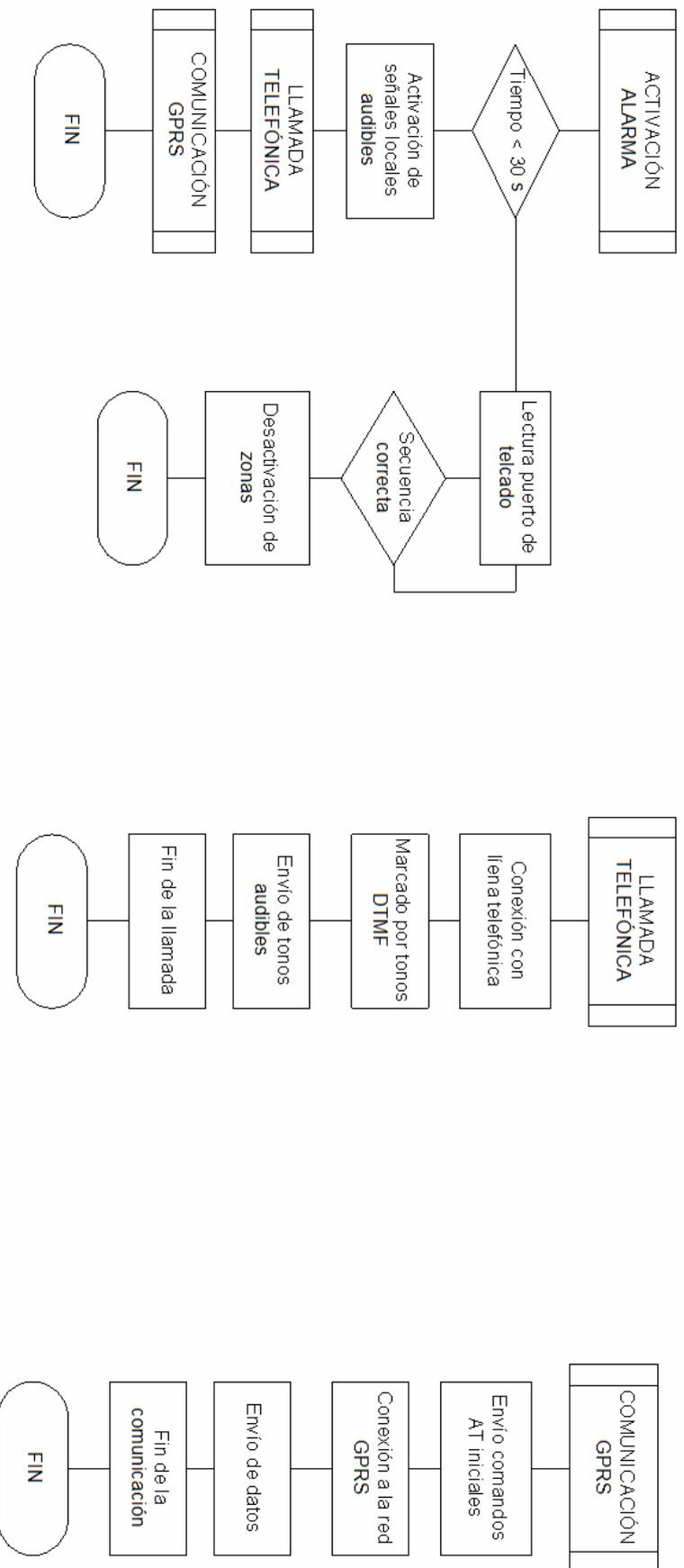


Diagrama de Flujo: Subrutinas en el

PIC

Anexo

II B

ESCALA

1:1

Proyecto
Sistema de Monitoreo Remoto

HOJA

3 DE 3

REV.

ANEXO III: DIAGRAMA DE LA BASE DE DATOS

Nivel
idnivel
desnivel

Alertas
codalerta
codcliente
estadoalerta
fechaalerta

Alertaszonas
codalertazona
codzona
codalerta

Zonas
codzona
descripcionzona

Clientezona
codcliente
codzona
descripcionzona
idclientezona

Cliente
Codcliente
Nombrecliente
direccioncliente
clavecliente
niccliente
telefono1cliente
telefono2cliente
movil1cliente
movil2cliente
ciudadcliente
provinciacliente
paiscliente
equipocliente
trabajocliente
trabajodircliente
sectordomiciliocliente
familiacliente
telefonofamiliarcliente
movilfamiliarcliente
apellidocliente
idnivel
emailcliente
idstatus
cedualcliente

Esquema de configuración de la Base de Datos	
Anexo	III

ESCALA	1:1	Proyecto	HOJA	1 DE 1	REV.
		Sistema de Monitoreo Remoto			

ANEXO IV: IMÁGENES DE SISTEMA IMPLEMENTADO

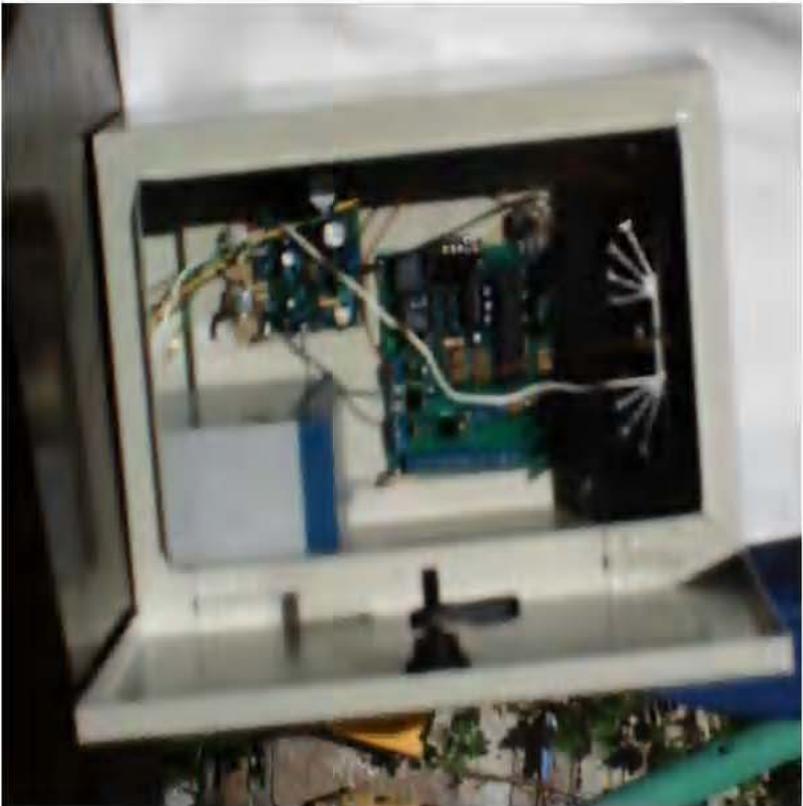


Imagen del sistema implementado	
Anexo	IV

ESCALA	1:1	Sistema de Monitoreo Remoto	PROYECTO	REV.
		HOJA	1 DE 1	



Imagen del sistema implementado	
Anexo	IV

ESCALA	1:1	Proyecto	1 DE 1	REV.
		Sistema de Monitoreo Remoto	HOJA	

**ANEXO V: PREGUNTAS PLANTEADAS EN ENCUESTA
TELEFÓNICA**

CUESTIONARIO PARA ENCUESTA TELEFÓNICA

Esta encuesta está preparada para efectuarse vía telefónica mediante una selección randómica de los encuestados según el directorio telefónico.

El tiempo aproximado de esta encuesta es de 10 minutos

1. Conoce usted de algún sistema de monitoreo domiciliario presente en el mercado?
2. Si su respuesta a la pregunta 1 es SI favor indique cual
3. Estaría usted interesado en adquirir un sistema de monitoreo remoto para su vivienda?
4. Que valor cree usted que es el adecuado para pagarse por un sistema de alarma domiciliaria que sea monitoreado remotamente y que le permita al usuario conocer el estado de su vivienda a través de Internet?

**ANEXO VI: CÓDIGO DEL PROGRAMA GRABADO E EL
MICROCONTROLADOR**

```

INCLUDE "modedefs.bas"
***** VISUALIZACION EN LCD*****
DEFINE    LCD_DREG  PORTC
DEFINE    LCD_DBIT  0
DEFINE    LCD_RSREG PORTD
DEFINE    LCD_RSBIT 1
DEFINE    LCD_EREG  PORTD
DEFINE    LCD_EBIT  0
'DefineOSC 20
*****CONFIGURANDO PUERTO A
NUMERO VAR BYTE 'VARIABLE PARA ALMACENAR NUMERO PULSADO
R VAR BYTE
Z VAR BYTE

AUX VAR BYTE 'VARIABLE PARA CONTROLAR ACTIVACIÓN O
DESACTIVACIÓN
AUX2 VAR BYTE
AUXD VAR BYTE
DES2 VAR BYTE
DES3 VAR BYTE
'VARIABLES PARA ACTIVACIÓN VARIADA DE ZONAS
A1 VAR BYTE
A2 VAR BYTE
A3 VAR BYTE
A4 VAR BYTE
A5 VAR BYTE
A6 VAR BYTE
'INICIALIZACIÓN DE VARIABLES
AUX = 0
AUX2 = 0
DES2 = 0
DES3 = 0
A1 = 0
A2 = 0
A3 = 0
A4 = 0
A5 = 0
A6 = 0

BIP VAR PORTE.1 'CONTOLA BIP CADA VEZ QUE SE PRESIONA UNA
TECLA
RFON VAR PORTA.4 'VARIABLE PARA CONECTAR EL RELE DE LINEA
TELEFÓNICA
ZAUX VAR PORTE.2 'SIRENA 1 DE ALARMA
ZAUX2 VAR PORTE.0 'SIRENA 2 DE ALARMA
'SENSORES PARA VALIDAR EVENTOS
SEN1 VAR PORTB.7
SEN2 VAR PORTB.6
SEN3 VAR PORTB.5
SEN4 VAR PORTB.4

```

```
SEN5 VAR PORTB.3
SEN6 VAR PORTB.2
'FILAS DEL TECLADO
F1 VAR PORTC.5
F2 VAR PORTC.6
F3 VAR PORTC.7
F4 VAR PORTD.4
'COLUMNAS DEL TECLADO
C1 VAR PORTD.2
C2 VAR PORTD.3
C3 VAR PORTC.4
'LEDS QUE INDICAN SI ESTA O NO ACTIVADA LA ALARMA
ACT VAR PORTA.2
DES VAR PORTA.1
'VARIABLES PARA MANEJAR ACTIVACIÓN ESCOJIDA DE ZONAS
D1 VAR PORTA.0
D2 VAR PORTD.5
D3 VAR PORTD.6
D4 VAR PORTD.7
D5 VAR PORTB.0 'CAMBIAR POR PORTC.4
D6 VAR PORTB.1
'VARIABLES PARA CLAVE
SETPRIME VAR BYTE
SETSEGUN VAR BYTE
SETERCER VAR BYTE
SETCUART VAR BYTE

'ESTADO INICIAL DE LA ALARMA
LOW RFON
'CLAVE INICIAL
EEPROM 0,[ 1,1,1,1 ] ;cargar la memoria EEPROM desde la dirección 0
```

```
'ESTADO INICIAL DE LA ALARMA DESACTIVADA
```

```
RESET:
```

```
    LOW ZAUX : LOW ZAUX2
    PAUSE 2000
    'LCDOUT $fe,1,"BIENVENIDO"
    'LCDOUT $fe,$c0,"CLAVE:"
    GOSUB SALUDO
    FOR R = 1 TO 3
        HIGH BIP
        PAUSE 50
        LOW BIP
        PAUSE 50
    NEXT
    GOSUB PTECLA
```

```
AUX = AUX + 1
```

```
IF AUX = 1 THEN
  LOW DES : HIGH ACT
  TRISD = %11111100
  TRISA = 1
  TRISB = %11111111
ENDIF
  PAUSE 1000
```

```
'LEE CLAVE DE ROM
READ 0,SETPRIME
READ 1,SETSEGUN
READ 2,SETERCER
READ 3,SETCUART
```

```
GOTO TECLAUNO
```

```
GRABAUNO:
```

```
  LCDOUT $fe, 1,"CAMBIO"
  LCDOUT $fe,$c0,"NUEVA CLAVE:"
```

```
  GOSUB PTECLA
  GOSUB BARRIDO : GOSUB PTECLA
  WRITE 0,NUMERO
  LCDOUT $fe,$cc,"*"
```

```
GRABADOS:
```

```
  GOSUB BARRIDO : GOSUB PTECLA
  WRITE 1,NUMERO
  LCDOUT $fe,$cd,"*"
```

```
GRABATRES:
```

```
  GOSUB BARRIDO : GOSUB PTECLA
  WRITE 2,NUMERO
  LCDOUT $fe,$ce,"*"
```

```
GRABACUATRO:
```

```
  GOSUB BARRIDO : GOSUB PTECLA
  WRITE 3,NUMERO
  LCDOUT $fe,$cf,"*"
  GOTO CAMBIO 'CAMBIO 18-09-2006
  'RETURN  'REGRESA LUEGO DE CAMBIAR CLAVE
```

```
BARRIDO:
```

```
  IF (AUX=1)OR(AUX=2) THEN
  IF (DES2=0) THEN
  GOSUB SENSOR
  ENDIF
  ENDIF
  IF (DES2=1) THEN
  AUXD = AUXD+1
  ENDIF
  IF AUXD>250 THEN
  HIGH ZAUX
  HIGH ZAUX2
```

```
AUXD=0
ENDIF
```

```
    LOW F1
    IF C1 = 0 THEN NUMERO =1 :RETURN
    IF C2 = 0 THEN NUMERO =2 :RETURN
    IF C3 = 0 THEN NUMERO =3 :RETURN
    HIGH F1
    LOW F2
    IF C1 = 0 THEN NUMERO =4 :RETURN
    IF C2 = 0 THEN NUMERO =5 :RETURN
    IF C3 = 0 THEN NUMERO =6 :RETURN
    HIGH F2
    LOW F3
    IF C1 = 0 THEN NUMERO =7 :RETURN
    IF C2 = 0 THEN NUMERO =8 :RETURN
    IF C3 = 0 THEN NUMERO =9 :RETURN
    HIGH F3
    LOW F4
    IF C1 = 0 THEN NUMERO =10 :RETURN
    IF C2 = 0 THEN NUMERO =0 :RETURN
    IF C3 = 0 THEN NUMERO =11 :RETURN
    HIGH F4
```

```
    PAUSE 40
SENSOR:
    IF
(SEN1=0)OR(SEN2=0)OR(SEN3=0)OR(SEN4=0)OR(SEN5=0)OR(SEN6=0)THEN
MAL:
```

```
    GOSUB ADVERTENCIA
    HIGH BIP
    GOSUB ESPERAE
    HIGH ZAUX
    HIGH ZAUX2
    'LCDOUT $fe, 1,"ASALTO"
    'GOSUB ESPERA
```

```
    HIGH F1: HIGH F2 :HIGH F3 :LOW F4
    IF (C3=0)AND(C1=0) THEN EMPEZAR 'CON # Y * PRESIONADOS A LA VEZ
```

```
    GOSUB TRANSMITIR
    GOSUB LLAMADA1
```

```
ASALTO:
    'LCDOUT $fe, 1,"ASALTO"
    GOSUB ADVERTENCIA
    """"""""""GOSUB TX
    WHILE AUX2 <= 600
    HIGH BIP
    PAUSE 500
```

```
LOW BIP
PAUSE 500
AUX2 = AUX2 + 1
IF AUX2 = 301 THEN
  LOW ZAUX2
ENDIF
WEND
```

```
IF AUX2 = 601 THEN
  LOW ZAUX
ENDIF
```

```
HIGH F1: HIGH F2 :HIGH F3 :LOW F4
IF (C3=0) AND (C1=0) THEN EMPEZAR ' CON # Y *
  GOTO ASALTO 'CICLO INFINITO SI NO SE PRESIONA COMBINACION
DE TECLAS
ELSE
  RETURN 'REGRESA A BARRIDO
ENDIF
```

LLAMADA1:

```
LCDOUT $fe, 1,"VOY LLAMAR"
PAUSE 7000
GOSUB ADVERTENCIA
'PRIMERA LLAMADA
PAUSE 300
HIGH RFON
PAUSE 2000
HIGH BIP
HIGH F1: HIGH F2 :HIGH F3 :LOW F4
IF (C3=0) THEN EMPEZAR
  DTMFOUT PORTA.3,[0,9,6,4,0,0,4,8,8]
  PAUSE 10
  LOW BIP
  PAUSE 6000 ;ESPERA HASTA QUE CONSTESTEN
```

```
FOR Z = 1 TO 40
  SOUND PORTA.3,[100,10,50,10]
NEXT
```

```
LOW RFON
'FIN PRIMERA LLAMADA
'PAUSE 500
RETURN
```

PTECLA:

```
HIGH BIP
PAUSE 100
LOW BIP
```

ESPACIO:

```
IF C1 = 0 THEN ESPACIO
IF C2 = 0 THEN ESPACIO
IF C3 = 0 THEN ESPACIO
PAUSE 25
RETURN
```

EQUIVOCADA:

```
'LOW ZAUX : LOW ZAUX2
PAUSE 2000
'LCDOUT $fe,1,"REPITA"
'LCDOUT $fe,$c0,"CLAVE:"
GOSUB SALUDO
FOR R = 1 TO 3
HIGH BIP
PAUSE 50
LOW BIP
PAUSE 50
NEXT
```

GOTO TECLAUNO

CAMBIO:

```
LOW ZAUX : LOW ZAUX2
PAUSE 2000
GOSUB SALUDO
FOR R = 1 TO 3
HIGH BIP
PAUSE 50
LOW BIP
PAUSE 50
NEXT
```

```
'LEE CLAVE DE ROM
READ 0,SETPRIME
READ 1,SETSEGUN
READ 2,SETERCER
READ 3,SETCUART
```

GOTO TECLAUNO

TECLAUNO:

```
GOSUB BARRIDO :GOSUB PTECLA
LCDOUT $fe,$c7,"*"
IF NUMERO = SETPRIME THEN TECLADOS
GOTO FALSO
```

TECLADOS:

```
GOSUB BARRIDO :GOSUB PTECLA
LCDOUT $fe,$c8,"*"

```

```
IF NUMERO = SETSEGUN THEN TECLATRES
GOTO FALSO1
```

TECLATRES:

```
GOSUB BARRIDO :GOSUB PTECLA
LCDOUT $fe,$c9,"*"
IF NUMERO = SETERCER THEN TECLACUATRO
GOTO FALSO2
```

TECLACUATRO:

```
GOSUB BARRIDO :GOSUB PTECLA
LCDOUT $fe,$ca,"*"
PAUSE 1000
IF NUMERO = SETCUART THEN OPENGE
GOTO FALSO3
```

OPENGE:

```
'DES2=1
DES2=0
AUX = AUX + 1
FOR R = 1 TO 2
PAUSE 100
HIGH BIP
PAUSE 100
LOW BIP
NEXT
```

```
IF AUX = 3 THEN
A2 = 0
LCDOUT $fe,1,"PRESIONAR *7*"
LCDOUT $fe,$c0,"ACTIVAR TODAS"
```

TODAS:

```
LOW F3
IF C1 = 0 THEN TODAS1 'SI PRESIONO 7
HIGH F1: HIGH F2 :HIGH F3 :LOW F4
IF C3 = 0 THEN SIGUIENTE 'A ESCOJER ZONAS INDIVIDUALMENTE
GOTO TODAS
```

TODAS1:

```
DES2 = 0
GOSUB PTECLA
LCDOUT $fe,1,"ZON: 1-2-3-4-5-6"
GOSUB ESPERA
HIGH F3
GOSUB LISTO
```

SIGUIENTE:

```
DES2 = 0
GOSUB PTECLA
```

```
LCDOUT $fe,1,"ACTIVAR:"
```

```
***BARRIDO PARA ACTIVAR
```

```
BARRIDO1:
```

```
  LOW F1
```

```
  IF C1 = 0 THEN
```

```
    GOSUB PTECLA
```

```
    A1 = 1
```

```
  ENDIF
```

```
  IF C2 = 0 THEN
```

```
    GOSUB PTECLA
```

```
    A2 = 1
```

```
  ENDIF
```

```
  IF C3 = 0 THEN
```

```
    GOSUB PTECLA
```

```
    A3 = 1
```

```
  ENDIF
```

```
  HIGH F1
```

```
  LOW F2
```

```
  IF C1 = 0 THEN
```

```
    GOSUB PTECLA
```

```
    A4 = 1
```

```
  ENDIF
```

```
  IF C2 = 0 THEN
```

```
    GOSUB PTECLA
```

```
    A5 = 1
```

```
  ENDIF
```

```
  PAUSE 10
```

```
  HIGH F1: HIGH F2 : LOW F4
```

```
'LUEGO DE HABER ESCOJIDO ZONAS
```

```
IF C3=0 THEN
```

```
  GOSUB PTECLA
```

```
  GOSUB ACTIVAR
```

```
ENDIF
```

```
  GOTO BARRIDO1
```

```
ACTIVAR:
```

```
  HIGH F4
```

```
'S6
```

```
IF(A1 = 0)AND(A2 = 0)AND(A3 = 0)AND(A4 = 0)AND(A5 = 0) THEN
```

```
  LCDOUT $fe,1,"ZON: 6"
```

```
  GOSUB ESPERA
```

```
  TRISD = %00011100
```

```
  TRISA = 0
```

```

HIGH D1 : HIGH D2 : HIGH D3 : HIGH D4 : HIGH D5 : LOW D6
ENDIF
'S1
IF(A1 = 1)AND(A2 = 0)AND(A3 = 0)AND(A4 = 0)AND(A5 = 0) THEN
LCDOUT $fe,1,"ZON: 1-6"
GOSUB ESPERA
HIGH D2 : HIGH D3 : HIGH D4 : HIGH D5

ENDIF
'S2
IF(A1 = 0)AND(A2 = 1)AND(A3 = 0)AND(A4 = 0)AND(A5 = 0) THEN
LCDOUT $fe,1,"ZON: 2-6"
GOSUB ESPERA
HIGH D1 : HIGH D3 : HIGH D4 : HIGH D5

ENDIF
'S1-S2
IF(A1=1)AND(A2=1)AND(A3=0)AND(A4=0)AND(A5=0) THEN
LCDOUT $fe,1,"ZON: 1-2-6"
GOSUB ESPERA
HIGH D3 : HIGH D4 : HIGH D5

ENDIF
'S3
IF(A1=0)AND(A2=0)AND(A3=1)AND(A4=0)AND(A5=0) THEN
LCDOUT $fe,1,"ZON: 3-6"
GOSUB ESPERA
HIGH D1 : HIGH D2 : HIGH D4 : HIGH D5

ENDIF
'S1-S3
IF(A1=1)AND(A2=0)AND(A3=1)AND(A4=0)AND(A5=0) THEN
LCDOUT $fe,1,"ZON: 1-3-6"
GOSUB ESPERA
HIGH D2 : HIGH D4 : HIGH D5

ENDIF
'S2-S3
IF(A1=0)AND(A2=1)AND(A3=1)AND(A4=0)AND(A5=0) THEN
LCDOUT $fe,1,"ZON: 2-3-6"
GOSUB ESPERA
HIGH D1 : HIGH D4 : HIGH D5

ENDIF
'S1-S2-S3
IF(A1=1)AND(A2=1)AND(A3=1)AND(A4=0)AND(A5=0) THEN
LCDOUT $fe,1,"ZON: 1-2-3-6"
GOSUB ESPERA
HIGH D4 : HIGH D5

ENDIF
'S4
IF(A1=0)AND(A2=0)AND(A3=0)AND(A4=1)AND(A5=0) THEN
LCDOUT $fe,1,"ZON: 4-6"
GOSUB ESPERA
HIGH D1 : HIGH D2 : HIGH D3 : HIGH D5

```

```

ENDIF
'S1-S4
IF(A1=1)AND(A2=0)AND(A3=0)AND(A4=1)AND(A5=0) THEN
  LCDOUT $fe,1,"ZON: 1-4-6"
  GOSUB ESPERA
  HIGH D2 : HIGH D3 : HIGH D5
ENDIF
'S2-S4
IF(A1=0)AND(A2=1)AND(A3=0)AND(A4=1)AND(A5=0) THEN
  LCDOUT $fe,1,"ZON: 2-4-6"
  GOSUB ESPERA
  HIGH D1 : HIGH D3 : HIGH D5
ENDIF
'S1-S2-S4
IF(A1=1)AND(A2=1)AND(A3=0)AND(A4=1)AND(A5=0) THEN
  LCDOUT $fe,1,"ZON: 1-2-4-6"
  GOSUB ESPERA
  HIGH D3 : HIGH D5
ENDIF
'S3-S4
IF(A1=0)AND(A2=0)AND(A3=1)AND(A4=1)AND(A5=0)THEN
  LCDOUT $fe,1,"ZON: 3-4-6"
  GOSUB ESPERA
  HIGH D1 : HIGH D2 : HIGH D5
ENDIF
'S1-S3-S4
IF(A1=1)AND(A2=0)AND(A3=1)AND(A4=1)AND(A5=0)THEN
  LCDOUT $fe,1,"ZON: 1-3-4-6"
  GOSUB ESPERA
  HIGH D2 : HIGH D5
ENDIF
'S2-S3-S4
IF(A1=0)AND(A2=1)AND(A3=1)AND(A4=1)AND(A5=0)THEN
  LCDOUT $fe,1,"ZON: 2-3-4-6"
  GOSUB ESPERA
  HIGH D1 : HIGH D5
ENDIF
'S1-S2-S3-S4
IF(A1=1)AND(A2=1)AND(A3=1)AND(A4=1)AND(A5=0)THEN
  LCDOUT $fe,1,"ZON: 1-2-3-4-6"
  GOSUB ESPERA
  HIGH D5
ENDIF
'S5
IF(A1=0)AND(A2=0)AND(A3=0)AND(A4=0)AND(A5=1)THEN
  LCDOUT $fe,1,"ZON: 5-6"
  GOSUB ESPERA
  HIGH D1 : HIGH D2 : HIGH D3 : HIGH D4
ENDIF
'S1-S5

```

```

IF(A1=1)AND(A2=0)AND(A3=0)AND(A4=0)AND(A5=1)THEN
  LCDOUT $fe,1,"ZON: 1-5-6"
  GOSUB ESPERA
  HIGH D2 : HIGH D3 : HIGH D4
ENDIF
'S2-S5
IF(A1=0)AND(A2=1)AND(A3=0)AND(A4=0)AND(A5=1)THEN
  LCDOUT $fe,1,"ZON: 2-5-6"
  GOSUB ESPERA
  HIGH D1 : HIGH D3 : HIGH D4
ENDIF
'S1-S2-S5
IF(A1=1)AND(A2=1)AND(A3=0)AND(A4=0)AND(A5=1)THEN
  LCDOUT $fe,1,"ZON: 1-2-5-6"
  GOSUB ESPERA
  HIGH D3 : HIGH D4
ENDIF
'S3-S5
IF(A1=0)AND(A2=0)AND(A3=1)AND(A4=0)AND(A5=1)THEN
  LCDOUT $fe,1,"ZON: 3-5-6"
  GOSUB ESPERA
  HIGH D1 : HIGH D2 : HIGH D4
ENDIF
'S1-S3-S5
IF(A1=1)AND(A2=0)AND(A3=1)AND(A4=0)AND(A5=1)THEN
  LCDOUT $fe,1,"ZON: 1-3-5-6"
  GOSUB ESPERA
  HIGH D2 : HIGH D4
ENDIF
'S2-S3-S5
IF(A1=0)AND(A2=1)AND(A3=1)AND(A4=0)AND(A5=1)THEN
  LCDOUT $fe,1,"ZON: 2-3-5-6"
  GOSUB ESPERA
  HIGH D1 : HIGH D4
ENDIF
'S1-S2-S3-S5
IF(A1=1)AND(A2=1)AND(A3=1)AND(A4=0)AND(A5=1)THEN
  LCDOUT $fe,1,"ZON: 1-2-3-5-6"
  GOSUB ESPERA
  HIGH D4
ENDIF
'S4-S5
IF(A1=0)AND(A2=0)AND(A3=0)AND(A4=1)AND(A5=1)THEN
  LCDOUT $fe,1,"ZON: 4-5-6"
  GOSUB ESPERA
  HIGH D1 : HIGH D2 : HIGH D3
ENDIF
'S1-S4-S5
IF(A1=1)AND(A2=0)AND(A3=0)AND(A4=1)AND(A5=1)THEN
  LCDOUT $fe,1,"ZON: 1-4-5-6"

```

```

    GOSUB ESPERA
    HIGH D2 : HIGH D3
ENDIF
'S2-S4-S5
IF(A1=0)AND(A2=1)AND(A3=0)AND(A4=1)AND(A5=1)THEN
    LCDOUT $fe,1,"ZON: 2-4-5-6"
    GOSUB ESPERA
    HIGH D1 : HIGH D3
ENDIF
'S1-S2-S4-S5
IF(A1=1)AND(A2=1)AND(A3=0)AND(A4=1)AND(A5=1)THEN
    LCDOUT $fe,1,"ZON: 1-2-4-5-6"
    GOSUB ESPERA
    HIGH D3
ENDIF
'S3-S4-S5
IF(A1=0)AND(A2=0)AND(A3=1)AND(A4=1)AND(A5=1)THEN
    LCDOUT $fe,1,"ZON: 3-4-5-6"
    GOSUB ESPERA
    HIGH D1 : HIGH D2
ENDIF
'S1-S3-S4-S5
IF(A1=1)AND(A2=0)AND(A3=1)AND(A4=1)AND(A5=1)THEN
    LCDOUT $fe,1,"ZON: 1-3-4-5-6"
    GOSUB ESPERA
    HIGH D2
ENDIF
'S2-S3-S4-S5
IF(A1=0)AND(A2=1)AND(A3=1)AND(A4=1)AND(A5=1)THEN
    LCDOUT $fe,1,"ZON: 2-3-4-5-6"
    GOSUB ESPERA
    HIGH D1
ENDIF
'S1-S2-S3-S4-S5
IF(A1=1)AND(A2=1)AND(A3=1)AND(A4=1)AND(A5=1)THEN
    LCDOUT $fe,1,"ZON: 1-2-3-4-5-6"
    GOSUB ESPERA
ENDIF

```

LISTO:

```

    A1 = 0
    A2 = 0
    A3 = 0
    A4 = 0
    A5 = 0
    AUX = 1

```

ENDIF

```

IF AUX = 2 THEN    'DESACTIVA ALARMA
    TRISD = %00011100

```

HIGH D1 : HIGH D2 : HIGH D3 : HIGH D4 : HIGH D5 : HIGH D6
LOW ACT
HIGH DES
LOW ZAUX : LOW ZAUX2

LCDOUT \$fe, 1,"DESACTIVADA"
LCDOUT \$fe,\$c0,"CLAVE CORRECTA"
'AUX = 1

'PARA CAMBIAR CLAVE SOLO CUANDO SE DESACTIVA ALARMA
PAUSE 1000
LCDOUT \$fe, 1, "PRESIONAR *0* PA"
LCDOUT \$fe,\$c0,"RA CAMBIAR CLAVE"
PAUSE 2500

HIGH F1: HIGH F2 :HIGH F3 :LOW F4
IF C2 = 0 THEN GRABAUNO

ENDIF

'IF C2 = 0 THEN 'CAMBIO 18-09-2006
' GOSUB GRABAUNO
'ENDIF

PAUSE 700
'LCDOUT \$fe, 1,"BIENVENIDO"
'LCDOUT \$fe,\$c0,"CLAVE:"
GOSUB SALUDO
GOTO TECLAUNO

DFALSO3:
FOR R = 1 TO 20
'IF (AUX=1)OR(AUX=3) THEN
'IF
(SEN1=0)OR(SEN2=0)OR(SEN3=0)OR(SEN4=0)OR(SEN5=0)OR(SEN6=0)THEN
MAL
'ENDIF
PAUSE 150
HIGH BIP
PAUSE 150
LOW BIP
HIGH F1: HIGH F2 :HIGH F4 :LOW F3
IF (C3=0)AND(C1=0)THEN 'CON 7 Y 9
GOSUB PTECLA
GOSUB SALUDO
GOTO TECLAUNO
ENDIF
DES3 = 1
NEXT

```
IF (DES3=1) THEN
HIGH ZAUX
HIGH ZAUX2
ENDIF
```

```
DPANICO: 'ALARMA DESACTIVADA ERROR AL INGRESAR CLAVE PARA
ACTIVAR
```

```
    LCDOUT $fe, 1,"ERROR"
    LCDOUT $fe,$c0,"CLAVE INCORRECTA"
    LCDOUT $fe, 1
    HIGH F1: HIGH F2 :HIGH F4 :LOW F3
    IF (C3=0)AND(C1=0)THEN 'DESACTIVAUNO   'HAY QUE REVISAR
    GOSUB PTECLA
    GOTO TECLAUNO
ENDIF
    GOTO DPANICO
```

```
FALSO:
    GOSUB BARRIDO :GOSUB PTECLA
    LCDOUT $fe,$c8,"*"
```

```
FALSO1:
    GOSUB BARRIDO :GOSUB PTECLA
    LCDOUT $fe,$c9,"*"
```

```
FALSO2:
    GOSUB BARRIDO :GOSUB PTECLA
    LCDOUT $fe,$ca,"*"
```

```
FALSO3:
    IF (AUX=0)OR(AUX=2)OR(DES2=1) THEN DFALSO3
    PAUSE 1500
    FOR R = 1 TO 10
    IF (AUX=1 THEN
    IF
    (SEN1=0)OR(SEN2=0)OR(SEN3=0)OR(SEN4=0)OR(SEN5=0)OR(SEN6=0)THEN
MAL
    ENDIF
    PAUSE 150
    HIGH BIP
    PAUSE 150
    LOW BIP
    HIGH F1: HIGH F2 :HIGH F4 :LOW F3
    IF (C3=0)AND(C1=0)THEN EQUIVOCADA 'CON 7 Y 9
    NEXT
```

```
'HAY QUE REVISAR
PANICO:
    HIGH ZAUX
```

```

IF (AUX=1)OR(AUX=3) THEN
IF
(SEN1=0)OR(SEN2=0)OR(SEN3=0)OR(SEN4=0)OR(SEN5=0)OR(SEN6=0)THEN
MAL
ENDIF
LCDOUT $fe, 1,"ERROR"
LCDOUT $fe,$c0,"CLAVE INCORRECTA"
HIGH F1: HIGH F2 :HIGH F4 :LOW F3
IF (C3=0)AND(C1=0)THEN 'DESACTIVAUNO 'HAY QUE REVISAR
GOSUB PTECLA
LCDOUT $fe,1,"DESACTIVACION"
GOTO DESACTIVAUNO
ENDIF
GOTO PANICO

```

ESPERA:

```

HIGH ACT
LOW DES
HIGH F1: HIGH F2 :HIGH F3 :LOW F4
FOR R=1 TO 300
"PAUSE 200
PAUSE 20
IF (C3=0)AND(C1=0)THEN 'CON # Y * PRESIONADOS A LA VEZ
GOSUB EMPEZAR
ENDIF
NEXT
""LINEA DE ABAJO AUMENTE 2006-16-09
HIGH F4
RETURN

```

ESPERAE:

```

HIGH F1: HIGH F2 :HIGH F3 :LOW F4

```

```

FOR R=1 TO 100
PAUSE 100

```

```

IF (C3=0)AND(C1=0)THEN 'CON # Y * PRESIONADOS A LA VEZ
GOSUB EMPEZAR
ENDIF
NEXT
""LINEA DE ABAJO AUMENTE 2006-16-09
HIGH F4
RETURN

```

EMPEZAR:

```

LCDOUT $fe,1,"DESACTIVACION"
LCDOUT $fe,$c0,"CLAVE:"
GOSUB PTECLA
LOW RFON

```

ENCENDIDA

DESACTIVAUNO:

```

GOSUB BARRIDO :GOSUB PTECLA

```

```
LCDOUT $fe,$c7,"*"
IF NUMERO = SETPRIME THEN TECLADOS
GOTO FALSO
```

```
DESACTIVADOS:
GOSUB BARRIDO :GOSUB PTECLA
LCDOUT $fe,$c8,"*"
IF NUMERO = SETSEGUN THEN TECLATRES
GOTO FALSO1
```

```
DESACTIVATRES:
GOSUB BARRIDO :GOSUB PTECLA
LCDOUT $fe,$c9,"*"
IF NUMERO = SETERCER THEN TECLACUATRO
GOTO FALSO2
```

```
DESACTIVACUATRO:
GOSUB BARRIDO :GOSUB PTECLA
LCDOUT $fe,$ca,"*"
PAUSE 1000
IF NUMERO = SETCUART THEN DESACTIVAOPENGE
GOTO FALSO3
```

```
DESACTIVAOPENGE:
DES2 = 0
LOW ZAUX : LOW ZAUX2
HIGH DES : LOW ACT
AUX = AUX + 1
FOR R = 1 TO 2
PAUSE 100
HIGH BIP
PAUSE 100
LOW BIP
NEXT
LCDOUT $fe,1,"BIENVENIDO"
LCDOUT $fe,$c0,"CLAVE:"
```

```
SALUDO:
LCDOUT $fe,1,"BIENVENIDO"
LCDOUT $fe,$c0,"CLAVE:"
RETURN
```

```
ADVERTENCIA:
LCDOUT $fe,1,"ASALTO"
RETURN
```

```
TRANSMITIR:
LCDOUT $fe,1,"SE VA A"
LCDOUT $fe,$c0,"TX:"
PAUSE 500
```

```
SEROUT PORTA.5,T9600,["ARVPT"]

IF SEN1=0 THEN
  SEROUT PORTA.5,T9600,["Z1"]
ELSE
  SEROUT PORTA.5,T9600,["XX"]
ENDIF
IF SEN2=0 THEN
  SEROUT PORTA.5,T9600,["Z2"]
ELSE
  SEROUT PORTA.5,T9600,["XX"]
ENDIF
IF SEN3=0 THEN
  SEROUT PORTA.5,T9600,["Z3"]
ELSE
  SEROUT PORTA.5,T9600,["XX"]
ENDIF
IF SEN4=0 THEN
  SEROUT PORTA.5,T9600,["Z4"]
ELSE
  SEROUT PORTA.5,T9600,["XX"]
ENDIF
IF SEN5=0 THEN
  SEROUT PORTA.5,T9600,["Z5"]
ELSE
  SEROUT PORTA.5,T9600,["XX"]
ENDIF
IF SEN6=0 THEN
  SEROUT PORTA.5,T9600,["Z6"]
ELSE
  SEROUT PORTA.5,T9600,["XX"]
ENDIF
RETURN

END
```

ANEXO VII: SCRIPT PARA LA BASE DE DATOS

SCRIPT BASE DE DATOS

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id =
OBJECT_ID(N'[dbo].[nivel]') AND type in (N'U'))
BEGIN
CREATE TABLE [dbo].[nivel](
[idnivel] [int] NOT NULL,
[descnivel] [varchar](20) NOT NULL
) ON [PRIMARY]
END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id =
OBJECT_ID(N'[dbo].[cliente]') AND type in (N'U'))
BEGIN
CREATE TABLE [dbo].[cliente](
[codcliente] [varchar](5) NOT NULL,
[nombrecliente] [varchar](25) NOT NULL,
[apellidocliente] [varchar](25) NOT NULL,
[direccioncliente] [varchar](60) NOT NULL,
[cedulacliente] [varchar](11) NOT NULL,
[clavecliente] [varchar](20) NOT NULL,
[nikcliente] [varchar](20) NOT NULL,
[emailcliente] [varchar](100) NOT NULL,
[telefono1cliente] [varchar](9) NOT NULL,
[telefono2cliente] [varchar](9) NOT NULL,
[movil1cliente] [varchar](9) NOT NULL,
[movil2cliente] [varchar](9) NOT NULL,
[ciudadcliente] [varchar](20) NOT NULL,
[provinciacliente] [varchar](20) NOT NULL,
[paiscliente] [varchar](20) NOT NULL,
[equipocliente] [varchar](20) NOT NULL,
[trabajocliente] [varchar](40) NOT NULL,
[trabajodircliente] [varchar](60) NOT NULL,
[telefonotrabajocliente] [varchar](9) NOT NULL,
[sectordomiciliocliente] [varchar](60) NOT NULL,
[familiarcliente] [varchar](40) NOT NULL,
[telefonofamiliarcliente] [varchar](9) NOT NULL,
[movilfamiliarcliente] [varchar](9) NOT NULL,
[imagendomiciliocliente] [image] NULL,
[idnivel] [int] NOT NULL,
[idstatus] [int] NOT NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
```

```

IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id =
OBJECT_ID(N'[dbo].[zonas]') AND type in (N'U'))
BEGIN
CREATE TABLE [dbo].[zonas](
        [codzona] [varchar](3) NOT NULL,
        [descripcionzona] [varchar](50) NOT NULL
) ON [PRIMARY]
END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id =
OBJECT_ID(N'[dbo].[alertas]') AND type in (N'U'))
BEGIN
CREATE TABLE [dbo].[alertas](
        [codalerta] [int] IDENTITY(1,1) NOT NULL,
        [codcliente] [varchar](50) NOT NULL,
        [mensajealerta] [varchar](17) NOT NULL,
        [estadoalerta] [int] NOT NULL CONSTRAINT [DF_alertas_estadoalerta] DEFAULT
((0)),
        [fechaalerta] [varchar](50) NOT NULL,
        [horaalerta] [varchar](10) NOT NULL
) ON [PRIMARY]
END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id =
OBJECT_ID(N'[dbo].[clientezona]') AND type in (N'U'))
BEGIN
CREATE TABLE [dbo].[clientezona](
        [idclientezona] [int] IDENTITY(1,1) NOT NULL,
        [codcliente] [varchar](5) NOT NULL,
        [codzona] [varchar](2) NOT NULL,
        [descripcionzona] [varchar](50) NOT NULL
) ON [PRIMARY]
END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
IF NOT EXISTS (SELECT * FROM sys.objects WHERE object_id =
OBJECT_ID(N'[dbo].[alertaszonas]') AND type in (N'U'))
BEGIN
CREATE TABLE [dbo].[alertaszonas](
        [codalertazona] [int] NOT NULL,
        [codzona] [varchar](2) NOT NULL,
        [codalerta] [int] NOT NULL
) ON [PRIMARY]
END

```

INDICE DE FIGURAS

Figura 1.1. Subsistemas de GSM.....	10
Figura 1.2. Arquitectura de un Sistema GSM	11
Figura 1.3. Canales Lógicos de un sistema GSM.....	17
Figura 1. 4. Esquema de una red GPRS	26
Figura 2.1. Esquema de la solución presentada.....	51
Figura 2.2. Diagrama de bloques del sistema presentado	56
Figura 2.3. Sub-bloques del bloque de Toma y Procesamiento de Información.....	57
Figura 2.4. Sub-bloques del bloque de Transmisión inalámbrica de Información.....	58
Figura 2.5. Esquema de comportamiento del Sensor Modelo.....	62
Figura 2.6. Esquema de conexión para ejecutar llamada telefónica.....	66
Figura 2.7. Esquema de conexión del conversor LM338.....	67
Figura 2.8. Esquema de conexión para una etapa de potencia	68
Figura 2.9. Sub-bloques del bloque de Transmisión inalámbrica de Información.....	69
Figura 3.1. Interconexión entre los bloque de la fuente de alimentación	88
Figura 3.2. Esquema de conexión a la red GPRS	93
Figura 3.3. Diagrama de flujo dentro del aplicativo web.....	99
Figura 4.1. Proceso de pruebas para comunicación PC - modem	106
Figura 4.2. Desarrollo de las capas de datos	110
Figura 4.3. Tabla dentro de la base de datos Alarma.....	114
Figura 4.4. Desarrollo de la Capa de Datos.....	114
Figura 4.5. Campos de la capa cliente dentro de la base de	115
Figura 4.6. Desarrollo de la capa de presentación	119
Figura 4.7. Modelo de las tres capas vistas en el entorno	120

INDICE DE TABLAS

Tabla 1.1. Capacidad de Fichero en diversos sistemas operativos	35
Tabla 2.1 Posibles salidas de Sensor Modelo.....	62
Tabla 2.2 Frecuencias generadas para tonos DTMF.....	65
Tabla 2.3. Trama de Zonas	72
Tabla 5.1. Costo de Equipos del Centro de Monitoreo y Base de Datos	135
Tabla 5.2. Costo de Servicios consumidos para el Sistema	135
Tabla 5.3. Costo de Materiales para elaboración de Tarjeta Electrónica.....	136
Tabla 5.4. Costo de Materiales y Equipos para Enlace GPRS	136
Tabla 5.5. Valor de servicio GPRS según la cantidad de MB consumidos	137
Tabla 5.6. Costo de Servicios de Montaje y configuración en vivienda.....	138
Tabla 5.7. Resumen de Costos iniciales.....	138
Tabla 5.8. Resumen de Costos Mensuales	139
Tabla 5.9. Resultado de consulta efectuada al público	141
Tabla 5.10. Valor a ganar según el número de usuarios	142

GLOSARIO

24/7 o 24/7/365

Hace referencia al tiempo de trabajo: 24 horas al día, 7 días a la semana, 365 días al año

CLIENTE

Se refiere al propietario de la vivienda que adquiere el servicio de monitoreo.

CENTRO DE CONTROL

Es el lugar donde se instalará un computador en el que se presentará el estado de todas las viviendas controladas, este centro será atendido por un supervisor en el formato 24/7

DLL

Es el acrónimo de Dynamic Linking Library (Bibliotecas de Enlace Dinámico), término con el que se refiere a los archivos con código ejecutable que se cargan bajo demanda del programa por parte del sistema operativo. Esta denominación se refiere a los sistemas operativos Windows siendo la extensión con la que se identifican los ficheros, aunque el concepto existe en prácticamente todos los sistemas operativos modernos.

ESTADO DE ALERTA

Termino utilizado para indicar que uno de los sensores de la vivienda ha detectado un evento

GNU

El proyecto GNU fue iniciado por Richard Stallman con el objetivo de crear un sistema operativo completo libre: el sistema GNU . El 27 de septiembre de 1983

se anunció públicamente GNU es un acrónimo recursivo que significa "GNU No es Unix".

GPRS

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. 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.

SERVICIO DE MONITOREO

Hace referencia al control desde un centro de monitoreo de todos los eventos ocurridos en una vivienda

SERVICIO WINDOWS

Archivo ejecutable que se instala en un computador y se lo puede configurar para que se active cada vez que dicho computador se encienda

SQL

El Lenguaje de Consulta Estructurado (Structured Query Language) es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas.

La presente documentación correspondiente al Proyecto de Tesis de Grado: “Diseño e implementación de un sistema de vigilancia remota para una residencia utilizando plataformas GPRS e Internet” ha sido presentada el día _____ del mes de Febrero del 2007.

Quedando como constancia de esta entrega firman los involucrados

Autores del documento

A. Soledad Zapata Y

Andrés R. Vallejo P.

Ing Gonzalo Olmedo

Ab. Jorge Carvajal

Director de Carrera

Secretario Académico