

# CAPÍTULO VI

## DESARROLLO DEL APLICATIVO



## **6.1. INTRODUCCIÓN.**

En el transcurso de la investigación se encontró dentro del "Hospital San Vicente de Paúl", cuatro estaciones de enfermería las cuales funcionan en diferentes pisos y son independientes la una de la otra, además tienen una estructura organizacional común en cada una de ellas.

Dentro de la estación de enfermería se puede llevar el control manual de enfermeros de turno, así como la planificación de horarios, pacientes hospitalizados y kardex de cada uno de ellos, también se lleva a cabo un gran número de informes de las actividades de enfermería.

El monitoreo manual resulta engorroso y poco práctico a la hora de llevar un control del personal de enfermería o de los pacientes hospitalizados. Además si el control depende de varias personas es muy probable que se cometan errores y descuidos con los pacientes.

En cambio si se desea monitorear toda la estación de enfermería de una manera más eficiente es necesario contar con un sistema de monitoreo que automatice el control de los procesos que se llevan a cabo dentro de esta.

Tras el estudio de las herramientas que gestionan el ciclo de vida de las aplicaciones, se puede decir que tanto Microsoft Visual Studio Team System como la Plataforma de Desarrollo de IBM Rational, son alternativas muy buenas para el desarrollo completo de sistemas en general, y una característica principal que las diferencia es el factor económico que implica la utilización de cada una de ellas.

Según las características principales del departamento de enfermería, la mejor opción es desarrollar con herramientas de Microsoft, ya que en esta institución hospitalaria no se cuenta con un presupuesto económico establecido para este proyecto, además en el aspecto tecnológico del Hospital San Vicente de Paúl actualmente se encuentran basados en servidores Windows 2000 y base de datos SQL Server 2000, siendo así, el departamento de enfermería puede hacer

uso de estos recursos para implementar el Sistema de Monitoreo de pacientes para salas de hospitalización de una manera exitosa.

Por lo anteriormente expuesto y en virtud de la investigación desarrollada en anteriores capítulos, desarrollar con herramientas de tecnología Microsoft es la mejor opción para este proyecto, a continuación se señalan las herramientas a utilizar:

- Microsoft Visual Studio Team System.
  - Microsoft Visual Studio Team Suite.
    - Visual Studio 2005 Team Edition for Software Architects.
    - Visual Studio 2005 Team Edition for Software Developers.
    - Visual Studio 2005 Team Edition for Software Testers.
  - Visual Studio 2005 Team Foundation Server.
  - Microsoft SQL Server 2005.

Visual Studio 2005 Team Edition for Data Base Professionals, no se utilizó en el desarrollo de este proyecto por encontrarse en un estado de pruebas, y para la fecha no es muy estable, pero se debe recalcar que la próxima versión de Team System se encuentra cerca de salir al mercado con mejoras significativas, incluyendo esta herramienta.

Además para que el sistema funcione correctamente al implantarlo en los servidores del HSVP, debemos utilizar el lenguaje SQL/92 Standar el cual nos permite ejecutar los script de la base de datos tanto en SQL Server 2000 y 2005.

## **6.2. ESPECIFICACIONES DEL APLICATIVO.**

El tema planteado para el desarrollo del presente Aplicativo es:

“Sistema de Monitoreo de Pacientes para Salas de Hospitalización, del Hospital San Vicente de Paúl.”

Tras analizar y definir los requerimientos de las estaciones de enfermería del HSVP, se concluye que el sistema se debe dividir en:

- Módulo de Seguridad.

- Módulo Administración de actividades de monitoreo.
- Módulo de Monitoreo de pacientes, para las salas de enfermería.
- Módulo para Alertas Tempranas.

### **6.2.1. MÓDULO DE SEGURIDAD.**

El Módulo de Seguridad es el encargado de gestionar los siguientes ítems:

- Control de ingreso mediante una interfaz que solicita, usuario y clave.
- Agente de Vigilancia de turno y horario programado, para verificar el acceso por horario de usuario registrado.
- Filtrado de información por usuario y estación.
- Registrar la hora y fecha en la cual ingresa al sistema un usuario.
- Informes del Ingreso de Personal de estación, fecha y hora de registro.

Además la fecha y hora se toman desde el servidor de base de datos para sincronizar las actividades de las estaciones dentro del sistema.

Como algo adicional el usuario solo posee tres intentos para ingresar al sistema después de los cuales el sistema se cierra automáticamente.

### **6.2.2. MÓDULO ADMINISTRACIÓN DE ACTIVIDAD DE MONITOREO.**

Este Módulo tiene a dos tipos de usuarios, el súper administrador y los líderes de enfermería.

El Súper Administrador es el rol que permite:

- Crear, Modificar y Eliminar la estructura de Pisos, Áreas, Estaciones, y Camas disponibles, dentro del Sistema.
- Crear, modificar, eliminar, deshabilitar los líderes de enfermería y asignar los usuarios y claves para que ingresen.
- Puede revisar los pacientes de todo el hospital, además de crear, modificar, eliminar y deshabilitar cualquier información en cualquier momento.

- Puede revisar, imprimir informes del módulo de seguridad, tales como las fechas y horas en las cuales ingresaron los usuarios y cuando salieron del sistema.

Líderes de Enfermería se encargan de:

- Crear, modificar, eliminar y deshabilitar los auxiliares de enfermería exclusivos de su estación.
- Crear, modificar, eliminar y deshabilitar pacientes hospitalizados en su estación.
- Ingresar, modificar y eliminar planes y actividades de enfermería.
- Incluir, cambiar, borrar y discontinuar kardex que incluye alertas de kardex.
- Tipificar los medicamentos utilizados para suministrar a los pacientes.
- Incluir, cambiar, borrar y discontinuar medicamentos dentro de los kardex.
- Crear, cambiar y eliminar los horarios para cada uno de los auxiliares de enfermería.

### **6.2.3. MÓDULO DE MONITOREO DE PACIENTES.**

En este módulo se encuentra un rol de usuario el cual es auxiliar de enfermería que es el encargado de las siguientes funciones dentro del sistema:

- Crear, modificar, eliminar y deshabilitar pacientes hospitalizados en su estación.
- Ingresar, modificar y eliminar planes y actividades de enfermería.
- Incluir, cambiar, borrar y discontinuar kardex que incluye alertas de kardex.
- Incluir, cambiar, borrar y discontinuar medicamentos dentro de los kardex.
- Permite buscar pacientes, así como personal de enfermería.

- Imprime reportes de kardex en el cambio de turno.

Los auxiliares de enfermería serán los responsables de los pacientes en su horario de turno al que fueron asignados, ya que se pueden generar reportes en los cuales se indican las actividades mediante observaciones, así como de medicamento suministrado, gastos médicos, los cuales indican la situación real de las actividades del departamento de enfermería.

#### **6.2.4. MODULO PARA ALERTAS TEMPRANAS.**

Este modulo tiene dos tipos de alertas:

##### Alertas Personales

Las alertas personales se manejan mediante un pulsador que envía una señal hacia la placa electrónica que ha sido diseñada para enviar información a través del puerto de comunicaciones COM1 hacia la computadora.

Esta placa se encarga de:

- Capturar o recibir la pulsación de un interruptor que el paciente posee en la cabecera de su cama.
- Transforma la señal mediante el PIC16F877A y las envía mediante el puerto serial de comunicaciones COM1.

En el sistema de monitoreo existe una parte de software:

- Existe un agente de búsqueda de peticiones de alertas personales, el cual es el encargado de recibir cualquier señal que llegue al puerto serial.
- Una vez que el agente encontró la petición se dirige con esa información hacia la base de datos y busca la información correspondiente a ese pulsador, la misma que está relacionada con los datos del paciente hospitalizado.
- Muestra la información en una ventana que es presentada automáticamente, la que corresponde a los datos del paciente

hospitalizado que pulso dicho interruptor. En esta ventana se pedirá que se ingrese una observación de la actividad que realizó el personal de enfermería encargado del turno.

#### Alertas Médicas Generales

- Los líderes de enfermería como los auxiliares son los encargados de planificar los kardex los mismos que contienen información del horario en el cual se suministrará los medicamentos, conjuntamente a esto se encuentra un agente el cual busca en la base de datos los kardex planificados cada hora, los cuales deben coincidir en fecha y hora en las que se programaron para cada paciente.
- Después de que el agente encuentra el o los kardex planificados, esta muestra una ventana con los datos de información correspondiente al paciente y a su debida medicación, conjuntamente a esto se debe incluir datos en observaciones para el seguimiento del progreso del paciente hospitalizado.

### **6.3. METODOLOGÍA.**

Con respecto al diseño y desarrollo del sistema de monitoreo de pacientes para salas de hospitalización del HSVP, específicamente en este sistema se aconseja seguir una metodología orientada a objetos la misma que influirá en todo el proceso de desarrollo, afectando la división de tareas y así facilitando el trabajo.

En el desarrollo de este aplicativo es preciso realizar una aclaración, al momento de la elección de la herramienta ALM estamos tomando por sentado su metodología que en este caso es Microsoft Solution Framework.

Microsoft Solution Framework incluye dos instancias de procesos predeterminados:

- MSF para Desarrolladores de Software Ágil. (Agile Software Development)
- MSF para CMMI Process Improvement.

De estas dos instancias se ha elegido a MSF para Agile Software Development, en la misma que podemos encontrar la guía de procesos que nos muestra una serie de tareas predefinidas que se debe seguir para asegurar la calidad del sistema, después de crear el proyecto de equipo, tenemos las primeras tareas:

**WorkItems administrados en Team System Web Access**

The screenshot displays the 'Work Items' section of the Microsoft Team System Web Access interface. The main content area shows a table titled 'Tareas Assigned To Me' with the following data:

ID	Título	Estado	Tipo de
1	Configuración: definir permisos	Cerrado	Tarea
2	Configuración: migración del código fuente	Cerrado	Tarea
3	Configuración: migración de elementos de trabajo	Cerrado	Tarea
4	Configuración: definir directivas de protección	Cerrado	Tarea
5	Configuración: configurar la generación	Activo	Tarea
6	Configuración: enviar correo electrónico a los usuarios para la instalación e ...	Cerrado	Tarea
7	Crear definición de la visión	Cerrado	Tarea
8	Configuración: crear descripción del proyecto en el portal del proyecto de e...	Activo	Tarea
9	Crear personas	Activo	Tarea
10	Definir longitud de iteración	Activo	Tarea
11	Crear hoja de cálculo del enfoque de prueba con umbrales para la prueba	Activo	Tarea
12	Enumerar y establecer la prioridad de la lista de escenarios	Activo	Tarea
13	Pensar y establecer la prioridad de la lista de requisitos de calidad de servicio	Activo	Tarea
14	Configuración: crear estructura del proyecto	Activo	Tarea
15	Crear plan de iteración	Activo	Tarea

The interface also includes a search bar, navigation tabs (Home, Work Items, Reports, Documents, Source, Build), and a sidebar with sections like Favorites, New Work Item, My Queries, Team Queries, and Reports.

Fuente[Team System Web Access]

**Figura 6.1**

### 6.3.1. CONFIGURACIÓN INICIAL DE TAREAS ASIGNADAS.

Durante estas actividades se deben elaborar la visión del proyecto, también se construye, descripción del proyecto y otros parámetros de la aplicación, estos documentos se generan siguiendo las tareas que se encuentran detalladas en Team System Web Access, estas tareas se generan automáticamente a partir de la creación de un proyecto de equipo.



Todas las tareas mencionadas anteriormente se deben ir construyendo iterativamente ya que la metodología MSF Agile trabaja con este tipo de planificación, al asignar tareas a cada uno de los roles.

Algunos documentos se debe subir al portal principal para que todos los involucrados estén al tanto de los detalles del mismo, estas configuraciones se detallan en los anexos de esta investigación.

### **6.3.2. DISEÑO DE INTERFAZ DE USUARIO.**

La interface de usuario es importante, ya que de esta dependen las iteraciones del usuario con la aplicación.

La aplicación consta de entidades como:

- Un súper administrador.
- Líderes de enfermería.
- Auxiliares de enfermería.
- Reportes.
- Administración de personal de enfermería.
- Administración de alertas
- Administración de la estructura hospitalaria (pisos, áreas, estaciones, camas).
- Administración de pacientes.
- Generación de kardex.
- Gestión de planes de enfermería.
- Tipificado de medicamentos.
- Control de Sesiones.

Para cumplir con lo dicho anteriormente la aplicación está basada en unas interfaces de usuario selectivas o basadas en menús, que hace el uso de técnicas tipo WIMP<sup>60</sup>, las mismas que contienen una barra principal o menú

---

<sup>60</sup> WIMP.- Interfaces que hacen uso de cuatro técnicas: ventanas (Windows), Iconos, Menús y dispositivos apuntadores (Pointing devices).

principal, y otra barra de herramientas con iconos representativos de todas las funciones disponibles.

Como un requerimiento las funciones primarias del sistema de monitoreo de pacientes cuenta las siguientes funciones principales en la interfaz de usuario:

- Personal.
  - Gestión personal.
- Estructura.
  - Estructura hospitalaria.
- Paciente.
  - Gestión paciente.
- Alarma kardex.
  - Activar alarma.
- Medicamentos.
  - Ingreso.
  - Reportes gastos totales.
  - Reportes entrega de turnos.
- Horarios.
  - Turnos.
- Sesiones.
  - Cerrar sesión.
  - Ingresar como usuario.
- Ver horario.
  - Horario mensual.
- Cumplimiento.
  - Verificaciones.
- Ayuda.
  - Tutorial.
  - Acerca de.

La anterior estructura corresponde a la barra de menú principal, las cuales están contenidas dentro de una aplicación MDI, esta permite llamar a sub ventanas dentro de la misma aplicación, para mayor comodidad los reportes

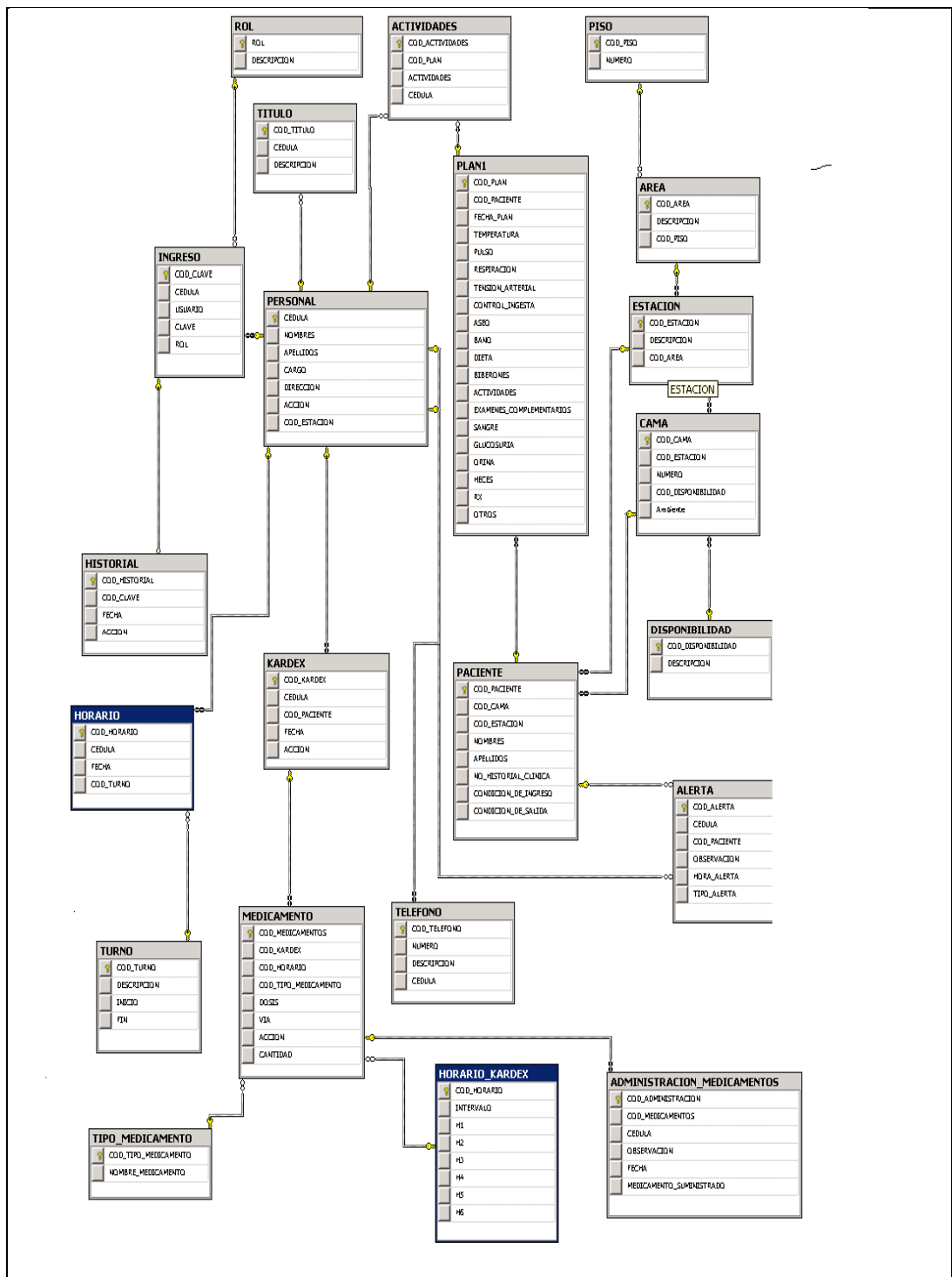
están desplegados en ventanas independientes de la aplicación MDI. En cuanto al lenguaje de desarrollo se ha utilizado C# por sus características netamente orientadas a objetos.

### 6.3.3. MODELO DE DATOS.

El modelo de base de datos SQL Server efectuado en la aplicación, es un modelo entidad relación, para el cual se puede utilizar Visual Studio 2005 Team Edition para profesionales de base de datos, pero esta herramienta por encontrarse en fases de prueba, no se utilizó, por lo cual se diagramó y administró desde SQL Server 2005.

<b>NOMBRE DE LA TABLA</b>	<b>DESCRIPCION</b>
INGRESO	Esta sirve para registrar un usuario autorizado para el ingreso al sistema.
ROL	Contiene el rol que se le asigna a un usuario de sistema en la tabla INGRESO.
HISTORIAL	Contiene el historial de ingresos / salidas del sistema que está registrado en la tabla INGRESO
PERSONAL	Incluye referente al personal de enfermería.
TITULO	Contiene información sobre el título que posee personal de enfermería, registrado en PERSONAL.
TELEFONO	Contiene información sobre el teléfono que posee personal de enfermería, registrado en PERSONAL.
HORARIO	Incluye información del horario del personal registrado en la tabla PERSONAL.
TURNO	Inicio y fin del turno de la tabla HORARIO.
PACIENTE	Contiene pacientes hospitalizados y ubicación.
ALERTA	Alertas personales generadas por el modulo electrónico de PIC16F877A.
PISO	Describe el número de piso al que corresponde con la estructura física del hospital.
AREA	Incluye datos sobre del área relacionada con el piso en que se encuentra.

ESTACION	Contiene datos relacionada con las estaciones de enfermería de cada área.
CAMA	Incluye datos del número de cama que se encuentran relacionada con cama.
DISPONIBILIDAD	Contiene datos de tipificado Habilitado y Deshabilitado de camas.
ACTIVIDADES	Corresponde a datos que se incluye dentro de un plan de actividades.
PLAN1	Contiene el plan de actividades de enfermería programadas para cada uno de los pacientes y relacionada con la tabla PACIENTE.
KARDEX	Kardex que son planificaciones de administración de medicamentos en determinados horarios hacia determinados pacientes.
TIPO_MEDICAMENTO	Incluye datos de tipificado de medicamentos.
MEDICAMENTO	Contiene los medicamentos que se deben suministrar de acuerdo con un determinado kardex.
HORARIO_KARDEX	Datos de tipos de horarios en los cuales se activa determinados kardex de acuerdo a una planificación.
ADMINISTRACION_ MEDICAMENTOS	Corresponde a registros que genera una alerta médica general provocada por un kardex planificado en la tabla HORARIO_KARDEX, y relacionada con la tabla MEDICAMENTO ya que registra el medicamento suministrado.
Fuente[Autores]	
Tabla 6.1 Descripción de Tablas de la Base de Datos Relacional.	
Diagrama de base de datos realizado con Team Edition para profesionales de Bases de datos.	

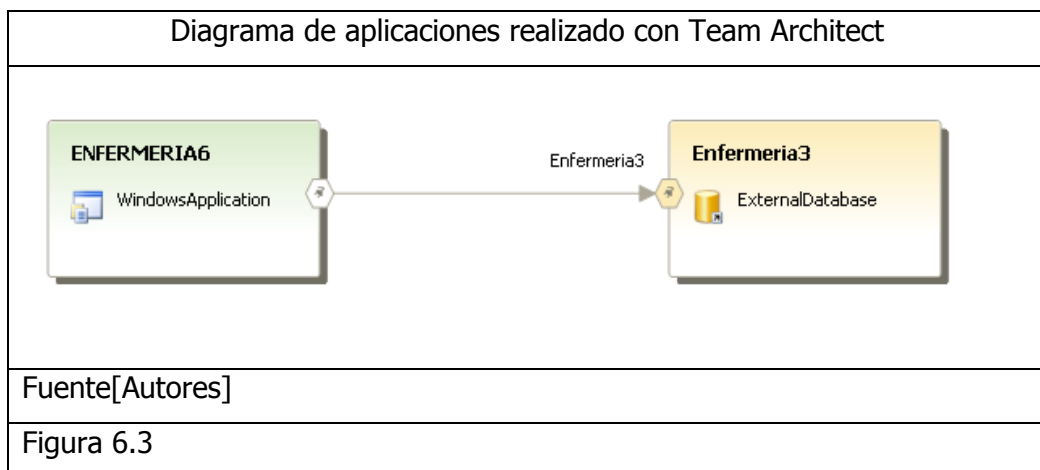


Fuente[Autores]

Figura 6.2

### 6.3.4. ARQUITECTURA DEL SISTEMA

El siguiente gráfico representa la arquitectura del sistema, el cual se creó como un diagrama necesario en el ciclo de vida de un proyecto dentro de MVSTS, para esto se utilizó la herramienta integrada Microsoft Visual Team Architect. Para crear el siguiente diagrama se hace click derecho sobre el proyecto, agregar nuevo diagrama de sistemas distribuidos, en la sub ventana que nos muestra elegimos diagrama de aplicaciones.



Una aplicación en el diseñador es análoga a un componente en UML y un extremo es equivalente a una interfaz UML.

El diseñador de aplicaciones contiene conceptos como métodos web y protocolos mediante el mecanismo de prototipos que es más potente que el mecanismo de estereotipos de UML.

El diseñador de aplicaciones posee las siguientes herramientas de diseño las cuales puede observarse en la Figura 6.4:

#### **Diseño General**

Puntero.- Selecciona los objetos dentro de un diagrama.

Conexión.- Utilizada para conectar las aplicaciones entre ellas mediante sus extremos.

Comentario.- Elemento libre de texto para aclaraciones.

## **Extremos**

WebServiceEndpoint.- Representa un punto de conexión para un servicio web.

WebContentEndpoint.- Representa un punto de conexión en el cual el contenido web se expone, ejemplo las conexiones de clientes.

GenericEndpoint.-Mecanismo para conectar aplicaciones genéricas

## **Aplicaciones**

WindowsApplication.- Representan una aplicación Windows.

ASP.NetWebServices.- Representa una aplicación ASP .NETWebApplication con un WebServiceEndpoint.

ASP .NET WebApplication.- Representa ASP .NETWebApplication con un WebContentEndpoint por defecto.

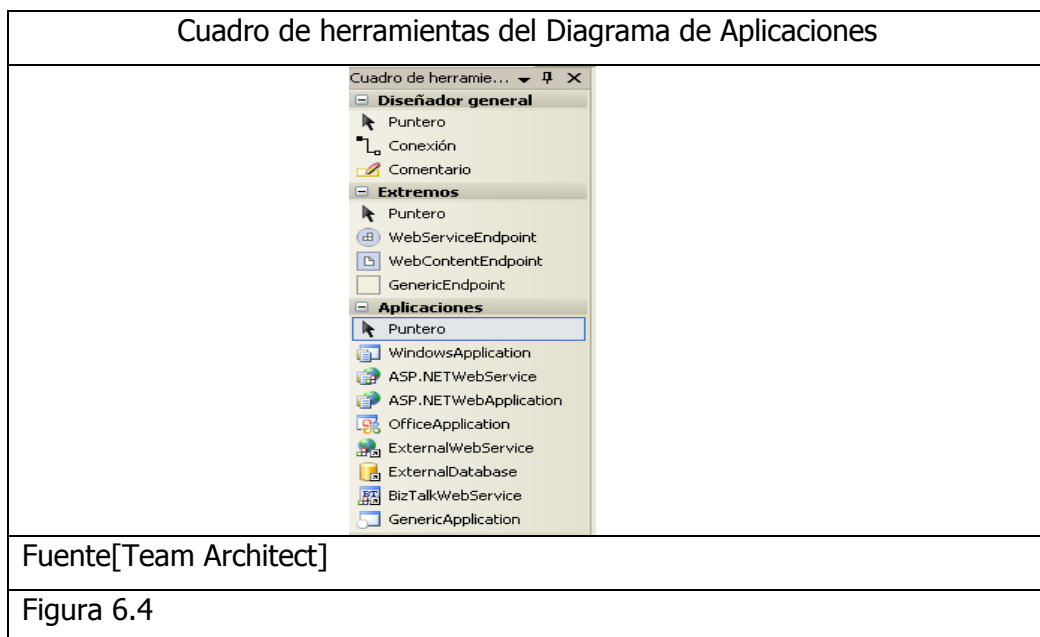
OfficeApplication.- Representa una aplicación de Microsoft Office.

ExternalWebService.- Representa un servicio Web externo, WSDL.

ExternalDatabase.- Representa una base de datos existente.

BizTalkWebService.- Representa un servicio Web BizTalk con un WebServiceEndpoint.

GenericApplication.- Representa una aplicación sin base o sin especificar.



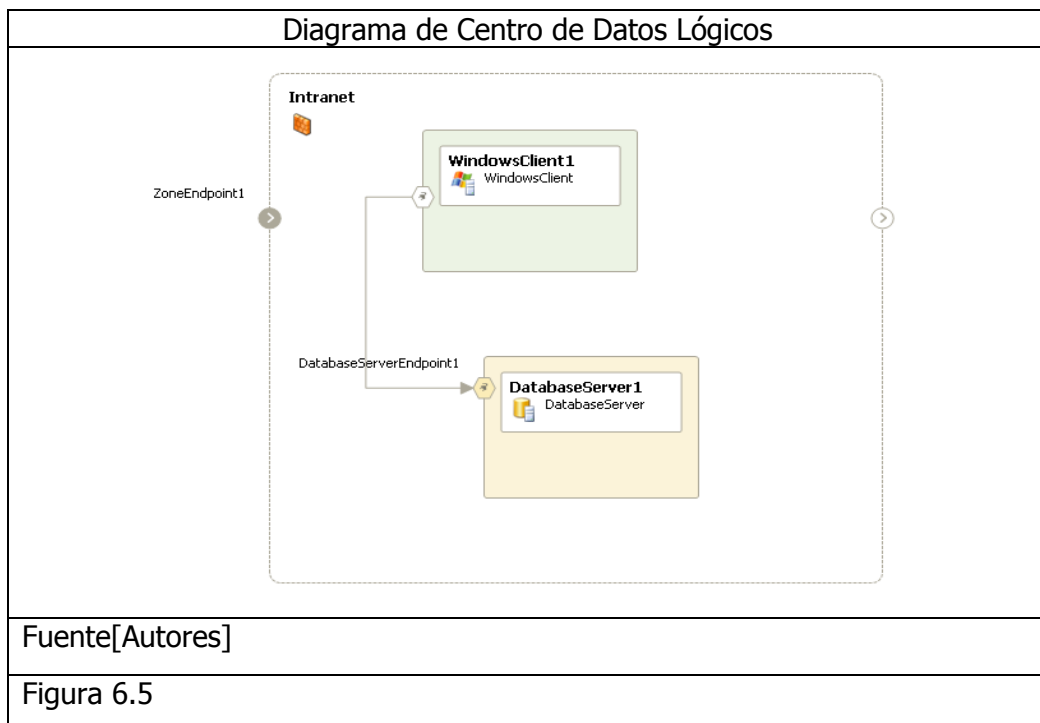
### 6.3.5. IMPLEMENTACIÓN.

El diagrama de centro de Datos Lógicos es casi como un diagrama de implementación de UML, porque muestra los nodos de implementación y las conexiones entre ellos.

Una diferencia importante es que mientras el diagrama de implementación de UML está predispuesto para el modelado de los nodos de implementación físicos, el diagrama de centros de datos lógicos es estrictamente lógico.

En la práctica los diagramas de implementación de UML se encuentran limitados en dos aspectos importantes:

- La mayoría de información que muestra no tiene uso real para los desarrolladores.
- Falta de integración entre la perspectiva de implementación de UML y las demás perspectivas de UML.
- MVSTS puede utilizar un metamodelo común SDM como un vínculo entre la infraestructura lógica y la arquitectura de aplicación.



A continuación se detalla las herramientas para diagramas de centro de datos lógicos que se pueden observar en la siguiente figura.



## **General**

Conexión.- Permite conectar extremos que pertenecen a servidores y zonas.

Comentario.- Es tan solo una nota de texto libre.

## **Extremos**

WebSiteEndpoint.- Tipo extreme con fuerza, a través del cual un cliente puede comunicarse con un servidor Web IIS utilizando el protocolo HTML.

HTTPClientEndpoint.- Este es el homologo del lado del cliente del WebSiteEndpoint del lado del servidor.

DatabaseClientEndpoint.- Este extremo del cliente representa el lado del cliente de una conexión a un servidor de base de datos.

GenericServerEndpoint.- Extremo generalizado para aquellas situaciones en las cuales un servidor genérico acepta comunicaciones entrantes a través de un protocolo no especificado.

GenericClientEndpoint.- Este es el homologo del lado del cliente del GenericServerEndpoint.

ZonePoint.- Este prototipo representa un extremo que restringe el tipo de comunicación que puede salir o entrar de una zona.

## **Servidores Lógicos**

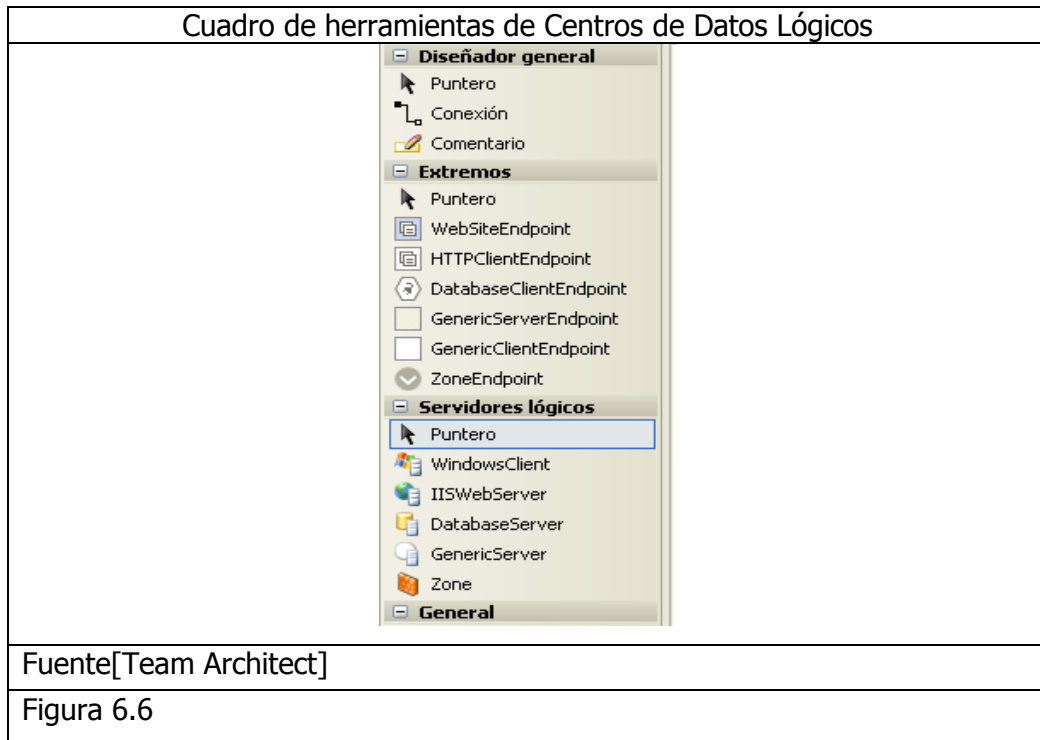
WindowsClient.- Representa una máquina que ejecuta un sistema operativo Windows en el cual se puede implementar una aplicación Windows.

IISWebServer.- Representa un servidor web IIS capaz de alojar aplicaciones web ASP.NET o servicios web.

DataServer.- Representa un servidor de base de datos.

GenericServer.- Representa un servidor genérico y es útil cuando se necesita mostrar un hosting en el que se implementara una aplicación de tecnologías que no son de Microsoft.

Zone.- Representa una zona de red para la cual se restringe la comunicación, hacia y desde los servidores en dicha zona, aun conjunto de protocolos específicos. En realidad puede no haber ningún mecanismo de restricción.



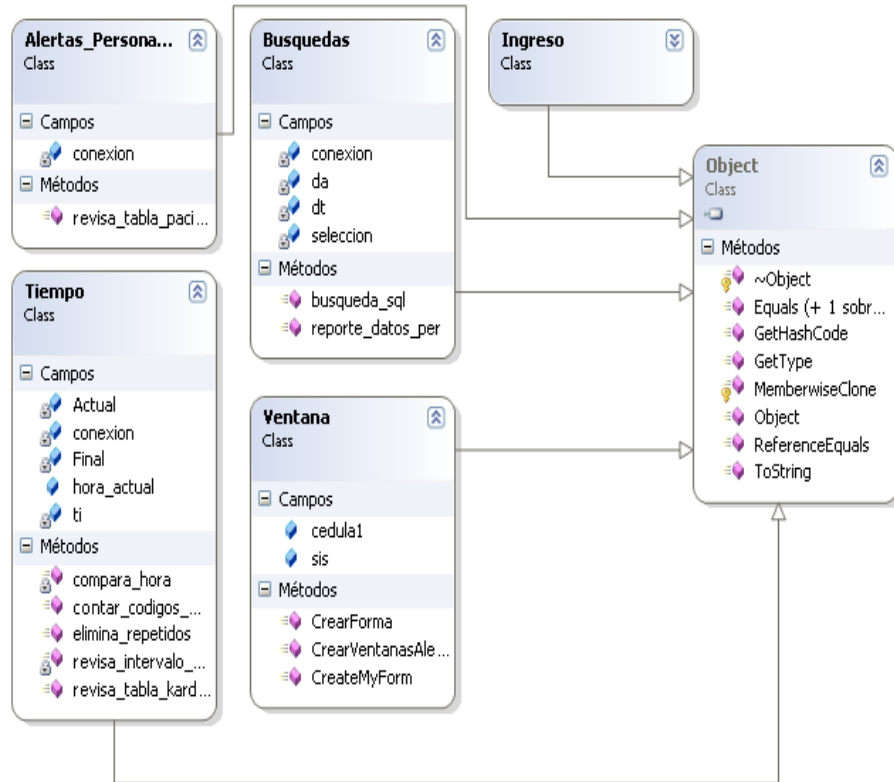
### 6.3.6. DISEÑO DE CLASES.

El diseñador de clases de MVSTS permite diagramar las clases del sistema. Puede plantearse de dos formas como una herramienta para visualizar código existente y como una herramienta para definir la estructura de dicho código.

El diseñador es optimizado para permitir distintos escenarios de uso, incluyendo la visualización de código existente, la refactorización y el diseño de diagramas para propósitos documentales.

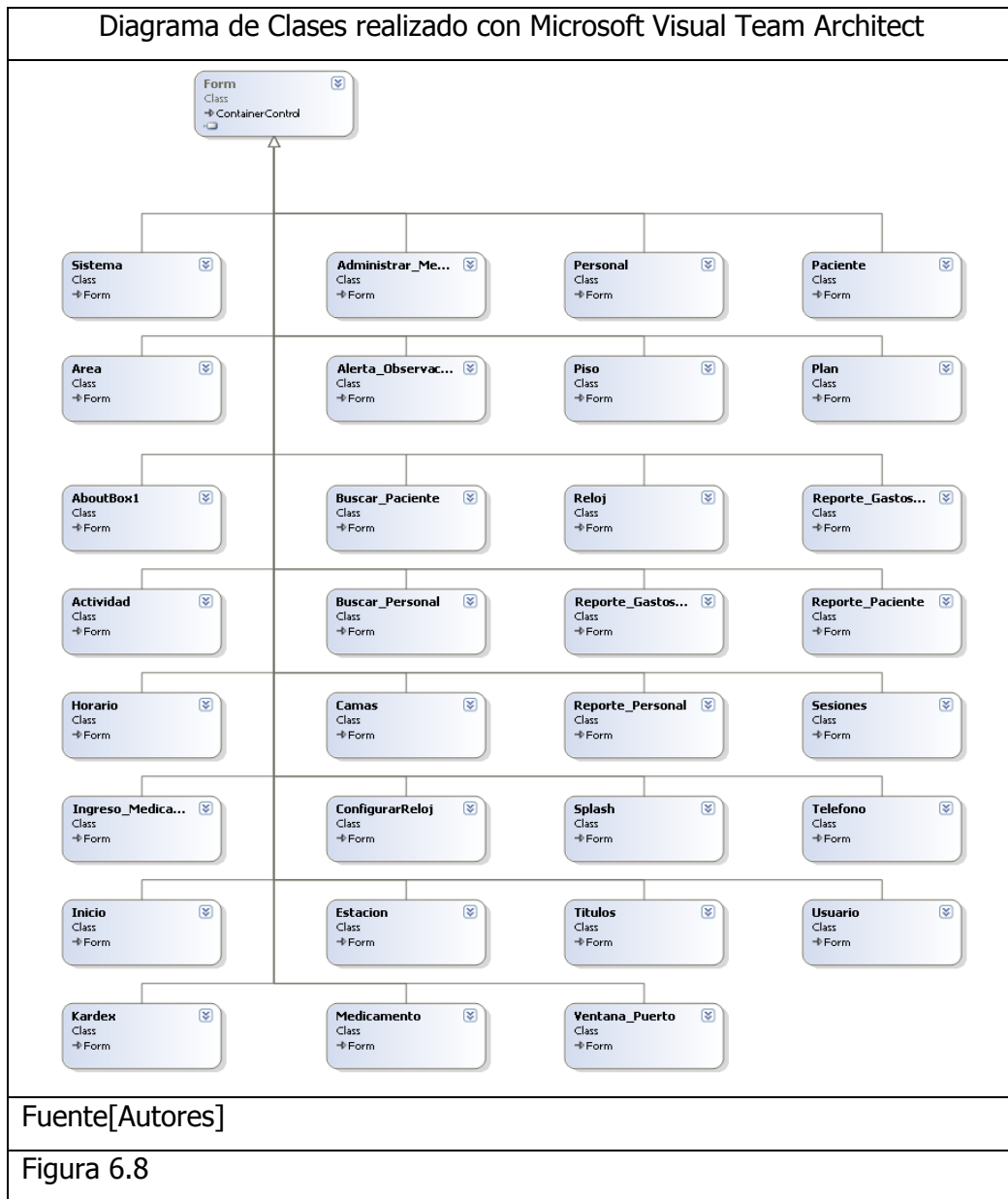
Algunas notaciones y procesos, sobre todo la combinación UML/RUP animan a una actividad de modelado de experiencia de usuario, basada en definir pantallas, formas y campos en sentido abstracto antes de codificarlos en cualquier lenguaje. Esto puede suponer algún beneficio limitado durante la fase de análisis, pero no resulta útil, sobre todo en diseño .NET, cuando se proporcionan herramientas más adecuadas.

### Diagrama General de Clases realizado con Microsoft Team Architect



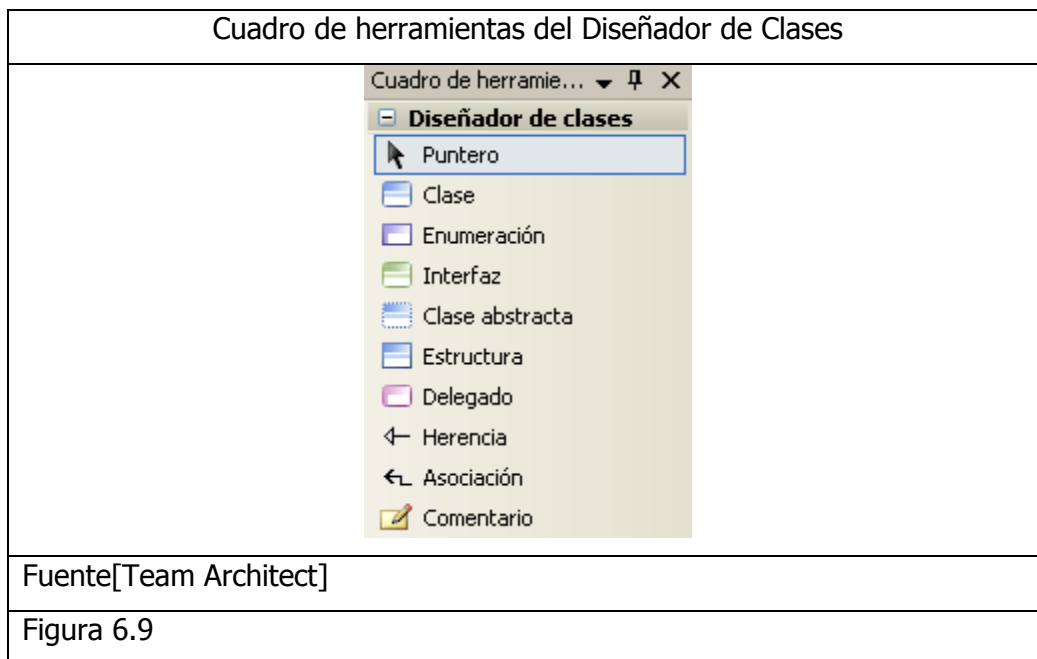
Fuente[Autores]

Figura 6.7



En las figuras 6.7 y 6.8 se pueden observar las relaciones de herencias de todas las clases que conforman el Sistema de Monitoreo de pacientes para salas de hospitalización, del Hospital San Vicente de Paúl.

En la siguiente figura se pueden observar las herramientas del diseñador de clases.



A continuación describimos la notación del Diseñador de Clases:

**Diseñador de clases.**

Clase.- Proporciona una breve representación, en forma de diagrama, de una clase en código; esta se sincroniza con el código de forma que los cambios de código se reflejan en el diagrama y viceversa.

Enumeración.- Proporciona una representación en forma de diagrama, de una enumeración en código.

Interfaz.- El tipo interfaz proporciona una representación en forma de diagrama de una interfaz en código.

Clase Abstracta.- Proporciona implementaciones concretas de algunos métodos.

Estructuras.- Proporciona una representación, en forma de diagrama, de una struct (tipo de valor) en código, con compartimientos para mostrar campos, métodos y demás.

Delegación.- Proporciona una representación de un método delegado, con los parámetros delegados mostrados como miembros.

Módulo.- Este tipo representa un módulo de Visual Basic, no disponible en C#.

Herencia.- Representa una relación o conexión entre tipos. La relación conecta un tipo derivado a un tipo base, y es equivalente a la relación de generalización de UML.

Asociación.- Permite que cualquier miembro se muestre dentro de un tipo o como una asociación a otro tipo, de forma completamente intercambiable según se prefiera. A diferencia de UML, las formas de asociación más fuertes no se distinguen en los diagramas, puesto que no se distinguen en código.

Comentario.- Es simplemente una nota de texto libre que añade claridad a sus diagramas.

### **6.3.7. Diseño de la placa electrónica**

Se compone de 20 pulsadores que simulan las camas, hay un pulsador que hace la función de reset que está conectado al pin número 1 del PIC16F877A.

Pulsador.- Manejan señales digitales, lo que permite llevar al microcontrolador un voltaje de 5 voltios (1 lógico) o 0 voltios (0 lógico)

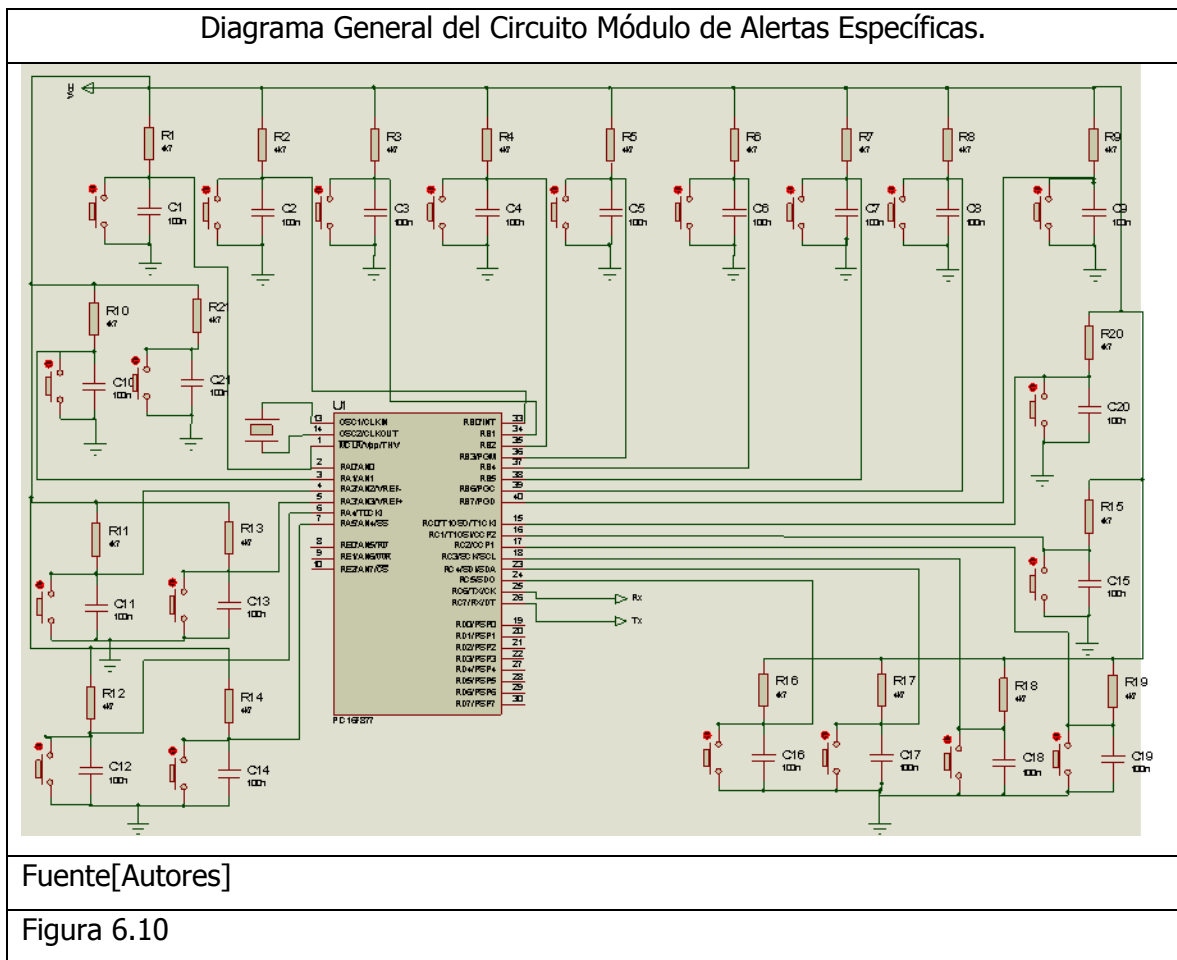
El pulsador está compuesto de:

- Resistencia.- Limita la corriente de ingreso al microcontrolador, 10 k $\Omega$ .
- Capacitador Cerámico.- (0,01 $\mu$ f) Elimina rebotes al momento de bajar o subir una señal digital.
- Pulsador.- Es un dispositivo que permite el paso o no de corriente.
- Cable de conexión con el microcontrolador es el encargado de enviar las señales que el PIC recibe por cada uno de los pines.

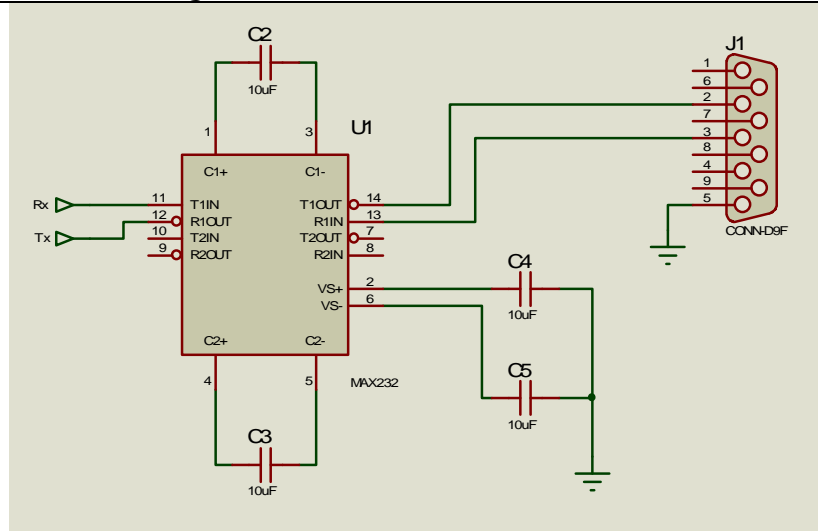
- VCC.- (voltaje de corriente continua).- Es el cable, que permite la conexión que envía 5 voltios.
- Para que haya cualquier voltaje debe haber 2 cables uno para el voltaje y el otro tierra.
- Cristal de cuarzo.- Nos permite dar la velocidad de operación del microcontrolador y es de 4Mhz.

El Microcontrolador PIC16F877A se encarga de recibir los datos de los pulsadores procesarlos y enviarlos mediante el MAX232 a la base de datos del computador. El PIC está programado solo para enviar datos.

La grabación del Microcontrolador PIC16F877A, se efectuó a través del software IC PROC.



### Diagrama de comunicación Serial RS-232.



Fuente[Autores]

Figura 6.11

El MAX232 convierte los niveles TTL (0 a +5 V) a nivel de energía RS-232 (cerca de +10 y -10 V) y viceversa.