

**Universidad Nacional Autónoma de Nicaragua  
UNAN-LEON  
Facultad de Ciencias y Tecnología**



**Tesis para optar al Título de Ingeniería en Sistema de Información.**

**TEMA: AUTOMATIZACIÓN DEL BANCO DE SANGRE DEL HOSPITAL  
H.E.O.D.R.A. DE LA CIUDAD DE LEÓN.**

**Elaborado por:**

-  **Br. Dixon José Rodríguez Sanchez.**
-  **Br. Isayra Hernández Reyes.**
-  **Br. Heriberto José Méndez Díaz.**

**Tutor:**

**Msc. Ernesto Espinoza Montenegro.**

**León – Nicaragua ,7 de noviembre del 2008**



## **Agradecimiento**

A DIOS por darme la fuerza, conocimiento, sabiduría y permitirme llegar hasta el final de esta etapa del camino de la enseñanza y el saber.

A mi Madre María Jesús Reyes y Abuelo Ricardo Reyes que siempre me apoyaron y me dieron el ejemplo de que con esfuerzo y tenacidad podía ser una persona útil en la vida.

A mi familia que de una u otra manera me dieron su apoyo.

A todos mis amigos que me apoyaron y me brindaron su ayuda incondicional.

A todos ellos, Muchas Gracias.

**Isayra Engracia Hernández Reyes.**



## Dedicatoria

A:

María Jesús Reyes.  
Mi Madre

A:

Ricardo Reyes.  
Mi abuelo

Por apoyarme en los momentos difíciles de mi vida.  
**Isayra Engracia Hernández Reyes.**



## **Agradecimiento**

A nuestro padre celestial por darme la fuerza, conocimiento, sabiduría.

A mi familia por haberme dado su apoyo.

A todos mis amigos, que me brindaron su amistad y ayuda incondicional.

A todos ellos, Muchas Gracias.  
**Heriberto José Méndez Díaz.**



## **Dedicatoria**

A:  
Rosa Argentina Díaz  
Mi Madre

A:  
Fundación Amigos de Holanda

A:  
Mis hermanos y demás familiares.

Por apoyarme durante toda mi vida estudiantil.  
**Heriberto José Méndez Díaz.**



## **Agradecimiento**

A nuestro padre celestial por darme la fuerza, conocimiento, sabiduría.

A mi familia por haberme dado su apoyo.

A todos mis compañeros que me brindaron su amistad y ayuda incondicional.

A todos ellos, Muchas Gracias.  
**Dixon José Rodríguez Sánchez**



## Dedicatoria

A:

Amparo Sánchez.  
Mi Madre

A:

Toribio Vásquez Barrera.  
Mi padre

A:

Mis hermanos y demás familiares.

Por apoyarme en los momentos difíciles de mi vida.  
**Dixon José Rodríguez Sánchez.**



## Indice

<b>I.</b>	<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>II.</b>	<b>ANTECEDENTES .....</b>	<b>2</b>
<b>III.</b>	<b>JUSTIFICACIÓN .....</b>	<b>3</b>
<b>IV.</b>	<b>OBJETIVOS .....</b>	<b>4</b>
<b>V.</b>	<b>MARCO TEÓRICO.....</b>	<b>5</b>
5.1	Terminología de banco de sangre.....	5
5.2	Donantes de Sangre .....	7
5.3	Componentes Sanguíneos .....	12
5.4	Solicitudes transfusionales.....	13
5.5	Ciclo de vida clásico .....	14
5.6	Terminología de base de datos .....	15
5.7	¿Porque MySQL?.....	17
5.8	ODBC. ....	18
5.9	¿Porque Visual Basic?.....	18
5.10	Funcionamiento de la Aplicación .....	19
5.11	Proyección del Sistema.....	20
<b>VI.</b>	<b>ANÁLISIS .....</b>	<b>21</b>
6.1.	Especificación de Requisitos de Software.....	21
6.2.	Diseño Arquitectónico .....	31
6.3.	Diagrama Entidad Relacion .....	32
6.4.	Descripción de los atributos del Diagrama Entidad Relación .....	33
6.5.	Esquema Relacional(Parte 1) .....	35
6.6.	Esquema Realcional (Parte 2) .....	36
6.7.	Diagrama de Flujo de datos.....	37



---

---

<b>VII.</b>	<b>DISEÑO METODOLÓGICO .....</b>	<b>40</b>
<b>VIII.</b>	<b>DIAGRAMA DE GANT .....</b>	<b>42</b>
<b>IX.</b>	<b>CONCLUSIÓN.....</b>	<b>43</b>
<b>X.</b>	<b>RECOMENDACIÓN .....</b>	<b>44</b>
<b>XI.</b>	<b>BIBLIOGRAFÍA.....</b>	<b>45</b>
<b>XII.</b>	<b>ANEXOS.....</b>	<b>46</b>



## I.Introducción

En la actualidad la tecnología evoluciona agigantadamente por lo que nos hemos visto obligados a sustituir unas herramientas por otras. Ejemplo de ello son las filmas o diapositivas que han sido reemplazadas por los videos, los cuales muestran la realidad de forma vivida; Las máquinas de escribir y los libros donde se guardaba información han sido sustituidos por computadoras.

El banco de sangre del hospital H.E.O.D.R.A presenta una difícil labor en cuanto al guardado de datos y generación de informes. Debido a que los datos son manejados en libros, los cuales son introducidos manualmente, y los informes se realizan usando programas como Excel y Word, lo que genera un desorden y pérdida de tiempo en la búsqueda de un archivo o creación de uno nuevo.

Nosotros como estudiantes de Ingeniería de Sistema participaremos en el desarrollo de un sistema que facilite el manejo de la información al personal del banco de sangre acerca de las solicitudes realizadas por las diferentes salas del hospital y así como donaciones de sangre realizadas al mismo, mejorando la calidad del servicio a sus pacientes.

Este software llevará por nombre: Banco de Sangre, el cual deberá recepcionar datos de solicitudes y donantes de sangre, almacenarla en una base de datos y acceder a ella a través de diferentes interfaces graficas, para posteriormente ser procesada y generar informes.

Para llevar a cabo el desarrollo de dicha aplicación será necesario especificar los requisitos de software que el sistema debe cumplir, también utilizaremos MYSQL como gestor de base de datos ya que es un software potente y de libre distribución.



## II. Antecedentes

En muchos países del mundo se han desarrollado sistemas de automatización para bancos de sangre, tales como España, Cuba, EEUU y Venezuela quien el día 9 de agosto del 2006 puso en funcionamiento el sistema automatizado de operación e información del Banco Municipal de Sangre, fase final de un proyecto iniciado en el año 1992. La ventaja de este sistema automatizado es que se podrán obtener y verificar rápidamente los datos de los donantes en diversas unidades.

En Nicaragua no existe actualmente ningún precedente de sistemas automatizado para llevar el control de la información de los Bancos de Sangre de los diferentes hospitales, es por esta razón que desarrollaremos un sistema de información que automatice las peticiones de sangre hechas por el paciente y las diferentes salas del hospital H.E.O.D.R.A de la ciudad de León.

El control de la información en el banco de sangre del hospital H.E.O.D.R.A en la actualidad es realizado manualmente, guardando los datos en libros, siendo este un trabajo pesado ya que se realizan 700 preparaciones de sangre por mes, previo a la transfusión de sangre al paciente, las que oscilan alrededor de 500 por mes.

Además de llevar un control de solicitudes al banco de sangre del hospital también se maneja un control de la sangre que ingresa a dicho hospital proveniente del Centro Nacional de Sangre (CNS) Cruz roja Managua.



### III. Justificación

Este proyecto nace de la necesidad de crear un sistema que facilite el control de la información al personal del banco de sangre del hospital H.E.O.D.R.A, ya que actualmente no cuenta con un sistema automatizado que permita llevar un control de los datos.

A raíz de esta problemática expuesta anteriormente, desarrollaremos un sistema de automatización para el banco de sangre de este hospital que permita dar soluciones a los problemas que presentan en cuanto al guardado y actualización de datos.

Este proyecto beneficiará de forma directa a:

1. Pacientes: Brindándole la mejor calidad de servicio, reduciendo el tiempo de espera a un 50% del actual.

#### 2. Personal del Banco de Sangre del H.E.O.D.R.A:

##### ***Mejorando la calidad del servicio ofertado:***

Trabajo: (Simplificación de actividades médicas y administrativas) reduciendo el tiempo de búsqueda y el guardado de información

Costo: Se reducirá la compra de materiales de oficina (Libros, lápices, borrador, regla, etc.)

##### ***Mejor control:***

- Bolsas de sangre disponibles en el inventario.
- Egresos de sangre a las distintas áreas.
- Información actualizada en el momento.
- Reportes generados en forma oportuna y veraz.

Este proyecto también beneficiará de manera indirecta a familiares del paciente y al hospital mismo con la mejora de calidad de servicios mencionados anteriormente.



## **IV.Objetivos**

### **4.1 Objetivo General**

- Desarrollar un software que automatice el control de la información de los donantes y solicitantes de sangre de las diferentes salas del hospital H.E.O.D.R.A de la ciudad de león.Objetivos Específicos

### **4.2 Objetivos Especificos**

- Desarrollar un sistema de automatización utilizando sistemas de base de datos de libre distribución.
- Desarrollar gráficas de usuario que proporcionen el control de las solicitudes de sangre realizadas al CNS por parte del banco de sangre del H.E.O.D.R.A. , así como las solicitudes realizadas por los pacientes o por las diferentes salas al banco de sangre del hospital.
- Llevar el control de las entradas y salidas de bolsas de sangres, así como la fecha de expiración de las mismas.
- Generar reportes de transfusiones realizadas por salas y grupos sanguíneos que se realizan por mes.
- Agilizar la atención a los solicitantes y donantes de sangre reduciendo el tiempo de espera en un 50 % del realizado actualmente



## V.Marco Teórico

### 5.1 Terminología de banco de sangre

#### **Bancos de sangre**

Los bancos de sangre son establecimientos, legalmente autorizados, donde se realizan los procedimientos necesarios para la utilización de la sangre humana y sus derivados con fines terapéuticos y de investigación los cuales deben estar debidamente habilitados por el ministerio de salud.

Para garantizar el funcionamiento y desarrollo científico-técnico de los bancos de sangre estos deben ser entes descentralizados administrativa y financieramente para que le permita la autosostenibilidad e incorporación de una nueva tecnología.

Por su capacidad científico-técnico, el tipo de actividad que realizan y su grado de complejidad los Bancos de sangre estarán en tres categorías:

#### **Banco de sangre A.**

Donde se efectúa la promoción, extracción, fraccionamiento, procesamiento, pruebas pretransfusionales, almacenamiento y transporte de hemocomponentes.

#### **Bancos de sangre B.**

Donde se efectúan la extracción, fraccionamiento, procesamiento, pruebas pretransfusionales, almacenamiento de hemocomponentes.

#### **Centros transfusionales.**

Donde se efectúan la transfusión, pruebas pretransfusionales, recepción y almacenamiento de hemocomponentes.

#### **Responsabilidad y autoridad.**

Los servicios de bancos de sangre definirán y documentarán la responsabilidad, la autoridad y las relaciones del personal que dirige, realiza y verifica la calidad en los servicios del Banco de Sangre, para asegurar que la recolección, procesamiento y transfusión de sangre y componentes sanguíneos y otros servicios similares se realicen de acuerdo a los requerimientos específicos.

#### **Plan de contingencia.**

Todo Banco de Sangre deberá establecer y desarrollar planes de contingencia para situaciones de desastres naturales o catástrofes causadas por el hombre.



### **Bioseguridad.**

Implantar todas las medidas de bioseguridad que protejan al donante, al receptor y al personal relacionado con la actividad.

### **Sistemas de Calidad.**

Los bancos de sangre deben contar con un sistema de calidad debidamente documentado y permanentemente actualizado.

Este sistema de calidad es el conjunto de elementos que define la estructura, organización, responsabilidades, sistemas documentales y medios necesarios para la implantación y gestión de la calidad basada en los siguientes aspectos:

- Organización de los recursos humanos y técnicos.
- Documentación de los procesos.
- Procedimientos e instrucciones de trabajo
- Valoración de los resultados y mediciones de la calidad de los procesos.
- Auditorías internas de la calidad.
- Calibración de los instrumentos empleados.
- Capacitación de los recursos humanos.
- Funcionamiento del comité de calidad.

### **Recursos Humanos:**

#### **•Cargo de Dirección**

▪ Los Bancos de sangre estarán bajo la dirección de profesionales de la salud en el siguiente orden de prioridad:

▪ *Para los Bancos de sangre de categoría A:* Médico Especialista( Hematólogo, Inmunólogo, Patólogo Clínico, Capacitados en Medicina Transfusional).

▪ *Para los Bancos de sangre de categoría B y Centros Transfusionales:* Médico Especialista( Hematólogo, Inmunólogo, Patólogo Clínico, Médico Internista, Médico General, Licenciado en Bioanálisis Clínico o Tecnólogo Médico, capacitados en Medicina Transfusional).

#### **•Personal Profesional y Técnico que labora en el laboratorio**

El Personal que labora en Bancos de Sangre deberá recibir educación continua acorde al desarrollo científico-técnico del área de banco de sangre y estará sujeto a evaluaciones periódicas, que garanticen el adecuado cumplimiento de las normas técnicas-administrativas.



## •Personal Administrativo

Los Bancos de Sangre **A Y B** deberán tener suficiente personal como para atender los aspectos administrativos del servicio, recursos humanos, el tratamiento y la notificación de datos, promoción de donantes, planificación, las gestiones, transporte y relaciones públicas.

En los centros Transfusionales se contará con los recursos humanos propios que garanticen tanto el aspecto científico-técnico y administrativo y el cumplimiento de estos estándares.

### 5.2 Donantes de Sangre

La donación se rige por una serie de principios médicos y éticos, plasmados en disposiciones legales, orientados a evitar perjuicios los y efectos no deseados al receptor.

Para la donación se deberá contar con un sistema de orientación claramente definido, donde se señale confidencialidad, registros sistemáticos de cada donación, recomendaciones post-donación, plan de emergencia en caso de reacción a la extracción y competencia del personal que atiende a los donantes.

#### ***Requisitos para la donación.***

Todo donante deberá identificarse adecuadamente. Todo programa educativo que pretenda estimular la donación tiene que informar al posible donante de las condiciones mínimas para poder donar y de las conductas de riesgo que impedirían esta donación, explicando el procedimiento de la donación, la venipunción, cuidados post-donación, los análisis que se realizarán a la unidad y las razones para estas determinaciones.

#### ***Selección del donante.***

El donante será seleccionado a través de una ficha de selección en que consten los antecedentes, examen físico y prueba de laboratorio. La entrevista y consejería deberá ser realizada con ética profesional, por personal entrenado y en un ambiente privado.

La ficha de selección del donante es una declaración jurada con firma y huella digital del donante y debiera ser firmada por el personal que lo atendió.

#### ***Autoexclusión***

Se deberá brindar a cada donante la oportunidad de indicar confidencialmente que la unidad recolectada puede ser inadecuada para transfusión.

El donante deberá tener la opción de autoexcluirse para ello se facilita una ficha de autoexclusión en la que se indican los aspectos básicos de aceptación o rechazo del donador.



### **Criterios para la Protección del Donante**

En el día de la donación con supervisión de una persona calificada y concedora de las fichas de selección del donante, se evaluarán los antecedentes y el estado actual del donante para determinar si se le puede realizar la extracción sin causar perjuicio.

#### **Enfermedades**

Deberán ser excluidos los donantes potenciales con enfermedades hematológicas, cardíacas, hepáticas o pulmonares, autoinmunes, hipertiroidismo, asmáticos severos, diabéticos, psoriásicos tratados con psoralenos y los que tienen antecedentes de: cáncer, tendencia al sangrado hormonal, convulsiones o pérdida de conocimiento.

#### **Medicamentos**

La terapéutica con medicamentos puede hacer que la donación sea perjudicial al donante o receptor, por cuyo motivo deberá evaluarse el tipo de droga y el cuadro clínico que determinó la indicación

La ingestión de ácido acetilsalicílico y otras drogas que alteren la función plaquetaria dentro de los tres días previos a la donación excluirá la unidad para ser destinada a la preparación de plaquetas como única frente para un receptor.

#### **Edad**

El donante de sangre o componentes sanguíneos deben tener entre 17 y 65 años de edad. Los donantes no incluidos entre estos límites, para su aceptación, deberán ser previamente evaluados por un médico y en el caso de los menores a 17 años se exigirá una autorización escrita del responsable legal

#### **Pulso**

El pulso debe ser rítmico y su frecuencia no deberá ser menos a 50 ni mayor a 100 latidos por minuto. En los atletas con alta potencia al ejercicio, una frecuencia menor puede ser aceptable, bajo previa evaluación médica.

#### **Presión Arterial**

La tensión sistólica no deberá ser mayor de 180mm Hg y la tensión Diastólica no deberá ser menor de 60 mm Hg ni mayor de 100 mm Hg. Los donantes potenciales con valores de presión arterial no comprendidos dentro de los valores mencionados solo podrán ser aceptados con la aprobación de un médico calificado.

#### **Embarazo y Menstruación**

Las donantes embarazadas no pueden ser aceptadas hasta las 6 semanas posteriores al parto. Así mismo no se aceptan como donantes las mujeres durante el periodo de lactancia.



La donación antológica será aceptada si se cuenta con la aprobación del médico obstetra de la gestante y del médico del banco de sangre.

Durante el periodo menstrual normal se puede donar sangre. La hipermenorrea u otras patologías de la menstruación deberán ser evaluadas por el profesional médico.

### ***Alergias***

El donante alérgico solamente será aceptado si en el momento de la donación no presenta síntomas, excepto aquellos que padezcan enfermedades atópicas graves, asma bronquial severa o alergia a drogas, los cuales serán rechazadas en todo momento. Los que estuvieran recibiendo tratamiento desensibilizante postergarán la donación hasta 72 horas después de la última aplicación

### ***Recolección y procesamiento de sangre y componentes***

La extracción de sangre o de algún componente, se llevará a cabo por el personal capacitado, quien deberá realizar los controles vitales al donante durante la extracción en cualquiera de los sitios que estén establecidos para la misma.

### ***Recolección de la sangre del donante***

- Una vez finalizada la etapa de selección, la entrevista médica y la exploración física del donante, pasará a la sala de extracción, donde se le entregará toda la documentación y la identificación del donante.
- El responsable del banco de sangre o el encargado del puesto de extracción, vigilará y supervisará al personal profesional o técnico que lleve a cabo los procedimientos de recolección.
- La extracción de sangre deberá ser realizada con condiciones asépticas mediante una sola venipuntura, con un sistema de recolección cerrado y estéril.

### ***Identificación del donante***

- Antes de la extracción se deberá realizar la comprobación de la identidad del donante con la ficha y numeración asignada para la donación, luego se procede a identificar las bolsas y etiquetar los tubos "**Pilotos**" destinados para los análisis.
- El nombre del donante no deberá constar en la etiqueta de la unidad de sangre o componentes, solo podrán imprimirse las iniciales de los nombres y apellidos del donante.

### ***Tiempo de llenado de bolsa***

- Durante el llenado deberá vigilarse que el flujo sea constante, al igual que la homogenización del anticoagulante.



- La duración de la extracción no deberá exceder a 12 minutos para no afectar negativamente la calidad de los componentes que se
- obtienen del fraccionamiento de la bolsa y ocasionar reacciones adversas a la donación.

### ***Volumen de Extracción***

- En cada extracción el volumen de sangre extraído al donante deberá de ser de:  $50 \pm 10\%$  (405-495 ml).
- Si se ha recogido un volumen entre 300-405 ml de sangre en un volumen de anticoagulante calculado para 450 ml, los glóbulos rojos desplasmalizados podrán ser usados para transfusión pero deberán ser etiquetados como: **“Unidades de bajo Volumen”**. No se deberán de preparar otros componentes a partir de las unidades de bajo volumen.

### ***Muestras de Laboratorio***

La guía de la bolsa de extracción deberá ser llenada con sangre anticoagulada y por cada numeración de la guía deberá realizarse un segmento para futuras pruebas de compatibilidad transfusional. Estos segmentos deberán ser separables de la bolsa sin pérdida de su esterilidad. Los tubos Pilotos para obtener sangre con destino a las pruebas de laboratorio, serán rotulados previa a la extracción. Las muestras serán recogidas directamente del donante por el extremo de la guía conectada a la vena, luego de haber sido separada de la bolsa con sangre. Finalmente volverán a corroborarse los rótulos de la bolsa y el tubo.

### ***Reacciones en Donante***

- Se dispone de instrucciones específicas concernientes a los procedimientos a ser seguidos en cuanto a la prevención y tratamiento de las reacciones en donantes.
- Se deberá contar con los medicamentos y equipos necesarios para brindar asistencia médica al donante que presente reacciones adversas.
- Se le brinda al donante algún alimento y bebida luego de la donación al retirarse del ámbito de la institución.

### ***Procesamiento de Sangre y Componentes***

- Los Componentes Sanguíneos son preparaciones obtenidas por separación a partir de unidades de sangre total que son utilizadas para transfusión.
- La esterilidad del componente deberá mantenerse durante el procesamiento, mediante el empleo de métodos asépticos, equipo y soluciones estériles, libres de pirógenos.



### **Investigación de la Sangre del Donante**

Deberán realizarse obligatoriamente las siguientes pruebas:

- Determinación del grupo ABO: La tipificación del grupo ABO deberá hacerse en tubo realizando la prueba celular con antisueros específicos (Anti- A y Anti -B) y la sérica frente a eritrocitos A1 y B. La sangre no deberá ser usada para transfusión, a menos que cualquier discrepancia haya sido resuelta.
- Determinación del Factor Rho(D): Este factor será determinado enfrentando los eritrocitos con suero anti Rho(D), si la reacción fuese negativa se deberá efectuar la técnica para la detención de Du.
- La esterilidad del componente deberá mantenerse durante el procesamiento, mediante el empleo de métodos asépticos, equipo y soluciones estériles, libres de pirógenos.
- Cuando la prueba D o Du resultare positiva, la sangre será rotulada "Rh Positivo". Cuando ambas pruebas resulten negativas, la sangre será rotulada como: "Rh Negativo".
- Se utiliza un control Rho(D) de los hematíes que será el específico del proveedor comercial o una suspensión de albúmina del 2 al 3%.

### **Investigación de anticuerpos irregulares.**

- En los donantes se deberá investigar la presencia de anticuerpos séricos irregulares, en caso de ser positivos se deberá identificar anticuerpos clínicamente significativos. Las unidades de sangre que contuvieran tales anticuerpos deberán ser procesadas en componentes con mínimas cantidades de plasma.

### **Datos de la Bolsa durante la fase de recolección**

- Identificación numérica o alfa-numérica.
- Nombre y cantidad de la mezcla preservadora.
- Fecha de la recolección.
- Consignar claramente si la donación fue antóloga o dirigida.

### **Datos de la etiqueta en la fase de fraccionamiento**

- Identificación de la institución recolectora.
- Nombre y Volumen aproximado del componente.
- Grupo ABO
- Rh en los Componentes que lo requieren.
- Identificación numérica o alfa-numérica de la donación
- Fecha de extracción y de vencimiento y la hora en los casos que aplicare el horario de vencimiento.



- Resultado de las pruebas de detección de cada agente infeccioso tamizado
- En la etiqueta deberá quedar consignado “No agregar medicamentos”

### **Contenido de etiquetas de las preparaciones especiales.**

- En los componentes liberados en forma de “pool”(Concentrado de plaquetas, crioprecipitados y sangre entera reconstituida) además de las especificaciones antes descritas se deberá informar:
  - Nombre del Componente, especificar que se trata de un “pool”, cantidad de unidades y su correspondiente identificación.
  - Grupo ABO y RH de la unidades “pool”, volumen final, fecha, horario de vencimiento.

## **5.3 Componentes Sanguíneos**

### **Componentes Eritrocitarios**

▪ **Paquete Globular:** Son los Eritrocitos remanentes luego de remover el plasma de una unidad de sangre total sedimentada o sometida a centrifugación. Los eritrocitos se pueden separar del plasma en cualquier momento antes de la fecha de expiración de la sangre.

▪ **Glóbulos Rojos Lavados:** Son eritrocitos que se obtienen después de efectuar lavados con solución isotónica de cloruro de sodio (NaCl), con la finalidad de eliminar la mayor cantidad posible de plasmas según el método usado, la preparación podrá contener cantidades variables de leucocitos y plaquetas de la unidad original.

▪ **Glóbulos Rojos pobres en Leucocitos:** Son los glóbulos rojos preparados por un método que asegure la retención de por lo menos el 80% de los eritrocitos originales. Cuando estuvieren destinados a la prevención de reacciones Transfusionales febriles no hemolíticas, deberán ser preparados por un método que reduzca el número de leucositos en el componente final a menos de  $5 \times 10^6$ .

### **Componentes plasmáticos**

▪ **Plasma Fresco Congelado:** Es el plasma separado de una unidad de sangre total y almacenado a temperatura de  $-20^{\circ}\text{C}$  o inferior, dentro de las 8 horas de realizada la recolección del donante.



▪ **Plasma Simple:** Es el plasma separado de una unidad de sangre total antes de los 5 días después de la fecha de expiración. Alternativamente el plasma simple puede resultar del plasma fresco congelado que hubiere superado su fecha de expiración o que hubiese sido

desprovisto del crioprecipitado. El plasma simple podrá ser almacenado en estado líquido a  $4 \pm -2^{\circ}\text{C}$  o inferior.

▪ **Crioprecipitado:** Es la fracción insoluble del plasma frío, obtenida a partir del plasma fresco congelado. Este deberá ser descongelado de  $2-6^{\circ}\text{C}$  inmediatamente después de completado el descongelamiento, el plasma deberá ser centrifugado a la temperatura de  $2-6^{\circ}\text{C}$  y separado del material insoluble en frío bajo condiciones estériles. El crioprecipitado resultante deberá ser recongelado dentro de la hora posterior a su obtención. El producto final deberá contener como mínimo 80 unidades internacionales de factor VII por unidad en por lo menos el 75% de las unidades evaluadas.

#### 5.4 Solicitudes transfusionales

Todas las transfusiones de sangre o componentes deberán ser prescritas por un médico en un formato de solicitud debidamente firmado con la información correcta para la identificación del receptor.

El formulario impreso deberá constar por lo menos de lo siguiente:

- Fecha de solicitud.
- Servicio donde se encuentra el paciente.
- Fecha y hora que se requiere la sangre.
- Nombres y Apellidos del Receptor.
- Edad y Sexo del paciente.
- Número de Expediente del paciente.
- Número de unidades de sangre y productos sanguíneos requeridas.
- Razón por la cual se solicita la transfusión.
- Grado de urgencia de la Solicitud.
- Nombre, firma, código del médico que solicita la sangre.
- Nombre del responsable de la extracción de la muestra.
- Grupo sanguíneo del paciente si se conoce.

#### ***Muestras de sangre***

▪ Al personal de servicio transfusional u otra persona capacitada del laboratorio deberá extraer las muestras de sangre del receptor identificándolo pacientemente.

▪ Antes de la toma de muestra se deberá corroborar que los datos de la solicitud de transfusión coinciden con los datos del paciente interrogando la identidad del mismo



paciente. En caso en que el estado de conciencia del paciente no lo permitiera, se deberá identificar los registros preguntando al personal de enfermería o a familiares que estuvieren presentes.

- Si los datos del paciente son diferentes a los de solicitud no se deberá tomar la muestra y se informará a quien corresponda.
- Se deberá obtener una muestra en forma aséptica, de preferencia en tubos plásticos al vacío con o sin anticoagulante según procedimiento.
- Si existe una nueva solicitud de sangre con un plazo superior a las 72 horas respecto a la anterior, se deberá requerir otra muestra para efectuar las pruebas de compatibilidad o si la muestra anterior es insuficiente para efectuar las pruebas pre-transfusionales.
- Todos los tubos empleados son rotulados en el momento mismo de la extracción con:
  - Nombre y Apellidos del Receptor.
  - Número de Expediente.
  - Nombre de la Sala.
  - Fecha de Extracción.
  - Firma del Febotomista.

## 5.5 Ciclo de vida clásico

Llamado también “modelo de cascada”. Este ciclo de vida exige un enfoque ordenado y secuencial del desarrollo del software que comienza en el nivel del sistema y progresa a través del análisis, diseño, codificación, prueba y mantenimiento.

**Ingeniería y análisis del sistema.** El trabajo comienza estableciendo los requisitos de todos los elementos del sistema y luego asignando algún subconjunto de estos requisitos al software. Abarca los requisitos globales a nivel del sistema con una pequeña cantidad de análisis y diseño a un nivel superior.

**Análisis de los requisitos del software.** El ingeniero del software debe comprender el ámbito de la información del software, así como la función, el rendimiento y las interfaces requeridas. Los requisitos tanto del sistema como del software se documentan y se revisan con el cliente.

**Diseño.** Es realmente un proceso multipaso que se enfoca sobre cuatro atributos distintos del programa: la estructura de los datos, la arquitectura del software, el detalle procedimental y la caracterización de la interfaz. El diseño se documenta y forma parte de la configuración del sistema.

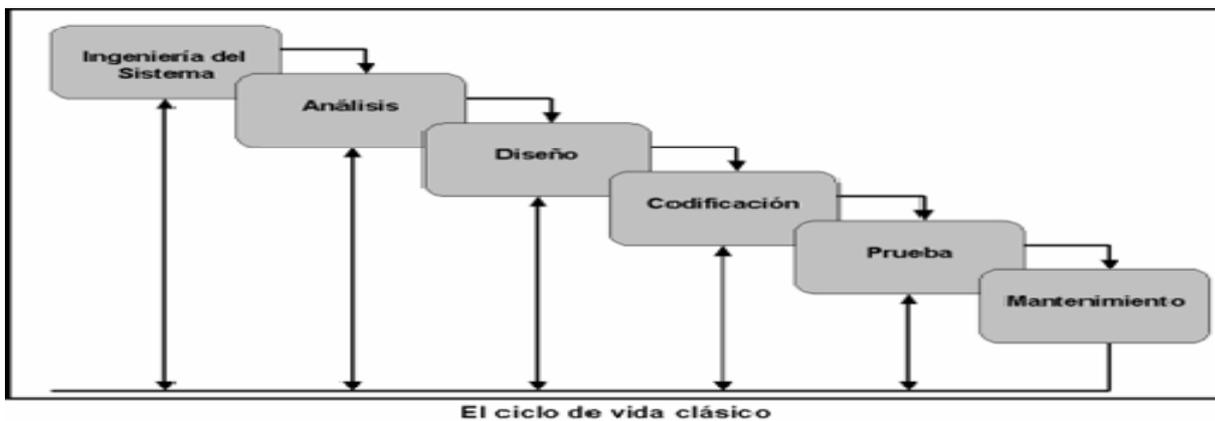
**Codificación.** El diseño debe traducirse en una forma legible para la máquina. El paso de codificación realiza esta tarea.

**Prueba.** Una vez que se ha generado el código, comienza la prueba del programa.



La prueba se centra en la lógica interna del software, asegurando que todas las sentencias se han probadas y en las funciones externas, realizando pruebas que aseguren que la entrada definida produce los resultados que realmente se requieren.

**Mantenimiento.** El software sufrirá cambios después de que se entregue al cliente. Los cambios ocurrirán debido a posibles errores que se hayan encontrado, a que el software debe adaptarse a cambios en el entorno externo, o debido a que el cliente requiera ampliaciones funcionales o de rendimiento. Posteriormente, ha de ser descartado y debe construirse el software real, con los ojos puesto en la calidad y en el mantenimiento.



## 5.6 Terminología de base de datos

### Dato

Podemos decir que un dato es una información que refleja el valor de una característica de un objeto real, sea concreto, abstracto o imaginario. Debe cumplir algunas condiciones, por ejemplo, debe permanecer en el tiempo. En ese sentido, estrictamente hablando, una edad no es un dato, ya que varía con el tiempo. El dato sería la fecha de nacimiento, y la edad se calcula a partir de ese dato y de la fecha actual. Además, debe tener un Significado y debe ser manipulable mediante operadores: comparaciones, sumas, restas, etc. (por supuesto, no todos los datos admiten todos los operadores).

### Base de datos

Podemos considerar que es un conjunto de datos de varios tipos, organizados e interrelacionados. Estos datos deben estar libres de redundancias innecesarias y ser independientes de los programas que los usan.

### SGBD (DBMS)

Son las siglas que significan *Sistema de Gestión de Bases de Datos*, en inglés DBMS, *DataBase Manager System*. En este caso, MySQL es un SGBD, o mejor dicho: nuestro SGBD.



## **Consulta**

Es una petición al SGBD para que procese un determinado comando SQL. Esto incluye tanto peticiones de datos como creación de bases de datos, tablas, modificaciones, inserciones, etc.

## **Redundancia de datos**

Decimos que hay redundancia de datos cuando la misma información es almacenada varias veces en la misma base de datos. Esto es siempre algo a evitar, la redundancia dificulta la tarea de modificación de datos y es el motivo más frecuente de inconsistencia

de datos. Además requiere un mayor espacio de almacenamiento, que influye en mayor coste y mayor tiempo de acceso a los datos.

## **Inconsistencia de datos**

Sólo se produce cuando existe redundancia de datos. La inconsistencia consiste en que no todas las copias redundantes contienen la misma información. Así, si existen diferentes modos de obtener la misma información y esas formas pueden conducir a datos almacenados en distintos sitios. El problema surge al modificar esa información, si sólo cambiamos esos valores en algunos de los lugares en que se guardan, las consultas que hagamos más tarde podrán dar como resultado respuestas inconsistentes (es decir, diferentes). Puede darse el caso de que dos aplicaciones diferentes proporcionen resultados distintos para el mismo dato.

## **Integridad de datos**

Cuando se trabaja con bases de datos, generalmente los datos se reparten entre varios ficheros. Si, como pasa con MySQL, la base de datos está disponible para varios usuarios de forma simultánea, deben existir mecanismos que aseguren que las interrelaciones entre registros se mantienen coherentes, que se respetan las dependencias de existencia y que las claves únicas no se repitan.

Por ejemplo, un usuario no debe poder borrar una entidad de una base de datos, si otro usuario está usando los datos de esa entidad. Este tipo de situaciones son potencialmente peligrosas, ya que provocan situaciones con frecuencia imprevistas. Ciertos errores de integridad pueden provocar que una base de datos deje de ser usable

Los problemas de integridad se suelen producir cuando varios usuarios están editando datos de la misma base de datos de forma simultánea. Por ejemplo, un usuario crea un nuevo registro, mientras otro edita uno de los existentes, y un tercero borra otro. El DBMS debe asegurar que se pueden realizar estas tareas sin que se produzcan errores que afecten a la integridad de la base de datos.



## 5.7 ¿Porque MySQL?

Para la realización de este proyecto utilizaremos como SGBD MySQL, este es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU.

MySQL fue creada por la empresa sueca MySQLAB, que mantiene el copyright del código fuente, así como también de la marca, sin embargo el software es de libre distribución.

Además de ser un software libre, hemos elegido este Gestor por las características que nos brinda, las cuales mencionamos a continuación:

1. MySQL nos brinda Acceso a las bases de datos de forma simultánea por varios usuarios y/o aplicaciones.
2. **Seguridad en forma de permisos y privilegios:** Determinados usuarios tendrán permisos para consulta o modificación de determinadas tablas. Esto permite compartir datos sin que peligre la integridad de la bases de datos, o protegiendo determinados contenidos.
3. **Potencia:** SQL es un lenguaje muy potente para consulta de bases de datos, usar un motor nos ahorra una enorme cantidad de trabajo.
4. **Portabilidad:** SQL es también un lenguaje estandarizado, de modo que las consultas hechas usando SQL son fácilmente portables a otros sistemas y plataformas. Esto unido al uso de C/C++ proporciona una portabilidad enorme.
5. **Escalabilidad:** Es posible manipular bases de datos enormes del orden de seis mil tablas por base y alrededor de cincuenta millones de registros por tabla y hasta 32 índices por tabla.
6. MySQL esta escrito en C y C++ probado con multitud de compiladores y dispone de APIS para muchas plataformas diferentes.
7. **Conectividad:** Permite conexiones entre diferentes máquinas con distintos sistemas operativos. Es corriente que servidores Linux o Unix, usando MySQL, sirvan datos para ordenadores con Windows, Linux, Solares, etc. Para ello se usa TCP/IP, tuberías o sockets Unix.
8. Permite manejar multitud de tipos para columnas.
9. permite manejar registros de longitud fija o variable.



## 5.8 ODBC.

El objetivo de ODBC (Open Database Connectivity) es hacer posible acceder a los datos de una base de datos desde cualquier aplicación, sin importar el sistema gestor de base de datos (SGBD) que se este utilizando, ODBC logra esto al insertar una capa intermedia llamada manejador de Bases de Datos, entre la aplicación y el SGBD, el propósito de esta capa es traducir las consultas de datos de la aplicación en comandos que el SGBD entienda. Para que esto funcione tanto la aplicación como el SGBD deben ser compatibles con ODBC, esto es que la aplicación debe ser capaz de producir comandos ODBC y el SGBD debe ser capaz de responder a ellos.

## 5.9 ¿Porque Visual Basic?

- Es un lenguaje de fácil aprendizaje pensado tanto para programadores principiantes como expertos, guiado por eventos, y centrado en un motor de formularios que facilita el rápido desarrollo de aplicaciones gráficas. Su sintaxis, derivada del antiguo BASIC, ha
- sido ampliada con el tiempo al agregarse las características típicas de los lenguajes estructurados modernos.
- Posee varias bibliotecas para manejo de bases de datos, pudiendo conectar con cualquier base de datos a través de ODBC (Informix, DBase, Access, MySQL, SQL Server, PostgreSQL, etc) a través de ADO.
- Es utilizado principalmente para aplicaciones de gestión de empresas, debido a la rapidez con la que puede hacerse un programa que utilice una base de datos sencilla, además de la abundancia de programadores en este lenguaje.
- La facilidad del lenguaje permite crear aplicaciones para Windows en muy poco tiempo. En otras palabras, permite un desarrollo eficaz y menor inversión en tiempo que con otros lenguajes.
- Permite generar librerías dinámicas (DLL) ActiveX de forma nativa y Win32 (no ActiveX, sin interfaz COM) mediante una reconfiguración de su enlazador en el proceso de compilación.
- Permite la utilización de formularios (Forms) tanto a partir de recursos (como en otros lenguajes) como utilizando un IDE para diseñarlos.
- Posibilidad de desarrollar y ejecutar aplicaciones de Visual Basic 6.0, Windows Vista sin realizar cambios en la mayoría de los casos pero no se logra aprovechar al máximo las características de este sistema como permite hacerlo Visual Basic 2005 o Visual Basic 2008.



## 5.10 Funcionamiento de la Aplicación

El funcionamiento de la aplicación será el siguiente:

El programa mostrará una ventana con diferentes opciones. Las opciones disponibles son las siguientes:

En el menú Registrar el usuario podrá ingresar datos del paciente correspondientes a:

- Solicitud
- Registrar Kardex
- Datos del Donante

En el menú ver el usuario podrá seleccionar cualquiera de las siguientes opciones:

- Solicitudes Pendientes
- Sangre Próxima a vencerse.
- Sangre Descartada.
- Sangre Vencida.
- Kardex.
- Datos del Donante.
- Datos de Prueba.
- Visualizar Base de Datos.
- Personal Registrado.
- Resultado no Entregados: estos pueden ser:
  - Coombs Directo
  - Coombs Indirecto
  - Sangre Preparada
  - Grupo y Rh.

En el menú Informe podrá seleccionar cualquiera de las siguientes opciones:

- Total de Transfusiones.
- Componentes por Sala.
- Consumo por Médico.
- Prevalencia de Grupo por Sala.
- Transfusiones por Servicio.
- Producto Descartado.

En el menú Herramientas el usuario podrá seleccionar cualquiera de las siguientes opciones:

- Copia de Seguridad
- Restaurar Copia de Seguridad

En el menú Ayuda el usuario podrá buscar información sobre el funcionamiento del Sistema.



### **5.11 Proyección del Sistema**

Este software se proyectará bajo el desarrollo de un sistema de escritorio debido a que le será útil solo al laboratorio del Banco de Sangre del H.E.O.D.R.A, pero en el transcurso

del tiempo se podrá cambiar este sistema por el desarrollo de un sistema en Red para que sea utilizado por todos los laboratorios del Centro Nacional de Sangre a nivel Nacional



## VI.Análisis

### 6.1. Especificación de Requisitos de Software

#### 1. Introduccion

##### 1.1. Propósito

Definición del conjunto de especificaciones de requisitos software que debe cumplir la aplicación: Sistema de Banco de Sangre para el hospital H.E.O.D.R.A-LEON, el cual se diseñará con el propósito de llevar un mejor control de la sangre que ingresa y egresa del Banco de Sangre de dicho hospital.

##### 1.2. Alcance

El nombre con el que se conocerá esta aplicación será Banco de Sangre.

El software realizará las funciones siguientes:

- Introducir datos de la sangre del donante.
- Introducir datos del paciente.
- Modificar datos del paciente.
- Implementar búsqueda.
- Eliminar registro de donantes.
- Emisión de los siguientes listados de información:
  - Listados del paciente a los que se aplicó una o varias transfusiones
  - Listado diario de la sangre disponible agrupadas por tipos.
  - Listado de sangre preparada.
  - Listado diario de la sangre con fecha de expiración próxima a vencerse.
  - Listado diario de la sangre descartada ya sea por: contaminación, fecha de expiración vencida, etc.

##### 1.3. Definición, acrónimos y abreviaturas

▪ **Nombre del paciente:** Especifica el nombre de la persona a la cual se realizará una transfusión, este será identificada como: nombre \_ paciente .

▪ **Número de expediente:** Representa el identificador único del paciente y lo denotaremos como: No\_Exp.

▪ **Prueba ABO:** Prueba que se realiza con la sangre del paciente al que se le realizará la transfusión para determinar el grupo al que pertenece que puede ser cualquiera de los mencionados a continuación: A, B, AB, O.

▪ **Sistema RH:** Prueba que tiene como objetivos determinar el signo del tipo al que pertenece ya sea positivo o negativo, esta tesis hará uso de los elementos que a continuación se detallan: a- Anti D: El cual denotaremos como :



- a. Anti\_d.
- b. Autocontrol: El cual se ará referencia como autocontrol.
- c. C.C.Coombs: Identificado como C\_C\_Coombs.

▪ **Interpretaciones:** Definimos el tipo de sangre al que pertenece el paciente, al que se realizará la transfusión, para ello se hará uso de los siguientes atributos que almacenarán esta información:

**1. Grupo:** Se especifica el grupo al que pertenece el paciente y se hará referencia como grupo.

**2. Rh:** Especifica el signo del grupo a que pertenece el paciente a este lo llamaremos Rh

▪ **Prueba cruzada mayor:** Constituye una parte de las pruebas de compatibilidad que consiste poner en contacto el suero de receptor con los glóbulos rojos de los donantes. Para esta prueba utilizaremos una variable que almacenará el resultado de este experimento a la cual haremos referencia con el nombre result\_mayor.

▪ **Prueba cruzada menor:** En este caso se pone en contacto el suero de la sangre del donante con los glóbulos rojos del paciente así mismo utilizaremos una variables que guardará dicho resultado y haremos referencia a esta variable con el nombre de result\_menor.

#### 1.4. Referencia

Informe obtenido como resultado de la entrevista realizada desde 01/10/07-19/11/07, a la directora del laboratorio del Banco de Sangre: Rosa Emelina Alonso y la responsable de dicho laboratorio Lic. Marcia Noguera.

#### 1.5. Visión general del producto

Primeramente se realizará una discusión general del producto que se desea desarrollar, para pasar posteriormente a estudiar cada uno de los requisitos específico individualmente.

## 2. Descripción General

### 2.1. El equipo en el que se desarrollará e implementará el producto final será

- BIOS: Phoenix - AwardBIOS v6.00PG
- Processor: Intel(R) Pentium(R) 4 CPU 3.20GHz (2 CPUs)
- Memory: 1024MB RAM
- DiscoDuro: 80GB
- System Model: HP Compaq dx2300 Microtower



## 2.2. Funciones del producto

El producto software debe contener todas las tareas que realizan manualmente el personal encargado del laboratorio de Banco de Sangre del H.E.O.D.R.A. de forma diaria las cuales son:

Cuando el CNS envía la sangre esta debe ser registrada en la Base de Datos, los principales datos que se introducirá se mencionan a continuación:

- Fecha de llegada.
- Código de Bolsa.
- Código de Guía.
- Tipo de sangre.

Al llegar un paciente al hospital que necesita de una transfusión inmediatamente, el usuario debe introducir los datos del paciente.

Si una unidad de sangre, es preparada para un determinado paciente, el usuario debe registrarla como tal y eliminarla de la lista de sangre enviada por el CNS que están esperando a ser preparada.

Si una unidad de sangre preparada no es utilizada por el paciente que la iba a utilizar, el usuario eliminará dicha sangre de la lista de sangre preparada y deberá registrarla nuevamente como sangre disponible para ser preparada para otro paciente.

El usuario también deberá registrar el hemoderivado de la sangre para llevar un mejor control de la misma.

Todos los días se ejecutará un proceso automático que listará todas las unidades de sangre próximas a vencerse con el objetivo de que sean usadas a la brevedad.

El usuario podrá disponer de los siguientes listados de forma diaria:

- Listado de pacientes a los que se le realizó una o varias transfusiones.
- Listado de sangre disponible agrupadas por tipo.
- Listado de sangre preparada.
- Listado de sangre descartada y el motivo.

## 2.3. Características de usuario

Los usuarios finales de la aplicación serán personas cuya experiencia informática es escasa, por lo cual el sistema incluirá ayuda para poder buscar información relacionada con el uso y manejo del software.



## **2.4. Restricciones Generales**

El lenguaje de programación utilizado será Visual Basic 6.0 siguiendo siempre los estándares de programación estructurada.

## **2.5. Suposiciones y dependencia**

Durante las entrevista realizadas a la Lic. Rosa Emelina Alonso directora del laboratorio de Sangre y la Lic. Marcia Noguera responsable de dicho laboratorio indicaron las posibilidades de desarrollar en un futuro un software que funcione en red y que varios host puedan acceder al mismo y de esta manera poder ahorrar tiempo y dinero en la introducción de la información.

## **3. Requisito Específicos**

### **3.1. Requisitos Funcionales**

#### **3.1.1. INTRODUCIR DATOS DE LA SANGRE DEL DONANTE**

##### **3.1.1.1. Especificación**

###### **3.1.1.1.1. Introducción**

Este proceso deberá realizar la captura de los datos correspondiente a la sangre del donante.

###### **3.1.1.1.2. Entradas.**

- Por pantalla:
- Número de guía.
- Código de bolsa.
- Fecha de ingreso.
- Fecha de Vencimiento.
- Tipo de sangre.

Datos proporcionados por el sistema.

- Referente a la Sangre:
- Fecha de llegada.
- Fecha de Vencimiento.
- Tipo de sangre.

###### **3.1.1.1.3. Proceso.**

Se mostrará la pantalla de introducción de datos al usuario, los datos necesarios a introducir serán:

- Número de guía: Es un dato obligatorio que representa la identificación única de una bolsa de sangre, el cual es asignado por la fábrica que la elaboró.



- Código de bolsa: Es un dato obligatorio ya que es una identificación única de la bolsa de sangre, dicho identificador es asignado por el CNS (Centro Nacional de Sangre).
- Fecha de ingreso: Indica la fecha de ingreso de una unidad de sangre al laboratorio del Banco de Sangre del H.E.O.D.R.A.
- Fecha de Vencimiento: Indica la fecha de expiración de una unidad de sangre.
- Tipo de sangre. Representa el tipo de sangre del donante.

#### **3.1.1.1.4. Salidas.**

Todos los datos mencionados correspondientes a la sangre del donante serán almacenados en la Base de Datos.

### **3.1.2. INTRODUCIR DATOS DEL PACIENTE**

#### **3.1.2.1. Especificación.**

##### **3.1.2.1.1. Introducción.**

Este proceso deberá realizar la captura de los datos correspondiente al paciente que va realizarse una transfusión.

##### **3.1.2.1.2. Entradas.**

Por pantalla:

- Nombre del paciente.
- Apellidos del paciente.
- Sexo.
- Edad.
- No.Exp
- Sala.
- Cama.

Datos proporcionados por el sistema:  
Referente al Paciente:

- Nombre del paciente.
- Dirección del paciente.
- Tipo de sangre.

##### **3.1.2.1.3. Proceso.**

Se mostrará la pantalla de introducción de datos al usuario, los datos necesarios a introducir serán:



- Nombre y apellidos del paciente: Es un dato que referencia al nombre y los apellidos del paciente que va a realizar la transfusión.

- No.Exp: Es un dato que referencia unívocamente al paciente.

- Sala: Número de sala donde se encuentra hospitalizado el paciente.

- Cama: Número de cama donde se encuentra hospitalizado el paciente.

#### **3.1.2.1.4. Salidas.**

Todos los datos mencionados correspondientes al paciente serán almacenado en la Base de Datos.

### **3.1.3. MODIFICAR DATOS DEL PACIENTE**

#### **3.1.3.1. Especificación.**

##### **3.1.3.1.1. Introducción.**

Este proceso deberá realizar la captura de todos los datos a modificar del paciente.

##### **3.1.3.1.2. Entradas.**

Por pantalla:

Datos proporcionados por el sistema.

Referente al Paciente.

- Nombre del paciente.

- Dirección del paciente.

- Tipo de sangre.

##### **3.1.3.1.3. Proceso.**

Se mostrará la pantalla de visualización con los datos correspondiente de un determinado paciente para ser modificado. Los datos a modificar podrían ser los siguientes:

- Nombre y apellidos del paciente: Es un dato que referencia al nombre y los apellidos del paciente que va a realizar la transfusión.

- No.Exp: Es un dato que referencia unívocamente al paciente.

- Sala: Número de sala donde se encuentra hospitalizado el paciente.

- Cama: Número de cama donde se encuentra hospitalizado el paciente.

##### **3.1.3.1.4. Salidas.**

Todos los datos mencionados correspondientes al paciente serán almacenados en la Base de Datos.

### **3.1.4. ELIMINAR REGISTRO DEL DONANTE.**

#### **3.1.4.1. Especificación.**



### **3.1.4.1.1. Introducción.**

Este proceso deberá realizar la captura de los datos correspondiente a la sangre del donante a eliminar.

### **3.1.4.1.2. Entradas.**

Por pantalla:

- Número de guía.
- Código de bolsa.
- Fecha de ingreso.
- Fecha de Vencimiento.
- Tipo de sangre.

Datos proporcionados por el sistema.

Referente a la Sangre.

- Fecha de llegada.
- Fecha de Vencimiento.
- Tipo de sangre.

### **3.1.4.1.3. Proceso.**

Se mostrará la pantalla de introducción de datos al usuario, los datos necesarios a introducir serán:

- Número de guía: Es un dato obligatorio que representa la identificación única de una bolsa de sangre, el cual es asignado por la fábrica que la elaboró.
- Código de bolsa: Es un dato obligatorio ya que es una identificación única de la bolsa de sangre, dicho identificador es asignado por el CNS.
- Fecha de ingreso: Indica la fecha de ingreso de una unidad de sangre al laboratorio del Banco de Sangre del H.E.O.D.R.A.
- Fecha de Vencimiento: Indica la fecha de expiración de una unidad de sangre.
- Tipo de sangre. Representa el tipo de sangre del donante.

### **3.1.4.1.4. Salidas.**

Todos los datos mencionados corresponden al donante y serán eliminados de la Base de Datos.



### **3.1.5. LISTADOS DE PACIENTES A LOS QUE SE APLICÓ UNA O VARIAS TRANSFUSIONES.**

#### **3.1.5.1. Especificación.**

##### **3.1.5.1.1. Introducción.**

Este proceso mostrará por pantalla un listado de los paciente a los que se le aplicó una o varias transfusiones.

##### **3.1.5.1.2. Entradas.**

Por pantalla:

- Ninguno.

Datos proporcionados por el sistema.  
Referente al Paciente:

- Nombre del paciente.
- Dirección del paciente.
- Tipo de sangre.

##### **3.1.5.1.3. Proceso.**

- Ninguno

##### **3.1.5.1.4. Salidas.**

Lista de los pacientes que han realizado una o varias transfusiones.

### **3.1.6. LISTADO DIARIO DE LAS SANGRES DISPONIBLES AGRUPADAS POR TIPOS.**

#### **3.1.6.1. Especificación.**

##### **3.1.6.1.1. Introducción.**

Este proceso deberá realizar la captura de los datos correspondiente a la sangre del donante.

##### **3.1.6.1.2. Entradas.**

- Ninguno

Datos proporcionados por el sistema.

- Listado diario de la sangre disponible agrupadas por tipos.

##### **3.1.6.1.3. Proceso.**



- Ninguno.

#### **3.1.6.1.4. Salidas.**

Listado diario de la sangre disponible agrupadas por tipos.

### **3.1.7. LISTADO DE SANGRE PREPARADA.**

#### **3.1.7.1. Especificación.**

##### **3.1.7.1.1. Introducción.**

Este proceso realizará el listado de la sangre preparada para el paciente.

##### **3.1.7.1.2. Entradas.**

- Ninguno

Datos proporcionados por el sistema.

- Listado de la sangre preparada para el paciente.

##### **3.1.7.1.3. Proceso.**

- Seleccionar de un menú el tipo de reporte que desea visualizar.

##### **3.1.7.1.4. Salidas.**

Todos los datos mencionados correspondientes a la sangre del donante serán almacenados en la Base de Datos.

### **3.1.8. LISTADO DIARIO DE LA SANGRE CON FECHA DE EXPIRACIÓN PRÓXIMA A VENCERSE.**

#### **3.1.8.1. Especificación.**

##### **3.1.8.1.1. Introducción.**

Este proceso mostrará el listado diario con fecha de expiración próxima a vencerse.

##### **3.1.8.1.2. Entradas.**

- Ningun

Datos proporcionados por el sistema:

- Listado diario de la sangre con fecha de expiración próxima a vencerse.



### **3.1.8.1.3. Proceso.**

- Seleccionar de un menú el tipo de reporte que desea visualizar.

### **3.1.8.1.4. Salidas.**

Listado diario de la sangre con fecha de expiración próxima a vencerse.

## **3.1.9. LISTADO DIARIO DE LA SANGRE DESCARTADA YA SEA POR: CONTAMINACIÓN, FECHA DE EXPIRACIÓN VENCIDA.**

### **3.1.9.1. Especificación.**

#### **3.1.9.1.1. Introducción.**

Este proceso deberá mostrar el listado de sangre descartada ya sea por contaminación, fecha de expiración vencida.

#### **3.1.9.1.2. Entradas.**

- Ninguno.

Datos proporcionados por el sistema:

- Listado de sangre descartada ya sea por contaminación, fecha de expiración vencida.

#### **3.1.9.1.3. Proceso.**

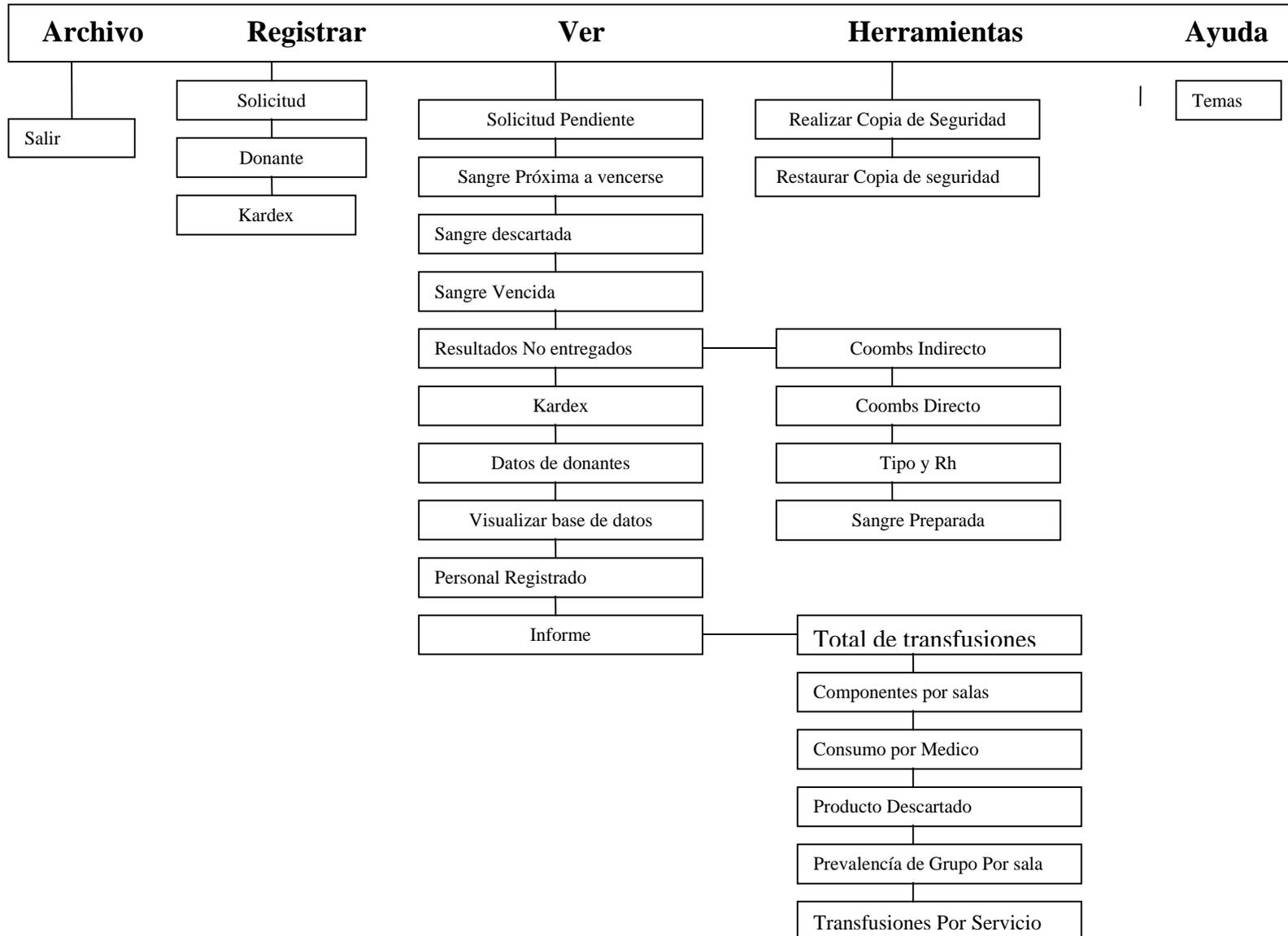
Seleccionar de un menú el tipo de reporte que desea visualizar.

#### **3.1.9.1.4. Salidas.**

- Listado de sangre descartada ya sea por contaminación, fecha de expiración vencida

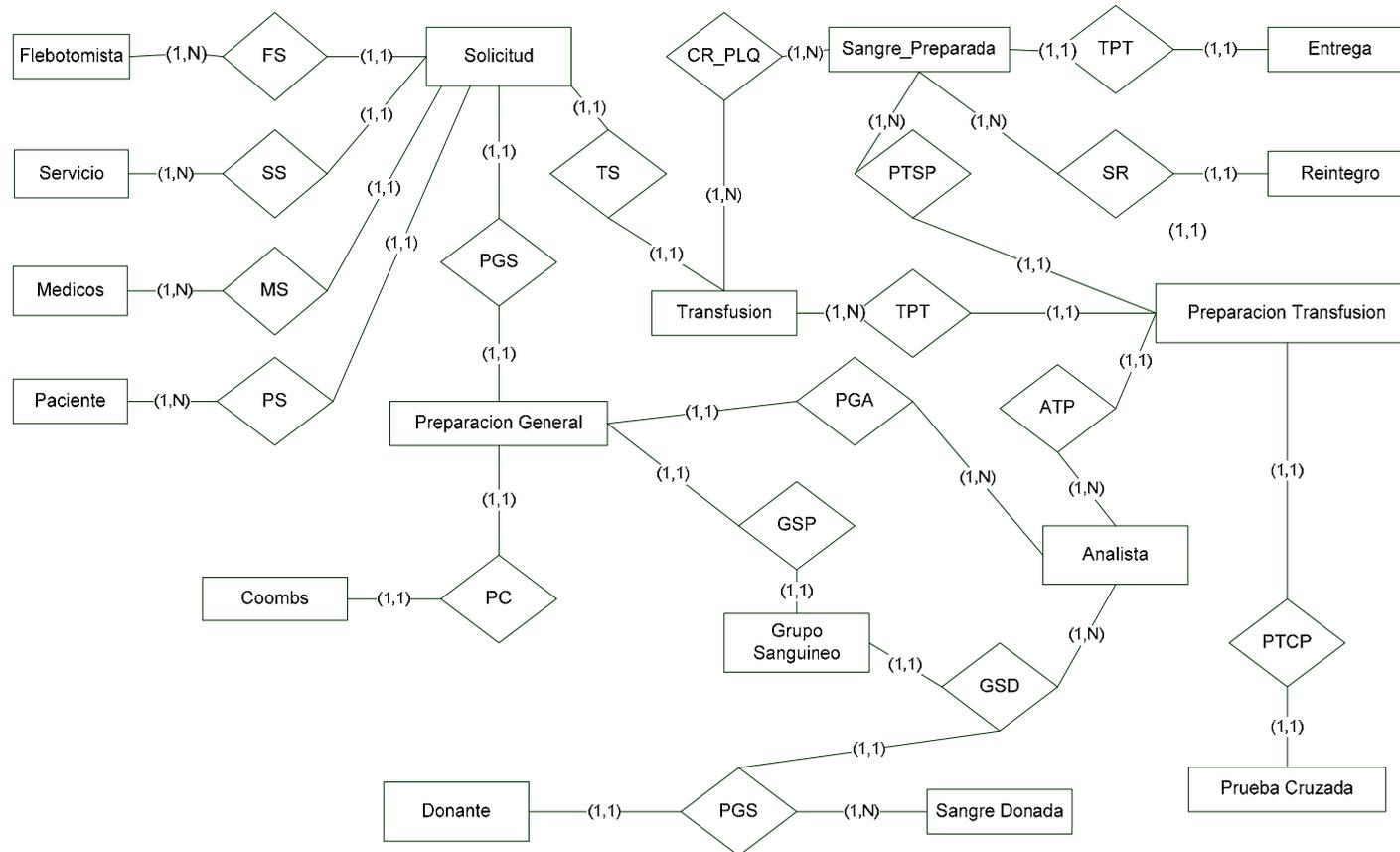


### 6.2. Diseño Arquitectónico





### 6.3. Diagrama Entidad Relacion





**6.4. Descripción de los atributos del Diagrama Entidad Relación**

<b>Flebotomista</b>	<b>Servicio</b>	<b>Médicos</b>	<b>Paciente</b>	<b>Preparación_General</b>
<b>Carnet</b> Nombre Apellido Especialidad	<b>ND</b> Servicio Sala	<b>Carnet</b> Nombre Apellido Especialidad	<b>Expediente</b> Nombre Apellido Edad Sexo FechaN Dirección Ocupación	<b>CodIPG</b> Hora _ preparación Fecha _ preparación

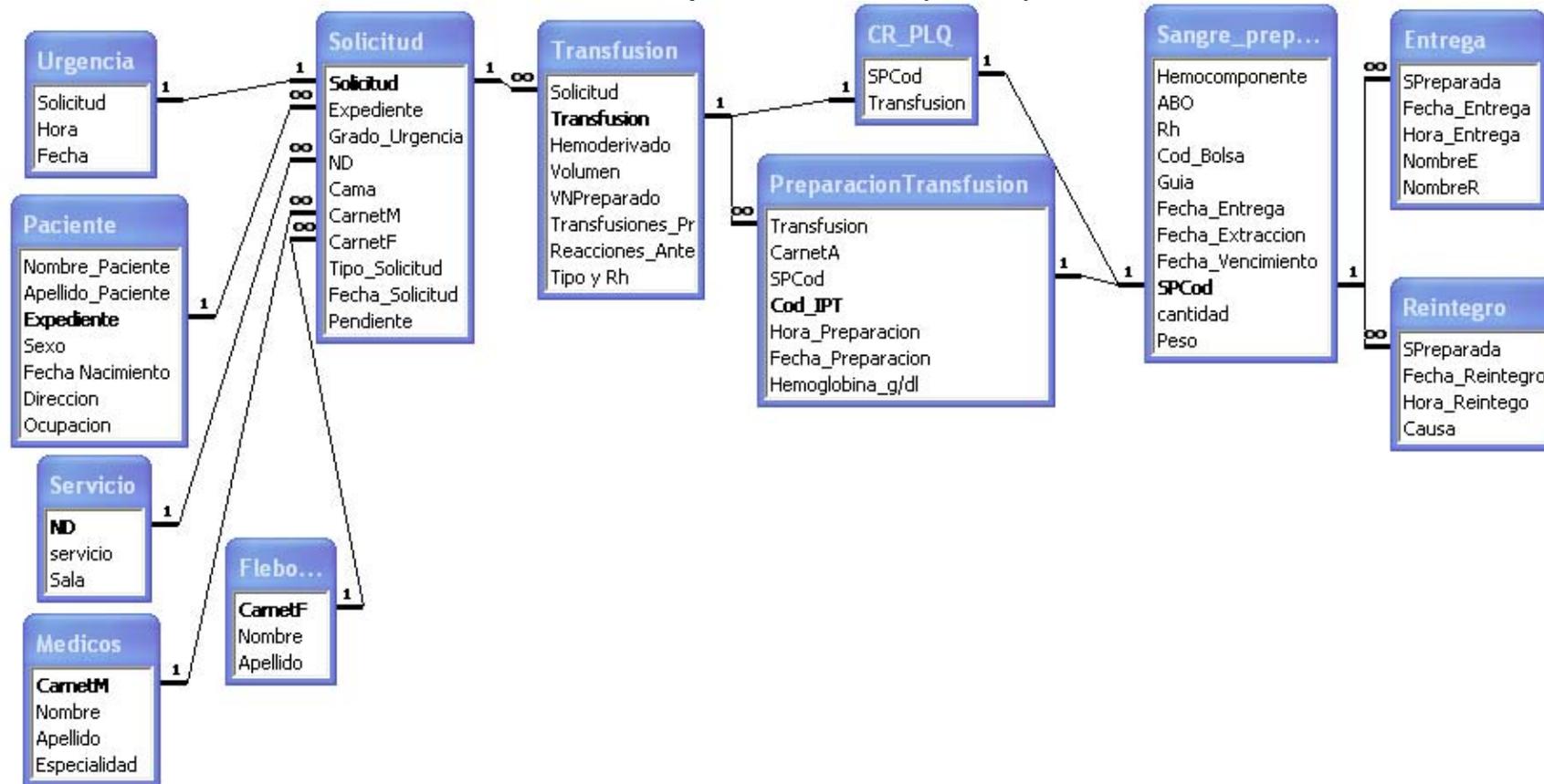
<b>Coombs</b>	<b>Grupo_Sanguineo</b>	<b>Transfusión</b>	<b>Solicitud</b>	<b>Sangre_Preparada</b>
<b>Nº_Reintegro</b> Albúmina AGH CCCoombs Tipo _ prueba Resultado	<b>Cod_GrupoS</b> Anti_A Anti_B Anti_AB Anti_D Autocontrol Variable_Du C.C.C Grupo Rh Cel_A Cel_O Cel_B Interpretacion Sifilis HIV HBSAg HCV Chagas Malaria	<b>Nº_Transfusión</b> Hemoderivado VNPreparado Transfusiones _ previas Reacciones_Anteriores	<b>Nº_Solicitud</b> Grado _ urgencia Cama Tipo _ solicitud Fecha _ solicitud	<b>SPCod</b> Hemocomponente ABO Rh Cod_Bolsa Guia Fecha_Entrega Fecha_Ectración Fecha_Vencimiento Cantidad Peso



<b>Analista</b> <b>Carnet</b> Nombre Apellido	<b>Sangre_Donada</b> Cod_Bolsa Cod_Guia Fecha_Extraccion Hora_Extraccion Rh Pendiente	<b>Prueba_Cruzada</b> <b>Nº_Reintegro</b> Salina Albúmina AGH CCCoombs Tipo _ prueba Resultado	<b>Preparación_Transf</b> <b>Cod_IPT</b> Hora _ preparación Fecha _ preparación Hemoglobina_g/dl	<b>Donante</b> Cedula Telefono Nombre Apellido Direccion Sexo Fecha Nacimiento Tipo Donacion <b>Cod_Donante</b>
<b>Reintegro</b> <b>Nº_Reintegro</b> Fecha _ reintegro Horra reintegro Causa NombreR	<b>Entrega</b> <b>Nº_Entrega</b> Fecha _ entrega Hora _ entrega NombreE NombreR			

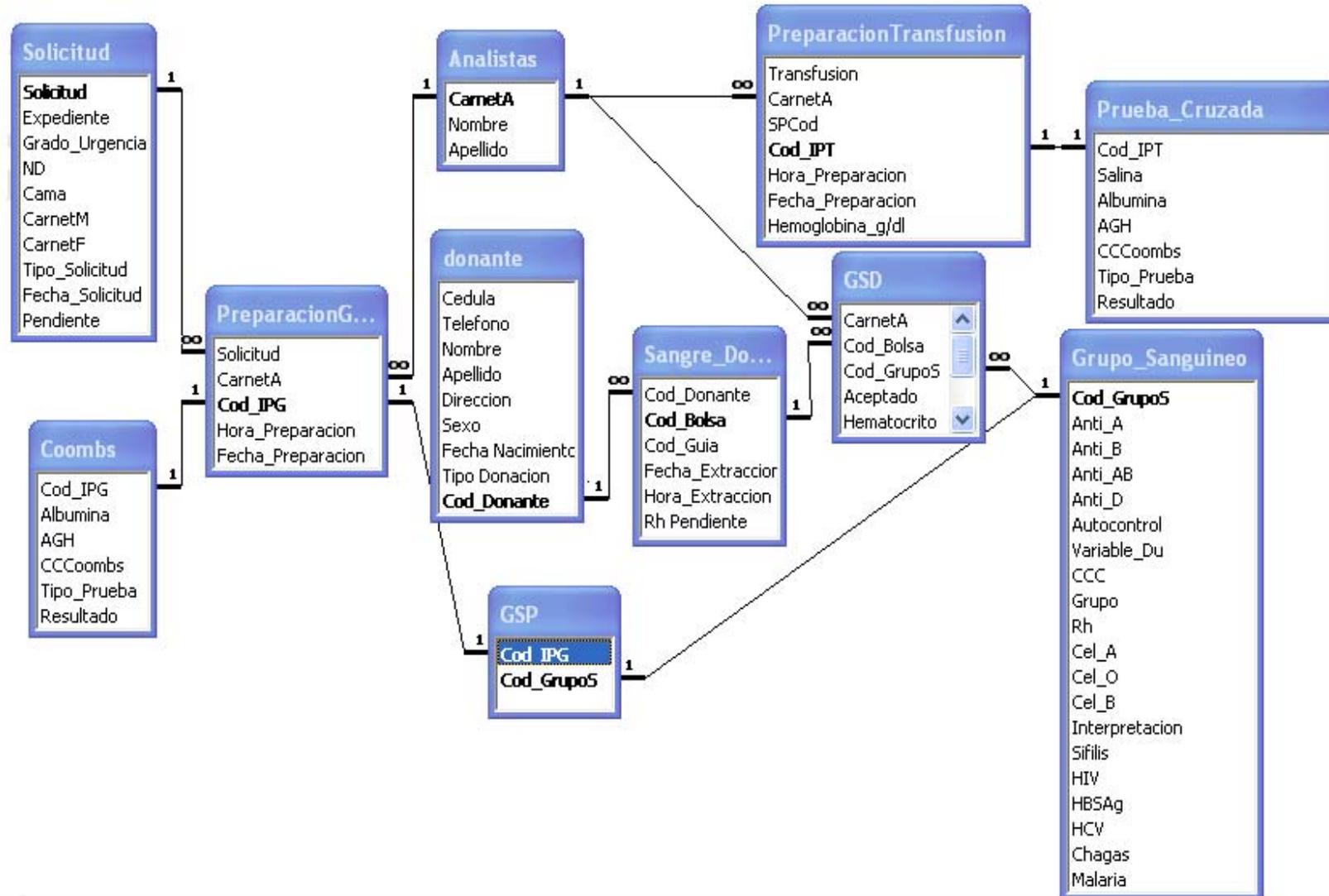


### 6.5. Esquema Relacional(Parte 1)



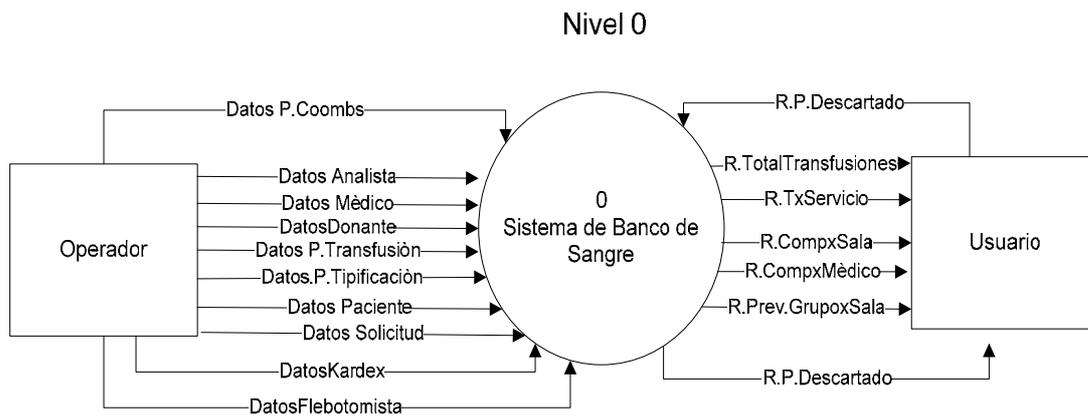


### 6.6. Esquema Realcional (Parte 2)



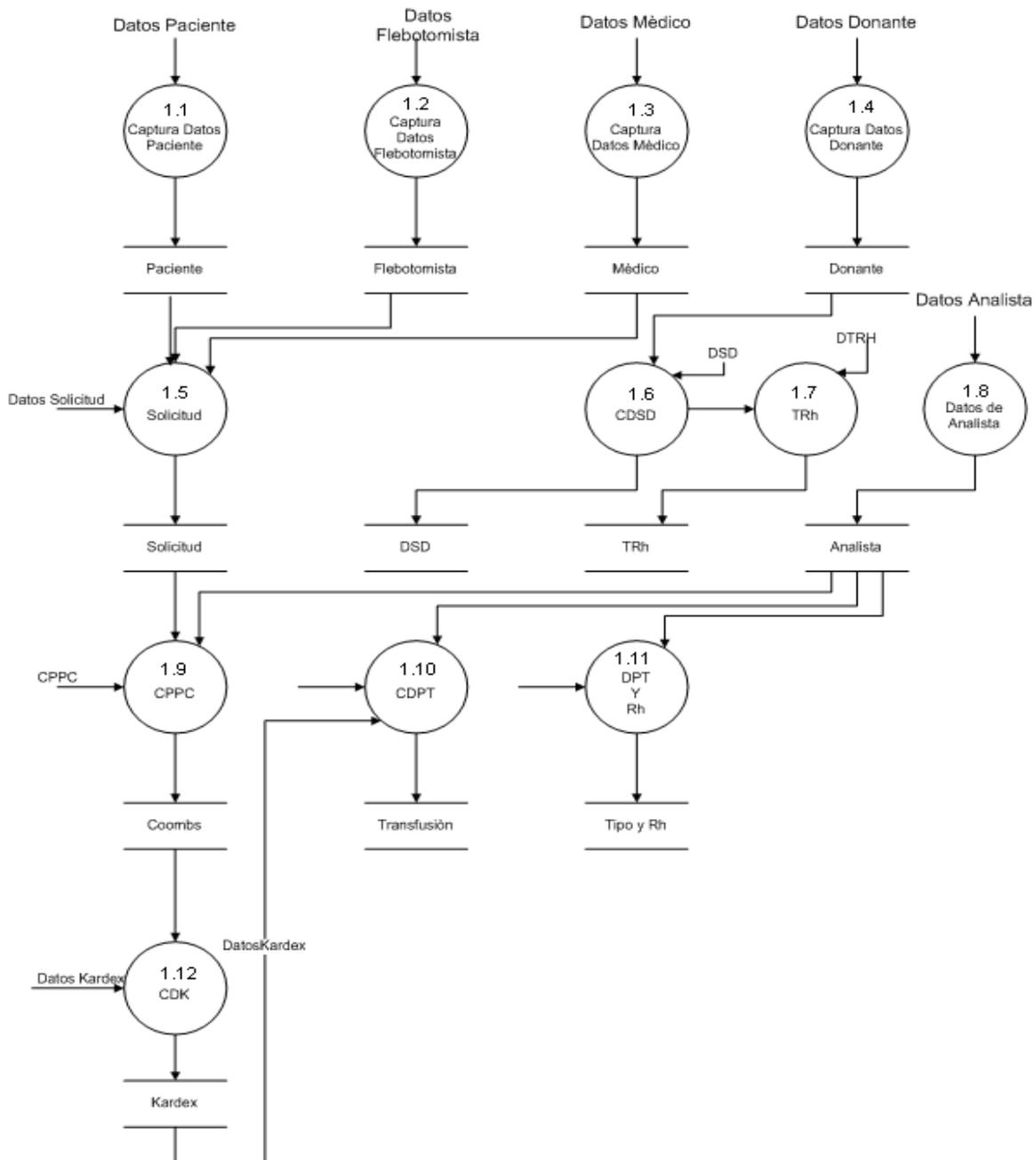


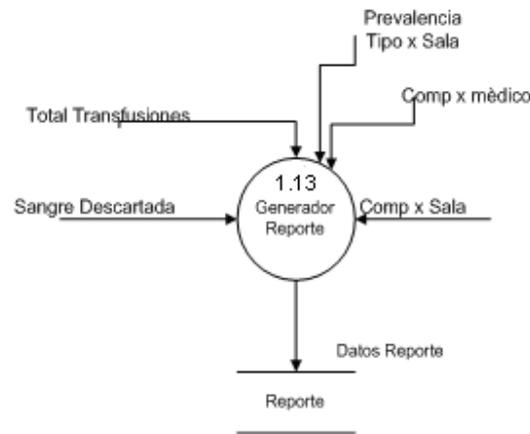
### 6.7. Diagrama de Flujo de datos





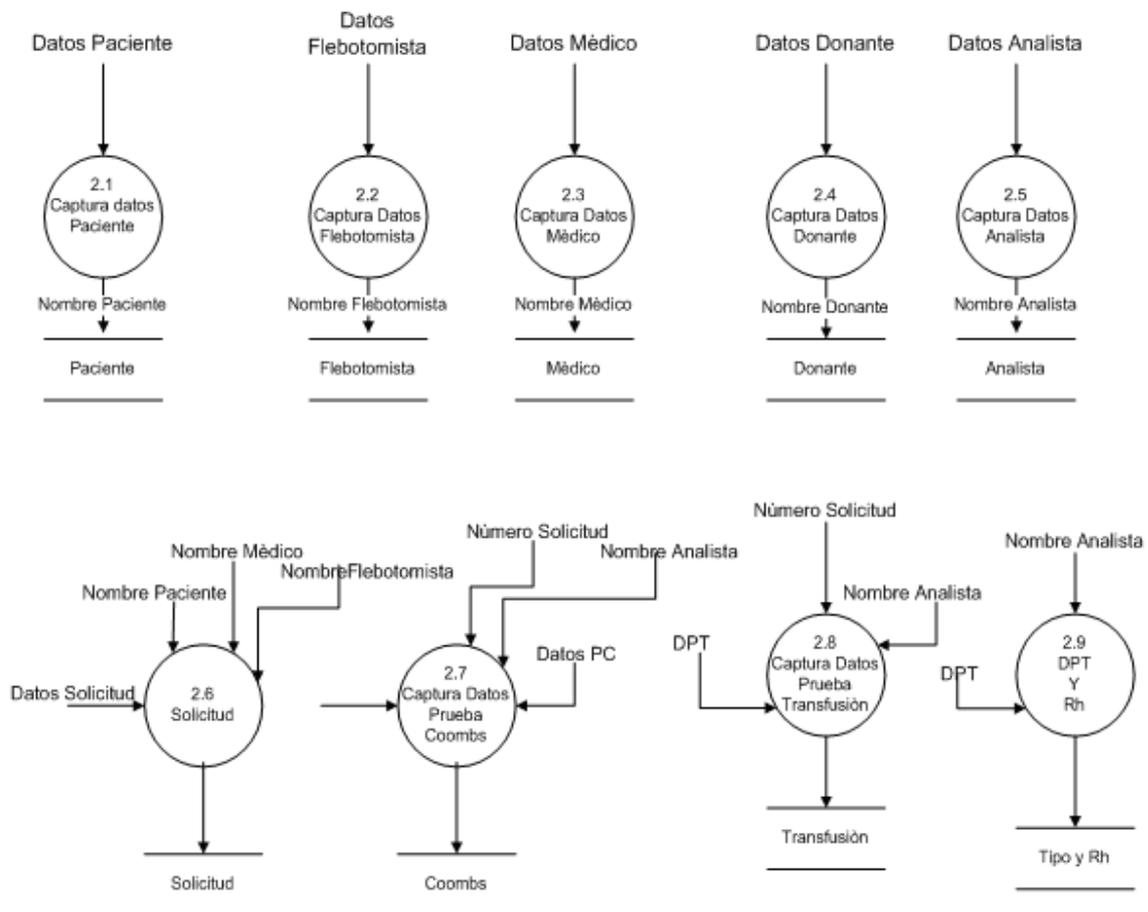
**Diagrama 1**  
Nivel 1





**Diagrama 2**

Nivel 2





## VII. Diseño Metodológico

El sistema informático se elaborará siguiendo el ciclo de vida clásico. Las etapas y los procedimientos para el desarrollo del software se detallan a continuación:

**1. Fase de Análisis (6 semanas):** Esta fase contempla identificación de requisitos del software y la definición de los alcances del sistema para el tiempo de desarrollo planificado. Las tareas contempladas para esta fase son:

a. Se realizarán entrevistas con la Directora del Laboratorio Lic. Rosa Emelina Alonso y la Lic. Marcia Noguera encargada del banco de sangre, para conocer los requisitos que debe satisfacer el sistema (funciones, limitaciones de contexto, etc). Además de entrevistar a la directora del Laboratorio, se solicitará documentación adicional para el diseño del sistema.

b. Definición de alcance y limitaciones del software. De los encuentros mencionados anteriormente se podrá definir los alcances y limitaciones para esta fase de desarrollo del sistema.

**2. Fase de Diseño (8 semanas):** En la fase de diseño se implantará la arquitectura software a seguir para la construcción del sistema, de igual manera en esta fase se instalarán y configurarán las herramientas de desarrollo necesarias para la construcción del sistema. Esta fase se divide en las tareas siguientes:

a. Definición e instalación de herramientas de soporte y desarrollo. Se utilizará software de libre distribución para esto se deberán realizar los pasos siguientes:

1. Instalar y configurar el gestor de Bases de Datos. El motor que se utilizará es MySQL 5.1 debido a que es una aplicación de libre distribución y funciona sobre la plataforma MS-Windows.

2. Instalar el entorno de programación. Las interfaces serán desarrolladas bajo el entorno Visual Basic 6.0. Este entorno permitirá realizar diseño que facilite la interacción Usuario-Sistema.

3. Proponer el modelo relacional que se utilizará para el diseño de la Base de Datos del Banco de Sangre del Hospital H.E.O.D.R.A.

b. Establecer métodos de validación del diseño. En este punto se debe involucrar a nuestro tutor M.sc. Ernesto Espinoza Montenegro y a la Directora del Banco de Sangre del Hospital H.E.O.D.R.A. Lic. Rosa Emelina Alonso para que afinen el primer prototipo de diseño estructural del sistema y acuerden los parámetros que validarán el modelo propuesto. Para esta fase se realizarán encuentros repetitivos.



c. Ajustar las Especificaciones del producto. Para finalizar la fase de diseño se propondrá un esquema de interfaces de usuarios que permitan recopilar la información de los Donantes y solicitantes de sangre.

.*Codificación:* La fase de codificación es la que demanda la mayor cantidad de recurso en el proyecto, debido a que se intensifica el trabajo en paralelo para la implementación de interfaces para este apartado se contemplan las siguientes acciones.

a. Se estima la primera etapa con una duración de 4 semanas. En esta etapa se crearán las interfaces de acceso a datos.

b. Se estima la segunda etapa con una duración de 8 semanas. Este período se dedicará a implementar la lógica de programación.

c. Se estima la tercera etapa con una duración de 3 semanas. En esta fase del proyecto se generaran los informes que deberá proporcionar el sistema.

d. La última semana se deberá utilizar para realizar los últimos ajustes. En esta etapa se deberá mantener encuentros consecutivos con nuestro tutor y la directora del Laboratorio del Hospital.

4. *Implantación.* En esta etapa realizaremos la instalación del sistema y la capacitación del personal que estará involucrado en la gestión del sistema.





## **IX. Conclusión**

- Este software agiliza la atención a los solicitantes y donantes de sangre reduciendo el tiempo de espera en un 50 % del realizado actualmente.
- Se lleva un mejor control de la sangre vencida y la que esta próxima a vencerse.
- Los reportes generados reducen el tiempo y costo ya que antes tenían que realizarse en varios días, ahora basta con pasar parámetros al generador de reporte para obtener los datos.
- Se agilizó el tiempo de búsqueda de un determinado paciente en un 50%.



## X.Recomendación

- Se recomienda a la UNAN-León registrar la autoría del software, pero bajo libre distribución del mismo
- Realizar mantenimiento del sistema de forma anual.
- Si se desea adaptar el sistema trabaje en red se recomienda migrar las interfaces de Visual Basic a Php.
- Mantener actualizada la fecha y hora del sistema operativo puesto que es utilizada por el programa en muchos procesos como por ejemplo en la elaboración de informes y en la generación de reportes.



## **XI. Bibliografía**

### **Libros**

Ceballos Sierra, Francisco Javier, Enciclopedia de programación Microsoft Visual Basic 6.0, Editorial RA-MA.

MalVorson, Michael, Aprende Microsoft Visual Basic 6.0 ya Editorial McGraw Hill, 1998.

Manual de estándares de medicina Transfusional.

### **Enlaces a Internet**

[http://es.tldp.org/ManualesLuCAS/manual\\_PHP/manual\\_PHP/odbc/instalar\\_myodbc.htm](http://es.tldp.org/ManualesLuCAS/manual_PHP/manual_PHP/odbc/instalar_myodbc.htm)

<http://dev.mysql.com/downloads/connector/odbc/3.51.html>

<http://dev.mysql.com/downloads/mysql/5.0.html>

<http://www.publispain.com/programas/programa-mysqlcontrol.htm>

<http://www.getec.etsit.upm.es/docencia/gproyectos/planificacion/cvida.htm>

[http://livedocs.adobe.com/dreamweaver/8\\_es/using/wwhelp/wwhimpl/common/html/wwhelp.htm?context=LiveDocs\\_Parts&file=28\\_con17.htm](http://livedocs.adobe.com/dreamweaver/8_es/using/wwhelp/wwhimpl/common/html/wwhelp.htm?context=LiveDocs_Parts&file=28_con17.htm)

<http://www.desarrolloweb.com/articulos/1202.php>

<http://www.telecable.es/personales/jrubi/index.htm?trucos.htm>



# ANEXOS



Fotografía que muestra la bolsa de sangre que se almacena en el Kardex.

Fecha	Número	Nombres y Apellidos del Paciente	Sexo	Edad	N° de Expediente
28/9/08		Darling Anacely Abanca (Quintero)	F		
28/9/08		Geraldine Gomez Monja	F		
28/9/08		Keyne Ramiro Ramiro Ruiz	M		
28/9/08		Mario José Jimín M	M		
28/9/08		Mario José Jimín M	M		
28/9/08		Blanca Flor Rocha	F		
28/9/08		Blanca Flor Rocha	F		
28/9/08		Leonilda Pacheco	F		
28/9/08		Ernstine Balacón Rocha	F		
28/9/08		Judith Rojas Flores	F		
28/9/08		Judith Rojas Flores	F		

Fotografía que muestra parte de los datos que se guardan en el libro de Transfusión.



Fotografía que muestra la realización de la preparación de la sangre para un paciente.



Fotografía que muestra el libro de registro de tipo y Rh.



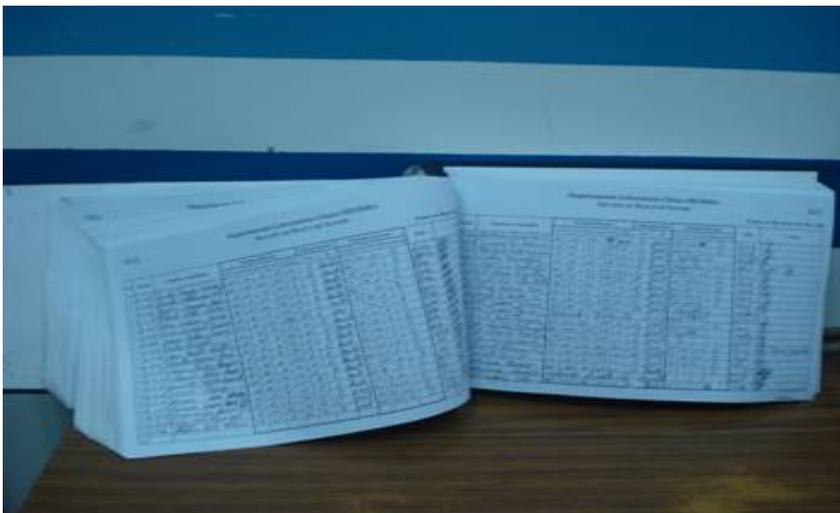
Carátula del libro de registro de transfusión



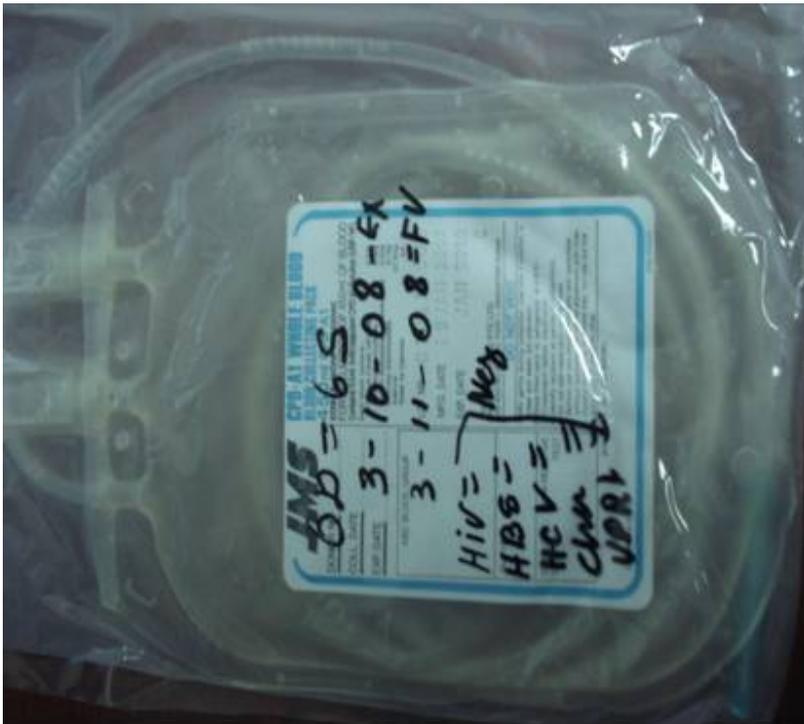
Libro de registro de donante.



Refrigeradora que contiene sangre disponible (Kardex).



Fotografía de libro de tipo y Rh



Bolsa de recolección de sangre del donante.



Váscula para pesar la sangre.



**Formulario Registrar Solicitud :** Este formulario permite seleccionar el tipo de solicitud que se desea realizar ya sea tipificación,transfusión,coombs directo,coombs indirecto.

**Formulario Registrar Solicitud de Tipificación, Coombs Directo ,Coombs Indirecto:** Este formulario permite registrar los datos de una solicitud. De determinación de Grupo y Rh., Coombs Directo y Coombs Indirecto.



**Formulario Registrar Solicitud de Transfusión:** Este formulario permite registrar los datos de una solicitud de Transfusión para un determinado paciente.

**SOLICITUD DE TRANSFUSION**

Nombre Paciente: MARIA ISABEL HERNÁNDEZ REY 00012

Flebotomista: ANA JULIA ESTRADA GURDIAN 0024-8969

Médico Solicitante: ANTONIO HERNANDEZ FLORES 0023-8979

Grado de Urgencia: URGENTE(15 min)

Volumen(ml): 250

Hemoderivado: ST

Tipo Y Rh: A POSITIVO

Transfuciones Previas: SI

N° Cama: 456

Reacción a Transf Anteriores: SI

Fecha de Solicitud : 30/09/08

Seleccione el Servicio y la sala

EMERGENCIAS

EMERGENCIAS

Guardar Agregar Dpto

**Formulario Registrar Paciente:** Este formulario permite registrar los datos de un paciente cuando este no se encuentra en la lista.

**Datos del Paciente**

N° Expediente:

Nombre Paciente:

Apellido Paciente:

Ocupación:

Dirección:

Sexo:

Fecha de Nacimiento: 30/09/08

Cancelar Guardar



**Formulario Registrar Flebotomista:** Este formulario permite registrar los datos de un nuevo flebotomista cuando este no se encuentra en la lista.

Formulario de registro de datos del paciente. El formulario tiene un título "Datos del Paciente" y un botón de cierre en la esquina superior derecha. Contiene los siguientes campos:

- N° Expediente:
- Nombre Paciente:
- Apellido Paciente:
- Ocupación:
- Dirección:
- Sexo:
- Fecha de Nacimiento:

En la parte inferior del formulario hay dos botones: "Cancelar" y "Guardar".

**Formulario Registrar Médico:** Este formulario permite registrar los datos del Médico cuando este no se encuentra en la lista.

Formulario de registro de datos del médico. El formulario tiene un título "Datos de Medicos" y un botón de cierre en la esquina superior derecha. Contiene los siguientes campos:

- Nombre:
- Apellido:
- Carnet:
- Especilidad:

En la parte inferior del formulario hay un botón: "Guardar".



**Formulario Kardex:** Este formulario permite visualizar los datos de la sangre que se encuentra en el inventario(Kardex), así mismo permite introducir una nueva sangre y descartar la sangre que esta vencida.

Vencida	Cod_Bolsa	Guia
Vencida	1420FG	1420PO
Vencida	1420H	1420H
Vencida	1420KL	1420HJ
Vencida	1420L	1420L
Vencida	1420P	1420P
Vencida	1420PP	1420FK
Vencida	1420R	1420YU
Vencida	1420UI	1420O
Vencida	1421FG	1421PO
Vencida	1421H	1421H
Vencida	1421KL	1421HJ
Vencida	1421L	1421L
Vencida	1421P	1421P
Vencida	1421DD	1421DK

**Formulario de Descarte:** Este formulario visualiza los datos de la sangre a descartar y la causa de descarte que debe ser seleccionada por el usuario a excepción de la sangre vencida que se selecciona automáticamente.



**Formulario de Donante:** En este formulario se registran los datos del donante, así mismo se muestran los datos de la bolsa de la sangre que se utiliza para almacenar la sangre recepcionada.

**Registro de Donante**

Nombre: ISA MARIA      Teléfono: 4568213  
 Apellido: HERNANDEZ REYES      Sexo: Femenino  
 Dirección: \*\* 2 CUADRAS AL NORT      Fecha de Nacimiento: 30/09/08  
 Cédula: 258-251286-000ZQ      Tipo Donación: Voluntaria

Cod Donante	Cod Bolsa	Cod Guia	Fecha Extraccion	Hora Extraccion	Rh Per
* 200893094556	0125	4526	30/09/08	9:30	si

Determinar Tipo y Rh      Resultado de Tipo y Rh

Primero	Anterior	Siguiente	último
Nuevo Donante	Guardar	Cancelar	Buscar

**Formulario Registrar Tipo y Rh:** Este formulario permite determinar el tipo y Rh de un donante, así mismo este formulario permite indagar la serología de la sangre para conocer si esta infectada.

**DETERMINAR GRUPO SANGUINEO DONANTE**

**Prueba Directa ABO**  
 Anti\_A: 1+  
 Anti\_B: 1+  
 Anti\_AB: NEGATIVO

**Sistema RH**  
 Autocontrol: NEGATIVO  
 Anti\_D: NEGATIVO  
 Variable\_D: NEGATIVO  
 C.C.Coombs: 2+

**Serología**  
 Sifilis: Negativo      HIV: Negativo  
 HCV: Negativo      Chagas: Negativo  
 Malaria: Negativo      HBsAg: Negativo

**Prueba Inversa ABO**  
 Cel\_0: 3+  
 Cel\_A: 3+  
 Cel\_B: 2+  
 Interpretación: B

**Interpretación ABO - Rh**  
 Grupo: B  
 Rh: NEGATIVO

**Fecha Preparación:** 30/09/08  
**Hora Preparación:** 09:30 AM

**Aceptado:** SI  
**Analista:** AA AA

**Hematocrito:** 0.00      **Hemoglobina:** 0.00

Guardar



**Formulario Ver Solicitudes Pendientes.** Este formulario visualiza las solicitudes que no se le han completado las pruebas ya sea de tipificación, transfusión, coombs directo e Indirecto, así mismo este formulario permite eliminar una solicitud que haya sido descartada.

**Solicitud Pendientes**

Nombre\_Paciente: HERIBERTO JOSE  
 Apellido\_Paciente: MENDEZ DIAZ  
 Expediente: 10  
 Sexo: MASCULINO  
 Fecha de Nacimiento: 08/02/79  
 Edad: 29

Volumen Solicitado (ml): 1000    Vol Pendiente: 1000    Tipo y Rh: A NEGATIVO    Hemoderivado: PG    Transf Previas: NO    Reacciones Anteriores: NO

**Datos de Transfusiones Programadas**

Fecha	Hora

**DATOS DE SOLICITUD**

Solicitud	Expediente	Grado Urgencia	Cama	Tipo_Solicitud
6	10	MUY URGENTE(5 min)	411	Transfusión
7	10	MUY URGENTE(5 min)	0012	Tipificación
8	11	URGENTE(15 min)	013	Coombs Directo
9	12	URGENTE(15 min)	0125	Tipificación

**Formulario Prueba Cruzada:** En este formulario se le realiza la prueba cruzada a un paciente cuando la solicitud es de Transfusión, así mismo se selecciona la sangre que va a ser preparada para dicho paciente.

**Prueba Cruzada**

**fases**      **Mayor**      **Autocontrol**

Salina:      2+      3+  
 Albúmina a 37° C:      3+      3+  
 AGH:      4+      4+  
 C.C.Coombs:      POSITIVO      POSITIVO

**DATOS DE BOLSA SELECCIONADA**  
 Fecha de Vencimiento : 03/11/2008    Fecha de Extracc  
 Código de bolsa: 6565  
 Hora\_Preparación: : :  
 Fecha\_Preparación: 30/09/08  
 Hemoglobina\_g/dl: 0.00  
 Analista isa reyes  
 Cant Preparada(ml)      Peso(g)  
 250      25

**Resultado:** COMPATIBLE       

**Formulario Coombs Directo:** En este formulario se realiza la prueba de Coombs Directo a un paciente, así mismo se registran los datos de preparación y el resultado de la prueba ya sea positivo o negativo.



Coombs Directo

**fases**

Albúmina a 37° C:

AGH:

C.C.Coombs:

**Coombs Directo**

2+

3+

POSITIVO

**Autocontrol**

4+

POSITIVO

POSITIVO

Hora Preparación  
10:05 AM

Fecha Preparación  
30/09/08

Analista  
AA AA

**Resultado:** POSITIVO

**Guardar**

**Cancelar**

**Formulario Sangre Próxima a vencerse:** Este formulario visualiza la sangre que su período de extracción está próxima a vencerse, esta sangre indica al analista según fecha de expiración cual es la que se debe dar prioridad de uso.

Sangre Próxima a Vencerse

Hemocomponente	ABO	Rh	Cod_Bolsa	Guia

**Imprimir**



**Formulario Sangre Vencida:** Este formulario visualiza la sangre que su fecha de vencimiento ha expirado, así mismo el usuario podrá descartar todas las sangres vencidas o solamente la seleccionada.

Hemocomponente	ABO	Rh	Cod_Bolsa	Guía
PG	B	POSITIVO	12	12
PG	O	POSITIVO	1420FG	1420PO
PG	AB	POSITIVO	1420H	1420H
PLQ	AB	POSITIVO	1420KL	1420HJ
PG	A	POSITIVO	1420L	1420L
PLQ	B	POSITIVO	1420P	1420P
PG	B	POSITIVO	1420PP	1420PK
PLQ	O	POSITIVO	1420R	1420YU

Buttons: Descartar Todas, Descartar Seleccionada

**Formulario Resultado de Pruebas de Coombs Indirecto:** Este formulario visualiza los resultados de las pruebas de Coombs Indirecto realizado a un determinado paciente, así mismo este formulario permite imprimir este resultado

**RESULTADO DE PRUEBAS DE COOMBS INDIRECTO**

Nombre\_Paciente:  Sexo:

Apellido\_Paciente:  Edad:

Expediente:

Resultado	Expediente	servicio

Buttons: Buscar, Actualizar, Entregar Resultado





**Formulario para entregar resultados de Sangre Preparada:** En este formulario se visualizan los datos de la sangre que ha sido preparada y que serán impresos.

**REMISION DE HEMOCOMPONENTES A TRANSFUNDIR**

Departamento de Laboratorio Clínico  
Sección Banco de Sangre

Fecha de Entrega: \_\_\_\_\_ Hora de Entrega: \_\_\_\_\_

**Datos del Paciente**

Nombre del Paciente: JAIRO HERNENDEZ RIOS RIOS  
 N° Expediente: 12      Departamento: ASEGURADOS  
 Sala: ASEGURADOS      Cama: # 0004

**Datos del Donador**

Código de bolsa: L74658U  
 Grupo sanguíneo: B NEGATIVO  
 Hemoderivado: PG      Cantidad (ml): 450  
 Fecha de Vencimiento: 27/10/2008      Fecha de Preparación: 23/09/2008  
 Resultado de la Prueba Cruzada: INCOMPATIBLE

Entregado Por: \_\_\_\_\_      Recibido Por: \_\_\_\_\_

**Formulario de Transfusiones con pruebas Cruzadas:** Este formulario visualiza todas las Transfusiones con pruebas Cruzadas que se han realizado. así mismo muestra los datos de solicitud y los datos de entrega.

**Pruebas Realizadas**

Nombre Paciente: JAIRO HERNENDEZ      Dirección: \*\*\* 1C AL NORTE  
 Apellido Paciente: RIOS RIOS      Ocupación: ECONOMISTA  
 Sexo: MASCULIN      Fecha Nacimiento: 08/04/86      Expediente: 12

Solicitud	Expediente	Servicio	Sala	Grado de Urgencia	Cama	Tipo de S
3	12	ASEGURADOS	ASEGURADOS	PROGRAMADA	0004	Transfus

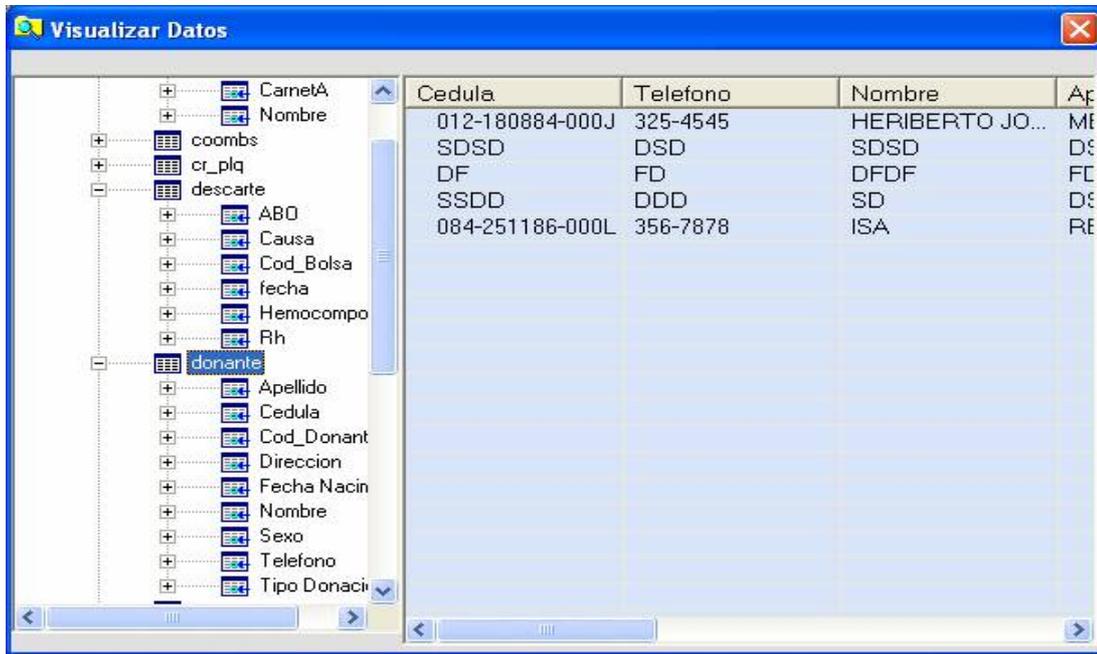
**Datos de Solicitudes de Transfusión:**

Tipo y Rh: B NEGATIVO      Reacciones Anteriores: NO      Hemo: PG      Volumen: 450      Transf Previas: NO

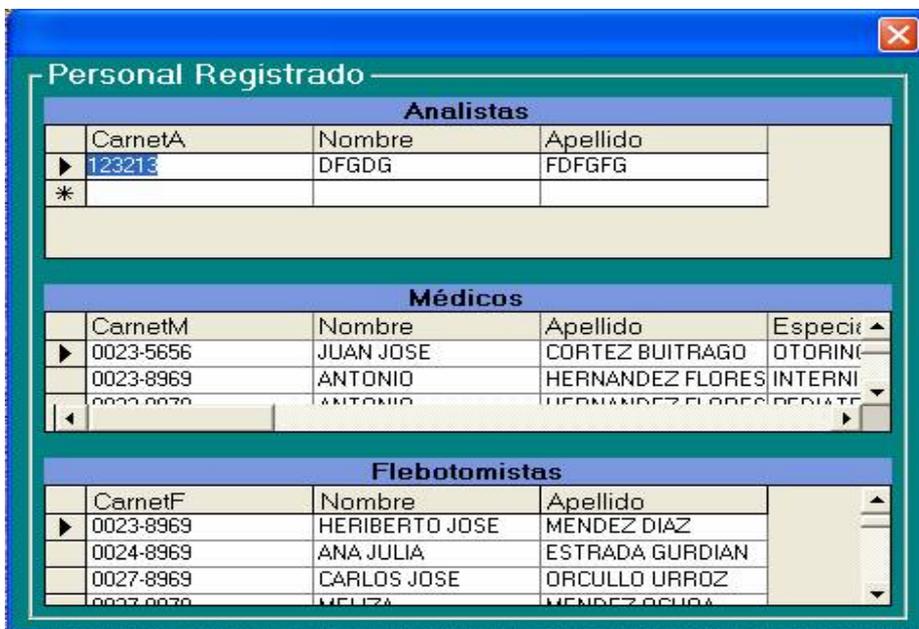
<b>Datos de Entrega:</b>		<b>Datos de Preparación:</b>	
Fecha de Entrega:	Recibido Por:	Hora de Preparación:	Cantidad Preparada:
Entregado Por:	Entregado Por:	Peso:	Hemoglobina:
Hora de Entrega:	Entregado Por:	Analista:	Código Bolsa:
		N° de Guía:	



**Formulario Visualizar Datos:** En este formulario se visualizan todos los datos de la Base de Datos.



**Formulario Personal Registrado:** En este formulario se visualizan todos los datos del personal registrado (Analista, Médico y Flebotomista).





**Informe de Total de Transfusiones:** En este informe se visualizan la cantidad de hemocomponente transfundido ya sea mensual, anual, semestral y trimestral.

Total Transfusiones

## HOSPITAL HEODRA

LEON - LEON

### Banco de Sangre

Total Transfusiones  
Informe Mensual  
Febrero Del 2000

Fecha de Elaboración: 30/09/2008

Regresar

Componente	Cantidad
Paquete Globular	0
Plasma Fresco	0
Plasma Simple	0
Plaquetas	0
Crioprecipitado	0
Sangre Total	0
Exanguinotransfusion	0
<b>Total</b>	<b>0</b>

---

RESP. DE LABORATORIO



### Código correspondiente a Tipo de Solicitud

```
Private Sub btacceptar_Click()
If cbTSolicitud.ListIndex > (-1) Then
Me.Tag = cbTSolicitud.Text
If cbTSolicitud.Text = "Transfusión" Then
frmsolicitudTransf.Show
Else
frmsolicitudTipo.Show
End If
End If
End Sub
```

```
Private Sub btcancelar_Click()
Unload Me
End Sub
```

```
Private Sub Form_Load()
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
End Sub
```

### Código Correspondiente al formulario Solicitud de Tipificación, Coombs Directo ,Coombs Indirecto

```
Private VarTSolicitud As String
Private RsSalas As ADODB.Recordset
```

```
Private Function RecargarDpto()
Set rs = databs.Connection1.Execute("SELECT Servicio FROM Servicio GROUP BY Servicio")
cbdepartamento.Clear
While Not rs.EOF
With cbdepartamento
.AddItem rs("Servicio").Value
rs.MoveNext
End With
Wend
End Function
```

```
Private Function RecargarFlebo()
Set rs = databs.Connection1.Execute("SELECT CONCAT(Nombre, ' ', Apellido) AS Nombre, CarnetF FROM flebotomistas")
cbflebo.Clear
cbNombreF.Clear
While Not rs.EOF
With cbflebo
```



```
.AddItem rs("CarnetF").Value
cbNombreF.AddItem rs("Nombre").Value
rs.MoveNext
End With
Wend
End Function
```

```
Private Function RecargarMedico()
Set rs = databs.Connection1.Execute("SELECT { fn CONCAT(Nombre, ' ', Apellido) } AS
Nombre,CarnetM FROM medicos")
cbmedico.Clear
cbNombreM.Clear
While Not rs.EOF
With cbmedico
.AddItem rs("CarnetM").Value
cbNombreM.AddItem rs("Nombre").Value
rs.MoveNext
End With
Wend
End Function
```

```
Private Function RecargarPaciente()
Set rs = databs.Connection1.Execute("SELECT { fn CONCAT(Nombre_Paciente, ' ',
Apellido_Paciente) } AS Nombre, Expediente FROM paciente")
cbNombre.Clear
cbexpediente.Clear
While Not rs.EOF
With cbNombre
.AddItem rs("Nombre").Value
cbexpediente.AddItem rs("Expediente").Value
rs.MoveNext
End With
Wend
```

End Function

```
Private Sub btcancelarsolicitud_Click()
btguardarsolicitud.Enabled = False
btcancelarsolicitud.Enabled = False
btnuevasolicitud.Enabled = True
LimpiarCajas
adosolicitud.Recordset.CancelUpdate
End Sub
```

```
Private Sub btdpto_Click()
frmDptoSala.Show vbModal
RecargarDpto
```



End Sub

Private Sub btguardarsolicitud\_Click()

Dim listb As ComboBox

On Error GoTo ControlError

If cbexpediente.ListIndex < 0 Then

MsgBox "Seleccione o Ingrese al Paciente", vbInformation, "Banco de Sangre"

Exit Sub

End If

If cbmedico.ListIndex < 0 Then

MsgBox "Seleccione o Ingrese al Medico", vbInformation, "Banco de Sangre"

Exit Sub

End If

If cbflebo.ListIndex < 0 Then

MsgBox "Seleccione o Ingrese al Flebotomista", vbInformation, "Banco de Sangre"

Exit Sub

End If

If cburgencia.ListIndex < 0 Then

MsgBox "Seleccione el grado de urgencia", vbInformation, "Banco de Sangre"

Exit Sub

End If

If Not IsNumeric(txtcama.Text) Then

MsgBox "El número cama debe ser Numérico", vbInformation, "Banco de Sangre"

Exit Sub

End If

If IsNull(dlsalas.SelectedItem) Then

MsgBox "seleccione la sala", vbInformation, "Banco de Sangre"

Exit Sub

End If

With databs.Connection1

If Not (RsSalas.EOF And RsSalas.BOF) Then

RsSalas.AbsolutePosition = dlsalas.SelectedItem

End If

.Execute "INSERT INTO solicitud VALUES(NULL,'" + cbexpediente.Text + "','" + cburgencia.Text + "','" + CStr(RsSalas("ND").Value) + "','" + txtcama.Text + "','" + cbmedico.Text + "','" + cbflebo.Text + "','" + VarTSolicitud + "','" + Format(Calendar1.Text, "yyyy-mm-dd") + "',true)"

End UIT



```
MsgBox "Datos Guardados", , "Banco de Sangre"  
Unload Me
```

```
Exit Sub
```

```
ControlError:  
Select Case Err.Number
```

```
Case Err.Number  
MsgBox "Error Inesperado: " + Err.Description, vbCritical, "Error"  
End Select  
End Sub
```

```
Private Function LimpiarCajas()  
txtExpediente.Text = ""  
cburgencia.Text = ""  
cbsala.Text = ""  
cbdepartamento.Text = ""  
txtmedico.Text = ""  
txtcama.Text = ""  
txtflebotomista.Text = ""  
End Function
```

```
Private Sub cbdepartamento_click()  
With dlsalas  
Set RsSalas = databs.Connection1.Execute("Select * from Servicio where Servicio like ""  
+ cbdepartamento.Text + """)  
Set .RowSource = RsSalas  
.ListField = "Sala"  
.Refresh  
End With  
End Sub
```

```
Private Sub cbExpediente_Click()  
cbNombre.ListIndex = cbexpediente.ListIndex  
End Sub
```

```
Private Sub cbflebo_click()  
cbNombreF.ListIndex = cbflebo.ListIndex  
End Sub
```

```
Private Sub cbmedico_click()  
cbNombreM.ListIndex = cbmedico.ListIndex  
End Sub
```

```
Private Sub cbNombre_click()  
cbexpediente.ListIndex = cbNombre.ListIndex
```



```
End Sub
Private Sub cbNombreF_click()

cbflebo.ListIndex = cbNombreF.ListIndex
End Sub

Private Sub cbNombreM_click()
cbmedico.ListIndex = cbNombreM.ListIndex
End Sub

Private Sub Form_Load()
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
VarTSolicitud = frmtiposolicitud.Tag
With cburgencia
.AddItem "URGENTE(15 min)"
.AddItem "MUY URGENTE(5 min)"
End With

RecargarFlebo
RecargarMedico
RecargarPaciente
RecargarDpto
End Sub

Private Sub txtflebotomista_LostFocus()
txtflebotomista.Text = UCase(txtflebotomista.Text)
End Sub

Private Sub txtmedico_LostFocus()
txtmedico.Text = UCase(txtmedico.Text)
End Sub

Private Sub Label10_Click()
Me.Tag = "Flebotomistas"
frmFlebo.Show vbModal
RecargarFlebo
End Sub

Private Sub Label11_Click()
Me.Tag = "Medicos"
frmMédicos.Show vbModal
RecargarMedico
End Sub

Private Sub Label9_Click()
frmNuevoPaciente.Show vbModal
```



RecargarPaciente  
End Sub

### **Código Correspondiente al formulario Solicitud de Transfusión**

Private RsSalas As ADODB.Recordset

```
'genera un nuevo Código de registro
Function GenerarNumRegistro() As String
GenerarNumRegistro = "S" + CStr(Year(Date)) + CStr(Month(Date)) + CStr(Day(Date)) +
CStr(Hour(Time)) + CStr(Minute(Time)) + CStr(Second(Time))
End Function
```

```
Private Function RecargarDpto()
Set rs = databs.Connection1.Execute("SELECT Servicio FROM Servicio GROUP BY
Servicio")
cbdepartamento.Clear
While Not rs.EOF
With cbdepartamento
.AddItem rs("Servicio").Value
rs.MoveNext
End With
Wend
End Function
```

```
Private Function RecargarFlebo()
Set rs = databs.Connection1.Execute("SELECT CONCAT(Nombre, ' ', Apellido) AS
Nombre, CarnetF FROM flebotomistas")
cbflebo.Clear
cbNombreF.Clear
While Not rs.EOF
With cbflebo
.AddItem rs("CarnetF").Value
cbNombreF.AddItem rs("Nombre").Value
rs.MoveNext
End With
Wend
End Function
```

```
Private Function RecargarMedico()
Set rs = databs.Connection1.Execute("SELECT { fn CONCAT(Nombre, ' ', Apellido) } AS
Nombre,CarnetM FROM medicos")
cbmedico.Clear
cbNombreM.Clear
While Not rs.EOF
With cbmedico
.AddItem rs("CarnetM").Value
cbNombreM.AddItem rs("Nombre").Value
```



```
rs.MoveNext  
End With  
Wend  
End Function
```

```
Private Function RecargarPaciente()  
Set rs = databs.Connection1.Execute("SELECT { fn CONCAT(Nombre_Paciente, ' ',  
Apellido_Paciente) } AS Nombre, Expediente FROM paciente")  
cbNombre.Clear  
cbexpediente.Clear  
While Not rs.EOF  
With cbNombre  
.AddItem rs("Nombre").Value  
cbexpediente.AddItem rs("Expediente").Value  
rs.MoveNext  
End With  
Wend  
End Function
```

```
Private Sub btdpto_Click()  
frmDptoSala.Show vbModal  
RecargarDpto  
End Sub
```

```
Private Sub btguardarsolicitud_Click()  
Dim varfecha, varhora As String  
'On Error GoTo ControlError  
Dim rs As Variant
```

```
If cbexpediente.ListIndex < 0 Then  
MsgBox "Seleccione o Ingrese al Paciente", vbInformation, "Banco de Sangre"  
Exit Sub  
End If
```

```
If cbflebo.ListIndex < 0 Then  
MsgBox "Seleccione o Ingrese al Flebotomista", vbInformation, "Banco de Sangre"  
Exit Sub  
End If
```

```
If cbmedico.ListIndex < 0 Then  
MsgBox "Seleccione o Ingrese al Medico", vbInformation, "Banco de Sangre"  
Exit Sub  
End If
```

```
If cburgencia.ListIndex < 0 Then  
MsgBox "Seleccione el grado de urgencia", vbInformation, "Banco de Sangre"  
Exit Sub
```



```

End If
If Not IsNumeric(txtvolumen.Text) Then
MsgBox "El volumen debe ser numérico", vbInformation, "Banco de Sangre"
Exit Sub
End If

If Not IsNumeric(txtcama.Text) Then
MsgBox "El número cama debe es inválido", vbInformation, "Banco de Sangre"
Exit Sub
End If

If IsNull(dlsalas.SelectedItem) Then
MsgBox "Seleccione la sala", vbInformation, "Banco de Sangre"
Exit Sub
End If

If cbdepartamento.ListIndex < 0 Then
MsgBox "Seleccione el Servicio", vbInformation, "Banco de Sangre"
Exit Sub
End If

If cburgencia.ListIndex = 2 Then
frmUrgencia.Show vbModal
If frmUrgencia.OK = True Then
varfecha = frmUrgencia.txtfecha.Text
varhora = frmUrgencia.txthora.Text
Unload frmUrgencia
Else
MsgBox "Ingrese la Fecha Y la Hora para las transfusiones Programadas",
vbInformation, "Banco de Sangre"
Exit Sub
End If
End If

With databs.Connection1
.BeginTrans
If Not (RsSalas.BOF And RsSalas.EOF) Then
RsSalas.AbsolutePosition = dlsalas.SelectedItem
End If
.Execute "INSERT INTO solicitud VALUES(NULL,'" + cbexpediente.Text + "','" +
cburgencia.Text + "','" + CStr(RsSalas("ND").Value) + "','" + txtcama.Text + "','" +
cbmedico.Text + "','" + cbflebo.Text + "','" + 'Transfusión','" + Format(Calendar1.Text, "yyyy-
mm-dd") + "',true)"
Set rs = .Execute("select max(`Solicitud`) from solicitud")
.Execute "INSERT INTO transfusion VALUES(" + CStr(rs(0).Value) + ",NULL,'" +
cbhemoderivado.Text + "','" + txtvolumen.Text + "','" + txtvolumen.Text + "','" +
cbtransfp.Text + "','" + cbreacciones.Text + "','" + cbtipo.Text + '"")"

```



```

If cburgencia.ListIndex = 2 Then
.Execute "INSERT INTO urgencia VALUES(" + CStr(rs(0)) + ", " + varhora + ", " +
Format(varfecha, "yyyy-mm-dd") + ")")
End If

.CommitTrans

End With

MsgBox "Datos Guardados", , "Banco de Sangre"
Unload Me
Exit Sub
'ControlError:
databs.Connection1.RollbackTrans
Select Case Err.Number
Case Err.Number
MsgBox "Error Inesperado: " + Err.Description, vbCritical, "Error"
End Select

End Sub

Private Function LimpiarCajas()
cbexpediente.Text = ""
cbreacciones.Text = ""
cburgencia.Text = ""
cbsala.Text = ""
cbdepartamento.Text = ""
cbtransfp.Text = ""
cbhemoderivado.Text = ""
txtmedico.Text = ""
txtvolumen.Text = ""
txtN°_Cama.Text = ""
txtflebotomista.Text = ""
End Function

Private Sub cbdepartamento_click()
With dlsalas
Set RsSalas = databs.Connection1.Execute("Select * from Servicio where Servicio like ""
+ cbdepartamento.Text + """)
Set .RowSource = RsSalas
.ListField = "Sala"
.Refresh
End With
End Sub

Private Sub cbExpediente_Click()
cbNombre.ListIndex = cbexpediente.ListIndex

```



End Sub

```
Private Sub cbflebo_click()  
cbNombreF.ListIndex = cbflebo.ListIndex  
End Sub
```

```
Private Sub cbmedico_click()  
cbNombreM.ListIndex = cbmedico.ListIndex  
End Sub
```

```
Private Sub cbNombre_click()  
cbexpediente.ListIndex = cbNombre.ListIndex  
End Sub
```

```
Private Sub cbNombreF_click()  
cbflebo.ListIndex = cbNombreF.ListIndex  
End Sub
```

```
Private Sub cbNombreM_click()  
cbmedico.ListIndex = cbNombreM.ListIndex  
End Sub
```

```
Private Sub CommandButton1_Click()
```

End Sub

```
Private Sub Form_Load()  
Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2  
With cbhemoderivado  
.AddItem "ST"  
.AddItem "PG"  
.AddItem "PFC"  
.AddItem "PLQ"  
.AddItem "CRIO"  
.AddItem "PS"  
End With
```

```
With cbtipo  
.AddItem "A POSITIVO"  
.AddItem "A NEGATIVO"  
.AddItem "O POSITIVO"  
.AddItem "O NEGATIVO"  
.AddItem "B POSITIVO"  
.AddItem "B NEGATIVO"  
.AddItem "AB POSITIVO"  
.AddItem "AB NEGATIVO"  
.AddItem "DESCONOCIDO"
```



End With

With cbreacciones

.AddItem "SI"

.AddItem "NO"

.AddItem "DESCONOCIDO"

End With

With cbtransfp

.AddItem "SI"

.AddItem "NO"

.AddItem "DESCONOCIDO"

End With

With cburgencia

.AddItem "MUY URGENTE(5 min)"

.AddItem "URGENTE(15 min)"

.AddItem "PROGRAMADA"

End With

RecargarFlebo

RecargarMedico

RecargarPaciente

RecargarDpto

End Sub

Private Sub txtflebotomista\_LostFocus()

txtflebotomista.Text = UCase((txtflebotomista.Text))

End Sub

Private Sub txtmedico\_LostFocus()

txtmedico.Text = UCase(Trim(txtmedico.Text))

End Sub

Private Sub Label10\_Click()

Me.Tag = "Medicos"

frmMédicos.Show vbModal

RecargarMedico

End Sub

Private Sub Label11\_Click()

Me.Tag = "Flebotomistas"

frmFleme.Show vbModal

RecargarFlebo

End Sub



```
Private Sub Label9_Click()
frmNuevoPaciente.Show vbModal
RecargarPaciente
End Sub
```

```
Private Sub txtfecha_GotFocus()
txtfecha.SelStart = 0
End Sub
```

### Código de Formulario de Paciente

```
Private Sub btcancelar_Click()
Unload Me
End Sub
```

```
Private Sub BtGuardar_Click()
On Error GoTo ControlError
Dim txtB As TextBox
```

```
If cbsexo.ListIndex < 0 Then
MsgBox "Seleccione el sexo", vbInformation, "Banco de Sangre"
Exit Sub
End If
```

```
For i = 0 To txtDPaciente.Count - 1
If txtDPaciente(i).Text = "" Then
MsgBox "Ingrese todos los valores de entrada", vbInformation, "Banco de Sangre"
Exit Sub
End If
Next
```

```
databs.Connection1.Execute "INSERT INTO paciente VALUES (" + txtDPaciente(0).Text
+ "," + txtDPaciente(1).Text + "," + txtDPaciente(2).Text + "," + cbsexo.Text + "," +
Format(Calendar1.Text, "yyyy-mm-dd") + "," + txtDPaciente(3).Text + "," +
txtDPaciente(4).Text + ")"
MsgBox "Datos guardados", vbInformation, "Banco de Sangre"
Exit Sub
```

```
ControlError:
Select Case Err.Number
Case Err.Number
MsgBox "Erro inesperado: " + Err.Description, vbCritical, "Error"
End Select
Err.Clear
End Sub
```



```
Private Sub btsolicitud_Click()
frmtiposolicitud.Show
End Sub
```

```
Private Sub Form_Load()
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
End Sub
```

```
Private Sub txtDpaciente_LostFocus(Index As Integer)
txtDPaciente(Index).Text = UCase(Trim(txtDPaciente(Index).Text))
End Sub
```

### **Código de Formulario Registrar Flebotomista**

```
Private tablas As String
Private Sub BtGuardar_Click()
On Error GoTo ControlError
```

```
For i = 0 To txtdatos.Count - 1
If txtdatos(i).Text = "" Then
MsgBox "Ingrese Todos los Datos Solicitados", vbInformation, "Banco de Sangre"
Exit Sub
End If
Next i
```

```
With databs.Connection1
.Execute "INSERT INTO " + tablas + " VALUES(" + txtdatos(2).Text + "," +
txtdatos(0).Text + "," + txtdatos(1).Text + ")"
End With
MsgBox "Datos Guardados", vbInformation, "Banco de Sangre"
Unload Me
Exit Sub
```

```
ControlError:
On Error Resume Next
Select Case Err.Number
Case Err.Number
MsgBox "Error Inesperado:" + Err.Description, vbCritical, "Banco de Sangre"
End Select
Err.Clear
End Sub
```

```
Private Sub Form_Load()
```



```
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
Me.Caption = "Datos de " + Screen.ActiveForm.Tag
tablas = Screen.ActiveForm.Tag
End Sub
```

```
Private Sub txtdatos_LostFocus(Index As Integer)
txtdatos(Index).Text = UCase(txtdatos(Index).Text)
End Sub
```

### Código de Formulario Registrar Médico

```
Private tablas As String
Private Sub BtGuardar_Click()
On Error GoTo ControlError
```

```
For i = 0 To txtdatos.Count - 1
If txtdatos(i).Text = "" Then
MsgBox "Ingrese Todos los Datos Solicitados", vbInformation, "Banco de Sangre"
Exit Sub
End If
Next i
```

```
With databs.Connection1
.Execute "INSERT INTO " + tablas + " VALUES(" + txtdatos(2).Text + "," +
txtdatos(0).Text + "," + txtdatos(1).Text + "," + txtdatos(3).Text + ")"
End With
MsgBox "Datos Guardados", vbInformation, "Banco de Sangre"
Unload Me
Exit Sub
```

```
ControlError:
On Error Resume Next
Select Case Err.Number
Case Err.Number
MsgBox "Error Inesperado:" + Err.Description, vbCritical, "Banco de Sangre"
End Select
Err.Clear
End Sub
```

```
Private Sub Form_Load()
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
Me.Caption = "Datos de " + Screen.ActiveForm.Tag
tablas = Screen.ActiveForm.Tag
End Sub
```



```
Private Sub txtdatos_LostFocus(Index As Integer)
txtdatos(Index).Text = UCase(txtdatos(Index).Text)
End Sub
```

### Código de Formulario Kardex

```
Private Sub btdescartar_Click()
If Me.Tag = "" Then
If MsgBox("Desea descartar la sangre Código: " + DatosDescarte.CodB, vbYesNo) =
vbYes Then
frmdescarte.Show vbModal
End If
With GRID
.ClearStructure
Set .DataSource = databs.Connection1.Execute("SELECT
IF(datediff(fecha_Vencimiento, CURDATE()) >= 0, ' ', 'Vencida') as 'Vencida', Cod_Bolsa,
Guia, Hemocomponente, ABO, Rh, DATE_FORMAT(Fecha_Entrega, '%d/%m/%Y') AS
'Fecha_Entrega', DATE_FORMAT(Fecha_Extraccion, '%d/%m/%Y') AS
'Fecha_Extraccion',
DATE_FORMAT(Fecha_Vencimiento, '%d/%m/%Y') AS 'Fecha_Vencimiento' From
kardex")
.Refresh
End With
End If
End Sub
```

```
Private Sub BtGuardar_Click()
On Error GoTo ControlError
```

```
databs.Connection1.Execute "INSERT INTO kardex VALUES('' +
cbhemocomponente.Text + "','" + cbabo.Text + "','" + cbrh.Text + "','" + txtguia.Text + "','"
+ txtguia.Text + "','" + Format(Calendar1.Text, "yyyy-mm-dd") + "','" +
Format(Calendar2.Text, "yyyy-mm-dd") + "','" + Format(txtfv.Text, "yyyy-mm-dd") + "'") "
```

```
With GRID
.ClearStructure
Set .DataSource = databs.Connection1.Execute("SELECT
IF(datediff(fecha_Vencimiento, CURDATE()) >= 0, ' ', 'Vencida') as 'Vencida', Cod_Bolsa,
Guia, Hemocomponente, ABO, Rh, DATE_FORMAT(Fecha_Entrega, '%dd/%mm/%Y')
AS 'Fecha_Entrega', DATE_FORMAT(Fecha_Extraccion, '%d/%m/%Y') AS
'Fecha_Extracción', DATE_FORMAT(Fecha_Vencimiento, '%d/%m/%Y') AS
'Fecha_Vencimiento' From kardex")
.Refresh
End With
GRID.ColWidth(0) = 500
For i = 1 To GRID.Cols - 1
GRID.ColWidth(i) = 2000
```



```

Next
MsgBox "Datos Guardados", , "Banco de Sangre"
Exit Sub

ControlError:
Select Case Err.Number
Case -2147217887:
MsgBox "Verifique:" _
+ vbNewLine + "1. Valores incorrectos" + vbNewLine + "2. Cajas vacias", vbCritical,
"Error"
Case -2147467259:
MsgBox "Verifique que las fechas sean validas ", vbCritical, "Error"
Case -2147217900:
MsgBox "El Código de Bolsa ya existe", vbCritical, "Error"
Case Err.Number
MsgBox "Error inesperado: " + CStr(Err.Number) + Err.Description, vbCritical, "Error"
End Select
End Sub

Private Sub btlimpiar_Click()
cbabo.Text = ""
cbrh.Text = ""
cbhemocomponente.Text = ""
txtguia.Text = ""
txtfv.Text = "dd/mm/aaaa"
txtbolsa.Text = ""

End Sub

Private Sub Calendar2_LostFocus()
txtfv.Text = CStr(DateAdd("d", 34, Calendar2.Text))
End Sub

Private Sub Form_Load()
Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 3

With GRID
.ClearStructure
Set .DataSource = databs.Connection1.Execute("SELECT
IF(datediff(fecha_Vencimiento, CURDATE()) >= 0, ' ', 'Vencida') as 'Vencida', Cod_Bolsa,
Guia, Hemocomponente, ABO, Rh, DATE_FORMAT(Fecha_Entrega, '%d/%m/%Y') AS
'Fecha_Entrega', DATE_FORMAT(Fecha_Extraccion, '%d/%m/%Y') AS
'Fecha_Extracción', DATE_FORMAT(Fecha_Vencimiento, '%d/%m/%Y') AS
'Fecha_Vencimiento' From kardex")
.Refresh
End With
GRID.ColWidth(0) = 500
For i = 1 To GRID.Cols - 1

```



```
GRID.ColWidth(i) = 2000
```

```
Next
```

```
btlimpiar_Click
```

```
End Sub
```

```
Private Sub GRID_Click()
```

```
With DatosDescarte
```

```
.CodB = GRID.TextMatrix(GRID.Row, 2)
```

```
.Hemo = GRID.TextMatrix(GRID.Row, 4)
```

```
.ABO = GRID.TextMatrix(GRID.Row, 5)
```

```
.rh = GRID.TextMatrix(GRID.Row, 6)
```

```
.Vencida = GRID.TextMatrix(GRID.Row, 1)
```

```
End With
```

```
With GRID
```

```
.ColSel = 1
```

```
.ColSel = (GRID.Cols - 1)
```

```
End With
```

```
End Sub
```

```
Private Sub txtfe_GotFocus()
```

```
txtfe.SelStart = 0
```

```
txtfe.SelLength = Len(txtfe.Text)
```

```
End Sub
```

```
Private Sub txtfe_LostFocus()
```

```
'On Error GoTo ControlError
```

```
txtfe.Text = CStr(DateAdd("d", 0, txtfe.Text))
```

```
txtfe.Text = Format(txtfe.Text, "dd/mm/yyyy")
```

```
Exit Sub
```

```
'ControlError:
```

```
'MsgBox "La fecha de entrega no es Valida", vbExclamation, "Banco de Sangre"
```

```
'txtfe.Text = "dd/mm/aaaa"
```

```
'End Sub
```

```
Private Sub txtfext_GotFocus()
```

```
txtfext.SelStart = 0
```

```
txtfext.SelLength = Len(txtfext.Text)
```

```
End Sub
```

```
Private Sub CALENDAR1_LostFocus()
```

```
'On Error GoTo ControlError
```

```
'txtfext.Text = CStr(Format(txtfext.Text, "dd/mm/yyyy"))
```

```
Exit Sub
```

```
'ControlError:
```

```
'MsgBox "La fecha de extracción no es Valida", vbExclamation, "Banco de Sangre"
```

```
'txtfv.Text = "dd/mm/aaaa"
```



```
'txtfext.Text = "dd/mm/aaaa"
End Sub
```

```
Private Sub txtfv_GotFocus()
txtfv.SelStart = 0
txtfv.SelLength = Len(txtfv.Text)
End Sub
```

```
Private Sub txtfv_LostFocus()
On Error GoTo ControlError
txtfv.Text = CStr(DateAdd("d", 0, txtfv.Text))
txtfv.Text = Format(txtfv.Text, "dd/mm/yyyy")
Exit Sub
ControlError:
MsgBox "La fecha de vencimiento no es Valida", vbExclamation, "Banco de Sangre"
txtfv.Text = "dd/mm/aaaa"
End Sub
```

### **Código de Formulario de Descarte**

```
Private Sub btcancelar_Click()
btguardar.Enabled = False
btcancelar.Enabled = False
btnuevo.Enabled = True
End Sub
```

```
Private Sub BtGuardar_Click()
'On Error GoTo ControlError
MsgBox Calendar1.Text
If cbcausa.ListIndex < 0 Then
MsgBox "Seleccione una causa de descarte", vbInformation, "Banco de Sangre"
Exit Sub
```

```
Elseif CStr(Calendar1.Text) = "" Then
MsgBox " Seleccione la fecha de descarte", vbInformation, "Banco de Sangre"
txtfecha.Text = "dd/mm/aaaa"
Exit Sub
End If
```

```
With databs.Connection1
If (cbcausa.ListCount - 1) = cbcausa.ListIndex Then
frmOtraCausa.Show vbModal
If frmOtraCausa.LoginSucceeded = False Then
Exit Sub
Else
BeginTrans
```



```
.Execute "INSERT INTO descarte VALUES('" + lbcodB.Caption + "','" +
lbHemo.Caption + "','" + lbABO.Caption + "','" + lbRH.Caption + "','" +
Format(txtfecha.Text, "yyyy-mm-dd") + "','" + frmOtraCausa.txtcausa.Text + "')"
.Execute "Delete from kardex where Cod_Bolsa='" + lbcodB.Caption + "'"
.CommitTrans
End If
Unload frmOtraCausa
Else
.BeginTrans
.Execute "INSERT INTO descarte VALUES('" + lbcodB.Caption + "','" + lbHemo.Caption
+ "','" + lbABO.Caption + "','" + lbRH.Caption + "','" + Format(Calendar1.Text, "yyyy-mm-
dd") + "','" + cbcausa.Text + "')"
.Execute "Delete from kardex where Cod_Bolsa='" + lbcodB.Caption + "'"
.CommitTrans
End If

End With
MsgBox "Descarte Realizado", vbInformation, "Banco de Sangre"
Unload Me
Exit Sub

'ControlError:
databs.Connection1.RollbackTrans
Select Case Err.Number
Case -2147217887:
MsgBox "Verifique:" _
+ vbNewLine + "1. Valores incorrectos" + vbNewLine + "2. Cajas vacias", , "Error"
Case -2147217900:
MsgBox "El Código de Bolsa ya existe", , "Error"
Case Err.Number:
MsgBox Err.Description, , "Error"
End Select
cbcausa.ListIndex = -1
Err.Clear
End Sub

Private Sub btnuevo_Click()
btnuevo.Enabled = False
btcancelar.Enabled = True
btguardar.Enabled = True
txtCod_Bolsa.Text = ""
txtfecha.Text = ""
End Sub

Private Sub cbcausa_Click()
If DatosDescarte.Vencida = "Vencida" Then
cbcausa.ListIndex = 4
End If
```



End Sub

```
Private Sub Form_Load()
Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 3
With DatosDescarte
lbABO.Caption = .ABO
lbcodB.Caption = .CodB
lbHemo.Caption = .Hemo
lbRH.Caption = .rh
End With
```

```
If DatosDescarte.Vencida = "Vencida" Then
cbcausa.ListIndex = 4
End If
```

```
'txtfecha.Text
t = Format(Now, "dd/mm/yyyy")
End Sub
```

```
Private Sub txtfecha_LostFocus()
On Error GoTo ControlError
txtfecha.Text = DateAdd("d", 0, txtfecha.Text)
txtfecha.Text = Format(txtfecha.Text, "dd/mm/yyyy")
Exit Sub
ControlError:
MsgBox "Fecha de descarte no valida", vbInformation, "Banco de Sangre"
txtfecha.Text = "dd/mm/aaaa"
End Sub
```

```
Private Sub Text1_GotFocus()
MonthView1.Visible = True
Text1.Text = MonthView1.Value
MsgBox Text1.Text
End Sub
```

### **Código de Formulario de Donante**

```
Dim WithEvents rsDonante As Recordset
```

```
Private Sub btGRh_Click()
Me.Tag = txtdonante(0).Text
frmGSDonante.Show vbModal
End Sub
```

```
Private Sub btbuscar_Click()
cad = InputBox("Cédula del Donante", "Búsqueda del Donante")
If cad = "" Then Exit Sub
If databs.rsDonant.RecordCount = 0 Then Exit Sub
```



```
Pos = databs.rsDonant.Bookmark
databs.rsDonant.MoveFirst
databs.rsDonant.Find "Cedula like '" + Trim(UCCase(cad)) + "'"
If databs.rsDonant.EOF Then databs.rsDonant.Bookmark = Pos
End Sub
```

```
Private Sub btcancelar_Click()
databs.rsDonant.CancelUpdate
btguardar.Enabled = False
btcancelar.Enabled = False
btnuevo.Enabled = True
BTPrimero.Enabled = True
BTSiguiente.Enabled = True
BTAnterior.Enabled = True
BTUltimo.Enabled = True
btbuscar.Enabled = True
```

```
txtdonante(5).Visible = True
End Sub
```

```
Private Sub btDt_Click()
If GRID.Columns.Count > 3 Then
Me.Tag = GRID.Columns(1).Text
```

```
frmGSDonante.Show vbModal
Recargar
End If
End Sub
```

```
Private Sub BtGuardar_Click()
On Error GoTo ControlError
```

```
Código = GenerarNumRegistro()
databs.rsDonant("Cod_Donante").Value = Código
databs.rsDonant("Fecha Nacimiento").Value = Format(Calendar1.Text, "yyyy-mm-dd")
databs.rsDonant.Update
btguardar.Enabled = False
btcancelar.Enabled = False
btnuevo.Enabled = True
Calendar1.Visible = True
Calendar1.Visible = True
txtdonante(5).Visible = True
BTPrimero.Enabled = True
BTSiguiente.Enabled = True
BTAnterior.Enabled = True
BTUltimo.Enabled = True
btbuscar.Enabled = True
MsgBox "Datos Guardados", vbInformation, "Banco de Sangre"
```



```
Recargar
Exit Sub
ControlError:
Select Case Err.Number
Case -2147217900
MsgBox "Paciente ya registrado", vbInformation, "Banco de Sangre"
Case Err.Number
MsgBox "Ingrese Todos los datos Solicitados correctamente", vbInformation, "Banco de Sangre"
End Select
End Sub
```

```
Private Sub btlmpiar_Click()
txtdonante(0).Text = ""
txtdonante(1).Text = ""
txtdonante(2).Text = ""
txtdonante(3).Text = ""
txtdonante(4).Text = ""
cbsexo.Text = ""
cbtipoD.Text = ""
End Sub
```

```
Private Sub btresultado_Click()
If GRID.VisibleRows = 1 Then Exit Sub
Me.Tag = GRID.Columns(1).Text
frmResultadoD.Show vbModal
End Sub
```

```
Sub Recargar()
Dim txt As TextBox
With txtdonante
databs.rsDonant.Requery
For Each txt In txtdonante
Set txt.DataSource = databs.rsDonant
Next
End With
GRID.Refresh
Set cbsexo.DataSource = databs.rsDonant
Set cbtipoD.DataSource = databs.rsDonant
End Sub
```

```
Private Sub btnuevo_Click()
BTPrimero.Enabled = False
BTSiguiente.Enabled = False
BTAnterior.Enabled = False
BTUltimo.Enabled = False
btbuscar.Enabled = False
btguardar.Enabled = True
```



```
btcancelar.Enabled = True  
btnuevo.Enabled = False  
txtdonante(5).Visible = False  
databs.rsDonant.AddNew
```

End Sub

```
Private Sub BTPrimero_Click()  
If databs.rsDonant.RecordCount = 0 Then Exit Sub  
databs.rsDonant.MoveFirst  
End Sub
```

```
Private Sub BTSiguiente_Click()  
If databs.rsDonant.RecordCount = 0 Then Exit Sub  
databs.rsDonant.MoveNext  
If databs.rsDonant.EOF Then  
databs.rsDonant.MovePrevious  
End If  
End Sub
```

```
Private Sub btultimo_Click()  
If databs.rsDonant.RecordCount = 0 Then Exit Sub  
databs.rsDonant.MoveLast  
End Sub
```

```
Private Sub BTAnterior_Click()  
If databs.rsDonant.RecordCount = 0 Then Exit Sub  
databs.rsDonant.MovePrevious  
If databs.rsDonant.BOF Then  
databs.rsDonant.MoveNext  
End If  
End Sub
```

```
Private Sub Form_Load()  
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2  
btguardar.Enabled = False  
btcancelar.Enabled = False  
btnuevo.Enabled = True
```

End Sub

```
Private Sub Grid_AfterUpdate()  
GRID.Refresh  
End Sub
```

```
Private Sub txtDonante_LostFocus(Index As Integer)  
txtdonante(Index).Text = Trim(UCase(txtdonante(Index).Text))  
End Sub
```



## Código de Formulario Solicitudes Pendientes

Private seleccionado As Boolean

Public Sub btactualizar\_Click()  
Dim lb As Label

On Error Resume Next  
databs.rsDatosS.Requery

With Grid1  
Set .DataSource = Nothing  
Set .DataSource = databs  
.Refresh  
End With

For Each lb In lbdpaciente  
With lb  
Set .DataSource = Nothing  
.Caption = ""  
Set .DataSource = databs  
.Refresh  
End With  
Next

For Each lb In LBtransF  
With lb  
Set .DataSource = Nothing  
.Caption = ""  
Set .DataSource = databs  
.Refresh  
End With  
Next

seleccionado = False  
End Sub

Private Sub btanular\_Click()  
On Error Resume Next  
If databs.rsDatosS.RecordCount = 0 Then  
Exit Sub  
End If  
If vbYes = MsgBox("Realmente desea anular la solicitud", vbInformation + vbYesNo,  
"Banco de Sangre") Then  
databs.Connection1.Execute "UPDATE Solicitud set Pendiente=false where Solicitud=" +  
CStr(databs.rsDatosS("Solicitud").Value)  
MsgBox "Solicitud eliminada", vbInformation, "Banco de Sangre"



```
btactualizar_Click
End If
End Sub
```

```
Private Sub btgs_Click()
Dim strABO, strRh As String
If databs.rsDatosS.RecordCount = 0 Then Exit Sub
```

```
If seleccionado = False Then
MsgBox "Seleccione un elemento de las solicitudes pendiente", vbOKOnly +
vbMsgBoxHelpButton, "Error"
Exit Sub
End If
```

```
If databs.rsDatosS("Tipo_Solicitud").Value = "Transfusión" Then
```

```
strABO = Trim(UCCase(Mid(LBtransF(2).Caption, 1, 2)))
strRh = Trim(UCCase(Mid(LBtransF(2).Caption, 3, Len(LBtransF(2).Caption))))
```

```
Set rs = databs.Connection1.Execute("SELECT grupo_sanguineo.Cod_GrupoS FROM
paciente, solicitud, grupo_sanguineo, preparaciongeneral, gsp WHERE
(preparacionGeneral.solicitud = solicitud.solicitud) AND (Gsp.Cod_IPG =
preparacionGeneral.Cod_IPG) AND (solicitud.Expediente = paciente.Expediente) AND
(GSP.cod_GrupoS = grupo_sanguineo.cod_GrupoS) AND (paciente.Expediente = "" +
lbdpaciente(2).Caption + "") AND ({ fn CONCAT(grupo_sanguineo.Grupo, ' ',
grupo_sanguineo.Rh) } = "" + LBtransF(2).Caption + ""))")
```

```
If rs.EOF And rs.BOF Then
MsgBox "No se a Registrado Rh "
Me.Tag = databs.rsDatosS("Solicitud").Value
frmGSPaciente.Show vbModal
End If
```

```
rs.Requery
If rs.EOF And rs.BOF Then
seleccionado = False
Exit Sub
End If
```

```
Me.Tag = databs.rsDatosS("Solicitud").Value
If LBtransF(3).Caption = "PFC" Or LBtransF(3).Caption = "PS" Then
frmPCMEA.Show vbModal
Elseif LBtransF(3).Caption = "ST" Then
frmPCMAMEA.Show vbModal
Elseif LBtransF(3).Caption = "PG" Then
frmPCMAA.Show vbModal
Elseif LBtransF(3).Caption = "CRIO" Or LBtransF(3).Caption = "PLQ" Then
frmCRIO_PLQ.Show vbModal
```



```

End If
Elseif databs.rsDatosS("Tipo_Solicitud").Value = "Coombs Directo" Then
Me.Tag = databs.rsDatosS("Solicitud").Value
frmCoombsD.Show vbModal
Elseif databs.rsDatosS("Tipo_Solicitud").Value = "Coombs Indirecto" Then
Me.Tag = databs.rsDatosS("Solicitud").Value
frmCoombsI.Show vbModal
Elseif databs.rsDatosS("Tipo_Solicitud").Value = "Tipificación" Then
Me.Tag = databs.rsDatosS("Solicitud").Value
frmGSPaciente.Show vbModal
End If
btactualizar_Click
seleccionado = False
End Sub

```

```

Private Sub Form_Load()
Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
Grid1.ColWidth(0) = 500
For i = 1 To Grid1.Cols - 1
Grid1.ColWidth(i) = 2000
Next
Grid1.Row = 0
End Sub

```

```

Private Sub Grid1_Click()
If databs.rsDatosS.RecordCount > 0 Then
If Grid1.Row > 0 Then
databs.rsDatosS.AbsolutePosition = Grid1.Row
End If
End If
seleccionado = True
End Sub

```

```

Private Sub Grid1_SelChange()
If databs.rsDatosS.RecordCount > 0 Then
databs.rsDatosS.AbsolutePosition = Grid1.Row
End If
seleccionado = True
End Sub

```

```

Private Sub lbdPaciente_Change(Index As Integer)
On Error Resume Next
If Index = 4 Then
If IsDate(lbdpaciente(Index).Caption) Then
lbdEdad.Caption = DateDiff("yyyy", lbdpaciente(Index).Caption, Now)
Else
lbdpaciente(Index).Caption = ""
End If
End If

```



End If  
End Sub

### Código de Formulario Prueba Cruzada(Mayor y Autocontrol)

Private RsBolsa As Recordset

Private Sub btcancelar\_Click()  
Unload Me  
End Sub

Private Function CargarListas()  
Dim i As Integer, Texto, strABO, strRh As String  
Texto = frmppendiente.LBtransF(2).Caption

strABO = Trim(UCase(Mid(Texto, 1, 2)))  
strRh = Trim(UCase(Mid(Texto, 3, Len(Texto))))

Set RsBolsa = databs.Connection1.Execute("select Cod\_Bolsa,CONCAT(' Fecha de Vencimiento : ',DATE\_FORMAT(Fecha\_Vencimiento, '%d/%m/%Y'),' Fecha de Extraccion : ', DATE\_FORMAT(Fecha\_Extraccion, '%d/%m/%Y'),'Guia: ',Guia )AS Datos, Guia from kardex where datediff(fecha\_Vencimiento, CURDATE()) >(-1) and ABO='" + strABO + "' and Rh='" + strRh + "' and Hemocomponente='" + frmppendiente.LBtransF(3).Caption + "'" + "ORDER BY Fecha\_Vencimiento")

With dcbolsa  
.Clear  
If RsBolsa.RecordCount > 0 Then  
i = 0  
While Not RsBolsa.EOF  
i = i + 1  
.AddItem RsBolsa("Cod\_Bolsa").Value  
.ItemData(.NewIndex) = i  
RsBolsa.MoveNext  
Wend  
End If  
End With  
End Function

Private Function CargarAnalistas()  
Set rs = databs.Connection1.Execute("Select concat(Nombre, ' ',Apellido)as Nombre,CarnetA from Analistas")  
With CbAnalista  
.Item(0).Clear  
.Item(1).Clear  
While Not rs.EOF  
.Item(0).AddItem rs("Nombre").Value  
.Item(1).AddItem rs("CarnetA").Value



```
rs.MoveNext
Wend
End With
End Function
```

```
Private Sub BtGuardar_Click()
Dim VarCódigo As String
'On Error GoTo ControlError
Dim rs As ADODB.Recordset
```

```
For Each CB In mayor
If CB.Text = "" Then
MsgBox "Ingrese Todos los datos en Mayor", vbInformation, "Banco de Sangre"
Exit Sub
End If
Next
```

```
For Each CB In autocontrol
If CB.Text = "" Then
MsgBox "Ingrese Todos los datos en Autocontrol", vbInformation, "Banco de Sangre"
Exit Sub
End If
Next
```

```
If cbresultado.Text = "" Then
MsgBox "Ingrese el resultado", vbInformation, "Banco de Sangre"
Exit Sub
End If
```

```
If dcbolsa.ListIndex < 0 Then
MsgBox "Seleccione el Código de bolsa a utilizar", vbInformation, "Banco de Sangre"
Exit Sub
End If
```

```
If Not IsNumeric(txthemoglobina.Text) Then
MsgBox "La Hemoglobina debe ser numerica", vbInformation, "Banco de Sangre"
Exit Sub
End If
```

```
If Not IsNumeric(txtpeso.Text) Then
MsgBox "El peso debe ser numerico", vbInformation, "Banco de Sangre"
Exit Sub
Elseif CInt(txtpeso.Text) <= 0 Then
MsgBox "El peso no es valido", vbInformation, "Banco de Sangre"
Exit Sub
End If
```

```
If Not IsNumeric(txtcantidad.Text) Then
MsgBox "La cantidad debe ser numerica", vbInformation, "Banco de Sangre"
```



```

Exit Sub
Elseif CInt(txtcantidad.Text) <= 0 Then
MsgBox "La cantidad no es valida", vbInformation, "Banco de Sangre"
Exit Sub
End If

If Not IsDate(mkhp.Text) Then
MsgBox "Ingrese correctamente la Hora", vbInformation, "Banco de sangre"
Exit Sub End If

If CbAnalista(0).ListIndex < 0 Then
MsgBox "seleccione en anlista ", vbInformation, "Banco de sangre"
Exit Sub
End If

Set rs = databs.Connection1.Execute("SELECT VNPreparado,Volumen,Transfusion from
Transfusion WHERE Solicitud=" + Me.Tag)

If rs.Fields("VNPreparado").Value < CInt(txtcantidad.Text) Then
MsgBox "La cantidad Preparada es mayor que el volumen Pendiente", vbInformation,
"Banco de Sangre"
Exit Sub
End If

VarCódigo = GenerarNumRegistro()
With databs.Connection1
.BeginTrans
.Execute "INSERT INTO `sangre_preparada` SELECT " + "SP" + VarCódigo +
",kardex.*," + txtcantidad.Text + "," + txtpeso.Text + " From kardex Where
kardex.cod_Bolsa =" + dcbolsa.Text + ""
.Execute "INSERT INTO preparaciontransfusion VALUES(" +
CStr(rs("Transfusion").Value) + "," + CbAnalista(1).Text + "," + "SP" + VarCódigo + "," +
"PT" + VarCódigo + "," + mkhp.Text + "," + Format(Calendar1.Text, "yyyy-mm-dd") + "," +
txthemoglobina.Text + ")"

.Execute "INSERT INTO prueba_cruzada VALUES(" + "PT" + VarCódigo + "," +
mayor(0).Text + "," + mayor(1).Text + "," + mayor(2).Text + "," + mayor(3).Text +
", 'Mayor'," + cbresultado + ")"
.Execute "INSERT INTO prueba_cruzada VALUES(" + "PT" + VarCódigo + "," +
autocontrol(0).Text + "," + autocontrol(1).Text + "," + autocontrol(2).Text + "," +
autocontrol(3).Text + ", 'Autocontrol'," + cbresultado + ")"

.Execute "DELETE FROM kardex WHERE Cod_Bolsa IN (SELECT Cod_Bolsa FROM
Sangre_preparada)"
.Execute "UPDATE Transfusion SET VNPreparado=" + CStr(rs("VNPreparado").Value -
CInt(txtcantidad.Text)) + " WHERE Transfusion.Solicitud=" + Me.Tag

```



```

Set rs = .Execute("SELECT VNPreparado from Transfusion WHERE Solicitud=" +
Me.Tag)

If rs.Fields("VNPreparado").Value = 0 Then
.Execute "UPDATE Solicitud SET Pendiente=false WHERE Solicitud=" + Me.Tag
End If

.CommitTrans
End With
MsgBox "Datos Guardados", , "Banco de Sangre"
CargarListas
frmpendiente.btactualizar_Click

If rs.Fields("VNPreparado").Value > 0 Then
databs.rsDatosS.AbsolutePosition = frmppendiente.Grid1.Row
Else
Unload Me
End If
Exit Sub

'ControlError:
databs.Connection1.RollbackTrans
Select Case Err.Number
Case -2147217887:
MsgBox "Verifique:" + vbNewLine + "1. Valores incorrectos" + vbNewLine + "2. Cajas
vacias", , "Error"
Case -2147217900:
MsgBox "El Código de Bolsa ya existe", vbCritical, "Error"
Case Err.Number:
MsgBox Err.Source + Err.Description, vbCritical, "Error"
End Select
End Sub

Private Sub cbanalista_Click(Index As Integer)
CbAnalista(1).ListIndex = CbAnalista(0).ListIndex
End Sub

Private Sub cbanalista_DbIclicK(Index As Integer)
Me.Tag = "analistas"
frmFleme.Show vbModal
CargarAnalistas
End Sub

Private Sub dcbolsa_Click()
If RsBolsa.RecordCount > 0 Then
RsBolsa.AbsolutePosition = dcbolsa.ItemData(dcbolsa.ListIndex)
txtdatos.Text = RsBolsa("Datos").Value
End If

```



End Sub

```
Private Sub Form_Load()  
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2  
Me.Tag = frmppendiente.Tag  
CargarListas  
CargarAnalistas  
End Sub
```

```
Private Sub mkfp_GotFocus()  
mkfp.SelStart = 0  
End Sub
```

```
Private Sub mkhp_GotFocus()  
mkhp.SelStart = 0  
End Sub  
Private Sub txtanalista_GotFocus()  
txtanalista.SelStart = 0  
End Sub
```

```
Private Sub mkfp_Validate(Cancel As Boolean)  
Beep  
If Not IsDate(mkfp.Text) Then  
MsgBox "Fecha no valida", , "Error"  
Cancel = True  
End If  
End Sub
```

```
Private Sub mkhp_Validate(Cancel As Boolean)  
Beep  
If Not IsDate(mkhp.Text) Then  
MsgBox "hora no valida" + vbNewLine + "Formato:hh:mm AM/PM", , "Error"  
Cancel = True  
End If  
End Sub
```

```
Private Sub txtanalista_LostFocus()  
txtanalista.Text = UCase(Trim(txtanalista.Text))  
End Sub
```

```
Private Sub btAgregA_Click()  
Me.Tag = "analistas"  
frmFleme.Show vbModal  
CargarAnalistas  
End Sub
```

### **Código de Formulario Coombs Directo(Coombs Directo y Autocontrol**



```
Dim rs As Recordset, VarSolicitud As String
Private Sub btAgegA_Click()
Me.Tag = "analistas"
frmFleme.Show vbModal
CargarAnalistas
End Sub
```

```
Private Function CargarAnalistas()
Set rs = databs.Connection1.Execute("Select concat(Nombre,' ',Apellido)as
Nombre,CarnetA from Analistas")
With CbAnalista
.Item(0).Clear
.Item(1).Clear
While Not rs.EOF
.Item(0).AddItem rs("Nombre").Value
.Item(1).AddItem rs("CarnetA").Value
rs.MoveNext
Wend
End With
End Function
```

```
Private Sub btcancelar_Click()
Unload Me
End Sub
```

```
Private Sub BtGuardar_Click()
Dim CB As ComboBox, VarCódigo As String
'On Error GoTo ControlError
```

```
For Each CB In mayor
If CB.Text = "" Then
MsgBox "Ingrese Todos los datos en coombs Directo", vbInformation, "Banco de Sangre"
Exit Sub
End If
Next
```

```
For Each CB In autocontrol
If CB.Text = "" Then
MsgBox "Ingrese Todos los datos en Autocontrol", vbInformation, "Banco de Sangre"
Exit Sub
End If
Next
```

```
If cbresultado.Text = "" Then
MsgBox "Ingrese el resultado", vbInformation, "Banco de Sangre"
Exit Sub
End If
```



```

If Not IsDate(mkhp.Text) Then
MsgBox "Ingrese correctamente la Hora", vbInformation, "Banco de sangre"
Exit Sub
End If

If CbAnalista(0).ListIndex < 0 Then
MsgBox "seleccione el analista ", vbInformation, "Banco de sangre"
Exit Sub
End If
VarCódigo = GenerarNumRegistro()
With databs.Connection1
.BeginTrans
.Execute "INSERT INTO preparaciongeneral VALUES(" + VarSolicitud + "," +
CbAnalista(1).Text + "," + "PG" + VarCódigo + "," + mkhp.Text + "," +
Format(Calendar1.Text, "yyyy-mm-dd") + ")"
.Execute "INSERT INTO coombs VALUES(" + "PG" + VarCódigo + "," +
mayor.Item(1).Text + "," + mayor.Item(2).Text + "," + mayor.Item(3).Text + "," + 'Coombs
Directo', " + cbresultado.Text + ")"
.Execute "INSERT INTO coombs VALUES(" + "PG" + VarCódigo + "," +
autocontrol(1).Text + "," + autocontrol(2).Text + "," + autocontrol(3).Text +
", 'Autocontrol', " + cbresultado.Text
+ ")"
.Execute "UPDATE Solicitud SET Pendiente=false WHERE Solicitud=" + VarSolicitud
.CommitTrans
End With

MsgBox "Datos Guardados", , "Banco de Sangre"
Unload Me
Exit Sub

'ControlError:
databs.Connection1.RollbackTrans
Select Case Err.Number
Case -2147217887:
MsgBox "Verifique:" _
+ vbNewLine + "1. Valores incorrectos" + vbNewLine + "2. Cajas vacias", , "Error"
Case -2147217900:
MsgBox "El Código de Bolsa ya existe", , "Error"
Case Err.Number:
MsgBox Err.Description, , "Error"
End Select

End Sub

Private Sub cbanalista_Click(Index As Integer)
CbAnalista(1).ListIndex = CbAnalista(0).ListIndex
End Sub

```



```

Private Sub Form_Load()
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
VarSolicitud = frmppendiente.Tag
CargarAnalistas
End Sub
Private Sub mkfp_GotFocus()
mkfp.SelStart = 0
End Sub

Private Sub mkhp_GotFocus()
mkhp.SelStart = 0
End Sub

Private Sub txtanalista_GotFocus()
txtanalista.SelStart = 0
End Sub

Private Sub txtanalista_LostFocus()
txtanalista.Text = UCase(Trim(txtanalista.Text)) End Sub
    
```

### **Código de Formulario Sangre Próxima a vencerse**

```

Private Function RecargarDatos()
Dim FL As Field
With GRID
Set .DataSource = databs.Connection1.Execute("SELECT Hemocomponente, ABO, Rh,
Cod_Bolsa, Guia, DATE_FORMAT(Fecha_Entrega, '%d/%m/%Y') AS 'Fecha de
Entrega', DATE_FORMAT(Fecha_Extraccion, '%d/%m/%Y') AS 'Fecha de Extraccion',
DATE_FORMAT(Fecha_Vencimiento, '%d/%m/%Y') AS 'Fecha de Vencimiento' From
kardex WHERE datediff(fecha_Vencimiento, CURDATE()) < 8 AND
datediff(fecha_Vencimiento, CURDATE()) > 0")
.Refresh
End With
End Function
Private Sub Form_Load()
Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
RecargarDatos
GRID.ColWidth(0) = 500
For i = 1 To GRID.Cols - 1
GRID.ColWidth(i) = 2000
Next
End Sub
    
```

### **Código de Sangre Vencida**



```
Private rs As Recordset
Private varDS As Integer
Private Rowinicio As Integer
```

```
Private Sub btaceptar_Click()
On Error GoTo ControlError
```

```
If Not IsDate(txtfecha.Text) Then
MsgBox "La fecha no es valida", vbInformation, "Banco de Sangre"
Exit Sub
End If
```

```
Frame1.Visible = False
```

```
Select Case varDS
Case 0
For i = 1 To (GRID.Rows - 1)
With databs.Connection1
.BeginTrans
.Execute "INSERT INTO descarte VALUES('" + GRID.TextMatrix(i, 4) + "','" +
GRID.TextMatrix(i, 1) + "','" + GRID.TextMatrix(i, 2) + "','" + GRID.TextMatrix(i, 3) + "','" +
Format(txtfecha.Text, "yyyy-mm-dd") + "','" + Vencimiento) "
.Execute "delete from kardex where Cod_Bolsa= '" + GRID.TextMatrix(i, 4) + "'"
.CommitTrans
End With
Next i
```

```
Case 1
If GRID.ColSel < (GRID.Cols - 1) Then
MsgBox "Seleccione las Filas a eliminar", vbInformation, "Banco de Sangre"
Exit Sub
End If
```

```
For i = GRID.Row To GRID.RowSel
With databs.Connection1
.BeginTrans
.Execute "INSERT INTO descarte VALUES('" + GRID.TextMatrix(i, 4) + "','" +
GRID.TextMatrix(i, 1) + "','" + GRID.TextMatrix(i, 2) + "','" + GRID.TextMatrix(i, 3) + "','" +
Format(txtfecha.Text, "yyyy-mm-dd") + "','" + Vencimiento) "
.Execute "DELETE FROM kardex WHERE Cod_Bolsa= '" + GRID.TextMatrix(i, 4) + "'"
.CommitTrans
End With
Next i
End Select
```

```
Set rs = databs.Connection1.Execute("SELECT Hemocomponente, ABO, Rh,
Cod_Bolsa, Guia, DATE_FORMAT.Fecha_Entrega, '%d/%m/%Y') AS 'Fecha de
Entrega', DATE_FORMAT.Fecha_Extraccion, '%d/%m/%Y') AS 'Fecha de Extraccion',
```



```
DATE_FORMAT(Fecha_Vencimiento, '%d/%m/%Y') AS 'Fecha de Vencimiento'
From kardex WHERE datediff(Fecha_Vencimiento, CURDATE()) < 0")
```

```
Set GRID.DataSource = rs
GRID.Refresh
```

```
Exit Sub
```

```
ControlError:
```

```
databs.Connection1.RollbackTrans
```

```
MsgBox Err.Description, vbExclamation, "Banco de Sangre"
```

```
End Sub
```

```
Private Sub btcancelar_Click()
```

```
Frame1.Visible = False
```

```
End Sub
```

```
Private Sub btDt_Click(Index As Integer)
```

```
Select Case Index
```

```
Case 0
```

```
varDS = Index
```

```
Case 1
```

```
varDS = Index
```

```
End Select
```

```
Frame1.Visible = True
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
```

```
Set rs = databs.Connection1.Execute("SELECT Hemocomponente, ABO, Rh,
Cod_Bolsa, Guia, DATE_FORMAT(Fecha_Entrega, '%d/%m/%Y') AS 'Fecha de
Entrega', DATE_FORMAT(Fecha_Extraccion, '%d/%m/%Y') AS 'Fecha de Extraccion',
DATE_FORMAT(Fecha_Vencimiento, '%d/%m/%Y') AS 'Fecha de Vencimiento' From
kardex WHERE datediff(Fecha_Vencimiento, CURDATE()) < 0")
```

```
Set GRID.DataSource = rs
```

```
GRID.Refresh
```

```
GRID.ColWidth(0) = 500
```

```
For i = 1 To GRID.Cols - 1
```

```
GRID.ColWidth(i) = 2000
```

```
Next
```

```
End Sub
```

```
Private Sub Frame1_DragOver(Source As Control, X As Single, Y As Single, State As
Integer)
```

```
If State = vbLeave Then
```

```
Frame1.Move X, Y
```

```
End If
```

```
End Sub
```



```
Private Sub txtfecha_Click()  
On Error GoTo ControlError
```

```
Exit Sub  
ControlError:  
MsgBox "Fecha no valida", vbInformation, "Banco de Sangre"  
End Sub
```

### **Código de Formulario Resultado de Pruebas de Coombs Indirecto**

```
Private Sub btactualizar_Click()  
Dim lbDP As Label  
On Error Resume Next  
databs.rsResultadoCI.Requery
```

```
For Each lbDP In lbDpaciente  
With lbDP  
Set .DataSource = Nothing  
Set .DataSource = databs  
.Refresh  
End With  
Next
```

```
Set Label1.DataSource = Nothing  
Set Label1.DataSource = databs
```

```
With GRID  
Set .DataSource = Nothing  
Set .DataSource = databs  
Refresh  
End With  
End Sub  
Private Sub btbuscar_Click()
```

```
If databs.rsResultadoCI.RecordCount > 0 Then  
databs.rsResultadoCI.MoveFirst  
Else  
Exit Sub  
End If
```

```
Pos = databs.rsResultadoCI.Bookmark
```

```
Select Case cbbuscar.ListIndex  
Case 0  
databs.rsResultadoCI.Find "Nombre_Paciente like '" + txtbuscar.Text + "'"<br>Case 1  
databs.rsResultadoCI.Find "Expediente like '" + txtbuscar.Text + "'"
```



End Select

```
If databs.rsResultadoCI.EOF Then
MsgBox cbbuscar.Text + UCase(" no encontrado"), vbInformation, "Banco de Sangre"
databs.rsResultadoCI.Bookmark = Pos
End If
End Sub
```

```
Private Sub BTEntregar_Click()
frmImpRCI.Show vbModal
End Sub
```

```
Private Sub Form_Load()
Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
GRID.ColWidth(0) = 500
For i = 1 To GRID.Cols - 1
GRID.ColWidth(i) = 2000
Next
End Sub
```

```
Private Sub GRID_Click()
If databs.rsResultadoCI.RecordCount > 0 Then
databs.rsResultadoCI.AbsolutePosition = GRID.Row
End If
End Sub
```

```
Private Sub Grid_SelChange()
If databs.rsResultadoCI.RecordCount > 0 Then
databs.rsResultadoCI.AbsolutePosition = GRID.Row
End If
End Sub
Private Sub Label1_Change()
On Error Resume Next
If IsDate(Label1.Caption) Then
lbedad.Caption = DateDiff("yyyy", Label1.Caption, Now)
Else
Label11.Caption = ""
End If
End Sub
```

### **Código de Formulario para Entrega de resultado de Coombs Indirecto**

```
Private Sub btImprimir_Click()
Dim AuxColor
On Error GoTo ControlError
```

```
With txtHospital
```



```

.BorderStyle = 0
AuxColor = .ForeColor
.ForeColor = &H40&
.Text = "Hospital - " + Trim(.Text)
End With

btimprimir.Visible = False
Picture2.SetFocus
Picture1.AutoRedraw = True
rv = SendMessage(Picture2.hWnd, WM_PAINT, Picture1.hDC, 0)
rv = SendMessage(Picture2.hWnd, WM_PRINT, Picture1.hDC, _
PRF_CHILDREN + PRF_CLIENT + PRF_OWNED)
Picture1.Picture = Picture1.Image
Picture1.AutoRedraw = False

Printer.Copies = 1
Printer.Zoom = 90
Printer.FillColor = vbWhite
Printer.PrintQuality = 300
Printer.ScaleMode = Picture1.ScaleMode
Printer.Print ""
Printer.PaintPicture Picture1.Picture, (Printer.ScaleWidth - Me.ScaleWidth) / 2, 0
Printer.PaintPicture Picture1.Picture, (Printer.ScaleWidth - Me.ScaleWidth) / 2,
(Printer.ScaleHeight / 2)
Printer.EndDoc

btimprimir.Visible = True
With txtHospital
.BorderStyle = 1
.ForeColor = AuxColor
End With
Exit Sub

ControlError:
btimprimir.Visible = True
With txtHospital
.BorderStyle = 1
.ForeColor = AuxColor
End With

MsgBox "Erro: " + Err.Description, vbCritical, "Error"
End Sub

Private Sub Form_Load()
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
With lbnombre

End With

```



End Sub

```
Private Sub Form_Resize()
Picture1.Move 0, 0, Me.ScaleWidth, Me.ScaleHeight
Picture2.Move 0, 0, Me.ScaleWidth, Me.ScaleHeight
End Sub
```

```
Private Sub txtHospital_Validate(Cancel As Boolean)
If txtHospital = "" Then
MsgBox "Ingrese el nombre del hospital", vbInformation, "Banco de Sangre"
Cancel = True
Exit Sub End If End Sub
```

### Código de Formulario Visualizar Datos

```
Private mNode As Node ' Variable de nodo a nivel de módulo.
Private mListem As ListItem ' Variable ListItem a nivel de módulo.
Private EventFlag As Integer ' Para señalar qué evento se ha producido.
Private mCurrentIndex As Integer ' Indicador para asegurar que no se ha hecho clic en el
nodo.
Private mStatusBarStyle As Integer ' Cambia el estilo de la barra de estado.
Private rs As ADODB.Recordset ' Consultas Globales.
```

```
Private Sub Form_Load()
Dim bd As DAO.Database
Dim tbl As DAO.TableDef
Dim prt As DAO.Property
Dim fld As DAO.Field
On Error Resume Next
Me.MousePointer = vbHourglass
Set bd = OpenDatabase("bancosangre", 0, 0, CadenaConexion)
' Abre primero el objeto Connection global.
With twdDB
.Sorted = True
.LabelEdit = False
.LineStyle = twwRootLines
.Nodes.Add , , "BD", bd.Name, "BD"

For Each tbl In bd.TableDefs
.Nodes.Add "BD", twwChild, tbl.Name, tbl.Name, "Table"

For Each fld In tbl.Fields
.Nodes.Add tbl.Name, twwChild, tbl.Name + fld.Name, fld.Name, "Field"
For Each prt In prt.Properties
Var = prt.Value
If Err.Number = 0 Then
```



```
.Nodes.Add tbl.Name + fld.Name, twChild, , prt.Name + "=" + CStr(prt.Value),
"Property"
```

```
Else: .Nodes.Add tbl.Name + fld.Name, twChild, , "N/A", "Property"
```

```
End If
```

```
Err.Clear
```

```
Next
```

```
Next
```

```
Next
```

```
End With
```

```
Me.MousePointer = vbNormal
```

```
Exit Sub
```

```
End Sub
```

```
Private Sub lvwDB_BeforeLabelEdit(Cancel As Integer)
```

```
lvwDB.SortKey = 1
```

```
End Sub
```

```
Private Sub twwDB_NodeClick(ByVal Node As MSComctlLib.Node)
```

```
Dim rs As Recordset
```

```
Dim ItmX As ListItem
```

```
On Error GoTo ControlError
```

```
Me.MousePointer = vbHourglass
```

```
If databs.Connection1.State = 0 Then
```

```
databs.Connection1.Open
```

```
End If
```

```
If Node.Parent.Key <> "BD" Then
```

```
Me.MousePointer = vbNormal
```

```
Exit Sub
```

```
End If
```

```
Set rs = databs.Connection1.Execute("SELECT * FROM " + Node.Text)
```

```
' Borrar la colección ColumnHeaders.
```

```
With lvwDB
```

```
.ListItems.Clear
```

```
.View = lvwReport
```

```
.ColumnHeaders.Clear
```

```
' Crear columnas de cabecera
```

```
For i = 0 To rs.Fields.Count - 1
```

```
.ColumnHeaders.Add , , rs.Fields(i).Name, 1800
```

```
Next
```

```
With prgLoad
```

```
.Value = 0
```

```
.Visible = True
```



```

End With
'CargarDatos en le listV
For i = 0 To rs.RecordCount - 1
prgLoad.Value = rs.AbsolutePosition

Set ItmX = .ListItems.Add( , CStr(rs(rs.Fields(0).Name).Value))
For j = 1 To rs.Fields.Count - 1
ItmX.SubItems(j) = CStr(rs(rs.Fields(j).Name).Value)
Next
rs.MoveNext
Next
With prgLoad
.Visible = False
End With
.Refresh
End With
Node.Sorted = True
Me.MousePointer = vbNormal
ControlError:
Me.MousePointer = vbNormal
End Sub
    
```

### **Código de Formulario Personal Registrado**

```

Private Sub Form_Load()
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2
End Sub

Private Sub GridPersonal_AfterColUpdate(Index As Integer, ByVal ColIndex As Integer)
GridPersonal(Index).Columns(ColIndex).Value =
Trim(UCase(GridPersonal(Index).Columns(ColIndex).Value))
End Sub
    
```

### **Código de Informe de Total de Transfusiones**

```

Private Function VisuBt(bool As Boolean)
LBUNIDAD.Visible = bool
LBMUNI.Visible = bool
LBDEPA.Visible = bool
Label3.Visible = bool
Label8.Visible = bool

txthospital.Visible = bool
TXTDEPARTAMENTO.Visible = bool
txtmunicipio.Visible = bool
    
```



```
Frame1.Visible = bool
```

```
btImprimir.Visible = bool
```

```
btprevia.Visible = bool
```

```
End Function
```

```
Private Sub btGInf_Click()
```

```
Dim VarAncho, varalto, VarTop, VarLeft As Single
```

```
Dim i, aux, aux2, total, aux3 As Integer
```

```
aux3 = 0
```

```
total = 0
```

```
NombreCompl = Array("Paquete Globular", "Plasma Fresco", "Plasma Simple",  
"Plaquetas", "Crioprecipitado", "Sangre Total", "Exanguinotransfusion", "Total")
```

```
arraycomponentes = Array("PG", "PFC", "PS", "PLQ", "CRIO", "ST",  
"Exanguinotransfusion", "Total")
```

```
TotalL = Array(0, 0, 0, 0, 0, 0, 0, 0)
```

```
TotalG = Array(0, 0, 0, 0, 0, 0, 0, 0)
```

```
Picture1.Cls
```

```
With Adodc1
```

```
.ConnectionString = CadenaConexion
```

```
.CommandType = adCmdText
```

```
End With
```

```
If cbTI.Text = "" Then
```

```
MsgBox "Seleccione el tipo de Informe", vbCritical + vbOKOnly, "Error"
```

```
Exit Sub
```

```
Elseif cbTI.Text = "Anual" Then
```

```
If txtanyo.Text = "" Then
```

```
MsgBox "Ingrese el Año"
```

```
Exit Sub
```

```
End If
```

```
Else
```

```
If txtanyo.Text = "" Then
```

```
MsgBox "Ingrese El Año"
```

```
Exit Sub
```

```
End If
```

```
If cbmeses.Text = "" Then
```

```
MsgBox "Ingrese el mes"
```

```
Exit Sub
```

```
End If
```

```
End If
```

```
If txthospital.Text = "" Then
```

```
MsgBox "Ingrese el Nombre del Hospital"
```

```
Exit Sub
```



```

Elseif txtmunicipio.Text = "" Then
MsgBox "Ingrese el Nombre del Municipio"
Exit Sub
Elseif TXTDEPARTAMENTO.Text = "" Then
MsgBox "Ingrese el Nombre del Departamento"
Exit Sub

End If

For i = 0 To Label4.Count - 1
Label4(i).Visible = False
Next i
Label10.Visible = False

If cbTI.Text = "Anual" Then
Label4(0).Visible = True

For j = 1 To (Label4.Count - 1)
VarAncho = Label4(j).Width
varalto = Label4(j).Height
VarLeft = Label4(j).Left
Label4(j).Visible = True

With Adodc1
.RecordSource = "SELECT Hemocomponente, COUNT(Hemocomponente)From
Transfusion, CR_PLQ, Sangre_Preparada, Entrega WHERE (Transfusion.Transfusion
LIKE CR_PLQ.Transfusion AND CR_PLQ.SPCod LIKE Sangre_Preparada.SPCod AND
Sangre_Preparada.SPCod LIKE Entrega.SPCod) AND
MONTH(Entrega.Fecha_Entrega) LIKE "" + CStr(j) + "" AND
YEAR(Entrega.Fecha_Entrega) LIKE "" + txtanyo.Text + "" Group
By(Sangre_Preparada.Hemocomponente)" + _
" Union " + _
" SELECT Hemocomponente, COUNT(Hemocomponente) FROM Transfusion,
Sangre_Preparada, Entrega, PreparacionTransfusion WHERE (Transfusion.Transfusion
LIKE PreparacionTransfusion.Transfusion AND PreparacionTransfusion.SPCod LIKE
Sangre_Preparada.SPCod AND Sangre_Preparada.SPCod LIKE Entrega.SPCod) AND
MONTH(Entrega.Fecha_Entrega) LIKE "" + CStr(j) + "" AND
YEAR(Entrega.Fecha_Entrega) LIKE "" + txtanyo.Text + "" Group
By(Sangre_Preparada.Hemocomponente)"
.Refresh
End With

Do While Not Adodc1.Recordset.EOF

For i = 0 To 6
If Adodc1.Recordset("Hemocomponente").Value Like arraycomponentes(i) Then
TotalL(i) = Adodc1.Recordset(1).Value
TotalL(7) = TotalL(7) + TotalL(i)

```



```
TotalG(i) = TotalG(i) + TotalL(i)
TotalG(7) = TotalG(7) + TotalL(i)
```

```
Exit For
End If
Next i
Adodc1.Recordset.MoveNext
Loop
```

```
For i = 0 To 7
VarTop = (Cdbl(i + 1) * (Label4(j).Height)) + Label4(j).Top
```

```
Picture1.Line ((Label4(0).Left), (Cdbl(i + 1) * (Label4(0).Height)) + Label4(0).Top - 20)-
Step(Label4(0).Width - 20, Label4(0).Height), , B
Picture1.Font.Bold = True
Picture1.Font.Name = "MS Sans Serif"
Picture1.Font.Size = 8
Picture1.CurrentY = VarTop + ((varalto - Picture1.TextHeight("exanguinotransfuion")) / 2)
Picture1.CurrentX = Label4(0).Left + ((Label4(0).Width -
Picture1.TextWidth("exanguinotransfuion")) / 2)
Picture1.Print NombreCompl(i)
```

```
'Dibujar el resto de las cajas
Picture1.Line (VarLeft, VarTop - 20)-Step(VarAncho - 20, varalto), , B
'Colocar las coordenadas donde se escribira el texto
Picture1.Font.Bold = False
Picture1.Font.Name = "Arial"
Picture1.CurrentY = VarTop + ((varalto - Picture1.TextHeight("Hola" + CStr(i))) / 2)
Picture1.CurrentX = VarLeft + ((VarAncho - Picture1.TextWidth("Hola" + CStr(i))) / 2)
If j = (Label4.Count - 1) Then
Picture1.Print TotalG(i)
Else:
Picture1.Print TotalL(i)
End If
Next i
```

```
'inicializar array
For i = 0 To 7
TotalL(i) = 0
Next i
```

```
Next j
```

```
Picture1.Refresh
```

```
Elseif cbTI.Text = "Semestral" Then
```

```
With Adodc1
```



```
.RecordSource = "SELECT Hemocomponente, COUNT(Hemocomponente)From
Transfusión, CR_PLQ, Sangre_Preparada, Entrega WHERE (Transfusión.Transfusión
LIKE CR_PLQ.Transfusión AND CR_PLQ.SPCod LIKE Sangre_Preparada.SPCod AND
Sangre_Preparada.SPCod LIKE Entrega.SPCod) AND YEAR(Entrega.Fecha_Entrega)
= " + txtanyo.Text + " AND MONTH(Entrega.Fecha_Entrega) >= " +
CStr(cbmeses.ListIndex * 6 + 1) + " AND MONTH(Entrega.Fecha_Entrega) <= " +
CStr((cbmeses.ListIndex + 1) * 6) + " Group By(Sangre_Preparada.Hemocomponente)"
+ _
" Union " + _
" SELECT Hemocomponente, COUNT(Hemocomponente) FROM Transfusión,
Sangre_Preparada, Entrega, PreparaciónTransfusión WHERE (Transfusión.Transfusión
LIKE PreparaciónTransfusión.Transfusión AND PreparaciónTransfusión.SPCod LIKE
Sangre_Preparada.SPCod AND Sangre_Preparada.SPCod LIKE Entrega.SPCod) AND
YEAR(Entrega.Fecha_Entrega) = " + txtanyo.Text + " AND
MONTH(Entrega.Fecha_Entrega) >= " + CStr(cbmeses.ListIndex * 6 + 1) + " AND
MONTH(Entrega.Fecha_Entrega) <= " + CStr((cbmeses.ListIndex + 1) * 6) + " Group By
(Sangre_Preparada.Hemocomponente)"
.Refresh
End With
```

```
Do While Not Adodc1.Recordset.EOF
For i = 0 To 6
If Adodc1.Recordset("Hemocomponente").Value Like arraycomponentes(i) Then
TotalL(i) = Adodc1.Recordset(1).Value
TotalL(7) = TotalL(7) + Adodc1.Recordset(1)
Exit For
End If
Next i
Adodc1.Recordset.MoveNext
Loop
```

```
Label4(0).Visible = True
VarAncho = Label10.Width
varalto = Label10.Height
VarLeft = Label10.Left
Label10.Caption = "Cantidad"
Label10.Visible = True
```

```
For i = 0 To 7
VarTop = (Cdbl(i + 1) * (Label10.Height)) + Label10.Top
'dibujar la primera caja
Picture1.Line ((Label4(0).Left), VarTop - 20)-Step(Label4(0).Width, varalto), , B
Picture1.Font.Bold = True
Picture1.Font.Name = "MS Sans Serif"
Picture1.Font.Size = 8
Picture1.CurrentY = VarTop + ((varalto - Picture1.TextHeight("exanguinotransfuion")) / 2)
Picture1.CurrentX = Label4(0).Left + ((Label4(0).Width -
Picture1.TextWidth("exanguinotransfuion")) / 2)
```



Picture1.Print NombreCompl(i)

'dibujar el resto de las cajas

Picture1.Line (VarLeft, VarTop - 20)-Step(VarAncho - 20, varalto), , B

Picture1.Font.Bold = False

Picture1.Font.Name = "Arial"

Picture1.CurrentY = VarTop + ((varalto - Picture1.TextHeight("Hola" + CStr(i))) / 2)

Picture1.CurrentX = VarLeft + ((VarAncho - Picture1.TextWidth("Hola" + CStr(i))) / 2)

Picture1.Print CStr(TotalL(i))

Next i

Picture1.Refresh

Elseif cbTI.Text = "Trimestral" Then

With Adodc1

.RecordSource = "SELECT Hemocomponente, COUNT(Hemocomponente)From  
Transfucion, CR\_PLQ, Sangre\_Preparada, Entrega WHERE (Transfucion.Transfucion  
LIKE CR\_PLQ.Transfucion AND CR\_PLQ.SPCod LIKE Sangre\_Preparada.SPCod AND  
Sangre\_Preparada.SPCod LIKE Entrega.SPCod) AND

MONTH(Entrega.Fecha\_Entrega) LIKE '8' AND YEAR(Entrega.Fecha\_Entrega) LIKE  
'2000' Group By(Sangre\_Preparada.Hemocomponente)" + \_

" Union " + \_

" SELECT Hemocomponente, COUNT(Hemocomponente) FROM Transfucion,  
Sangre\_Preparada, Entrega, PreparacionTransfucion WHERE (Transfucion.Transfucion  
LIKE PreparacionTransfucion.Transfucion AND PreparacionTransfucion.SPCod LIKE  
Sangre\_Preparada.SPCod AND Sangre\_Preparada.SPCod LIKE Entrega.SPCod) AND  
MONTH(Entrega.Fecha\_Entrega) LIKE '8' AND YEAR(Entrega.Fecha\_Entrega) LIKE ""  
+ txtan + "" Group By(Sangre\_Preparada.Hemocomponente)"

.Refresh

End With

Do While Not Adodc1.Recordset.EOF

For i = 0 To 6

If Adodc1.Recordset("Hemoderivado").Value Like arraycomponentes(i) Then

TotalL(i) = Adodc1.Recordset(1).Value

TotalL(7) = TotalL(7) + Adodc1.Recordset(1)

Exit For

End If

Next i

Adodc1.Recordset.MoveNext

Loop

Label4(0).Visible = True

VarAncho = Label10.Width

varalto = Label10.Height

VarLeft = Label10.Left

Label10.Caption = "Cantidad"

Label10.Visible = True



```

For i = 0 To 7
VarTop = (CDBl(i + 1) * (Label10.Height)) + Label10.Top
'dibujar la primera caja
Picture1.Line ((Label4(0).Left), VarTop - 20)-Step(Label4(0).Width - 20, varalto), , B
Picture1.Font.Bold = True
Picture1.Font.Name = "MS Sans Serif"
Picture1.Font.Size = 8
Picture1.CurrentY = VarTop + ((varalto - Picture1.TextHeight("exanguinotransfuion")) / 2)
Picture1.CurrentX = Label4(0).Left + ((Label4(0).Width -
Picture1.TextWidth("exanguinotransfuion")) / 2)
Picture1.Print NombreCompl(i)

'dibujar el resto de las cajas
Picture1.Line (VarLeft, VarTop - 20)-Step(VarAncho - 20, varalto), , B
Picture1.Font.Bold = False
Picture1.Font.Name = "Arial"
Picture1.CurrentY = VarTop + ((varalto - Picture1.TextHeight("Hola" + CStr(i))) / 2)
Picture1.CurrentX = VarLeft + ((VarAncho - Picture1.TextWidth("Hola" + CStr(i))) / 2)
Picture1.Print CStr(TotalL(i))
Next i
Picture1.Refresh
Elseif cbTI.Text = "Mensual" Then

With Adodc1
.RecordSource = "SELECT Hemocomponente, COUNT(Hemocomponente)From
Transfusion, CR_PLQ, Sangre_Preparada, Entrega WHERE (Transfusion.Transfusion
LIKE CR_PLQ.Transfusion AND CR_PLQ.SPCod LIKE Sangre_Preparada.SPCod AND
Sangre_Preparada.SPCod LIKE Entrega.SPCod) AND
MONTH(Entrega.Fecha_Entrega) LIKE "" + CStr(cbmeses.ListIndex + 1) + "" AND
YEAR(Entrega.Fecha_Entrega) LIKE "" + txtanyo.Text + "" Group
By(Sangre_Preparada.Hemocomponente)" + _
" Union " + _
" SELECT Hemocomponente, COUNT(Hemocomponente) FROM Transfusion,
Sangre_Preparada, Entrega, PreparacionTransfusion WHERE (Transfusion.Transfusion
LIKE PreparacionTransfusion.Transfusion AND PreparacionTransfusion.SPCod LIKE
Sangre_Preparada.SPCod AND Sangre_Preparada.SPCod LIKE Entrega.SPCod) AND
MONTH(Entrega.Fecha_Entrega) LIKE "" + CStr(cbmeses.ListIndex + 1) + "" AND
YEAR(Entrega.Fecha_Entrega) LIKE "" + txtanyo.Text + "" Group
By(Sangre_Preparada.Hemocomponente)"
.Refresh
End With

Do While Not Adodc1.Recordset.EOF
For i = 0 To 6
If Adodc1.Recordset("Hemocomponente").Value Like arraycomponentes(i) Then
TotalL(i) = Adodc1.Recordset(1).Value
TotalL(7) = TotalL(7) + Adodc1.Recordset(1)

```



```

Exit For
End If
Next i
Adodc1.Recordset.MoveNext
Loop

Label4(0).Visible = True
VarAncho = Label10.Width
varalto = Label10.Height
VarLeft = Label10.Left
Label10.Caption = "Cantidad"
Label10.Visible = True

For i = 0 To 7
VarTop = (CDBl(i + 1) * (Label10.Height)) + Label10.Top
'dibujar la primera caja
Picture1.Line ((Label4(0).Left), VarTop - 20)-Step(Label4(0).Width - 20, varalto), , B
Picture1.Font.Bold = True
Picture1.Font.Name = "MS Sans Serif"
Picture1.Font.Size = 8
Picture1.CurrentY = VarTop + ((varalto - Picture1.TextHeight("exanguinotransfuion")) / 2)
Picture1.CurrentX = Label4(0).Left + ((Label4(0).Width -
Picture1.TextWidth("exanguinotransfuion")) / 2)
Picture1.Print NombreCompl(i)
'dibujar el resto de las cajas
Picture1.Line (VarLeft, VarTop - 20)-Step(VarAncho - 20, varalto), , B
Picture1.Font.Bold = False
Picture1.Font.Name = "Arial"
Picture1.CurrentY = VarTop + ((varalto - Picture1.TextHeight("Hola" + CStr(i))) / 2)
Picture1.CurrentX = VarLeft + ((VarAncho - Picture1.TextWidth("Hola" + CStr(i))) / 2)
Picture1.Print CStr(TotalL(i))
Next i
Picture1.Refresh

End If
LBHOSPITAL.Caption = Trim(UCase("HOSPITAL " + txthospital.Text))
Label9.Caption = UCase(Trim(txtmunicipio.Text + " - " + TXTDEPARTAMENTO.Text))
Label8.Caption = "Fecha de Elaboraciòn: " + Format(Calendar1.Text, "dd/mm/yyyy")
Label3.Caption = "Informe " + cbTI.Text + vbNewLine + cbmeses.Text + " Del " +
txtanyo.Text
End Sub

Private Sub btImprimir_Click()
On Error GoTo ControlError
btprevia_Click

btretornar.Visible = False

```



```
Picture1.SetFocus
Picture2.AutoRedraw = True
rv = SendMessage(Picture1.hWnd, WM_PAINT, Picture2.hDC, 0)
rv = SendMessage(Picture1.hWnd, WM_PRINT, Picture2.hDC, _
PRF_CHILDREN + PRF_CLIENT + PRF_OWNED)
Picture2.Picture = Picture2.Image
Picture2.AutoRedraw = False

Printer.Copies = 1
Printer.Zoom = 90
Printer.FillColor = vbWhite
Printer.PrintQuality = 300
Printer.ScaleMode = Picture2.ScaleMode
Printer.Print ""
Printer.Print ""
Printer.Print ""
Printer.PaintPicture Picture2.Picture, (Printer.ScaleWidth - Me.ScaleWidth) / 2, 0
Printer.EndDoc

btretornar_Click
Exit Sub

ControlError:

btretornar_Click

If Err.Number = 482 Then

Elseif MsgBox(" Error en la impresion " + vbNewLine + _
"Desea intentarlo de nuevo", vbYesNo + vbCritical, "Error") = vbYes Then
btImprimir_Click
End If
Err.Clear

End Sub

Private Sub btprevia_Click()

Label3.Visible = True

Frame2.Visible = False
Frame1.Visible = False

btImprimir.Visible = False
btprevia.Visible = False
btretornar.Visible = True
End Sub
```



```
Private Sub btretornar_Click()  
Label3.Visible = False
```

```
Frame2.Visible = True  
Frame1.Visible = True
```

```
btImprimir.Visible = True  
btprevia.Visible = True  
btretornar.Visible = False  
End Sub
```

```
Private Sub cbTI_Click()  
cbmeses.Clear  
If cbTI.Text = "Anual" Then  
Elseif cbTI.Text = "Mensual" Then
```

```
With cbmeses  
.AddItem "Enero", 0  
.AddItem "Febrero", 1  
.AddItem "Marzo", 2  
.AddItem "Abril", 3  
.AddItem "Mayo", 4  
.AddItem "Junio", 5  
.AddItem "Julio", 6  
.AddItem "Agosto", 7  
.AddItem "Septiembre", 8  
.AddItem "Octubre", 9  
.AddItem "Noviembre", 10  
.AddItem "Diciembre", 11  
End With  
Elseif cbTI.Text = "Semestral" Then
```

```
With cbmeses  
.AddItem "I Semestre", 0  
.AddItem "II Semestre", 1  
End With  
Elseif cbTI.Text = "Trimestral" Then
```

```
With cbmeses  
.AddItem "I Trimestre", 0  
.AddItem "II Trimestre", 1  
.AddItem "III Trimestre", 2  
.AddItem "IV Trimestre", 3  
End UIT  
End If  
End Sub
```



```
Private Sub Form_Load()  
Picture1.Move 0, 0  
Me.Move (frmprincipal.Width - Width) / 2, (frmprincipal.Height - Height) / 2  
End Sub
```

```
Private Sub Form_Resize()  
Picture1.Move 0, 0, Me.ScaleWidth, Me.ScaleHeight  
Shape1.Move Shape1.Left, Shape1.Top, Shape1.Width, Picture1.ScaleHeight - 200  
End Sub
```

```
Private Sub txtFentrega_LostFocus()  
txtFentrega.Text = UCase(Trim(txtFentrega.Text))  
End Sub
```

```
Private Sub txthospital_LostFocus()  
txthospital.Text = UCase(Trim(txthospital.Text))  
End Sub
```

```
Private Sub txtmunicipio_LostFocus()  
txtmunicipio.Text = UCase(Trim(txtmunicipio.Text))  
End Sub
```