

SLD111 SISTEMA PARA LA GESTIÓN DE TRASPLANTE HEPÁTICO

SLD111 MANAGEMENT SYSTEM FOR LIVER TRANSPLANT

Dunior Socarrás Benítez¹, Angélica de la Caridad Vázquez Rúa², Lisbel Caridad Gómez Poey³,
Rosbel Fernández León⁴, Dr. Fernando González Castillo⁵

1 Universidad de las Ciencias Informáticas, Cuba, dsocarras@uci.cu, Carretera a San Antonio de los Baños, km 2 ½, Boyeros, Ciudad de La Habana

2 Universidad de las Ciencias Informáticas, Cuba, alvarez@uci.cu

3 Universidad de las Ciencias Informáticas, Cuba, lcgomez@estudiantes.uci.cu

4 Universidad de las Ciencias Informáticas, Cuba, rleon@estudiantes.uci.cu

5 Hospital Hermanos Ameijeiras, Cuba, fernando.gonzales@infomed.sld.cu

RESUMEN: El área de trasplante de hígado de las instituciones hospitalarias gestiona grandes volúmenes de información. El procesamiento de la misma, usualmente se realiza de forma manual y se archiva en formato duro. Esto provoca que surja la necesidad de crear un sistema dedicado al trasplante de hígado, por la no existencia de un sistema informático que recogiera la información necesaria al respecto que le facilitara el trabajo al personal que allí labora. Por estas razones la investigación está encaminada a obtener un sistema que facilite la gestión de la información médica de los trasplantes de hígado en el área de trasplante del Hospital Clínico Quirúrgico Hermanos Ameijeiras. El desarrollo del sistema está guiado por el Proceso Unificado de Desarrollo y se basa en tecnologías libres para entornos Web, multiplataforma y sobre una arquitectura en capas, utilizando Java como lenguaje de programación e implementando el patrón de arquitectura Modelo Vista Controlador. Como Sistema de Gestión de Bases de Datos se utiliza PostgreSQL y como servidor de aplicaciones el JBoss Application Server. Entre los beneficios que aporta la aplicación se encuentran proveer una atención de salud con calidad, que garantice la seguridad y confiabilidad de la información médica y su registro en una Historia Clínica Electrónica única de hospitalización, así como disponer de una herramienta de gestión clínica y administrativa que facilite la toma de decisiones al personal que labora en el área.

Palabras Clave: Gestión, Historia Clínica Electrónica, Información Médica, Trasplante.

ABSTRACT: The area of liver transplantation in hospitals manages large volumes of information. The very processing is usually done manually and is filed in hard format. This causes the need for a dedicated system to liver transplantation, by the absence of a computer system to collect the necessary information about that enhanced the work to the staff who work there. For these reasons the research is aimed at obtaining a system that facilitates the management of medical information on liver transplants in the transplant area in Hermanos Ameijeiras Surgical Hospital. System development was guided by the Rational Unified Process and is based on free technologies for Web environments, multiplatform and over a layered architecture, using Java as a programming language and implementing the architecture pattern Model View Controller. PostgreSQL as Management System Database and JBoss Application Server is used as the application server. Among the benefits of the application are to provide a quality health care, to ensure the safety and reliability of medical information and registration in a single Electronic Health Record in hospital, as well as to have a tool for clinical and administrative management to facilitate decision-making to the people working in the area.

KeyWords: Electronic Health Record, Management, Medical Information, Transplant

1. INTRODUCCIÓN

Cuba está inmersa en la informatización de todos los sectores de la sociedad. Este proceso en la esfera de la salud es hoy un hecho, dentro de la atención terciaria se encuentran los servicios intrahospitalarios de alta especificación, los cuales están destinados a la ejecución de intervenciones quirúrgicas y atenciones de mayor complejidad, que requieran hospitalización, permitiendo el retraso de la aparición o el progreso de una enfermedad empleando diferentes medios para no prolongar su avance con cirugías y trasplantes. Las enfermedades hepáticas son las que se conocen como crónicas del hígado y entre sus tratamientos para su cura se encuentra el Trasplante Hepático (TH).

En Cuba se reportaron 963 defunciones en el año 2002 por cirrosis hepática y otras enfermedades crónicas del hígado, lo que representa una tasa de 8,5 por cada 100 000 habitantes. Esta cifra ubica a este grupo de enfermedades como la décima causa de muerte en nuestro país, lo cual justifica la creación de un programa de TH con el objetivo de ofrecer una alternativa terapéutica para este grupo importante de pacientes. [1]

Actualmente en las instituciones hospitalarias cubanas y en específico en el Hospital Clínico Quirúrgico Hermanos Ameijeiras, todo el proceso relacionado con la gestión de la información de trasplante de hígado se realiza de forma manual y se archiva en formato duro, desde que el paciente llega a la consulta hasta que es trasplantado. Todo esto implica que las hojas no se encuentren integradas y por tanto provocan pérdida de tiempo para los médicos, redundancia de información ya que deben de escribir y transcribir el análisis un número de veces, ocasionando pérdida de los datos o el deterioro de estos documentos y en el peor de los casos, que puedan extraviarse en caso de que el enfermo deba recurrir a otro centro, pues él sería el encargado de trasladar su historia clínica.

2. CARACTERÍSTICAS DEL SISTEMA

En el área de trasplante de hígado de los hospitales se lleva a cabo la atención del paciente, la gestión de la información referente al mismo en la consulta, de forma que se incorporen a su historia clínica todos aquellos documentos y actividades que se desencadenan durante su seguimiento en el hospital. Los pacientes que requieren de trasplantes de hígado necesitan de atención especializada, los mismos son atendidos en instituciones hospitalarias.

Dentro de la atención terciaria se encuentran los servicios intrahospitalarios de alta especificación los cuales están destinados a la ejecución de intervenciones quirúrgicas y atenciones de mayor complejidad que requieren hospitalización; así como retrasar la aparición o el progreso de una enfermedad, empleando diferentes medios para no prolongar su avance con cirugías, trasplantes, ginecología, entre otras.

El área de trasplante de hígado tiene dos procesos principales como son: consulta especializada de trasplante de hígado y Trasplante de hígado.

Estos procesos, de conjunto con otras funcionalidades de apoyo, fueron desarrollados en el módulo de Gestión de Trasplante de Hígado, integrado al Sistema de Gestión Hospitalaria alas HIS, aprovechando las funcionalidades que este ofrece, entre las cuales se pueden mencionar los módulos: Citas, Hospitalización, Historia Clínica, Laboratorio, Consulta Externa, Admisión entre otros. Todos estos de conjunto con el módulo de trasplante hepático, conforman un Sistema de Gestión de Trasplante.

2.1 Descripción de los modelos implementados para la inclusión y ordenamiento en la lista de trasplante.

Modelo Child-Pug

El modelo Child o Child-Turcotte fue propuesto por primera vez en 1964. La versión inicial del modelo pronóstico de Child-Turcotte, incluía 2 variables continuas (bilirrubina y albúmina) y 3 variables cuantitativas discretas (ascitis, encefalopatía y estado nutricional), como se observa en la tabla 1. La selección de estas 5 variables fue empírica así como los puntos de corte para bilirrubina y albúmina. Las 5 variables y sus respectivos puntos de corte fueron asignados para definir 3 grupos distintos con incremento de su severidad (A, B y C) los pacientes con valores individuales se asignan a diferentes grupos. Por tanto, a las variables se les asignan 1, 2 y 3 puntos según caigan dentro de los valores límites de cada grupo A, B y C, respectivamente; el puntaje es la suma de todos los puntos desde un rango de 5 a 15. Es generalmente aceptado que pacientes con puntuación entre 5 y 8 pertenecen al grupo A, pacientes con puntuación entre 9 y 11 pertenecen al grupo B y pacientes con puntuación entre 12 y 15, al grupo C.

Tabla I. Modelo Child-Turcotte

Puntos	1	2	3
Bilirrubina (mmol/L)	<34	34-51	>51
Albúmina (g/L)	>35	30-35	<30
Ascitis	Ausente	Controlada	Refractaria
Encefalopatía	Ausente	Mínima	Avanzada (coma)
Estado Nutricional	Bueno	Aceptable	Pobre

Una versión modificada fue propuesta casi 10 años después (tabla 2), denominada Child-Pugh (CP), esta versión es la que se utiliza actualmente y por ello es la que se implementó en el sistema. En esta versión modificada, el estado nutricional fue reemplazado por el tiempo de protrombina. El punto de corte más bajo de la albúmina fue cambiado de 30 a 28 g/L. El modelo correspondiente a la suma de todos los puntos individuales permite categorizar a los pacientes en diferentes grados de Child-Pugh, A (5-6 puntos, B (7-9 puntos) y C (10-15 puntos). De forma importante, el rango total de puntos (5-15) no es igualmente distribuido entre los grados A, B y C, probablemente tratando de reflejar más eficientemente el impacto clínico de cada grado en términos de pronóstico.

Tabla II. Modelo Child-Pug

Puntos	1	2	3
Bilirrubina (mmol/L)	<34	34-51	>51
Albúmina (g/L)	>35	28-35	<28
Ascitis	Ausente	Leve	Moderada
Encefalopatía	Ausente	Mínima	Avanzada (coma)
Tiempo protrombina (en segundos)	1-4	4-6	>6

Modelo MELD

El modelo Model for End-stage Liver Disease (MELD) es creado través del análisis de regresión de Cox, con el fin de determinar el pronóstico de pacientes sometidos a shunt intrahepático portosistémico, TIPS por sus siglas en inglés, y que después fue validado en pacientes con cirrosis hepática en estadio terminal y utilizado como sistema para la distribución de órganos en el trasplante hepático. Este modelo utiliza 3 variables objetivas de uso clínico diario: bilirrubina, internacional normalized ratio (INR) para tiempo de protrombina, creatinina sérica y etiología de la cirrosis, esta última variable fue removida posteriormente del modelo por no afectar de forma significativa el pronóstico de la supervivencia. A partir de estas variables se obtiene un puntaje que predice la supervivencia en un paciente dado.

$$\text{Puntaje MELD} = R = 0,957 \times \log e (\text{creatinina mg/dL}) + 0,38 \times \log e (\text{bilirrubina mg/dL}) + 1,120 \times \log e (\text{INR}) + 0,643 \times 10.$$

Con esta fórmula, un mayor puntaje indica mayor severidad de la enfermedad y por tanto, mayor riesgo de fallecer. Este nuevo modelo cambia los sistemas de distribución de órganos para trasplantes de hígado actuales que hasta hace muy poco dependían para la asignación de un órgano, en gran medida, del tiempo en lista de espera. Con el MELD, la asignación de un órgano a un paciente está determinada por el grado de severidad de su enfermedad hepática y sus posibilidades de supervivencia. [2]

2.2 Descripción del proceso atender paciente en la consulta especializada.

El paciente llega remitido al área de consulta externa del hospital, y es atendido por un especialista de trasplante de hígado, donde se le recogen una serie de datos en una hoja de consulta. Además se le orientan un conjunto de exámenes y estudios pertinentes para determinar el padecimiento que presenta el órgano en el paciente, así como pruebas de laboratorios las cuales se analizan entre un grupo de especialistas arrojando un resultado médico, que en conjunto con el algoritmo CHILD – PUGH, le permiten al especialista decidir si el órgano necesita ser trasplantado, en caso de que no sea necesario, el médico le indicará al paciente un método para su recuperación.

En caso de que el paciente no tenga remisión, ingresa según la gravedad que presente el órgano. El especialista puede decidir atenderlo fuera de consulta por la afección que presenta.

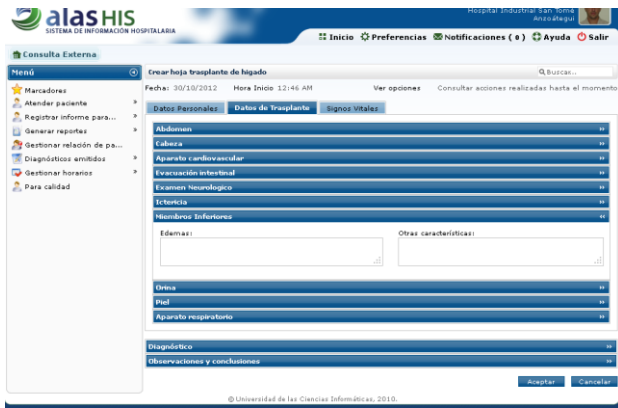


Fig. 1: Hoja de Consulta Externa de Trasplante de Hígado

En la hoja de consulta del sistema, el especialista recoge los datos de la consulta los cuales están conformados por los datos personales del paciente, los generales de la consulta de trasplante y se le realiza un examen físico donde se revisan cada una de las estructuras del mismo, aunque se hace más énfasis en las estructuras que corresponden con el hígado.

Recoger los datos generales de la consulta de trasplante, es una actividad de gran importancia, ya que perfecciona la atención al paciente basándose en una evaluación metódica mediante la frecuencia respiratoria, pulso, peso y reconocimiento del paciente que conducen al especialista a comprobar los signos generales de una enfermedad y que son de interés a la hora de emitir un diagnóstico. Luego, se realizan los exámenes de signos vitales, como son la inspección, palpación, edemas entre otros.

A continuación, el médico decide si es necesario que al paciente se le desarrollen los exámenes del algoritmo, estos exámenes arrojan cálculos que son necesarios en caso de que el paciente necesite intervención quirúrgica, que pueden servir para determinar la conducta a seguir del paciente y la emisión de una impresión diagnóstica.

La impresión diagnóstica, la emite el especialista de trasplante de hígado, luego de valorar todos los exámenes del paciente.

2.3 Descripción del proceso atender paciente en el módulo de trasplante de hígado.

Si la decisión del especialista, luego de analizar los resultados de los exámenes, es de trasplantarlo, el paciente debe ser incluido de una lista de pacientes a trasplantar, para lo cual es necesario que este posea una serie de exámenes realizados que le permitirán al especialista, a través del resultado del

algoritmo CHILD-PUG, si este está apto para trasplantar.

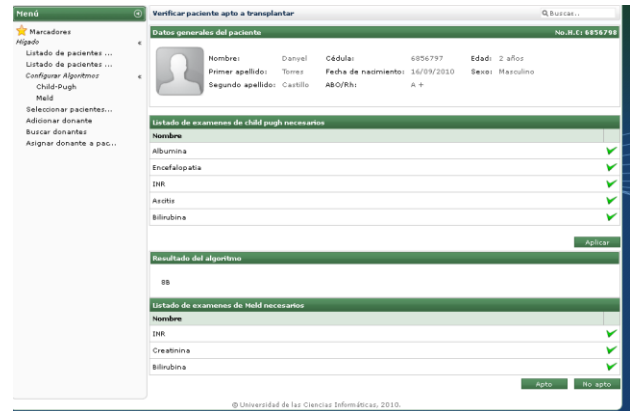


Fig. 2: Paciente apto para trasplantar

El orden y la prioridad de los receptores en la lista de espera pueden ir variando según la gravedad de cada órgano mediante el algoritmo MELD. Este algoritmo consta de diferentes análisis de laboratorio y se les aplica mensualmente, hasta que aparezca un donante. El sistema cuenta con una gestión de los donantes, en el cual se le permite al coordinador de trasplantes, insertar de forma digital, el consentimiento escaneado de los familiares concediendo el permiso de realizar la extracción del órgano para ser trasplantado en otro paciente.

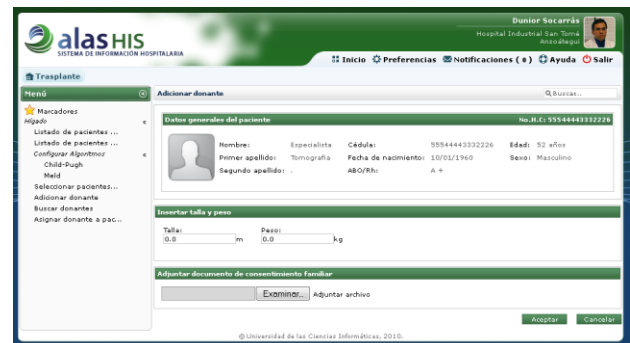


Fig. 3: Paciente apto para trasplantar.

Una vez que se conozca un donante se procede hacer las pruebas necesarias para determinar la compatibilidad entre el donante y el receptor. Así como las pruebas correspondientes para la selección del paciente que se le realizará el trasplante. El sistema le permite al especialista conocer los receptores compatibles con el donante por tipo de grupo sanguíneo, y los expone en el orden en que se encuentran en la lista de trasplante, ordenados por el algoritmo MELD.

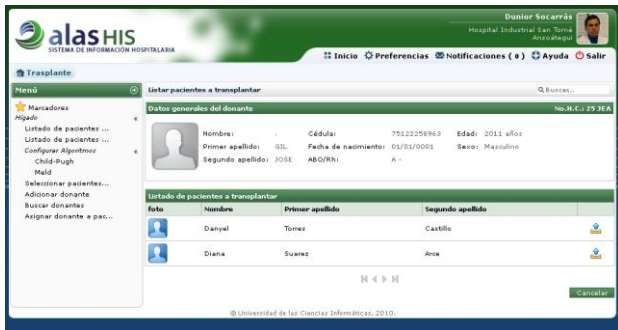


Fig. 4: Selección de receptor para donante seleccionado.

Luego que el paciente es trasplantado se elimina de la lista de espera y permanece bajo observación en vista de cómo responde su organismo con el nuevo órgano.

3. DESCRIPCIÓN DE LA ARQUITECTURA

La arquitectura de todo software es un elemento de vital importancia para su desarrollo por lo que debe tenerse en cuenta. Es una vista estructural de alto nivel, ocurre muy tempranamente en el ciclo de vida del software y define los estilos o grupos de estilos adecuados para cumplir con los requisitos no funcionales. Para lograr un sistema de calidad, es necesario que este se apoye en una arquitectura robusta y flexible.

Para ello se utiliza el patrón MVC (Modelo, Vista, Controlador), que tiene como objetivo separar la lógica del negocio de la lógica de presentación o interfaz y los datos para realizar más fácil los cambios en caso de que existan porque se atacaría al nivel requerido.

El MVC en una aplicación web, se representa a través de la Vista que constituyen las páginas Lenguaje de Marcado de Hipertexto (HTML) y el código que provee datos dinámicos a la página, que permite mostrarle la información al usuario, debido a la captura de datos del sistema al exterior y viceversa. El Controlador gestiona todos los eventos de entrada del usuario y traslada las peticiones de la Vista a la Modelo, mediante el intermediario entre ellas. El Modelo contiene toda la información persistente y reglas de negocio.

La Vista está desarrollada básicamente con JSF, usando componentes Seam de Interfaz de Usuario, la librería de componentes Richfaces y Ajax4jsf además, se utiliza Facelets como motor de plantillas. En el controlador se utiliza Seam como Framework de integración y Drools como gestor de las reglas del negocio. En el modelo se utiliza Hibernate

como herramienta de mapeo objeto relacional, que es la implementación de EJB 3.0 y JPA.

La integración de elementos existentes en la aplicación se logra mediante el framework JbossSeam que permite la utilización de anotaciones con las que es posible acceder al modelo de datos directamente desde la vista, característica que permite un mejor funcionamiento del sistema.

4. PRINCIPALES TECNOLOGÍAS Y HERRAMIENTAS UTILIZADAS EN EL DESARROLLO.

Java Server Faces (JSF): Es un framework que define un modelo de componentes de interfaz de usuario y de eventos. Permite manejar el estado de los componentes de interfaz de usuario, manejar sus eventos, la validación y conversión del lado del servidor y centralizar la navegabilidad de las páginas de la aplicación. JSF es el marco estándar que proporciona Java para construir aplicaciones web, sigue el patrón MVC, proporcionando una manera de validar datos, llamar a reglas de negocio y devolver los resultados al cliente. [3]

AJAX: Asynchronous JavaScript and XML (AJAX). No es un lenguaje de programación, sino un conjunto de tecnologías (HTML-Java Script-CSS-DHTML-PHP/ASP.NET/JSP-XML) que permiten hacer páginas web más interactivas. Su característica fundamental es permitir actualizar parte de una página con información que se encuentra en el servidor, sin tener que refrescar completamente la página. De modo similar se puede enviar información al servidor. [4]

RichFaces: Es una biblioteca de componentes para JSF y un avanzado framework para la integración de AJAX con facilidad en la capacidad de desarrollo de aplicaciones de negocio. Incluye un fuerte apoyo para la disponibilidad de interfaces de aplicaciones JSF, también aprovecha al máximo los beneficios del framework JSF, encerrando la validación y conversión de instalaciones, junto con la gestión de estática y dinámica los recursos.

JBossSeam 2.1.1: Es un framework de código libre desarrollado por JBoss. Integra la capa de presentación (JSF) con la capa de negocios y persistencia Enterprise JavaBeans (EJB3). Se puede acceder a cualquier componente EJB desde la capa de presentación, refiriéndose a él mediante su nombre de componente Seam, ya que integra la capa de presentación (JSF) con la capa de negocios y persistencia (EJB).[5]

Tiene como objetivo simplificar la arquitectura de las

aplicaciones, lo que permite integrar tecnologías de forma relativamente transparente y con herramientas de generación de código. Todo esto posibilita que se utilice como el framework para la implementación del software.

Hibernate: Es una herramienta de mapeo objeto-relacional (ORM) para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos XML, que permiten establecer estas relaciones. Permite expresar consultas en su propia expresión de SQL portátil (HQL, Hibernate Query Language), al mismo tiempo que una API para construir las consultas programáticamente (conocida como “criterio”).

Java RuntimeEnvironment (JRE1.6): Es la máquina virtual de Java más básica. Sin ella, ningún programa Java lograría arrancar, y lo mismo se aplica a los applets de las páginas web. Es un componente tan imprescindible como .NET o las librerías DirectX. Su instalador se encarga de integrar Java con el sistema y los principales navegadores (como Internet Explorer o Firefox). [6]

Lenguaje de Programación: El lenguaje de programación seleccionado es Java. Este es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90's [7]. Permite organizar el código en entidades como las clases compuestas de datos y funciones, y a través de la característica de la herencia se pueden organizar las clases en jerarquías. La programación en Java admite el desarrollo de aplicaciones bajo el esquema de Cliente – Servidor como de aplicaciones distribuidas, lo que lo hace capaz de conectar dos o más ordenadores, ejecutando tareas simultáneamente y de esta forma consigue distribuir el trabajo a cumplir.

Eclipse: Eclipse es un ambiente integrado del desarrollo (IDE) escrito sobre todo adentro Java. En su forma del defecto se significa para los reveladores de Java, consistiendo en las herramientas de desarrollo de Java (JDT). Los usuarios pueden ampliar sus capacidades instalando los plug-ins escritos para el marco del software del eclipse, tal como cajas de herramientas del desarrollo para otros lenguajes de programación y pueden escribir y contribuir sus propios módulos enchufables.[8]

JBoss Server o JBoss AS 4.2.2: JBoss es un Servidor de Aplicaciones Java EE de Software Libre implementado en Java puro. Al estar basado en Java, JBoss puede ser utilizado en cualquier sistema operativo que lo soporte. JBoss puede ser descargado, utilizado, incrustado, y distribuido sin res-

tricciones por la licencia. Por este motivo es la plataforma más popular de middleware para desarrolladores, vendedores independientes de software y, también, para grandes empresas. [9]

PostgreSQL 8.3: PostgreSQL es un SGBD de Objeto-Relacional (Object-RelationalDatabase Management System (ORDBMS)) libre. Es una versión que incluye características nuevas, optimizadas y entrega mayor consistencia en el rendimiento que hace de PostgreSQL. No tiene costo asociado por lo que cualquiera puede disponer de su código fuente, modificarlo a voluntad y redistribuirlo libremente. PostgreSQL presenta alta concurrencia, para esto utiliza la tecnología de Control de Concurrencia Multi-Versión (Multi versión concurrency control (MVCC)), con lo que se logra que ningún lector sea bloqueado por un escritor.

Es altamente extensible, soporta operadores, funciones, métodos de acceso y tipos de datos definidos por el usuario. Tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/PGSQL. Este lenguaje es comparable al lenguaje procedural del sistema de gestión de base de datos relacional Oracle, PL/SQL.

5. VISTA DE DESPLIEGUE

En el siguiente diagrama de despliegue del sistema se muestra el diseño de nodos unidos por conexiones de comunicación que modela la arquitectura en tiempo de ejecución del sistema

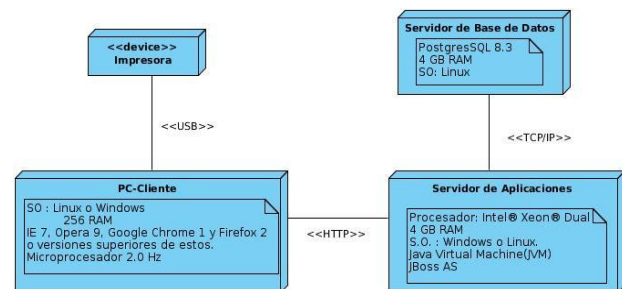


Fig. 5: Diagrama de Despliegue

6. CONCLUSIONES

La utilización del Sistema de Gestión de Trasplante le permitirá a los especialistas concentrarse mucho más en la atención médica y en los diagnósticos, dejándole toda la gestión de información y los cálculos al sistema informático. Con el se logra además acelerar el proceso de atención a los pacientes debido a que toda la información se encontrará automatizada, centralizada y accesible en todo momento. Todas las hojas de consulta se encontrarán integradas, permitiendo el estudio y comparación de

consultas anteriores, se podrán revisar los análisis realizados durante el tiempo de la enfermedad, no se contará con la pérdida de información ya que la historia clínica del paciente se encontrará resguardada en las bases de dato del sistema. Se podrá consultar la lista de pacientes a trasplantar en todo momento, sin necesidad de realizar los cálculos que exigen los algoritmos cada vez que los pacientes se realicen los exámenes, además el proceso de asignación de un donante a un receptor, se hace más fácil para los especialistas, debido a que el sistema le propone pacientes compatibles acorde con el grupo sanguíneo de ambos, dando la posibilidad además de visualizar los datos tanto del receptor como del donante para una mejor decisión.

7. REFERENCIAS BIBLIOGRÁFICAS

1. Castillo, Dr. Fernando F. González. TRASPLANTE HEPÁTICO ORTOTÓPICO. Trasplante. [En línea] [Citado el: 2012 de Abril de 2012.]
2. Bertot, Luis Calzadilla. Gómez, Eduardo Vilar. Enriquez, Laritza Lincheta. Modelos pronósticos para la cirrosis hepática. [1 de noviembre de 2010.] http://bvs.sld.cu/revistas/med/vol50_2_11/med09211.htm
3. Introducción a JSF (Java Server Faces). Primer artículo de un pequeño manual sobre esta tecnología. [En línea] [Citado el: 27 de Enero de 2012.] <http://www.desarrolloweb.com/articulos/2380.php>
4. Qué es AJAX. [En línea] [Citado el: 24 de Enero de 2012.] <http://www.ajaxya.com.ar/temarios/descripcion.php?cod=8&punto=1>.
5. Tecnologías de la Sociedad de la Información. JBoss Seam Framework. [En línea] [Citado el: 28 de Enero de 2012.] <http://tundidor.com/blog/?p=311>.
6. Sebastián, Juan. Modelo Vista Controlador – Definición y Características. [En línea] 13 de Noviembre de 2010. [Citado el: 1 de Febrero de 2012.] <http://java.softonic.com/>.
7. Historia del Lenguaje Java. [En línea] [Citado el: 31 de Enero de 2012.] http://www.cad.com.mx/historia_del_lenguaje_java.html.
8. Eclipse (software). [En línea] [Citado el: 1 de Febrero de 2012.] http://www.multilingualarchive.com/ma/enwiki/es/Eclipse_%28software%29.
9. JBoss Application Server. [En línea] [Citado el: 1 de Febrero de 2012.] http://www.dosideas.com/wiki/JBoss_Application_Server.

10. SÍNTESIS CURRICULARES DE LOS AUTORES

Ing. Dunior Socarrás Benítez: Nacido en Manzanillo, Granma, Cuba el 26 de Junio de 1984, graduado de Ingeniero en Ciencias Informáticas en la Universidad de las Ciencias Informáticas. Profesor Asistente desde el año 2012, ha sido profesor de las asignaturas Inglés, Administración de Servidores Linux e Historia de la Informática. Lee, escribe y habla el idioma Inglés. Actualmente cursa la Maestría de Informática en Salud con el CECAM, en la cual ya tiene culminado el primer diplomado del programa de esta. Ha sido tutor de 3 tesis de pregrado, obteniéndose en las mismas la máxima calificación. Actualmente se desempeña como Jefe del Departamento de Sistemas Especializados en Salud del Centro de Informática Médica, en la Facultad 7 de la Universidad de las Ciencias Informáticas.