Sistema de Reserva de Citas Médicas.

Diagrama de Arquitectura de la Herramienta
Diagrama físico de la base de datos.

DESARROLLO DE UN SISTEMA PARA RESERVA DE CITAS CLINICAS

Revisión	Fecha	Modificación Realizada	Realizado Por:
0	10/07/2018	Versión Inicial.	Maricela Escobar
1			
2			

TABLA DE CONTENIDO

Contenido

GLOSARIO DE TÉRMINOS

Definiciones:

Dentro del contexto del presente documento se definen los siguientes términos:

Consultorio: Establecimiento particular fundado por uno o varios profesores de medicina, generalmente especialistas para que las personas poco pudientes acudan a él, a consultar acerca de sus dolencias.

Historia clínica: Es un documento médico-legal que surge del contacto entre el profesional de la salud y el paciente donde se recoge la información necesaria para la correcta atención de los pacientes.

Medico: Es un profesional que practica la medicina que intenta mantener y recuperar la salud humana mediante el estudio, el diagnóstico y el tratamiento de la enfermedad o lesión del paciente.

Pacientes: Es aquella persona que sufre de dolor y malestar y, por ende, solicita asistencia médica y, está sometida a cuidados profesionales para la mejoría de su salud**.**

Paquetes de datos: Agrupa todos los datos de un paciente en un consultorio que pueden ser consultados a través del servicio web de Interoperabilidad.

Java Beans y Enterprise Beans. - facilitan la implementación de la lógica de negocio.

JEE 7: Es la versión más reciente y estable de un conjunto de especificaciones, funcionalidades y estándares orientadas al desarrollo de aplicaciones empresariales, escalables, portables, y móviles utilizando el lenguaje de programación Java.

EJB: Enterprise JavaBeans proporcionan un modelo de componentes distribuidos estándar del lado del servidor que le permite al programador abstraerse de los problemas generales de una aplicación empresarial para centrarse en el desarrollo de la lógica de negocio en sí.

JPA: Es una herramienta del estándar de Java que realiza el Mapeo objeto-relacional (ORM) entre una base de datos relacional tradicional y el modelo de objetos de una aplicación.

JSP: facilitan el desarrollo y mantenimiento del contenido HTML

1. INTRODUCCIÓN

1.1 Objeto del Documento.

Presenta la descripción de cada uno de los componentes de la arquitectura para la creación del sistema de reservas citas médicas.

Una Arquitectura de Software, consiste en un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software de un sistema de información.

La arquitectura de la aplicación permite definir la organización interna de la misma estableciendo una funcionalidad por capas y bien definida. Permiten tener mejor control de la ubicación de las reglas de negocio y por lo tanto facilita las tareas de mantenimiento de las mismas.

1.2 Equipo de Trabajo.

Las personas que directa o indirectamente intervinieron son las siguientes:

Nombre	Cargo
Maricela Escobar	Estudiante de la Carrera de Ingeniería en Software

Por parte del Consultorio Med participan:

- Marco Herrera

1.3 Material Revisado.

- Estándares y herramientas para el desarrollo de aplicaciones Java.
- Estándares y herramientas para el desarrollo de aplicaciones Java de Dimutza.

DESARROLLO DE UN SISTEMA PARA RESERVA DE CITAS CLINICAS

2. DIAGRAMA DE ROLES

Administrador

Puede:

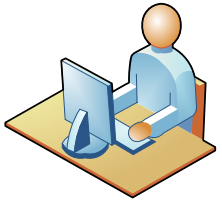
-Configurar Agenda

-Modificar datos personales

-Modificar disponibilidad

-Crear/ bloquear usuarios

-Crear jornadas



Recepción

Puede (acciones asignadas por el administrador):

-Reservar horas

-Anular horas

-Modificar disponibilidad´

-Asignar sobrecupos

-Bloqueo de horario.

Médico

Puede:

-Crear su agenda de trabajo.

-ver Nª de consultas tanto por día como totales.

-Modificar disponibilidad.

Paciente

Puede

-Explorar agenda

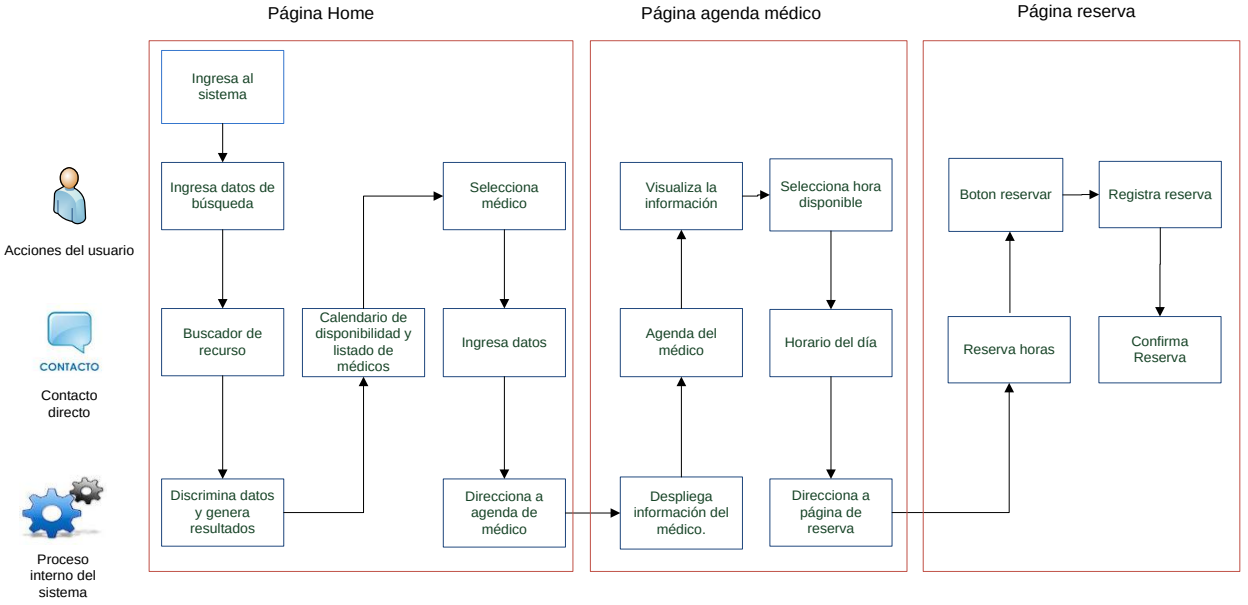
-Reservar horas

-Confirmar horas

-Anular horas

-Modificar horas

Reservas de horas médicas



3. CASO DE USO

3.1 Ingreso al sistema

ID RQ	ID CU	Descripción	Detalle	Restricciones
1	1.0	Pacientes que solicitan reserva por primera vez.	Al ingresar por primera vez, el sistema solicitara los siguientes datos: -Seleccionar tipo de documento. (ID)* -Cedula* -Nombres* -Apellido Paterno. * -Apellido Materno.* -Fecha de nacimiento. (DD/MM/YYYY) -Sexo. Deberá tener un combo box en el cual se podrá seleccionar masculino, femenino u otro.* -Nº Teléfono. -Celular.* -E-mail.* -Dirección. -Región. -Comuna. -Nacionalidad. -Ingresar Contraseña (máximo 6 dígitos). Una vez que se registran estos datos deberá emitir un mensaje que diga registro fue exitoso.	*Estos Datos son de carácter obligatorio, en caso de no ingresar todos los datos, o ingresar un dato no valido, el sistema deberá reconocer cual es el dato que falta o erróneo y solicitarlo.
2	1.0	Pacientes que se encuentran en la agenda médica.	Se deberá ingresar el DI (pasaporte) Al acceder al sistema, se podrá ingresar a las siguientes ventanas: -Atenciones. -Reserva de Horas	
3	1.0	Centros médicos con más de una sucursal	En caso de que el centro médico tenga más de una sucursal o centro, la agenda deberá tener la opción de escoger el centro médico por región y por centro, en la que la cual desea realizar una reserva.	La agenda deberá funcionar de forma independiente por cada centro médico. Esta selección deberá aparecer antes de comenzar la reserva de horas. Deberá aparecer seleccionado por defecto una región y un centro médico.
4	1.0	Historial de atenciones.	Al ingresar al historial de atenciones se deberán mostrar listadas todas las atenciones realizadas, además En pantalla se deberán	Solo se puede visualizar el historial de un paciente por vez. Al ingresar al historial se deberá mostrar

			visualizar todas las personas que estén anexados a ese usuario.	por defecto el paciente principal. (usuario registrado)
5	1.0	Ingreso al sistema de recepcionista, administrador, médico.	El ingreso al sistema por parte de estos usuarios se habilitará mediante un usuario y contraseña, estos serán proporcionados por cada servicio.	

2. Realizar Reserva

ID RQ	ID CU	Descripción	Detalle	Restricciones
1	2.0	Elección de personal al cual se asignará cita.	Para elegir un profesional al cuál se le va a asignar una cita con un paciente se deberán seguir los siguientes criterios: • Comenzar eligiendo una especialidad determinada, luego la prestación, y por último elegir entre todos los profesionales que cumplan con estas condiciones. Por ejemplo, alguien necesita un oftalmólogo, no le importa quién sea el médico que lo atienda. • Buscar por el nombre del médico, este campo se deberá poder escribir y al realizar la búsqueda deberá traer todos los datos que encuentre de médicos según coincidencia, deberá mostrar la hora más cercana de este y su horario en calendario.	
2	2.0	Futuras citas	Dentro de la agenda se deberá poder insertar citas futuras. La inserción de estas dependerá de cómo el servicio tenga programada su horario. Se visualizará el último calendario que se tenga disponible.	
3	2.0	Identificación de confirmación.	En recepción se deberá poder identificar a través de cambios de estado las reservas confirmadas.	

4	2.0	Reservas no confirmadas.	Las reservas que no se encuentren confirmadas se distinguirán de las confirmadas con un color, una vez confirmadas se les podrá realizar el cambio de estado.	
5	2.0	Confirmación de Reserva por paciente.	Al confirmar una cita se deberán desplegar en pantalla los siguientes datos: -fecha de reserva. -Hora de reserva. -Profesional. -Especialidad. Debe existir la opción de volver a la ventana de atenciones.	
6	2.0	Reserva de hora en recepción.	En recepción, se deberá visualizar el día actual, además deberá tener la opción de cambiar de fechas. En pantalla se mostrará la lista de trabajo. Deberá existir un botón el cual redireccione a la reserva de horas.	

2.1. Anulación de Reserva

ID RQ	ID CU	Descripción	Detalle	Restricciones
1	2.1	Anulación de reserva	Deberá existir un botón que permita anular dicha reserva.	

2.2 Consulta de Reserva

ID RQ	ID CU	Descripción	Detalle	Restricciones
1	2.2	Consulta de una reserva	El paciente cuando se encuentre dentro del historial de atenciones deberá poder consultar sobre su reserva. En caso de tener usuarios anexados a su cuenta, en pantalla se visualizaran todas las reservas separas por nombre de usuario. En caso de contar con una reserva se	

			visualizará la información respecto de la reserva.	
2	2.2	Cambio de reserva	Al ingresar a la vista atenciones, las reservas que aún se encuentran vigentes deberán tener la opción de modificarse. Deberá existir un botón el cual al presionarlo re direccionará a la reserva de horas, una vez realizada otra reserva, la que existía quedara completamente anulada.	
3	2.2	Consulta de reservas Recepción	En recepción se podrán consultar todas las reservas que se encuentren en la agenda.	
4	2.2	Consulta de reservas Doctor	Se deberá tener la opción de buscar Reservas en el tiempo. Se deberán mostrar las columnas de pacientes, procedimiento, hora, sala, disponibilidad, reserva, y la opción de levantar la ficha clínica.	El doctor solo podrá visualizar las reservas que se encuentran asignadas a él. Al ingresar se deberá mostrar por defecto el día actual.
5	2.2	Búsqueda de paciente en interfaz Doctor.	Deberá tener un buscador por coincidencia.	

2.3 Buscador de paciente

ID RQ	ID CU	Descripción	Detalle	Restricciones
1	2.3	Búsqueda en recepción	Se requiere que exista un buscador de pacientes, este deberá tener los mismos campos de búsqueda que el registro, deberá traer las búsquedas por coincidencia.	

3.0 Cambios de estado

ID RQ	ID CU	Descripción	Detalle	Restricciones
1	3.0	Cambio de estado en Recepción	En las listas de reservas deber existir por cada reserva que se tenga un botón el cual indique que dicha atención se encuentra pagada. Al realizar este	

			cambio de estado en la interfaz de doctor deberá aparecer en la columna disponibilidad como confirmado.	
2	3.0	Cambio de estado en Médico	Una vez finalizada la atención el médico deberá cambiar de estado a “atendido”. Con esto desaparecerá el paciente de la lista de trabajo.	

4. ARQUITECTURA

4.1 Arquitectura trabajada

Fue necesario organizar el código de forma que fuera fácil su desarrollo y para esto se utilizó el patrón de diseño MVC, donde se separa de las siguientes capas:

1. **Capa Modelo:** esta capa define la lógica del negocio y es donde se encuentra la base de conocimiento extraída de los modelos de minería de datos útiles.
2. **Capa Vista:** Esta capa contiene la interfaz de usuario o comúnmente llamada GUI, la cual hace que los usuarios puedan interactuar con el sistema.
3. **Capa controlador:** Aquí se encuentra las clases que contienen la ejecución del modelo de minería de datos con el fin de obtener los patrones requeridos por el usuario por medio de la capa vista. Esta capa actúa como un intermediador entre la GUI y la base de conocimiento.

En la siguiente ilustración se puede observar el comportamiento a nivel global de la arquitectura utilizada:

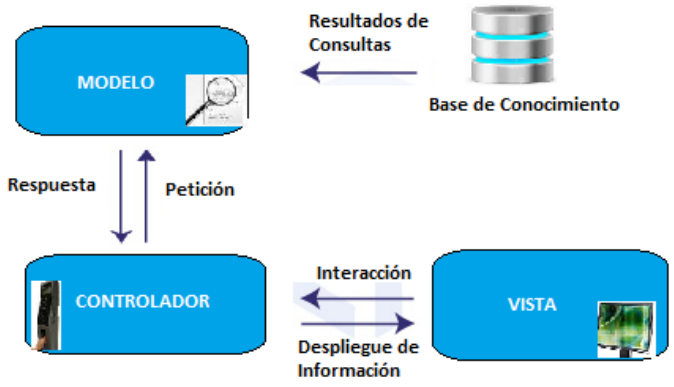


Ilustración 1 Vista global

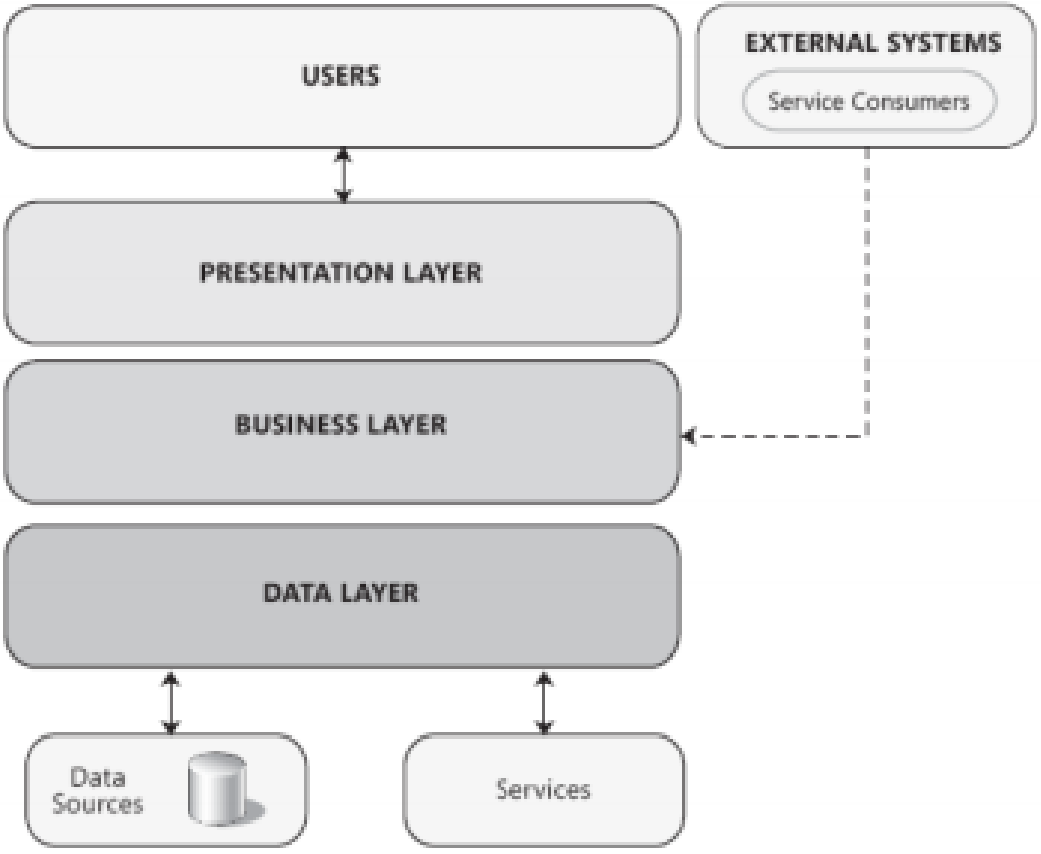
Se puede observar en la anterior ilustración que los componentes se comunican mediante un protocolo de petición – respuesta. Para el caso particular del proyecto, se asignaron los paquetes que componen la aplicación a cada uno de los componentes de la arquitectura de la siguiente forma:

- **Vista:** GUI
- **Controlador:** Lógica
- **Modelo:** Conexión

4.2 Arquitectura Lógica

El patrón de arquitectura elegido para la creación del sistema reserva de citas clínicas es el de niveles de abstracción o capas que recomienda el la Arquitectura de n-Niveles; este modelo de arquitectura se encuentra basado en la especificación JEE.

Diseño



2.2.1 Capa de Interfaz de Usuario:

La capa de presentación (vista y controlador) se ejecutarán en el contenedor web, las capas de negocio y persistencia (modelo) se ejecutarán en un servidor de aplicaciones. Para poder desplegar la aplicación es necesario contar con un servidor web como por ejemplo (apache) y un servidor de aplicaciones que posea servicios web

2.2.2 Capa de Aplicación:

La capa de aplicación es la responsable de manejar el flujo de la aplicación. Contiene La capa de presentación es el browser o la interfaz gráfica que le presenta el sistema al usuario al momento de abrir la aplicación, por lo que debe poseer las características principales de ser amigable, entendible y fácil de usar, esta capa también permite comunicar y capturar la información del usuario. Solo se puede comunicar con la capa del negocio.

2.2.3 La capa lógica del negocio:

Es la capa donde se encuentran los programas que están ejecutando, recibe las peticiones del usuario y envía la respuesta tras el proceso. En esta capa se establecen todas las normas o reglas que deben cumplirse en esta arquitectura.

2.2.4 Capa de Dominio

La capa de dominio o lógica de negocio contiene la implementación del ORM usando JPA. También contiene las validaciones y reglas propias de negocio

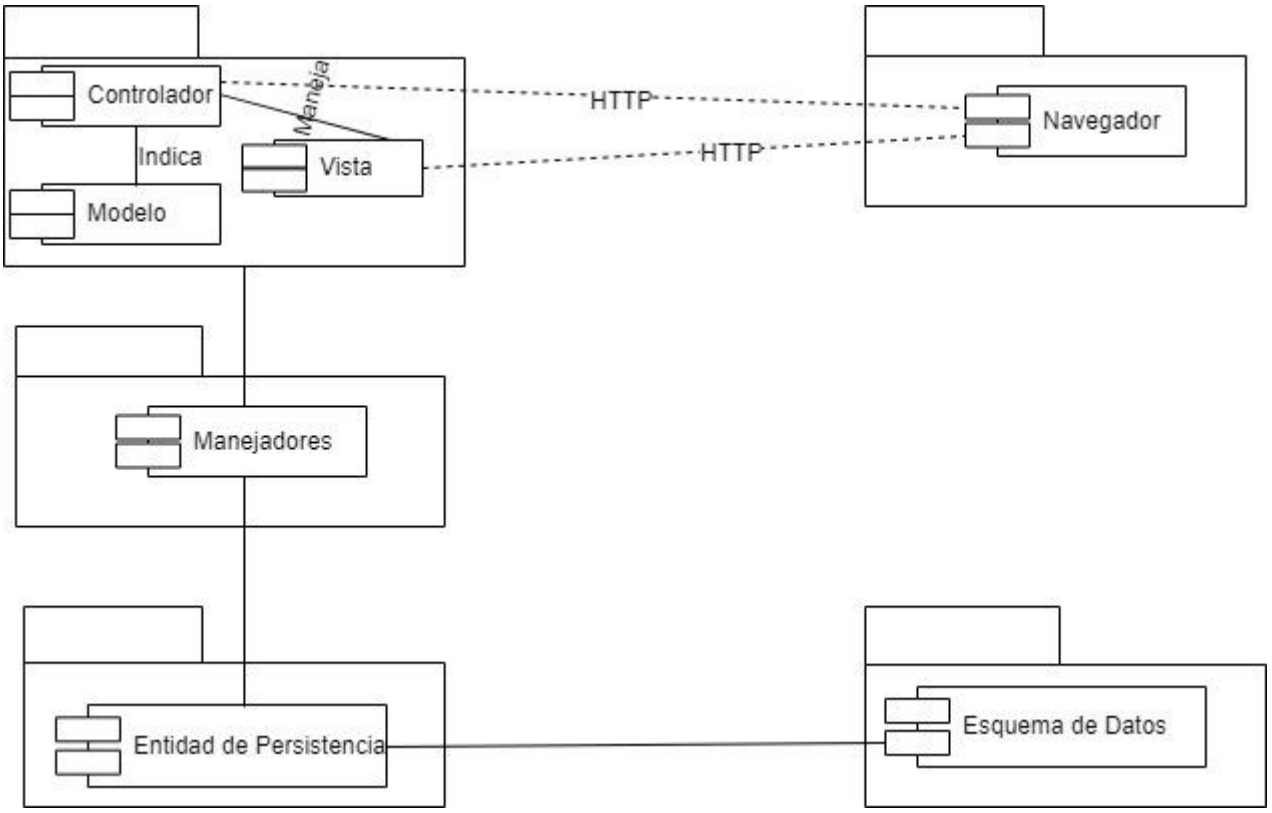
Esta capa se encuentra compuesta por:

Data.- JPA Entities que contienen además la lógica de negocio.

Repository.- Interfaces dispuestas por el dominio para que la infraestructura pueda consumir sus objetos.

2.2.5 Capa de Infraestructura:

Capa de datos es una base de datos donde se guarda toda la información, para después acceder a la misma. Por lo que su característica principal es almacenar y devolver datos a la capa de la lógica del negocio



2.3 Arquitectura Física

A nivel físico el sistema cuenta con un esquema distribuido entre múltiples plataformas:

Java DB.- Base de datos que contiene información.

JPA. - Es una herramienta del estándar de Java que realiza el Mapeo objeto-relacional (ORM) entre una base de datos relacional tradicional y el modelo de objetos de una aplicación.

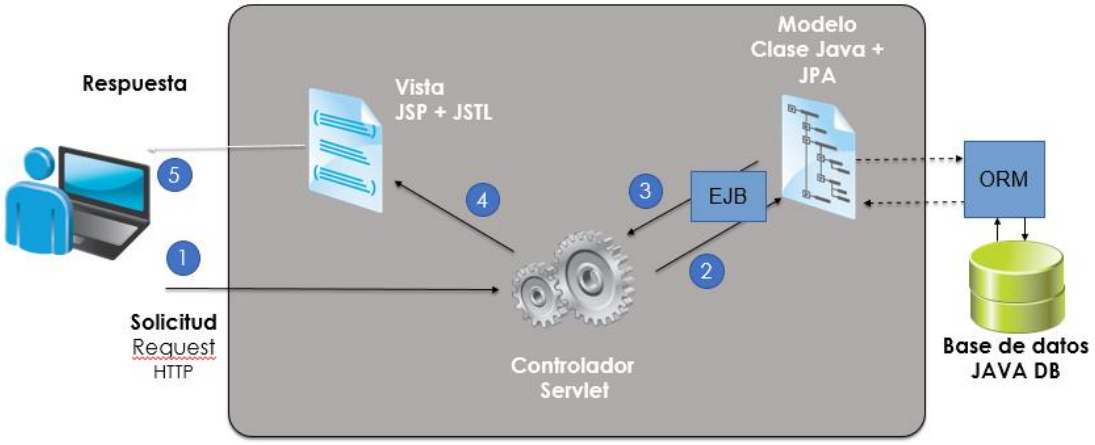
ORM. - es un JPA (persistence unit)

GlassFish Server 4.1, es un servidor de aplicaciones de software

Servlet(Request/Response).- evalúa los parámetros

Entity Manager. - CRUD reglas del negocio

Vista. - página web



Utiliza *GlassFish Server 4.*1, NetBeans 8.2 y si utilizamos Postgress 4

5. CALIDAD

Las consideraciones arquitectónicas y de diseño mostradas en este documento contribuyen a elevar la calidad del servicio de la siguiente manera:

Escalabilidad y Confiabilidad: apoyada por el diseño modular y la separación en componentes, esta permite distribuir los componentes entre varios servidores en caso de ser requerido.

Mantenimiento: Los patrones arquitectónicos que se usan permiten que la solución sea lo suficientemente flexible separando claramente cada parte funcional de la solución para soportar cambios en una manera consistente y sencilla.

6. DOCUMENTACIÓN

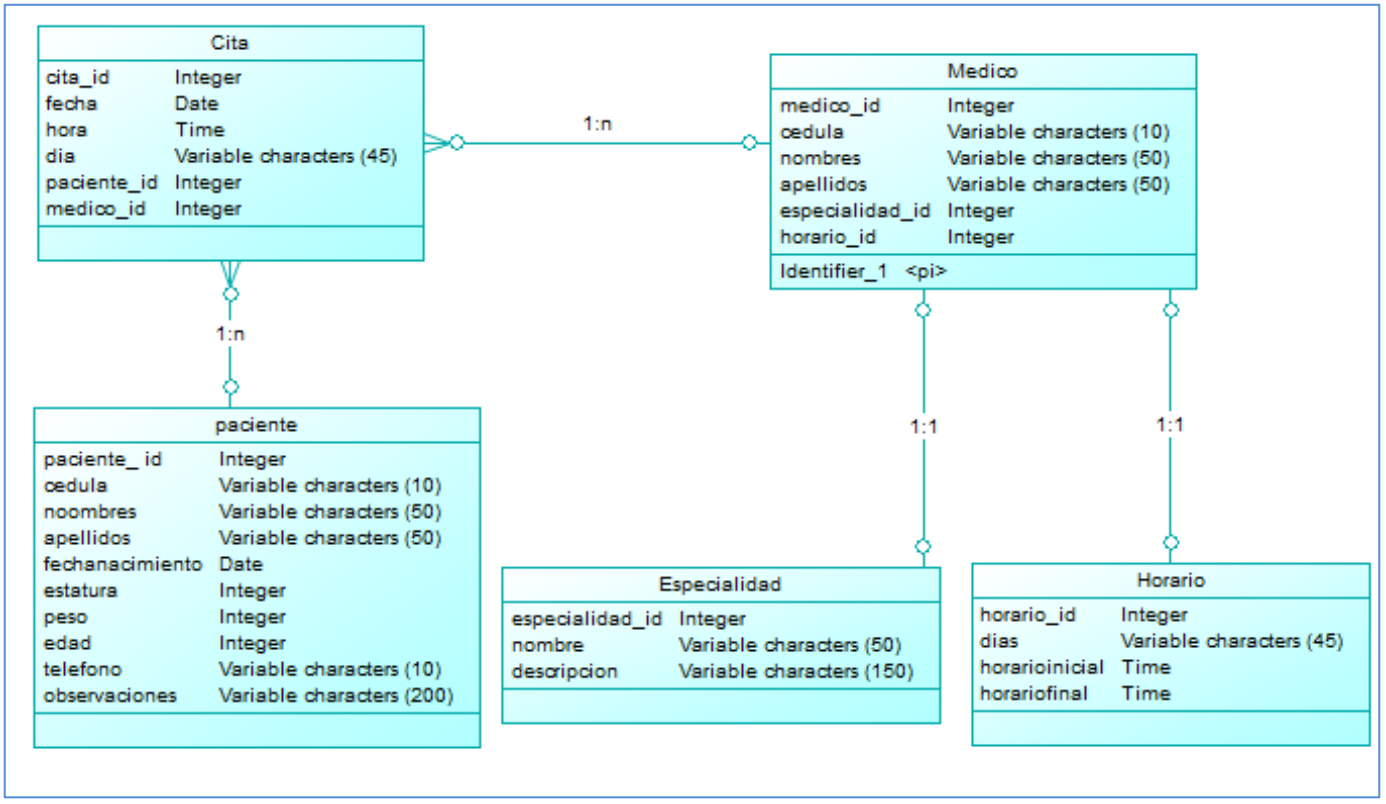
Se listan los documentos a entregar junto con el proyecto:

- Diagrama de Arquitectura.
- Diagrama Físico de Base de Datos.
- Manual de Usuario.
- Plan de Pruebas.

7. Anexos

7.1 Modelo Físico

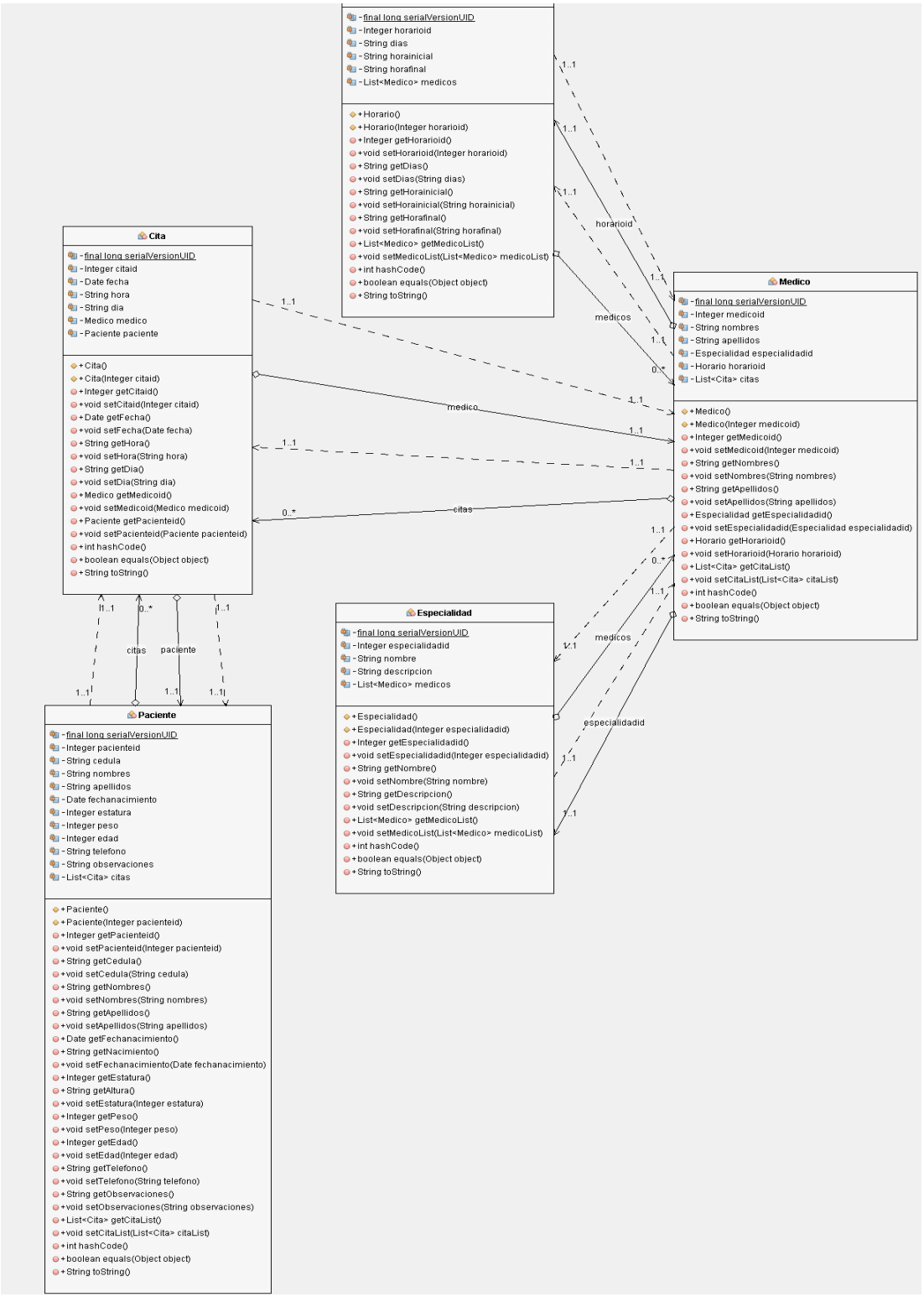
Modelo Fisico.png



7.2 Diagrama de Clases

5.2.1 Modelos

modelos.png



5.2.2 Controladores

Controladores.png

5.2.3 Servicios

Servicios.png