Proyecto Final

Fernely Andrés Ortiz Ochoa Código: 1091203555

Isabel Fernanda Angulo Martinez Código: 1092456876





Profesor: Moisés Jhonatan Renteria Campaña

Universidad del Quindío
Facultad de Ingeniería
Ingeniería de Sistemas y Computación
Teoría y Diseño Organizacional
Grupo 02 diurno
Armenia, Quindío, 2025









Pensamiento Computacional del Sistema de gestión hospitalaria

1. Abstracción

¿Qué se solicita finalmente?

Se solicita el desarrollo de un sistema eficiente para la gestión de pacientes, citas médicas y personal médico en el Hospital. Este sistema debe optimizar la administración de historiales médicos, programación de consultas y asignación de médicos, garantizando una interfaz intuitiva.

¿Qué información es relevante dado el problema anterior?

- Datos personales de pacientes, médicos y administradores.
- Información de citas médicas (fechas, horarios, médicos asignados).
- Historiales médicos con diagnósticos y tratamientos.
- Disponibilidad de médicos y salas de consulta.
- Notificaciones sobre citas y cambios en horarios.

¿Qué información es relevante dado el problema anterior?

- Datos personales de los pacientes: Nombre completo, identificación, fecha de nacimiento, dirección, teléfono, correo electrónico, tipo de sangre, antecedentes médicos y alergias.
- **Datos personales de los médicos:** Nombre completo, especialidad, número de licencia, identificación, contacto y horarios de atención.
- **Información de citas médicas:** Fecha, hora, médico asignado, paciente, motivo de la consulta, estado de la cita (pendiente, confirmada, cancelada), historial de citas previas.
- **Historiales médicos de los pacientes:** Diagnósticos, tratamientos, medicamentos recetados, exámenes médicos, evolución del paciente, notas médicas.
- **Disponibilidad de recursos:** Horarios y turnos de médicos, disponibilidad de salas de consulta, capacidad hospitalaria, número de pacientes atendidos por día.
- Notificaciones y recordatorios: De cambios de realizados en las citas.

¿Cómo se agrupa la información relevante?







Persona

- +nombre:string
- +apellido:string
- +id:string
- +telefono:string
- +email:string
- +direccion:string

Médico

- -especialidad:string
- -consultorio:string
- -numLicencia:strin

g

Administrador

-hospitalAsignado:s tring

Diagnostico

- -idDiagnostico:strin
- -descripcion:string
- -fecha:string

Cita

-idCita:string -fecha y hora:string

Medicamento

especialidad:string

consultorio:string

numLicencia:string

historialMedico

-idHistorial:strin

Sala

leam

-idSala:string

-capacidad:string

-disponibilidad:boo

Tratamiento

- -idTratamiento:stri
- -descripcion:string
- -duracion:string

Paciente

- -fechaNacimiento:s tring
- -grupoSanguinio:str







2.Descomposición:

qué funcionalidades se solicitan finalmente?
Funcionalidades Generales de Pacientes
Registro y actualización de datos personales.
Solicitud y cancelación de citas médicas.
Consulta de historial médico.
Recepción de notificaciones sobre citas programadas.
Funcionalidades Generales de Médicos
Acceso a los historiales médicos de sus pacientes.
Registro de diagnósticos y tratamientos.
Administración de horarios de consulta.
Notificación de cambios en las citas

Administradores

Registro, modificación y eliminación de médicos y pacientes. Gestión de salas y horarios de atención. Monitoreo de disponibilidad de médicos y asignación de pacientes.

Generación de reportes sobre citas médicas y ocupación del hospital.

3. Descomposición:

¿Cómo se distribuyen las funcionalidades?:

En el diagrama de clases.

¿Cómo se prueban? Con JUnit 5

Reconocimiento de patrones:

¿Qué puedo hacer para probar la solución en java?: No aplica.







Codificación:

como pruebo y como escribo la solución en java?

```
private List<Cita> citas; 5 usages
   private List<Notificacion> notificaciones; 5 usages
   this.citas = new ArrayList<>();
      this.notificaciones = new ArrayList<>();
   // Agregar la cita a la lista
      citas.add(cita);
      // Crear la notificación
      String mensaje = "Estimado/a " + cita.getPaciente().getNombre() +
            ", su cita fue programada para: " + cita.getFechaHoraCita();
      Notificacion notificacion = new Notificacion(mensaje, LocalDateTime.now());
      // Guardar la notificación sin mostrarla
      notificaciones.add(notificacion);
   for (Notificacion notificacion : notificaciones) {
         notificacion.mostrar(); // Se imprimen solo cuando el usuario lo decida
```







```
Administrador administrador; 34 usages
   Paciente paciente; 5 usages
   @BeforeEach
                Isabel Angulo
   void setUp() {
       administrador = new Administrador( nombre: "Carlos", apellido: "Gomez", id: "admin1");
       paciente = new Paciente( nombre: "123", apellido: "Juan", id: "Perez", java.time.LocalDate.of( year:
       administrador.agregarPaciente(paciente);
   }
   @Test
           Isabel Angulo
   void eliminarPaciente() {
       assertTrue(administrador.getPacienteAdministrados().contains(paciente));
       // Elimina el paciente
       administrador.eliminarPaciente(paciente);
       // Verifica que ya no está en la lista
       assertFalse(administrador.getPacienteAdministrados().contains(paciente));
   }
```







Universidad del Quindío Facultad de Ingeniería Tel: (57) 6 735 9300 Ext Carrera 15 Calle 12 Norte Armenia, Quindío – Colombia correoelectronico@uniquindio.edu.co

UNIQUINDÍO, en conexión territorial

Carrera 15 Calle 12 Norte Tel: (606) 7 35 93 00 Armenia - Quindío - Colombia