

Entregable 11 – Analista.

Casas Vázquez Luis Américo

22140750

Ingeniería en Sistemas Computacionales, Instituto Tecnológico de México campus

Querétaro.

Ingeniería de Software.

6A

SCC1007

Villeda Maldonado Julio Alejandro

26 de febrero de 2026



Listas de clases

A) Entidades

- Paciente

- **Atributos:** idPaciente, nombreP, apellidoPat, apellidoM, fecha_nac, sexo, curp, domicilio, estadoCivil, correo, ocupacion, telefono, contactoEmergencia, tipoSangre, alergias, enfCronicas, estatus.

- ExpedienteClinico

- **Atributos:** idExp, idPaciente, idMedico, fecha, motivo, antecedentesPer, antecedentesFam, presionArt, frecCardiaca, frecResp, temp, saturacionO2, peso, talla, altura, imc, diagnostico, tratamiento, observaciones, estatus.

- Usuario

- **Atributos:** idUsuario, idRol, nombreUsuario, passwordHash, status.

- **Métodos:**

- iniciarSesion(contrasenaIngresada)
- cerrarSesion()
- cambiarContrasena(hashNueva)

*La contraseña se cifra en el Service.

Clases Hijas

- Medico
 - **Atributos:** cedula, especialidad.

- Enfermero
 - **Atributos:** licencia.

- Rol

- **Atributos:** idRol, nombreRol, descr.

Datos

datosP = nombreP, apellidoPat, apellidoM, fecha_nac, sexo, curp, domicilio, estadoCivil, correo, ocupacion, telefono, contactoEmergencia, tipoSangre, alergias, enfCronicas.

datosExp = idPaciente, idMedico, fecha, motivo, antecedentesPer, antecedentesFam, presionArt, frecCardiaca, frecResp, temp, saturacionO2, peso, talla, altura, imc, diagnostico, tratamiento, observaciones.

nombreCompleto = (nombre + ‘ ‘ + apellidoPat + ‘ ‘ + apellidoM).

criteriosExp = nombreP | idPaciente| idExp | curp | idExp | fecha| idMedico.

B) Controladores

ControladorPaciente

- registrarPaciente(datosP)
- consultarPaciente(CURP|ID| nombreCompleto)



- actualizarPaciente(idPaciente, cambios)
- darDeBajaPaciente(idPaciente)

ControladorExpediente

- registrarExpediente(datosExp)
- consultarExpediente(criteriosExp)
- actualizarExpediente(idExp, cambios)
- eliminarExpediente(idExp)

ControladorUsuario

- crearUsuario(nombreUsuario, contrasena)
- asignarRol(idUsuario, idRol)

C) Servicios (Service)

- **ServicioPaciente**
 - registrarPaciente(datosP)
 - validarDatosPaciente(datosP)
 - consultarPaciente(CURP|ID| nombreCompleto)
 - actualizarDatosPaciente(idPaciente, cambios)
 - darDeBajaPaciente(idPaciente)
- **ServicioExpediente**
 - registrarExpediente(datosExp)
 - consultarExpediente(criteriosExp)
 - actualizarExpediente(idEx, cambios)
 - eliminarExpediente(idEx)
- **ServicioUsuario**
 - crearUsuario(nombreUsuario, contrasena)
 - asignarRol(idUsuario, idRol)
 - cambiarContrasena(idUsuario, nuevaContrasena)

D) Repositorios (Repository)

- **RepositoryPaciente**
 - guardar(p: Paciente)
 - buscar(CURP|ID| nombreCompleto)
 - actualizar(p: Paciente)
 - cambiarEstatusBaja(idPaciente)
- **RepositoryExpediente**
 - guardar(e: ExpedienteClinico)
 - buscar(criteriosExp)
 - actualizar(e: ExpedienteClinico)
 - eliminarPorId(idExp)
- **RepositoryUsuario**
 - guardar(u: Usuario)
 - buscar(idUsuario, nombreUsuario)
 - actualizar(u: Usuario)



- **RepositoryRol**

- buscar(idRol | nombreRol)

Relación Requisito – Clase

Requisito	Caso de uso	Clases involucradas
RF-01	Registrar paciente	Paciente, ControladorPaciente, ServicioPaciente, RepositoryPaciente, Medico, Enfermero
RF-02	Consultar paciente	Paciente, ControladorPaciente, ServicioPaciente, RepositoryPaciente, Medico, Enfermero.
RF-03	Actualizar datos del paciente	Paciente, ControladorPaciente, ServicioPaciente, RepositoryPaciente, Medico, Enfermero
RF-04	Dar de baja paciente	Paciente, ControladorPaciente, ServicioPaciente, RepositoryPaciente, Medico
RF-05	Registrar expediente clínico	ExpedienteClinico, ControladorExpediente, ServicioExpediente, RepositoryExpediente, Paciente, Medico
RF-06	Consultar expediente clínico	ExpedienteClinico, ControladorExpediente, ServicioExpediente, RepositoryExpediente, Paciente, Medico, Enfermero.
RF-07	Actualizar expediente clínico	ExpedienteClinico, ControladorExpediente, ServicioExpediente, RepositoryExpediente, Medico
RF-08	Eliminar expediente clínico	ExpedienteClinico, ControladorExpediente, ServicioExpediente, RepositoryExpediente, Medico
RF-09	Asignar roles	Usuario, Rol, ControladorUsuario, ServicioUsuario, RepositoryUsuario, RepositoryRol.
RF-10	Crear usuarios	Usuario, ControladorUsuario, ServicioUsuario, RepositoryUsuario.



Justificación

Se adoptó una separación por capas para mantener responsabilidades claras: los Controladores orquestan los casos de uso, los Servicios concentran validaciones y reglas del dominio, los Repositorios encapsulan el acceso y persistencia de datos, y las Entidades representan la información del negocio. Esta estructura reduce acoplamiento, mejora mantenibilidad y facilita la trazabilidad entre requisitos y componentes.

Se modeló la clase Usuario como superclase y se definieron las subclases Medico y Enfermero, que tienen atributos propios. Para los otros actores (Personal Administrativo, Director, Administrador del sistema) no se crearon subclases ya que no cuentan con atributos propios. Sus responsabilidades se gestionan mediante el uso de la entidad Rol.

La entidad Rol no cuenta con controlador ni servicio propio debido a que los roles se encuentran predefinidos dentro del sistema. La validación de permisos ocurre dentro de la capa de servicios.

Se modela que un Paciente puede tener múltiples ExpedienteClinico, porque el sistema maneja varios registros clínicos a lo largo del tiempo para el mismo paciente.