

***Entregable 11 – Diseñador.***

Ruiz Avilés Fernanda Michelle

22140756

*Ingeniería en Sistemas Computacionales, Instituto Tecnológico de México campus Querétaro.*

**DIAGRAMA DE CLASES UML – MODELADO ESTÁTICO COMPLETO**

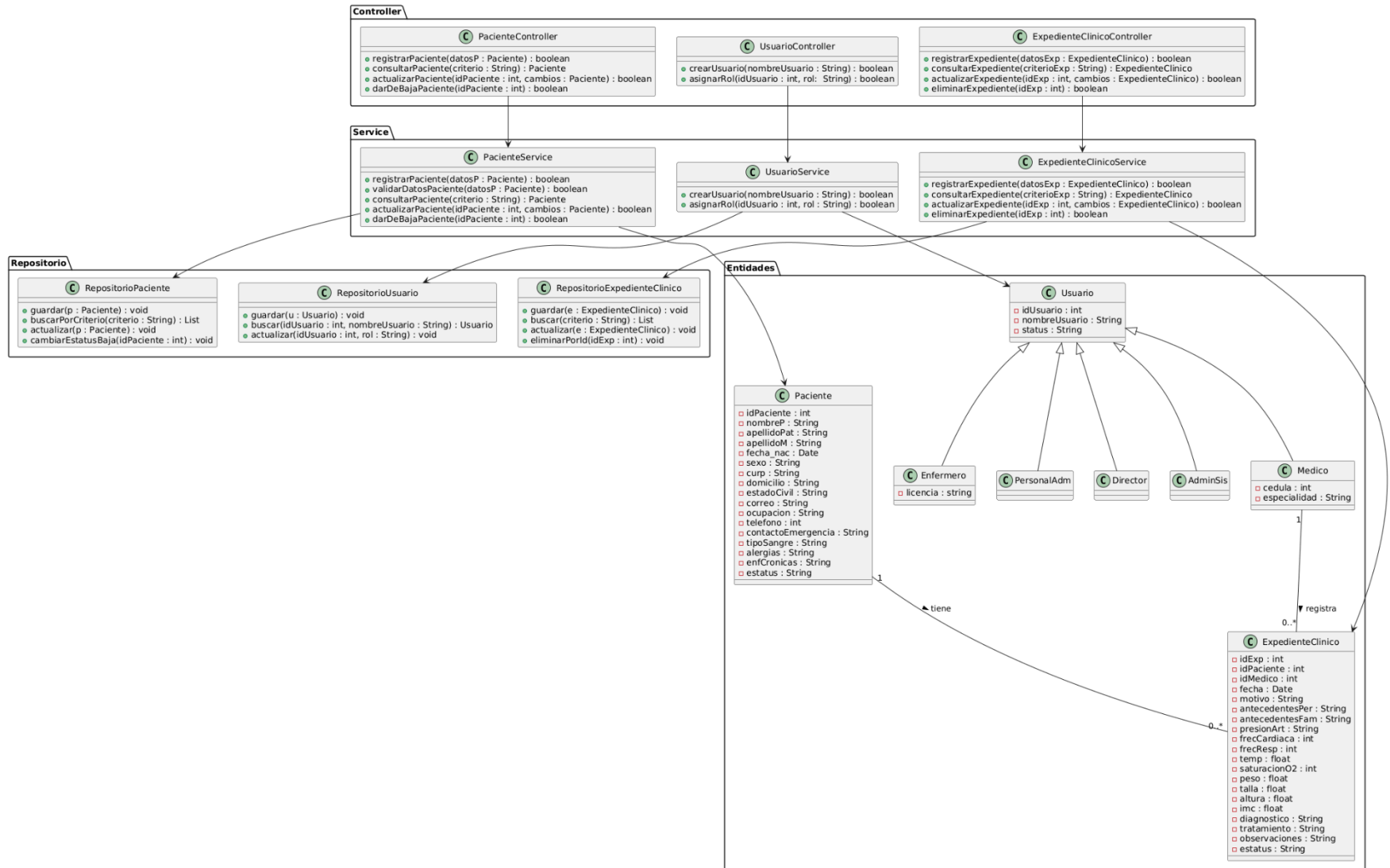
Ingeniería de Software.

6A

SCC1007

Villeda Maldonado Julio Alejandro

26 de febrero de 2026



## Datos

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

### consultarPaciente

- criterios = CURP, ID, nombreCompleto

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

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

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

@startuml

```
package "Controller"{
class PacienteController {
+ registrarPaciente(datosP : Paciente) : boolean
+ consultarPaciente(criterio : String) : Paciente
+ actualizarPaciente(idPaciente : int, cambios : Paciente) : boolean
+ darDeBajaPaciente(idPaciente : int) : boolean
}

class ExpedienteClinicoController {
+ registrarExpediente(datosExp : ExpedienteClinico) : boolean
+ consultarExpediente(criterioExp : String) : ExpedienteClinico
+ actualizarExpediente(idExp : int, cambios : ExpedienteClinico) : boolean
+ eliminarExpediente(idExp : int) : boolean
}

class UsuarioController {
+ crearUsuario(nombreUsuario : String) : boolean
+ asignarRol(idUsuario : int, rol : String) : boolean
}
}
```

```

package "Service"{
class PacienteService {
    + registrarPaciente(datosP : Paciente) : boolean
+ validarDatosPaciente(datosP : Paciente) : boolean
    + consultarPaciente(criterio : String) : Paciente
    + actualizarPaciente(idPaciente : int, cambios : Paciente) : boolean
    + darDeBajaPaciente(idPaciente : int) : boolean
}

class ExpedienteClinicoService {
    + registrarExpediente(datosExp : ExpedienteClinico) : boolean
    + consultarExpediente(criterioExp : String) : ExpedienteClinico
    + actualizarExpediente(idExp : int, cambios : ExpedienteClinico) : boolean
    + eliminarExpediente(idExp : int) : boolean
}

class UsuarioService{
    + crearUsuario(nombreUsuario : String) : boolean
    + asignarRol(idUsuario : int, rol: String) : boolean
}
}

package "Repositorio"{
class RepositorioPaciente{
    + guardar(p : Paciente) : void
    + buscarPorCriterio(criterio : String) : List
    + actualizar(p : Paciente) : void
    + cambiarEstatusBaja(idPaciente : int) : void
}

class RepositorioExpedienteClinico {
    + guardar(e : ExpedienteClinico) : void
    + buscar(criterio : String) : List
    + actualizar(e : ExpedienteClinico) : void
    + eliminarPorId(idExp : int) : void
}

class RepositorioUsuario {
    + guardar(u : Usuario) : void
    + buscar(idUsuario : int, nombreUsuario : String) : Usuario
    + actualizar(idUsuario : int, rol : String) : void
}
}

```

```
package "Entidades"{
class Paciente {
  - idPaciente : int
  - nombreP : String
  - apellidoPat : String
  - apellidoM : String
  - fecha_nac : Date
  - sexo : String
  - curp : String
  - domicilio : String
  - estadoCivil : String
  - correo : String
  - ocupacion : String
  - telefono : int
  - contactoEmergencia : string
  - tipoSangre : String
  - alergias : String
  - enfCronicas : String
  - estatus : String
}
```

```
class ExpedienteClinico {
  - idExp : int
  - idPaciente : int
  - idMedico : int
  - fecha : Date
  - motivo : String
  - antecedentesPer : String
  - antecedentesFam : String
  - presionArt : String
  - frecCardiaca : int
  - frecResp : int
  - temp : float
  - saturacionO2 : int
  - peso : float
  - talla : float
  - altura : float
  - imc : float
  - diagnostico : String
  - tratamiento : String
  - observaciones : String
  - estatus : String
}
```

```
class Usuario {  
  - idUsuario : int  
  - nombreUsuario : String  
  - status : String  
}
```

```
Usuario <|-- Medico  
Usuario <|-- Enfermero  
Usuario <|-- PersonalAdm  
Usuario <|-- Director  
Usuario <|-- AdminSis
```

```
class Medico {  
  - cedula : int  
  - especialidad : String  
}
```

```
class Enfermero {  
  - licencia : string  
}  
}
```

```
Paciente "1" -- "0..*" ExpedienteClinico : tiene >  
Medico "1" -- "0..*" ExpedienteClinico : registra >
```

```
PacienteController --> PacienteService  
PacienteService --> RepositorioPaciente  
PacienteService --> Paciente
```

```
ExpedienteClinicoController --> ExpedienteClinicoService  
ExpedienteClinicoService --> RepositorioExpedienteClinico  
ExpedienteClinicoService--> ExpedienteClinico
```

```
UsuarioController --> UsuarioService  
UsuarioService --> RepositorioUsuario  
UsuarioService --> Usuario  
@enduml
```