

# Contenidos



**01**

**Descripción del  
problema.**

**02**

**Diseño y descripción  
del modelo  
orientado a objetos.**

**03**

**Manejo de errores.**

**04**

**Pruebas**

**05**

**Conclusiones  
individuales**





# Descripción del problema.

- El problema se centra en la gestión de expedientes clínicos electrónicos para pacientes en una clínica médica.
- El sistema debe permitir un gestionar las consultas, la emisión de recetas, medicamentos y que los médicos estén autorizados con sus cédulas profesionales y de especialidad.

# Diseño y descripción del modelo orientado a objetos.



```
#Hacemos uso de excepciones
```

```
try:
```

```
    # Intenta abrir y leer el archivo de receta médica del paciente.
```

```
    with open(f"{paciente.get_nombre_paciente()}_receta_medica.txt", "r") as archivo:
```

```
        contenido = archivo.read()
```

```
        print(contenido)
```

```
except FileNotFoundError:
```

```
    # Si no se encuentra la receta médica, muestra un mensaje correspondiente.
```

```
    print(f"No se encontró una receta médica para {paciente.get_nombre_paciente()}.")
```

## Manejo de errores



Es la capacidad de tu programa para detectar, gestionar y responder a situaciones excepcionales o inesperadas que pueden ocurrir durante la ejecución de un programa

## Excepciones



Indican que algo inesperado o incorrecto ha sucedido, y permiten al programa responder de manera controlada en lugar de simplemente detenerse o colapsar.

# PRUEBAS



```
EXPLORADOR  ...  clinica.py  x  cirugia.py  especialidad.py  estudio_medico.py  expediente_clinico.py  historial_medico.py  historial_vacunas.py  medicamentos.py  ...

✓ SI SE PUEDE
  > __pycache__
  + cirugia.py
  ≡ cirujias.dat
  + clinica.py
  + especialidad.py
  ≡ especialidades_medicas.dat
  + estudio_medico.py
  ≡ estudios_medicos.dat
  + expediente_clinico.py
  + historial_medico.py
  + historial_vacunas.py
  ≡ medicamentos.dat
  + medicamentos.py
  + medico.py
  ≡ medicos.dat
  + paciente.py
  + receta.py

  clinica.py > Clinica > crear_expediente_clinico

520      # Utiliza la biblioteca pickle para serializar y guardar la lista '__expedientes_clinico' en el archivo.
521      pickle.dump(self.__expedientes_clinico, archivo)
522
523      # Este método permite a un paciente crear su propio expediente clínico proporcionando información personal, historial médico y registros de vacunación.
524      def crear_expediente_clinico(self):
525          #Se recopila la información sobre paciente.
526          print("\nBienvenido a la clínica. Por favor, complete su expediente clínico.")
527          nombre = input("Ingrese su nombre: ")
528          edad = input("Ingrese su edad: ")
529          fecha_nacimiento = input("Ingrese su fecha de nacimiento (YYYY-MM-DD): ")
530          genero = input("Ingrese su género: ")
531          num_seguro_social = input("Ingrese su número de seguro social: ")
532          direccion = input("Ingrese su dirección: ")
533          telefono = input("Ingrese su número de teléfono: ")
534          correo = input("Ingrese su correo electrónico: ")
535          nombre_contacto_emergencia = input("Ingrese el nombre del contacto de emergencia: ")
536
537          # Se crea un objeto 'Paciente' con la información personal proporcionada por el paciente.
538          paciente = Paciente(nombre, edad, fecha_nacimiento, genero, num_seguro_social, direccion, telefono, correo, nombre_contacto_emergencia)
539
540          print(f"Bienvenido, {nombre}! Ha sido registrado en la clínica.")
541
542          # Se recopila información adicional sobre el historial médico del paciente.
543          alergias = input("Alergias: ")
544          enfermedades_cronicas = input("Enfermedades crónicas: ")
545          procedimientos_quirurgicos = input("Procedimientos quirúrgicos previos: ")
546          enfermedades_hereditarias = input("Enfermedades hereditarias: ")
547          condiciones_familiares = input("Condiciones médicas en la familia: ")
548          peso = input("Peso: ")
549          talla = input("Talla: ")
550
551          # Se crea un objeto 'HistorialMedico' con la información recopilada.
552          historial_medico = HistorialMedico(alergias, enfermedades_cronicas, procedimientos_quirurgicos, enfermedades_hereditarias, condiciones_familiares,
553
554          # Se crea un objeto 'HistorialVacunas' para registrar las vacunas.
555          historial_vacunas = HistorialVacunas()
556
```



# PRUEBAS

---- Menú Principal ----

1. Administrador de sistema
2. Médico
3. Paciente
4. Salir

Seleccione una opción: 2

---- Menú Administrador ----

1. Eliminar Paciente
2. Eliminar Médico
3. Eliminar Medicamento
4. Eliminar Especialidad
5. Agregar Especialidad
6. Agregar Medicamento
7. Mostrar Medicamentos disponibles
8. Mostrar Especialidades disponibles
9. Mostrar Médicos Registrados
10. Consultar Expediente
11. Regresar

Seleccione una opción:

---- Menú Paciente ----

1. Nuevo usuario
2. Usuario registrado
3. Regresar

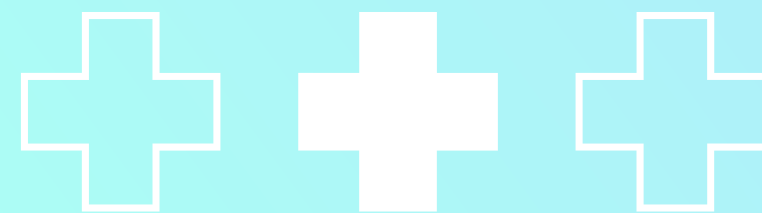
Seleccione una opción:



**Ustedes** son la presentación



**Conclusiones**



# Gracias

SAUL ROVELO  
SAMUEL PEREZ