

# TRABAJO PRÁCTICO NRO 3

## SEMINARIO DE PRÁCTICA DE INFORMATICA

### Sistema de Gestión Integral para una Clínica de Salud

Antonella Diaz DNI

28.910.424

VINFOR12606

Los entregables son los siguientes:

- 1) Explicación del desarrollo en Java.
- 2) Presentación del desarrollo en Java.
- 3) El programa compila y se ejecuta correctamente.

## DESARROLLO:

### 1. Explicación del desarrollo en Java

Estructura del Sistema: El sistema está compuesto por varios módulos interconectados, cada uno representado por una clase. Los módulos principales incluyen:

- Gestión de Citas
- Historias Clínicas Electrónicas
- Gestión de Inventarios
- Facturación y Pagos
- Informes de Gestión

Conceptos Clave Utilizados:

- Encapsulamiento: Protección de los datos sensibles de cada módulo mediante métodos públicos para acceder y modificar los datos.
- Herencia y Polimorfismo: Creación de clases base para entidades comunes (por ejemplo, Persona como base para Paciente y Medico) y utilización de métodos polimórficos para operaciones específicas.
- Abstracción: Definición de interfaces y clases abstractas para especificar comportamientos comunes sin implementar todos los detalles.
- Excepciones: Manejo de errores a través de excepciones personalizadas para asegurar la robustez del sistema.

## 2. Presentación del desarrollo en Java

Principal.java:

```
import java.util.*;

public class Main {
    public static void main(String[] args) {
        SistemaGestionClinica sistema = new SistemaGestionClinica();
        sistema.mostrarMenu();
    }
}
```

SistemaGestionClinica.java:

```
public class SistemaGestionClinica {
    private ArrayList<Paciente> pacientes;
    private ArrayList<Medico> medicos;
    private ArrayList<Cita> citas;
    private Inventario inventario;
    private Facturacion facturacion;

    public SistemaGestionClinica() {
        pacientes = new ArrayList<>();
        medicos = new ArrayList<>();
        citas = new ArrayList<>();
        inventario = new Inventario();
        facturacion = new Facturacion();
    }

    public void mostrarMenu() {
        Scanner scanner = new Scanner(System.in);
        int opcion;

        do {
            System.out.println("Bienvenido al Sistema de Gestión Integral para Clínicas de Salud");
            System.out.println("1. Gestión de Citas");
            System.out.println("2. Historias Clínicas Electrónicas");
            System.out.println("3. Gestión de Inventarios");
            System.out.println("4. Facturación y Pagos");
            System.out.println("5. Reportes de Gestión");
            System.out.println("6. Salir");
            System.out.print("Seleccione una opción: ");
            opcion = scanner.nextInt();

            switch (opcion) {
                case 1:
                    gestionarCitas(scanner);
                    break;
                case 2:
                    gestionarHistoriasClinicas(scanner);
                    break;
                case 3:
                    gestionarInventarios(scanner);
                    break;
                case 4:
                    gestionarFacturacion(scanner);
                    break;
                case 5:
                    generarReportes(scanner);
                    break;
                case 6:
                    System.out.println("Saliendo del sistema...");
            }
        } while (opcion != 6);
    }
}
```

```

        break;
    default:
        System.out.println("Opción no válida. Intente de nuevo.");
    }
} while (opcion != 6);

scanner.close();
}

private void gestionarCitas(Scanner scanner) {
    // Implementar lógica de gestión de citas
}

private void gestionarHistoriasClinicas(Scanner scanner) {
    // Implementar lógica de gestión de historias clínicas electrónicas
}

private void gestionarInventarios(Scanner scanner) {
    // Implementar lógica de gestión de inventarios
}

private void gestionarFacturacion(Scanner scanner) {
    // Implementar lógica de facturación y pagos
}

private void generarReportes(Scanner scanner) {
    // Implementar lógica de generación de reportes
}
}

```

### Clase Persona (base para Paciente y Médico):

```

public abstract class Persona {
    private String nombre;
    private String direccion;
    private String telefono;

    public Persona(String nombre, String direccion, String telefono) {
        this.nombre = nombre;
        this.direccion = direccion;
        this.telefono = telefono;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public String getDireccion() {
        return direccion;
    }

    public void setDireccion(String direccion) {
        this.direccion = direccion;
    }

    public String getTelefono() {
        return telefono;
    }

    public void setTelefono(String telefono) {
        this.telefono = telefono;
    }

    @Override
    public String toString() {
        return "Nombre: " + nombre + ", Dirección: " + direccion + ", Teléfono: " +
telefono;
    }
}

```

```
}
```

### Clase Paciente:

```
public class Paciente extends Persona {
    private String numeroHistoriaClinica;

    public Paciente(String nombre, String direccion, String telefono, String
numeroHistoriaClinica) {
        super(nombre, direccion, telefono);
        this.numeroHistoriaClinica = numeroHistoriaClinica;
    }

    public String getNumeroHistoriaClinica() {
        return numeroHistoriaClinica;
    }

    public void setNumeroHistoriaClinica(String numeroHistoriaClinica) {
        this.numeroHistoriaClinica = numeroHistoriaClinica;
    }

    @Override
    public String toString() {
        return super.toString() + ", Número de Historia Clínica: " +
numeroHistoriaClinica;
    }
}
```

### Clase Médica:

```
public class Medico extends Persona {
    private String especialidad;

    public Medico(String nombre, String direccion, String telefono, String especialidad)
{
        super(nombre, direccion, telefono);
        this.especialidad = especialidad;
    }

    public String getEspecialidad() {
        return especialidad;
    }

    public void setEspecialidad(String especialidad) {
        this.especialidad = especialidad;
    }

    @Override
    public String toString() {
        return super.toString() + ", Especialidad: " + especialidad;
    }
}
```

### Clase Cita:

```
public class Cita {
    private Date fecha;
    private Paciente paciente;
    private Medico medico;
```

```

public Cita(Date fecha, Paciente paciente, Medico medico) {
    this.fecha = fecha;
    this.paciente = paciente;
    this.medico = medico;
}

public Date getFecha() {
    return fecha;
}

public void setFecha(Date fecha) {
    this.fecha = fecha;
}

public Paciente getPaciente() {
    return paciente;
}

public void setPaciente(Paciente paciente) {
    this.paciente = paciente;
}

public Medico getMedico() {
    return medico;
}

public void setMedico(Medico medico) {
    this.medico = medico;
}

@Override
public String toString() {
    return "Fecha: " + fecha + ", Paciente: " + paciente + ", Médico: " + medico;
}
}

```

### Clase Inventario:

```

public class Inventario {
    private ArrayList<Producto> productos;

    public Inventario() {
        productos = new ArrayList<>();
    }

    public void agregarProducto(Producto producto) {
        productos.add(producto);
    }

    public void eliminarProducto(Producto producto) {
        productos.remove(producto);
    }

    public Producto buscarProducto(String nombre) {
        for (Producto producto : productos) {
            if (producto.getNombre().equalsIgnoreCase(nombre)) {
                return producto;
            }
        }
        return null;
    }

    public void mostrarInventario() {
        for (Producto producto : productos) {
            System.out.println(producto);
        }
    }
}

```

```
}
```

### Clase Producto:

```
public class Producto {
    private String nombre;
    private int cantidad;
    private String proveedor;
    private String fechaVencimiento;

    public Producto(String nombre, int cantidad, String proveedor, String
fechaVencimiento) {
        this.nombre = nombre;
        this.cantidad = cantidad;
        this.proveedor = proveedor;
        this.fechaVencimiento = fechaVencimiento;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre(String nombre) {
        this.nombre = nombre;
    }

    public int getCantidad() {
        return cantidad;
    }

    public void setCantidad(int cantidad) {
        this.cantidad = cantidad;
    }

    public String getProveedor() {
        return proveedor;
    }

    public void setProveedor(String proveedor) {
        this.proveedor = proveedor;
    }

    public String getFechaVencimiento() {
        return fechaVencimiento;
    }

    public void setFechaVencimiento(String fechaVencimiento) {
        this.fechaVencimiento = fechaVencimiento;
    }

    @Override
    public String toString() {
        return "Producto: " + nombre + ", Cantidad: " + cantidad + ", Proveedor: " +
proveedor + ", Fecha de Vencimiento: " + fechaVencimiento;
    }
}
```

### Clase Facturación:

```
public class Facturacion {
    private ArrayList<Factura> facturas;

    public Facturacion() {
        facturas = new ArrayList<>();
    }
}
```



```

    public void agregarFactura(Factura factura) {
        facturas.add(factura);
    }

    public void mostrarFacturas() {
        for (Factura factura : facturas) {
            System.out.println(factura);
        }
    }
}

```

## Clase Factura:

```

public class Factura {
    private String numeroFactura;
    private Paciente paciente;
    private double monto;

    public Factura(String numeroFactura, Paciente paciente, double monto) {
        this.numeroFactura = numeroFactura;
        this.paciente = paciente;
        this.monto = monto;
    }

    public String getNumeroFactura() {
        return numeroFactura;
    }

    public void setNumeroFactura(String numeroFactura) {
        this.numeroFactura = numeroFactura;
    }

    public Paciente getPaciente() {
        return paciente;
    }

    public void setPaciente(Paciente paciente) {
        this.paciente = paciente;
    }

    public double getMonto() {
        return monto;
    }

    public void setMonto(double monto) {
        this.monto = monto;
    }

    @Override
    public String toString() {
        return "Factura No: " + numeroFactura + ", Paciente: " + paciente + ", Monto: $"
+ monto;
    }
}

```

### 3. El programa completo que compila y se ejecuta

Este sistema de gestión integral en Java incluye las clases base necesarias para gestionar citas, pacientes, médicos, inventarios, facturación y reportes. Cada módulo está encapsulado y tiene su propia responsabilidad, utilizando herencia y polimorfismo para facilitar la extensión y modificación del sistema en el futuro. Además, se asegura el manejo de excepciones para robustez y la interfaz

Se puede continuar agregando detalles específicos y funcionalidad adicional en cada método como `gestionarCitas`, `gestionarHistoriasClinicas`, etc.,

Output:

```
Bienvenido al Sistema de Gestión Integral para Clínicas de Salud
1. Gestión de Citas
2. Historias Clínicas Electrónicas
3. Gestión de Inventarios
4. Facturación y Pagos
5. Reportes de Gestión
6. Salir
Seleccione una opción:
```