

# FASE DE PROTOTIPADO

En esta fase se puede evidenciar una pequeña idea o estructura del cómo se haría la aplicación y como debe funcionar, esto cada vez mejorándose con la fase de implementación para poder dar una mejor experiencia a los usuarios y a su vez ofrecer comodidad.

PROGRAMA UTILIZADO: NetBeans LENGUAJE DE PROGRAMACION: java

## CLASES UTILIZADAS

### CLASE 1 DISPOSITIVOS:

```
public class Dispositivos {

    private int ID;
    private String tipo;
    private String marca;
    private String estado;
    private String protocoloComunicacion;
    private String fechaInstalacion;
    private String estadoReciclaje;
    private float nivelBateria;

    public Dispositivos(int ID, String tipo, String marca, String estado, String
    protocoloComunicacion, String fechaInstalacion, String estadoReciclaje, float ni
    this.ID = ID;
    this.tipo = tipo;
    this.marca = marca;
    this.estado = estado;

        this.protocoloComunicacion = protocoloComunicacion;
        this.fechaInstalacion = fechaInstalacion;
        this.estadoReciclaje = estadoReciclaje;
        this.nivelBateria = nivelBateria;
    }

    public void reciclarDispositivo() {
        this.estadoReciclaje = "Reciclado";
        System.out.println("El dispositivo ha sido reciclado.");
    }

    // Método para mostrar los detalles del dispositivo
    public void mostrarDispositivo() {
        System.out.println("----- Detalles del Dispositivo -----");
        System.out.println("ID: " + ID);
        System.out.println("Tipo: " + tipo);
        System.out.println("Marca: " + marca);
        System.out.println("Estado: " + estado);
        System.out.println("Protocolo de Comunicación: " + protocoloComunicacio
        System.out.println("Fecha de Instalación: " + fechaInstalacion);
        System.out.println("Estado de Reciclaje: " + estadoReciclaje);
        System.out.println("Nivel de Bateria: " + nivelBateria + "%");
        System.out.println("-----");
    }
}
```

## CLASE 2 HOGAR:

```
import java.util.ArrayList;

public class HOGAR {
    private int ID;
    private String direccion;
    private String propietario;
    private String tipoEdificio;
    private float areaTotal;
    private String fuenteEnergiaRenovable;
    private ArrayList<Dispositivos> dispositivos;

    // Constructor
    public HOGAR(int ID, String direccion, String propietario, String tipoEdificio, float areaTotal,
String fuenteEnergiaRenovable) {
        this.ID = ID;
        this.direccion = direccion;
        this.propietario = propietario;
        this.tipoEdificio = tipoEdificio;
        this.areaTotal = areaTotal;
        this.fuenteEnergiaRenovable = fuenteEnergiaRenovable;
        this.dispositivos = new ArrayList<>();
    }

    // Método para agregar un dispositivo al hogar
    public void agregarDispositivo(Dispositivos dispositivo) {
        dispositivos.add(dispositivo);
        System.out.println("Dispositivo agregado.");
    }

    // Método para eliminar un dispositivo del hogar
    public void eliminarDispositivo(Dispositivos dispositivo) {
        dispositivos.remove(dispositivo);
        System.out.println("Dispositivo eliminado.");
    }

    // Método para optimizar el consumo de energía
    public void optimizarConsumoEnergia() {
        System.out.println("Optimizando el consumo de energía en el hogar...");
    }

    // Método para mostrar la información del hogar
    public void mostrarHogar() {
        System.out.println("----- Información del Hogar -----");
        System.out.println("ID: " + ID);
        System.out.println("Dirección: " + direccion);
        System.out.println("Propietario: " + propietario);
        System.out.println("Tipo de Edificio: " + tipoEdificio);
        System.out.println("Área Total: " + areaTotal + " m²");
        System.out.println("Fuente de Energía Renovable: " + fuenteEnergiaRenovable);

        System.out.println("Dispositivos en el hogar:");
        if (dispositivos.isEmpty()) {
            System.out.println("No hay dispositivos registrados en este hogar.");
        } else {
            for (Dispositivos dispositivo : dispositivos) {
                dispositivo.mostrarDispositivo(); // Mostrar detalles de cada dispositivo
            }
        }
    }
}
```

## CLASE 3 USUARIO:

```
public class USUARIO {
    private int ID;
    private String nombre;
    private String email;
    private String telefono;

    // Constructor
    public USUARIO(int ID, String nombre, String email, String telefono) {
        this.ID = ID;
        this.nombre = nombre;
        this.email = email;
        this.telefono = telefono;
    }

    // Método para registrar usuario
    public void registrar() {
        System.out.println("Usuario registrado.");
    }

    // Método para iniciar sesión
    public void iniciarSesion() {
        System.out.println(nombre + " ha iniciado sesión.");
    }
}

...

public USUARIO(int ID, String nombre, String email, String telefono) {
    this.ID = ID;
    this.nombre = nombre;
    this.email = email;
    this.telefono = telefono;
}

// Método para registrar usuario
public void registrar() {
    System.out.println("Usuario registrado.");
}

// Método para iniciar sesión
public void iniciarSesion() {
    System.out.println(nombre + " ha iniciado sesión.");
}

// Método para mostrar la información del usuario
public void mostrarUsuario() {
    System.out.println("----- Información del Usuario -----");
    System.out.println("ID: " + ID);
    System.out.println("Nombre: " + nombre);
    System.out.println("Email: " + email);
    System.out.println("Teléfono: " + telefono);
}
```

## CLASE 4 RUTINA AUTOMATIZACION:

```
public class RUTINA AUTOMATIZACION {
    private int ID;
    private String nombre;
    private String condiciones;
    private String acciones;
    private String frecuencia;

    // Constructor
    public RUTINA AUTOMATIZACION(int ID, String nombre, String condiciones, String acciones,
String frecuencia) {
        this.ID = ID;
        this.nombre = nombre;
        this.condiciones = condiciones;
        this.acciones = acciones;
        this.frecuencia = frecuencia;
    }

    // Método para crear una rutina
    public void crearRutina() {
        System.out.println("Rutina " + nombre + " creada.");
    }

    // Método para ejecutar una rutina
    public void ejecutarRutina() {
        System.out.println("Ejecutando la rutina " + nombre + "...");
    }

    // Método para modificar una rutina
    public void modificarRutina(String nuevasCondiciones, String nuevasAcciones) {
        this.condiciones = nuevasCondiciones;
        this.acciones = nuevasAcciones;
        System.out.println("Rutina " + nombre + " modificada.");
    }

    // Método para eliminar una rutina
    public void eliminarRutina() {
        System.out.println("Rutina " + nombre + " eliminada.");
    }

    // Método para mostrar la información de la rutina
    public void mostrarRutina() {
        System.out.println("----- Información de la Rutina de Automatización -----");
        System.out.println("ID: " + ID);
        System.out.println("Nombre: " + nombre);
        System.out.println("Condiciones: " + condiciones);
        System.out.println("Acciones: " + acciones);
        System.out.println("Frecuencia: " + frecuencia);
    }
}
```

## CLASE 5 ECOSISTEMACOMPATIBILIDAD:

```
public class ECOSISTEMACOMPATIBILIDAD {
    private int ID;
    private String marca;
    private String modelo;
    private String protocolo;
    private String versionFirmware;

    public ECOSISTEMACOMPATIBILIDAD(int ID, String marca, String modelo, String protocolo, String
versionFirmware) {
        this.ID = ID;
        this.marca = marca;
        this.modelo = modelo;
        this.protocolo = protocolo;
        this.versionFirmware = versionFirmware;
    }

    public boolean verificarCompatibilidad() {
        System.out.println("Verificando compatibilidad del dispositivo...");
        return true;
    }
}
```

## CLASE MAIN PRUEBA APLICACIÓN:

```
import java.util.ArrayList;
import java.util.Scanner;

public class Main {
    private static ArrayList<Dispositivos> dispositivos = new ArrayList<>();
    private static ArrayList<HOGAR> hogares = new ArrayList<>();
    private static ArrayList<USUARIO> usuarios = new ArrayList<>();
    private static ArrayList<RUTINA AUTOMATIZACION> rutinas = new ArrayList<>();

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int opcion;

        do {
            System.out.println("----- Menú Principal -----");
            System.out.println("1. Registrar Dispositivo");
            System.out.println("2. Registrar Hogar");
            System.out.println("3. Registrar Usuario");
            System.out.println("4. Crear Rutina de Automatización");
            System.out.println("5. Verificar Compatibilidad de Dispositivo");
            System.out.println("6. Mostrar Dispositivos");
            System.out.println("7. Mostrar Hogares");
            System.out.println("8. Mostrar Usuarios");
            System.out.println("9. Mostrar Rutinas de Automatización");
            System.out.println("10. Salir");
            System.out.print("Seleccione una opción: ");
            opcion = scanner.nextInt();
            scanner.nextLine(); // Limpiar buffer
        } while (opcion != 10);
    }
}
```

```

private static void registrarHogar(Scanner scanner) {
    System.out.println("----- Registrar Hogar -----");
    System.out.print("Ingrese ID del hogar: ");
    int ID = scanner.nextInt();
    scanner.nextLine(); // Limpiar buffer
    System.out.print("Ingrese dirección del hogar: ");
    String direccion = scanner.nextLine();
    System.out.print("Ingrese nombre del propietario: ");
    String propietario = scanner.nextLine();
    System.out.print("Ingrese tipo de edificio: ");
    String tipoEdificio = scanner.nextLine();
    System.out.print("Ingrese área total del hogar: ");
    float areaTotal = scanner.nextFloat();
    scanner.nextLine(); // Limpiar buffer
    System.out.print("Ingrese fuente de energía renovable: ");
    String fuenteEnergiaRenovable = scanner.nextLine();

    HOGAR hogar = new HOGAR(ID, direccion, propietario, tipoEdificio, areaTotal,
fuenteEnergiaRenovable);
    hogares.add(hogar);
    System.out.println("Hogar registrado con éxito.");
}

```

----- Menú Principal -----

1. Registrar Dispositivo
  2. Registrar Hogar
  3. Registrar Usuario
  4. Crear Rutina de Automatización
  5. Verificar Compatibilidad de Dispositivo
  6. Mostrar Dispositivos
  7. Mostrar Hogares
  8. Mostrar Usuarios
  9. Mostrar Rutinas de Automatización
  10. Salir
- Seleccione una opción: