

# Fase 5: Sistema de Búsqueda y Gestión Avanzada de Asistentes

## 1. Descripción General

En esta fase se amplían las capacidades del sistema para incluir una funcionalidad robusta de búsqueda y validación avanzada de asistentes. Los administradores pueden buscar a los residentes utilizando múltiples criterios como nombre, apellido, torre, apartamento y cédula. Además, se asegura que los datos sean consistentes y se gestionen de manera eficiente.

## 2. Actualizaciones de Código y Cambios Principales

### Lectordata.java - Actualización

Se ha actualizado la clase lectorData con métodos mejorados de búsqueda que permiten realizar validaciones avanzadas y mensajes detallados en consola:

#### Cambios Principales:

- Implementación de mensajes informativos que confirman si un asistente fue encontrado o no.
- Optimización del método personaBuscar para manejar mejor los errores en el archivo.

#### Código Actualizado:

```
public void personaBuscar(String buscarNombre, String buscarApellido, String buscarTorre
String buscarCc) {
    try (BufferedReader lector = new BufferedReader(new FileReader(rutaDelArchivo))) {
        // Leer la cabecera
        String lineaCabecera = lector.readLine();
        System.out.println(lineaCabecera);

        String linea;
        boolean encontrado = false;

        while ((linea = lector.readLine()) != null) {
            String[] valores = linea.trim().split(";");

            if (valores.length >= 5) {
                if (valores[0].equalsIgnoreCase(buscarNombre) &&
                    valores[1].equalsIgnoreCase(buscarApellido) &&
                    valores[2].equalsIgnoreCase(buscarTorre) &&
                    valores[3].equalsIgnoreCase(buscarApartamento) &&
                    valores[4].equalsIgnoreCase(buscarCc)) {

                    System.out.println("Nombre: " + valores[0] +
                        ", Apellido: " + valores[1] +
                        ", Torre: " + valores[2] +
                        ", Apartamento: " + valores[3] +
                        ", Cédula: " + valores[4]);

                    encontrado = true;
                    System.out.println("¡Persona encontrada correctamente!");
                    break;
                }
            }
        }

        if (!encontrado) {
            System.out.println("No se encontró ninguna persona que coincida con los cri
        } catch (IOException e) {
            System.out.println("Error al leer el archivo: " + e.getMessage());
        }
    }
}
```

## main.java - Actualización

Se actualizó el menú principal para integrar las funcionalidades nuevas, añadiendo opciones de búsqueda detallada.

### Código Actualizado:

```
case 2:
    TextoCentrado("REGISTRO ASISTENTES DE LA ASAMBLEA", 1);
    TextoCentrado("Digite el nombre de la persona a buscar: ", 0);
    String nombreBuscado = sc.nextLine();
    TextoCentrado("Digite el apellido de la persona a buscar: ", 0);
    String apellidoBuscado = sc.nextLine();
    TextoCentrado("Digite el número de cédula de la persona a buscar: ", 0);
    String buscarCc = sc.nextLine();
    TextoCentrado("Digite la torre a buscar: ", 0);
    String buscarTorre = sc.nextLine();
    TextoCentrado("Digite el apartamento a buscar: ", 0);
    String buscarApartamento = sc.nextLine();

    lector.personaBuscar(nombreBuscado, apellidoBuscado, buscarTorre, buscarApartamento,
    break;
```

## logeoAdmin.java - Actualización

Aunque no se realizaron cambios significativos, se revisaron las validaciones para asegurarse de que las opciones de búsqueda sean accesibles solo a administradores autorizados.

## Aspectos Técnicos y Consideraciones

### Implementación de Seguridad

El sistema ahora incluye validaciones estrictas en las búsquedas para garantizar que los datos de los asistentes sean correctos y completos, solo los administradores autorizados pueden realizar búsquedas avanzadas y gestionar los registros, aunque los datos siguen siendo almacenados en

archivos CSV sin encriptación, se prioriza la exactitud mediante verificaciones de múltiples campos

### **Manejo de Entrada de Usuario**

Se mejoró el manejo de entradas para búsquedas detalladas de asistentes, verificando criterios como nombre, apellido, torre, apartamento y cédula, mensajes de retroalimentación claros en consola informan al usuario si se encuentra o no un asistente que coincida con los criterios de búsqueda, las entradas incompletas o inválidas no son procesadas, evitando errores o inconsistencias en el sistema.

### **Estructura de Control**

La jerarquía del menú se expandió para incluir nuevas opciones de búsqueda avanzada, manteniendo una interfaz intuitiva y organizada, se implementó un bucle de control para repetir búsquedas si el asistente no es encontrado, evitando que el usuario deba reiniciar el sistema, la estructura modular del sistema facilita futuras expansiones, como la edición y eliminación de registros.

### **Limitaciones Actuales**

- Almacenamiento de datos: Las búsquedas y registros dependen del archivo CSV cargado, lo que limita la escalabilidad y el rendimiento del sistema.
- Falta de edición: No se implementaron funcionalidades para modificar o eliminar registros de asistentes.
- Formato de salida: Los resultados de búsqueda se muestran exclusivamente en consola, sin opción de exportación a formatos externos (PDF, Excel).
- Persistencia de datos: No se implementó un sistema de respaldo automático o una base de datos relacional para optimizar el acceso y manejo de los datos.