

Universidad de San Carlos de Guatemala  
Centro Universitario de Occidente  
División de Ciencias de la Ingeniería  
Introducción a la programación y computación 1 Sección "A"



**ENRIQUE ALEXANDER TEBALAN HERNANDEZ**

Carné: 202230026

## MANUAL TECNICO PRACTICA #2 2024

# { BIENVENIDO }

---

### ## Manual Técnico

**CREADO:** JAVA NETBEANS IDE 20

### ### Introducción

El Manual Técnico proporciona información detallada sobre la estructura del proyecto, las clases y métodos implementados, así como su funcionalidad y uso. Está dirigido a desarrolladores y personas interesadas en comprender el funcionamiento interno del sistema.

### ### Estructura del Proyecto

El proyecto se divide en varios paquetes y clases que cumplen funciones específicas. A continuación, se describe la estructura básica del proyecto:

- **\*\*Paquete com.mycompany.practica\_2\_enrique\_tebalan\*\***
  - Clase `GestorArchivos`: Maneja la lectura y escritura en archivos de texto.
  - Clase `FrontEnd`: Interfaz gráfica de usuario (GUI) para interactuar con el usuario.
  - Otras clases: Clases adicionales según la lógica de negocio.



### ### Clase GestorArchivos

La clase `GestorArchivos` se encarga de manejar operaciones relacionadas con archivos de texto, como leer desde un archivo existente y escribir en un archivo nuevo.



#### #### Métodos Principales

1. **conectarConCarpeta()**
  - **Descripción:** Conecta con la carpeta especificada y muestra información sobre ella.
  - **Parámetros:** Ninguno.
  - **Tipo de Retorno:** `void`.
  
2. **escribirEnArchivoDeTexto(String contenido)**
  - **Descripción:** Escribe el contenido proporcionado en el archivo de texto especificado.
  - **Parámetros:**
    - `contenido` (tipo String): Contenido a escribir en el archivo.
  - **Tipo de Retorno:** `void`.
  
3. **leerDesdeArchivoTexto()**
  - **Descripción:** Lee el contenido del archivo de texto y lo muestra en un cuadro de diálogo.
  - **Parámetros:** Ninguno.
  - **Tipo de Retorno:** `void`.

#### #### Uso de la Clase GestorArchivos

Para utilizar la clase `GestorArchivos`, se debe crear una instancia de la misma y llamar a sus métodos según sea necesario. Por ejemplo:

```
```java
GestorArchivos gestor = new GestorArchivos();
gestor.conectarConCarpeta();
gestor.escribirEnArchivoDeTexto("Contenido a escribir en el archivo.");
gestor.leerDesdeArchivoTexto();
```
```

### ### Clase FrontEnd

La clase `FrontEnd` representa la interfaz gráfica de usuario (GUI) del sistema. Incluye componentes como botones, campos de texto, cuadros de diálogo, etc.

#### #### Funciones Principales

##### 1. `**`iniciarInterfaz()`**`

- **Descripción:** Inicia la interfaz gráfica y muestra los componentes necesarios.

- **Parámetros:** Ninguno.

- **Tipo de Retorno:** ``void``.

##### 2. `**`accionGuardarArchivoActionPerformed(ActionEvent evt)`**`

- **Descripción:** Maneja la acción de guardar un archivo al hacer clic en un botón.

- **Parámetros:**

- ``evt`` (tipo `ActionEvent`): Evento que desencadena la acción.

- **Tipo de Retorno:** ``void``.

#### #### Uso de la Clase FrontEnd

La clase `FrontEnd` se encarga de la interacción con el usuario. Se debe crear una instancia de esta clase para iniciar la interfaz y manejar las acciones del usuario.

```
``java
```

```
FrontEnd interfaz = new FrontEnd();
```

```
interfaz.iniciarInterfaz();
```

```
``
```

### ### Conclusiones

El Manual Técnico proporciona una visión general de la estructura del proyecto, las clases principales y sus funciones. Es una guía útil para desarrolladores y personas interesadas en comprender el funcionamiento interno del sistema.

---