

## **Proyecto 1 entrega 3**

**No le sé a C**

**Marlon Londoño 202317352**

**Joban Mejia 202213845**

**German Jimenez 201817214**

**Nicolas Sua 202310949**

Para complementar este trabajo, se construyeron dos paquetes nuevos que son fundamentales para el funcionamiento del sistema. Uno de estos paquetes es el de "listeners", el cual está diseñado para gestionar las opciones en cada uno de los paneles nuevos que se han integrado. Estos paneles se encuentran dentro del paquete "interfaz".

El paquete de listeners contiene aproximadamente 18 clases distintas, cada una de las cuales desempeña un papel crucial en la administración y flujo de información según los diferentes tipos de usuarios. A continuación, se detalla cómo estas clases contribuyen al sistema:

1. **Modularidad y Organización:** La creación de un paquete específico para los listeners permite una mejor organización del código. Cada listener se encarga de una parte específica de la interfaz de usuario, lo que facilita la gestión y mantenimiento del código.
2. **Escalabilidad:** Al tener los listeners distribuidos en 18 clases, es más fácil añadir nuevas funcionalidades o modificar las existentes sin afectar significativamente otras partes del sistema. Cada listener puede ser actualizado de manera independiente.
3. **Flujo de Información:** Los listeners actúan como intermediarios entre los paneles de la interfaz y la lógica del sistema. Capturan las acciones de los usuarios (como clics en botones, selecciones en menús desplegables, etc.) y las traducen en operaciones específicas. Esto garantiza que las interacciones del usuario se manejen de manera eficiente y coherente.

4. Personalización según el Usuario: Dado que diferentes usuarios pueden tener diferentes roles y permisos, los listeners están diseñados para gestionar estas variaciones. Por ejemplo, un listener puede habilitar o deshabilitar ciertas opciones en la interfaz dependiendo del tipo de usuario que esté interactuando con el sistema.

5. Mantenibilidad: La separación de los listeners en un paquete distinto mejora la mantenibilidad del código. Los desarrolladores pueden localizar rápidamente las clases responsables de las interacciones de la interfaz, facilitando la identificación y corrección de errores, así como la implementación de mejoras.

6. Reutilización de Código: Al tener listeners específicos para diferentes paneles y funcionalidades, se promueve la reutilización de código. Si varias partes de la interfaz comparten comportamientos similares, se pueden reutilizar listeners **existentes en lugar de crear nuevos desde cero**.

En resumen, la incorporación del paquete de listeners con 18 clases en el paquete "interfaz" no solo mejora la organización y claridad del código, sino que también garantiza un flujo de información eficiente y adaptado a las necesidades de diferentes usuarios. Esta estructura modular y escalable facilita la futura expansión y mantenimiento del sistema, asegurando una experiencia de usuario óptima y consistente.



