*15 de Noviembre 2024*

*Dr. Amadeo Sabattini IPETyM n°246*

*Sistemas de Información: Marcelo Fedullo*

*Antonella Cattáneo Montes 5°C*

***Desarrollo de una Calculadora con Interfaz Gráfica en Java***

La creación de la interfaz gráfica de usuario (GUI) para una calculadora es una tarea que me permitió explorar las diferentes herramientas y lenguajes que facilitan la interacción entre el usuario y el software. En este proyecto, use Java Swing, una biblioteca ampliamente utilizada en el desarrollo de interfaces gráficas por su versatilidad y compatibilidad multiplataforma. El objetivo principal era diseñar una calculadora funcional, fácil de usar y compatible con todas las versiones de Windows.

Este proyecto surge como una evolución de los conocimientos adquiridos en herramientas como Raptor, Python y Pseint, donde pude crear calculadoras básicas, superar esas versiones simples y crear una calculadora con una interfaz gráfica fue un gran objetivo.

Este informe documenta el proceso de construcción de la interfaz, las pruebas realizadas con diferentes herramientas y las referencias consultadas para superar los obstáculos encontrados durante el desarrollo, que por cierto, fueron múltiples.

### **Prueba y Error con Distintas Herramientas**

Durante el desarrollo, probé varias herramientas para la construcción de la interfaz gráfica. A continuación, se describen las más relevantes:

#### **NetBeans Editor de Interfaces**

* **Ventajas**: NetBeans incluye un editor visual que permite diseñar interfaces gráficas de manera rápida, arrastrando y soltando los componentes. Esto redujo el tiempo de desarrollo y minimizó los errores.
* **Desventajas**: El código generado automáticamente por el editor visual es difícil de modificar manualmente.
* **Resultado**: Utilicé NetBeans como entorno de desarrollo principal, especialmente para organizar los componentes de la calculadora, aprovechando su editor visual para acelerar el proceso, en vez de hacerlo desde el block de notas.

**Más Ecuaciones a la Calculadora**

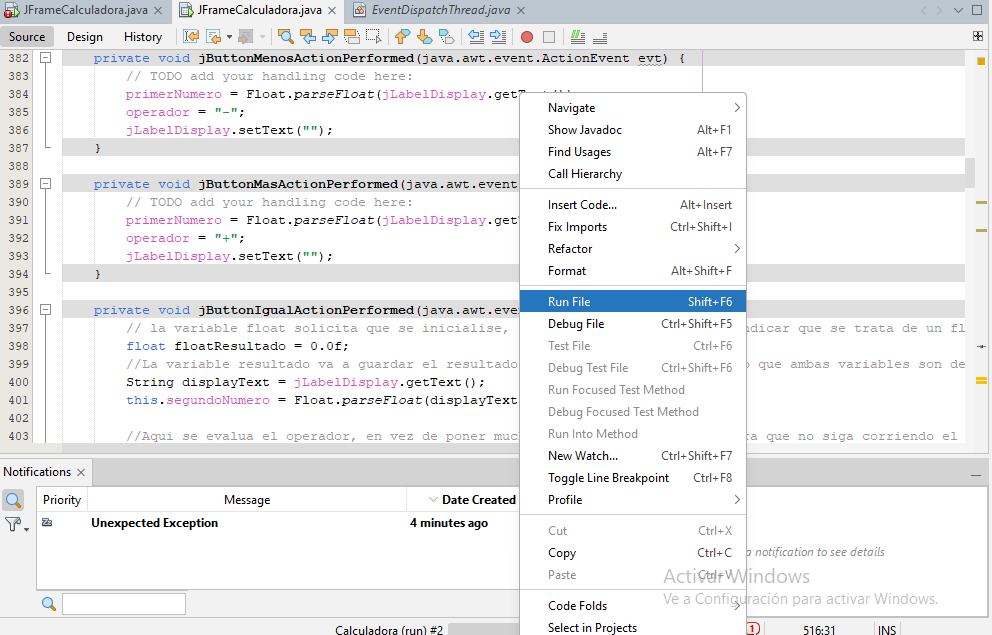
* **Ventajas:** Al agregarle más ecuaciones a mi calculadora podría tener mayor funcionalidad, llegar a un público más amplio y sería más requerida, sin mencionar que podría obtener mayor puntuación en la evaluación de mi trabajo.
* **Desventajas:** Como todo avance o mejora tenía cierto nivel de complejidad que no estaba dispuesta a explorar dadas las complicaciones que sufrí en el desarrollo de la misma, agregar estas ecuaciones significa modificar gran parte del código que anteriormente ya se había echado a perder.
* **Resultado:** Tomé la decisión de dejar la calculadora lo más simple posible ya que muchas veces lo mejor es ir a lo seguro; en algún otro momento estoy dispuesta a seguir explorando las posibilidades que ofrece el desarrollo activo de un proyecto.

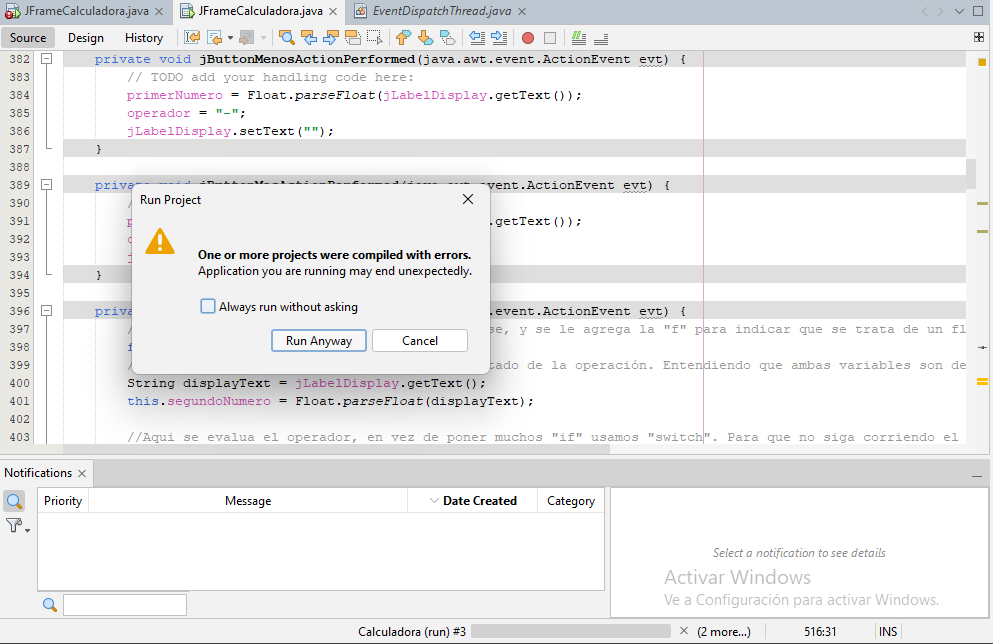
**Uso del Teclado para Realizar las Ecuaciones**

* **Ventajas:** Lograr que mi calculadora tuviera interacción con el teclado era una mejora significativa dada la simplicidad de la misma, no dependería solo del mouse o el puntero y sería un buen distintivo.
* **Desventajas:** La aplicación de esta función se vio bloqueada, para llevarlo a cabo debería haber planteado toda la aplicación en un jPanel en vez de un jFrame, si quisiera modificarlo tendría que plantear todo el código de nuevo, o así de drástico fué planteado por un compañero de 7mo año.
* **Resultado:** Eventualmente logré que la calculadora entendiera cuando yo tocaba un caracter numérico en el teclado, pero solo funcionaba luego de que se tocara al menos una vez en número con el puntero y no permitía que lo haga con distintos números; finalmente decidí dejarla como estaba por el poco tiempo que tenía y la frustración que sentí en su proceso.

**Dificultad para Correr el Programa**

Salta un aviso al intentar correr el programa, sin embargo funciona con normalidad al correrlo de todos modos.



****

**Manual Técnico**

El siguiente manual fue diseñado a fin de guiar a quien retome/mejore este proyecto y como guía para quien tenga la misma tarea.

**Búsqueda de Antecedentes:** Antes de escribir una sola línea de código, se debe investigar sobre proyectos similares y estudiar los fundamentos necesarios para el desarrollo, por ejemplo:

1. **Uso de Java para interfaces gráficas**:  
   Estudiar bibliotecas como Java Swing o JavaFX para entender cómo usar correctamente componentes gráficos como botones, cuadros de texto y etiquetas.
2. **Exploración de NetBeans**:  
   Familiarizarse con el entorno de desarrollo integrado (IDE) para crear interfaces gráficas. Al usar su editor visual ten en cuenta la vista y uso del usuario para la disposición de elementos.
3. **Implementación de eventos**:  
   Investigar cómo capturar acciones del usuario como los clics o pulsaciones de teclado
4. **Proyectos de referencia**:  
   Revisar calculadoras simples hechas en otros lenguajes (Python, Raptor, etc.) para inspirarse en cómo estructurar el proyecto.

**Pasos para el Desarrollo:**

1. **Inicio**:
   * Instalar JDK, NetBeans o cualquier otro IDE compatible con Java.
   * Crear un proyecto nuevo en NetBeans, seleccionar el uso de bibliotecas Swing y asegurarse de que el entorno este correctamente configurado
2. **Diseño de la interfaz gráfica (GUI)**:
   * Usar el editor visual de NetBeans para colocar los componentes básicos: botones, cuadros de texto y lo demas necesario
   * Organizar los elementos para que la interfaz sea clara y fácil de usar.
   * Diseñar una disposición clara y ordenada para los botones numéricos (0-9) y las operaciones (+, -, \*, /, etc.).
3. **Programación de eventos**:
   * Asignar funciones a los botones para realizar operaciones matemáticas.
   * Implementar validaciones básicas para evitar errores, como dividir entre cero o ingresar caracteres no válidos.
   * Opcional: Añadir soporte para el teclado.
4. **Prueba y Error**:
   * Ejecutar el programa en busca de errores lógicos y de interfaz.
   * Probar con diferentes entradas (números grandes, valores negativos, etc.) para garantizar que la calculadora sea lo más funcional posible.
5. **Documentación del código**:
   * Comentar cada sección del código explicando su propósito y funcionamiento.
   * Llevar un registro de cambios y problemas encontrados para facilitar la revisión futura.
6. **Creación del ejecutable**:
   * Exportar el proyecto como un archivo .jar o .exe.
   * Asegurarse de incluir lo necesario para que funcione en otros equipos.
7. **Opciones de mejora**:
   * Añadir funcionalidades avanzadas, como un historial de operaciones o soporte para ecuaciones complejas.
   * Mejorar la estética de la interfaz para que sea más moderna y atractiva para los usuarios.

**Importante:**

1. **Limitaciones de los entornos visuales**: Aunque NetBeans facilita el diseño de la GUI, el código generado automáticamente puede ser difícil de modificar manualmente.
2. **Interacción con el teclado:** Los eventos como keypressed requieren un manejo cuidadoso para evitar conflictos con otros controles.
3. **Errores al exportar:** Es necesario verificar la compatibilidad de la versión de Java utilizada con el sistema operativo de destino.

**Manual de Usuario**

**¿Qué hace esta calculadora?**

Es una aplicación que te permite sumar, restar, multiplicar y dividir números fácilmente.

#### **Pasos para usar la calculadora:**

1. **Abre la calculadora**: Haz doble clic en el archivo de la calculadora en tu computadora.
2. **Escribe los números**: Haz clic en los botones de los números que aparecen en la pantalla.
3. **Elige una operación**: Haz clic en el botón de la operación que quieras usar:

* “**+**” para sumar.
* “**-**” para restar.
* “**×**” para multiplicar.
* “**÷**” para dividir.

1. **Corroboración**: Después de elegir la operación y escribir los números, haz clic en el botón "=" para ver el resultado.
2. **Borra**: Si te equivocaste, haz clic en el botón "C" para empezar de nuevo.

### **Bibliografía de Referencia**

[Aprende a Programar tu Primera Calculadora Básica Paso a Paso | Java | NetBeans](https://www.youtube.com/watch?v=b7rC0BENpQw)

Fue esencial para entender los conceptos clave de Swing y cómo usar los componentes de la interfaz gráfica. Este tutorial ofreció ejemplos prácticos para trabajar con botones, cuadros de texto y otros elementos. Me ayudó a familiarizarme con el editor visual y a gestionar eventos dentro del entorno de desarrollo.

[Eventos de tipo Key en NetBeans "Keypressed, Keyreleased" - YouTube](https://www.youtube.com/watch?v=OnfIkdSsDRs)

[Como implementar el Evento KeyPressed Java](https://www.youtube.com/watch?v=DB47isOZJok)

En mi intento de hacer que mi calculadora tuviera interacción con el teclado realicé múltiples búsquedas que no lograron darme una solución práctica, finalmente no cumplieron con el requerimiento y fueron descartados.

### **Conclusión Personal sobre el Proyecto**

* **¿Qué aprendí al realizar este proyecto?**Aprendí a utilizar herramientas como Java Swing para crear interfaces gráficas funcionales, así como a resolver problemas que surgieron durante el desarrollo. También comprendí mejor cómo organizar y gestionar la interfaz de usuario para que sea intuitiva.
* **¿Qué mejoras implementaría si tuviera más tiempo?**Si tuviera más tiempo, implementaría una funcionalidad de historial de operaciones, para que el usuario pueda ver las operaciones anteriores y una barra que le vaya mostrando las ecuaciones que va planteando para no equivocarse, agregaría más funciones para tener una calculadora más completa. También mejoraría el diseño de la interfaz para lograr una apariencia más moderna. Me hubiera gustado terminar lo que empecé, que fue un inicio de código para que mi calculadora pudiera leer las teclas que se presionaban en el teclado y que así funcione también con el teclado de una computadora. Por otro lado me hubiera gustado hacer un desarrollo más profundo del código y sus modificaciones ya que lo que hice fue comentar los cambios en vez de llevar un registro por este medio.
* **Comentarios sobre la experiencia de desarrollar el ejecutable**Crear el ejecutable fue una experiencia desgastante. Utilicé herramientas múltiples que no fueron suficientes para generar el instalador. Luego me di cuenta que el motivo por el que no funciona es porque es una versión no compatible por lo avanzado que era NetBeans.