

# Documentación Técnica

## Herramientas

JFlex y CUP son herramientas de generación de analizadores léxicos y sintácticos para el lenguaje de programación Java. JFlex se utiliza para generar analizadores léxicos a partir de especificaciones de expresiones regulares, mientras que CUP se utiliza para generar analizadores sintácticos a partir de especificaciones de gramáticas.

Un **analizador léxico** se encarga de leer el código fuente del programa y dividirlo en tokens o símbolos, que son las unidades básicas de lenguaje. Los analizadores léxicos generados con JFlex utilizan expresiones regulares para definir los patrones de los tokens. Por ejemplo, una expresión regular que define un identificador de variable en Java podría ser `[a-zA-Z][a-zA-Z0-9]*`.

Un **analizador sintáctico** se encarga de analizar la estructura del programa a nivel de sintaxis, es decir, cómo se combinan los tokens para formar las sentencias y expresiones del lenguaje. Los analizadores sintácticos generados con CUP utilizan una gramática para definir las reglas sintácticas del lenguaje.

Un **analizador semántico** se encarga de analizar el significado del programa a nivel semántico, es decir, cómo se relacionan los tokens y las estructuras sintácticas con el comportamiento del programa. Los analizadores semánticos suelen implementarse manualmente y utilizan información adicional, como la tabla de símbolos y las reglas de tipado, para realizar el análisis semántico.

En resumen, JFlex y CUP son herramientas útiles para generar analizadores léxicos y sintácticos para un lenguaje de programación como Java. Los analizadores semánticos, por otro lado, suelen implementarse manualmente y requieren información adicional sobre el programa para realizar el análisis semántico.

## Lenguajes de programación a utilizar

**Java** es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems (ahora Oracle) en la década de 1990. Se trata de un lenguaje de propósito general que se utiliza en una gran variedad de aplicaciones, desde aplicaciones de escritorio y móviles hasta servidores y sistemas embebidos.

**Java Swing** es una biblioteca de interfaz gráfica de usuario (GUI) para Java. Fue desarrollada por Sun Microsystems y proporciona un conjunto de componentes y herramientas para construir interfaces de usuario en aplicaciones de escritorio. Swing es una biblioteca de clase

de nivel superior en la plataforma Java, y ofrece una amplia gama de componentes gráficos, incluyendo botones, cajas de texto, cuadros de diálogo, tablas, árboles, menús y paneles.

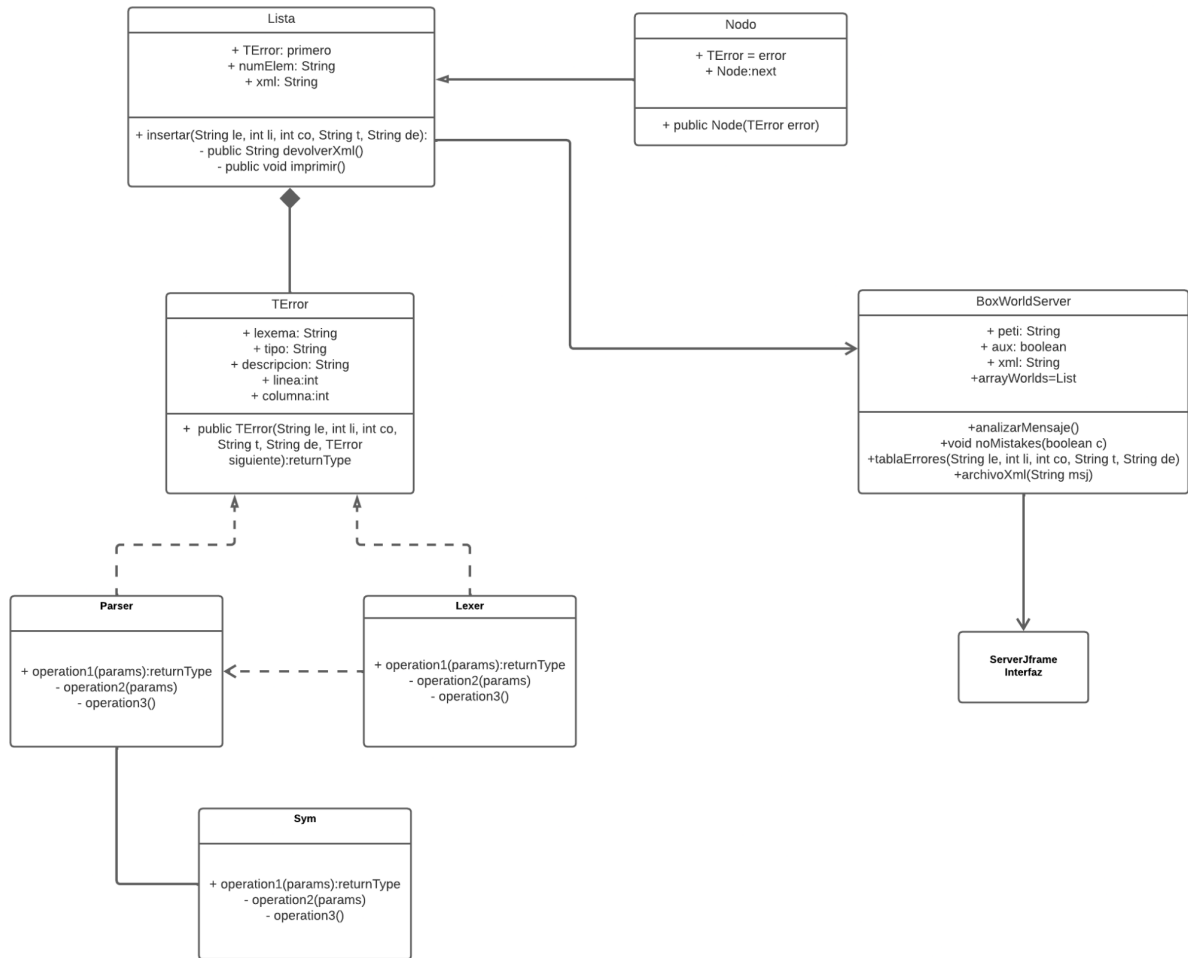
Para construir aplicaciones con Swing, los desarrolladores de Java utilizan el lenguaje de programación Java en conjunto con la biblioteca Swing. En general, las aplicaciones de Swing se construyen utilizando el patrón de diseño Modelo-Vista-Controlador (MVC), donde el modelo representa los datos de la aplicación, la vista representa la interfaz de usuario y el controlador maneja las interacciones entre los datos y la interfaz de usuario.

**Kotlin** es un lenguaje de programación de código abierto desarrollado por JetBrains, la empresa detrás de IntelliJ IDEA, un entorno de desarrollo integrado (IDE) para Java. Kotlin fue diseñado para ser interoperable con Java, lo que significa que los desarrolladores pueden utilizar Kotlin para desarrollar aplicaciones que se ejecutan en la máquina virtual de Java (JVM) y que pueden acceder a las bibliotecas y marcos de Java existentes.

**Android Studio** es un IDE oficial de desarrollo para el sistema operativo móvil Android, basado en IntelliJ IDEA. Android Studio está diseñado para proporcionar un entorno de desarrollo completo para construir aplicaciones móviles de alta calidad para Android. Android Studio es compatible con varios lenguajes de programación, incluyendo Java y Kotlin.

# DIAGRAMAS

## JAVA



## Kotlin

