DOCUMENTACIÓN MINIMAL REACT CMS

A continuación se presenta la documentación del proyecto de Minimal React CMS, se presenta la estructura a través de un diagrama de clases, y explicación de cada clase que compone el proyecto dividido en backend y frontend.

Tecnologías utilizadas

Se utilizó las siguientes tecnologías para el desarrollo de la aplicación.

* JDK 22 tanto para el backend y el frontend.
* Jflex para la creación del analizador léxico, jar utilizado: jflex-full-1.9.1.jar.
* Cup para la creación del analizador sintáctico por medio de gramáticas, jar utilizado: java-cup.11b.jar.
* Antlr4 para analizador tanto léxico como sintáctico versión 4.13.2.
* WebSockets para el servidor backend.
* JavaFx para el frontend.
* Intellij IDEA como IDE para la creación de la aplicación backend y frontend.
* PlantUML para crear el diagrama de clases.

# Diagrama de clases

El siguiente diagrama de clases contiene se divide en dos partes, una muestra cómo se estructura el servidor backend y la otra muestra cómo se estructura el frontend.

## Diagrama de clases del servidor (backend)

## Diagrama de clases de la aplicación de usuario (frontend)

# Descripción de clases de cada diagrama

A continuación se describe el funcionamiento de cada clase que compone el backend y el fronted.

## Backend

El backend es un servidor el cuál se comunica por medio de WebSockets, se compone de lo siguiente:

### Paquete antlr4

Este paquete contiene la carpeta que contiene la carpeta analizadores y este la carpeta antlr4, dentro de esta carpeta está el archivo Analizador.g4, esta estructura de carpetas coincide con la misma de la carpeta java, esto es necesario para que se pueda utilizar el analizador léxico y sintáctico de antlr4.

### Paquete cup

Este paquete contiene el archivo parser.cup que es el encargado de generar las gramáticas que deben cumplir los archivos de texto.

### Paquete java

Este paquete contiene las clases java y paquetes que permitirán el funcionamiento del servidor WebSocket.

#### Paquete analizadores

Este paquete contiene otro paquete donde se encuentra el paquete antlr4 que contiene las clases ErrorLexicoAntlr y ErrorSintacticoAntlr.

##### Clase ErrorLexicoAntlr

Esta clase se encarga de añadir todos los errores léxicos que puedan existir dentro del análisis.

##### Clase ErrorSintacticoAntlr

Esta clase se encarga de añadir todos los errores sintácticos que puedan existir dentro del análisis.

#### Clase ErrorLexico

Esta clase sirve únicamente para tener el molde de un error léxico.

#### Clase ErrorSintactico

Esta clase sirve únicamente para tener el molde de un error sintáctico.

#### Clase Token

Esta clase sirve para ser el modelo de un token que contiene número de línea, columna, tipo y lexema.

## Frontend