Escuela de Ciencias y Sistemas

Organización de Lenguajes y Compiladores 1

Proyecto 2

Manual Técnico

César André Ramírez Dávila

202010816

Fecha: 04/11/2022

Índice

[Introducción 3](#_Toc118494268)

[Objetivos 3](#_Toc118494269)

[Requerimientos 4](#_Toc118494270)

[Creación del programa 5](#_Toc118494271)

[1. Interfaz Gráfica 5](#_Toc118494272)

[2. Carga de Archivo 5](#_Toc118494273)

[3. Token de Errores 6](#_Toc118494274)

[4. Analizador Lexico 6](#_Toc118494275)

[5. Analizador Sintáctico 8](#_Toc118494276)

[6. Clases abstractas 10](#_Toc118494277)

[7. Gráfico del Árbol de sintaxis abstracta 12](#_Toc118494278)

[8. Servidor 13](#_Toc118494279)

[9. Frontend 15](#_Toc118494280)

# Introducción

El presente documento describe los aspectos técnicos informáticos del programa. El documento familiariza al desarrollador que hace uso de node.js, con ayuda de las herramientas Jison y typescript con temas como: uso de Clases, Gramática, Analizador léxico, Analizador Sintáctico, cargas de archivos, servidor rest.

# Objetivos

* Instruir el uso adecuado del Sistema de información, para el acceso adecuado en el uso de este, mostrando los pasos de desarrollo del programa, así como la descripción de las funciones y métodos usados para la realizacion del programa.
* Comprender uso de las herramientas Jison y typescript.
* Obtener mayor conocimiento en node.js, angular y typescript.

El presente manual está enfocado para el manejo de un cliente-servidor.

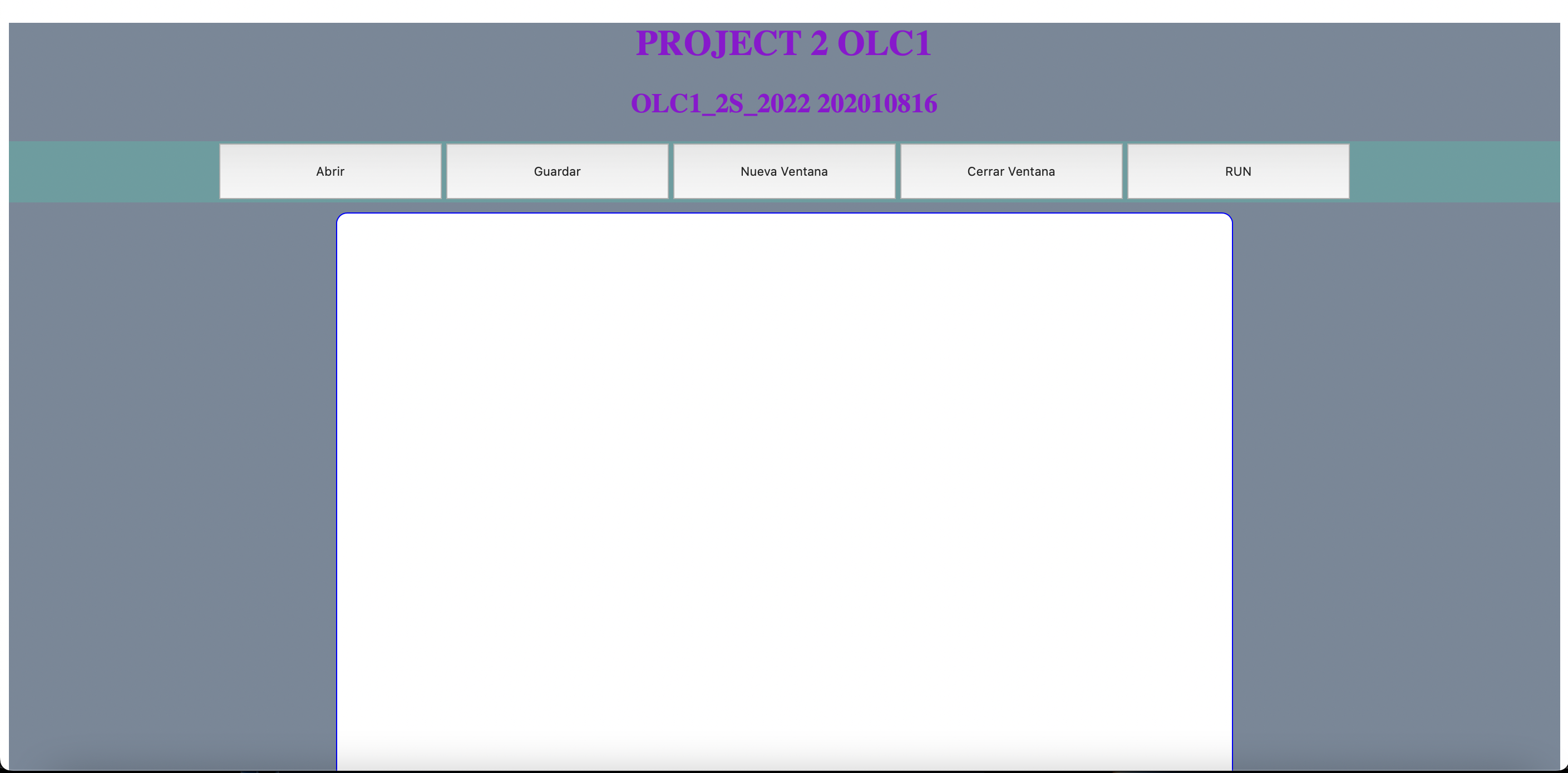
# **Requerimientos**

* La aplicación puede ser ejecutada en cualquier sistema operativo que tenga instalado node.js y jison en el sistema.
* IDE recomendado: Visual studio code
* Equipo Intel Pentium o superior
* Espacio en el disco duro, al menos 800 mb
* Memoria ram recomendada 4gb (por la conexión de frontend y backend)

# **Creación del programa**

1. **Interfaz Gráfica**

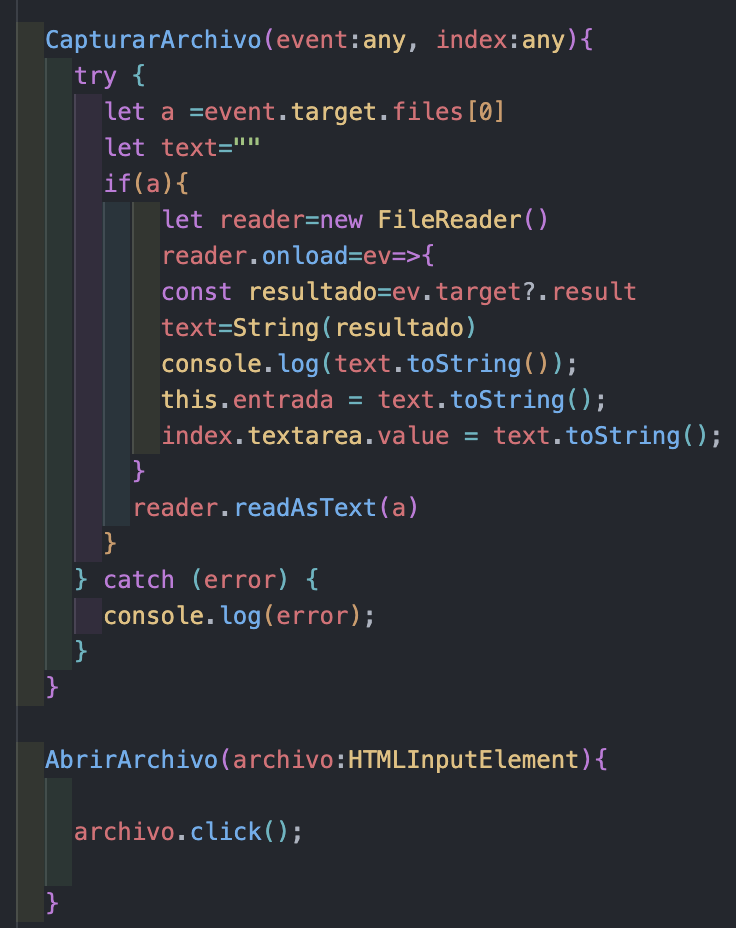
La primera parte del programa es tener una interfaz grafica donde se procederá a la ejecución de todo el programa.



Con la herramienta de Angular podemos hacer plantillas de html para el frontend

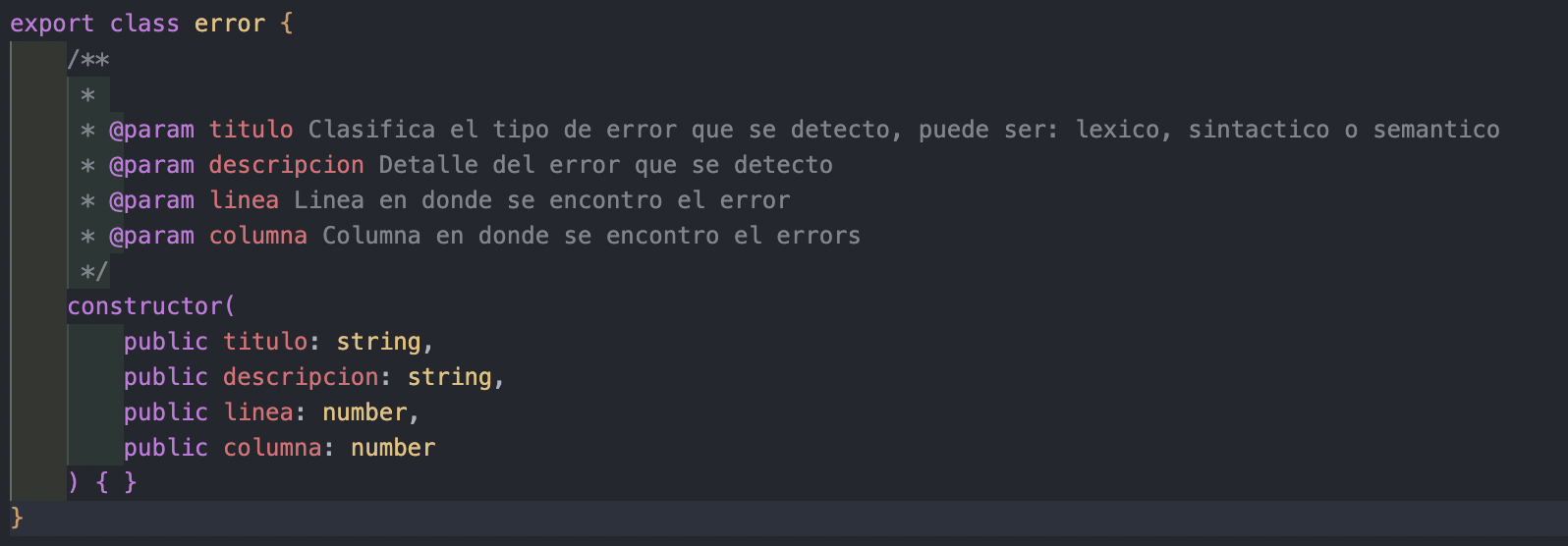
1. **Carga de Archivo**

Creamos la función que nos permite hacer la carga de un archivo con extension especifica .olc y poner su contenido en la caja de texto para su edicion posteriormente.

****

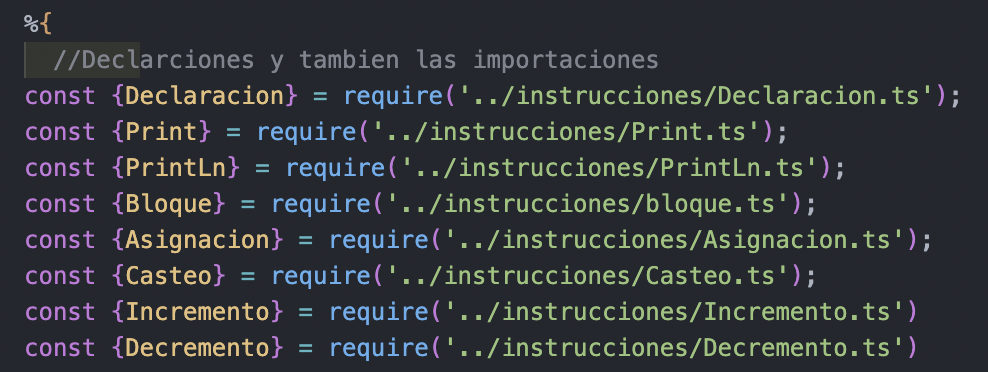
1. **Token de Errores**

Obtenemos su tipo, lexema, descripcion, linea y columna

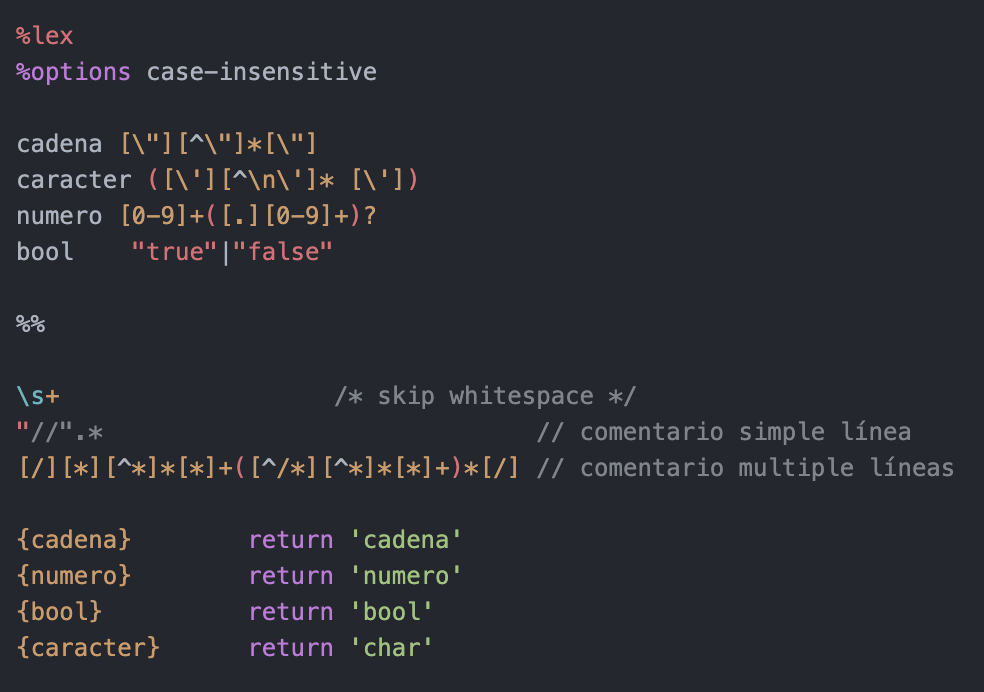
****

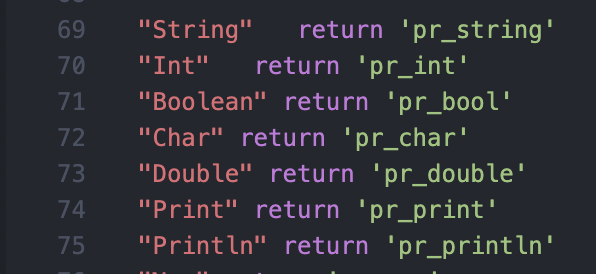
1. **Analizador Lexico**

El siguiente paso para el programa es la creación del Analizador léxico con la ayuda de Jison

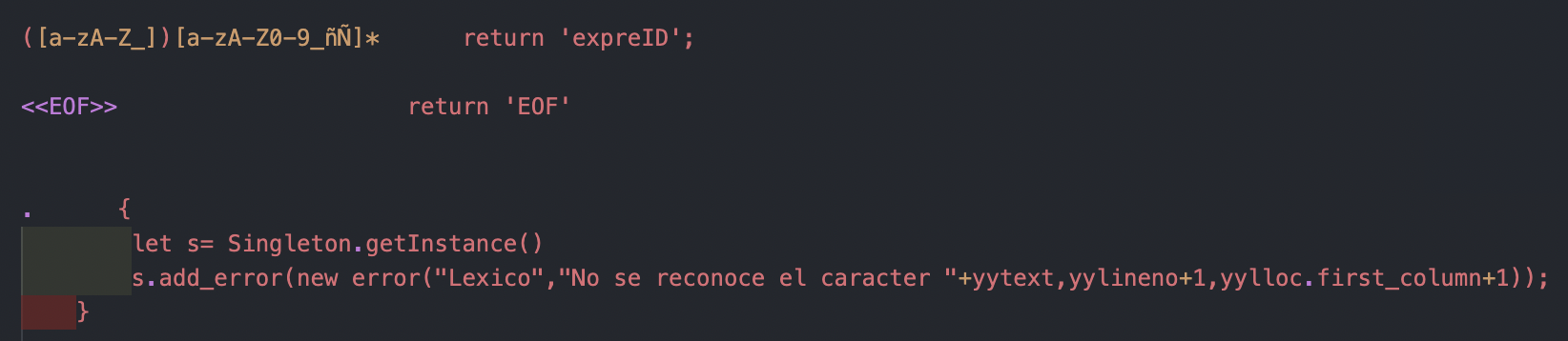


Declaraciones e importaciones que usaremos





Terminales para el programa.

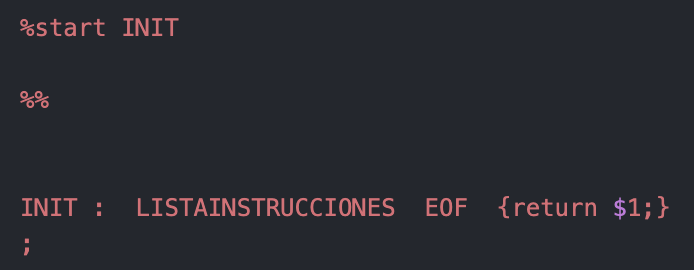


Almacenando errores en una lista.

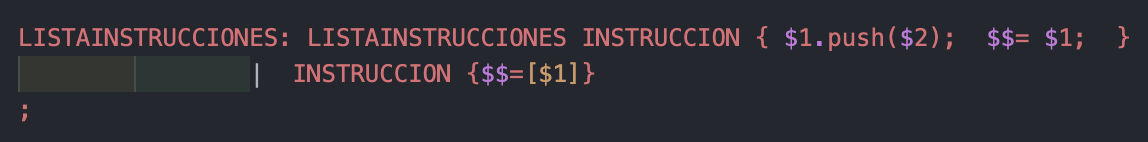
1. **Analizador Sintáctico**

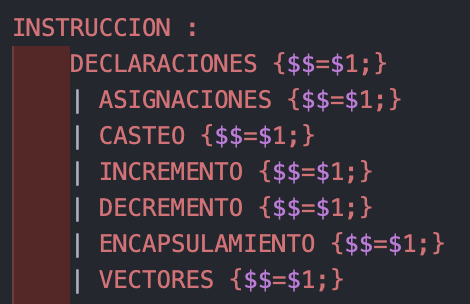


Precedencia de operadores

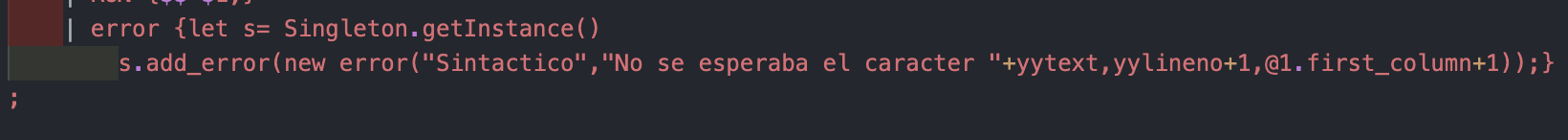


Iniciando Gramática

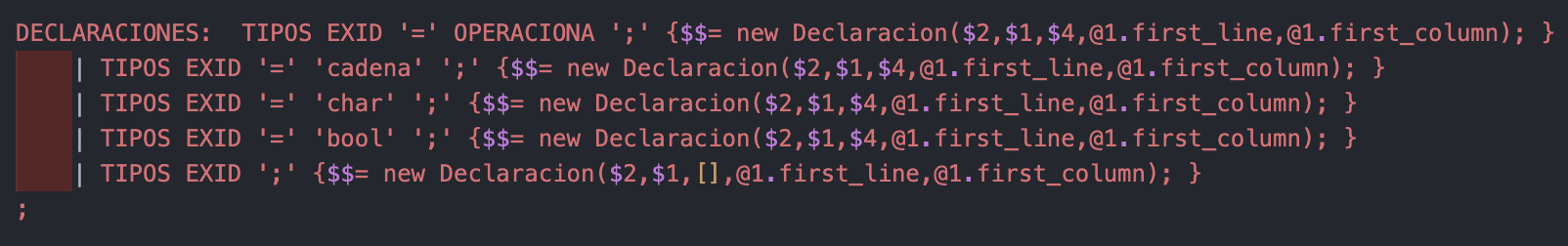


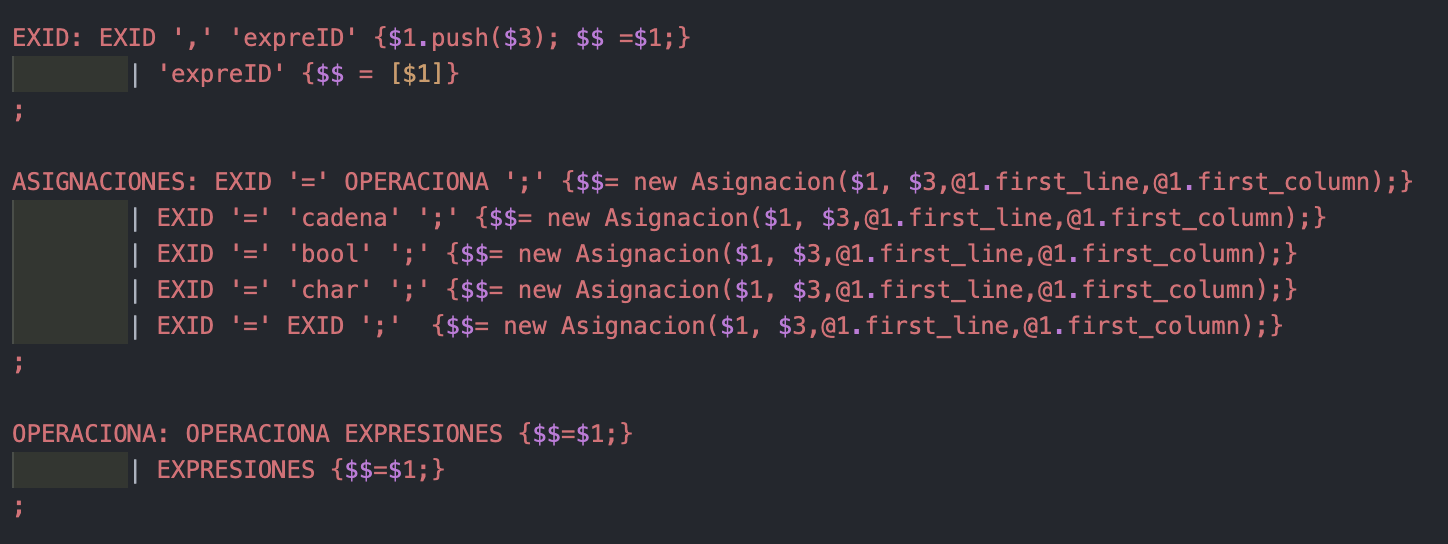


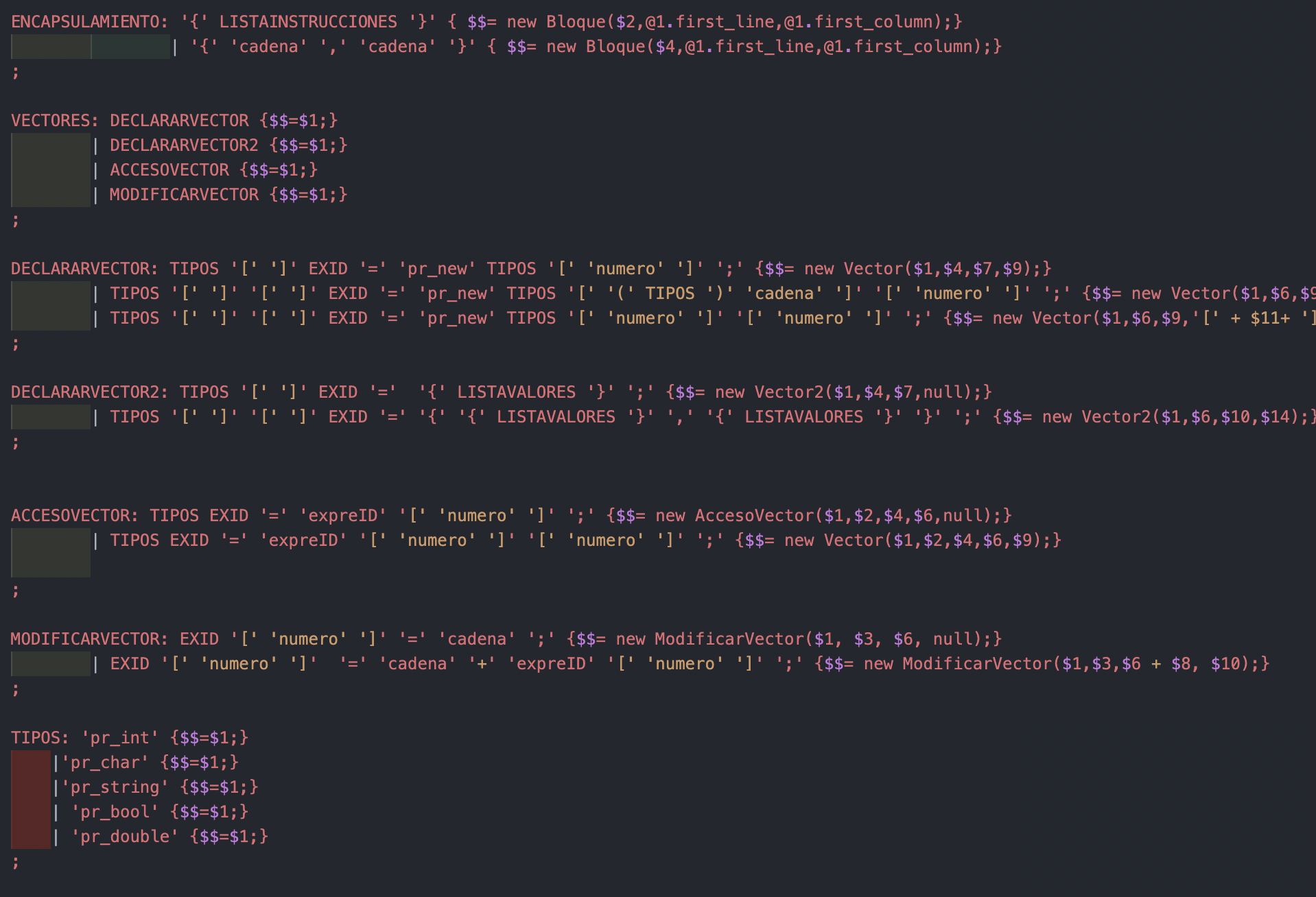
Instrucciones de la gramática.



Recuperando y almacenando errores sintácticos.

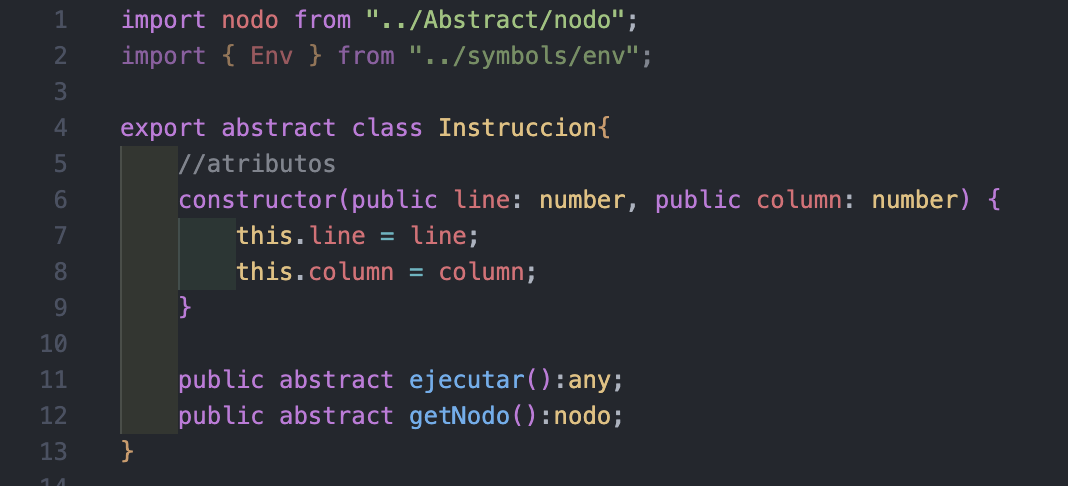






Continuando con todas las instrucciones sucesivamente

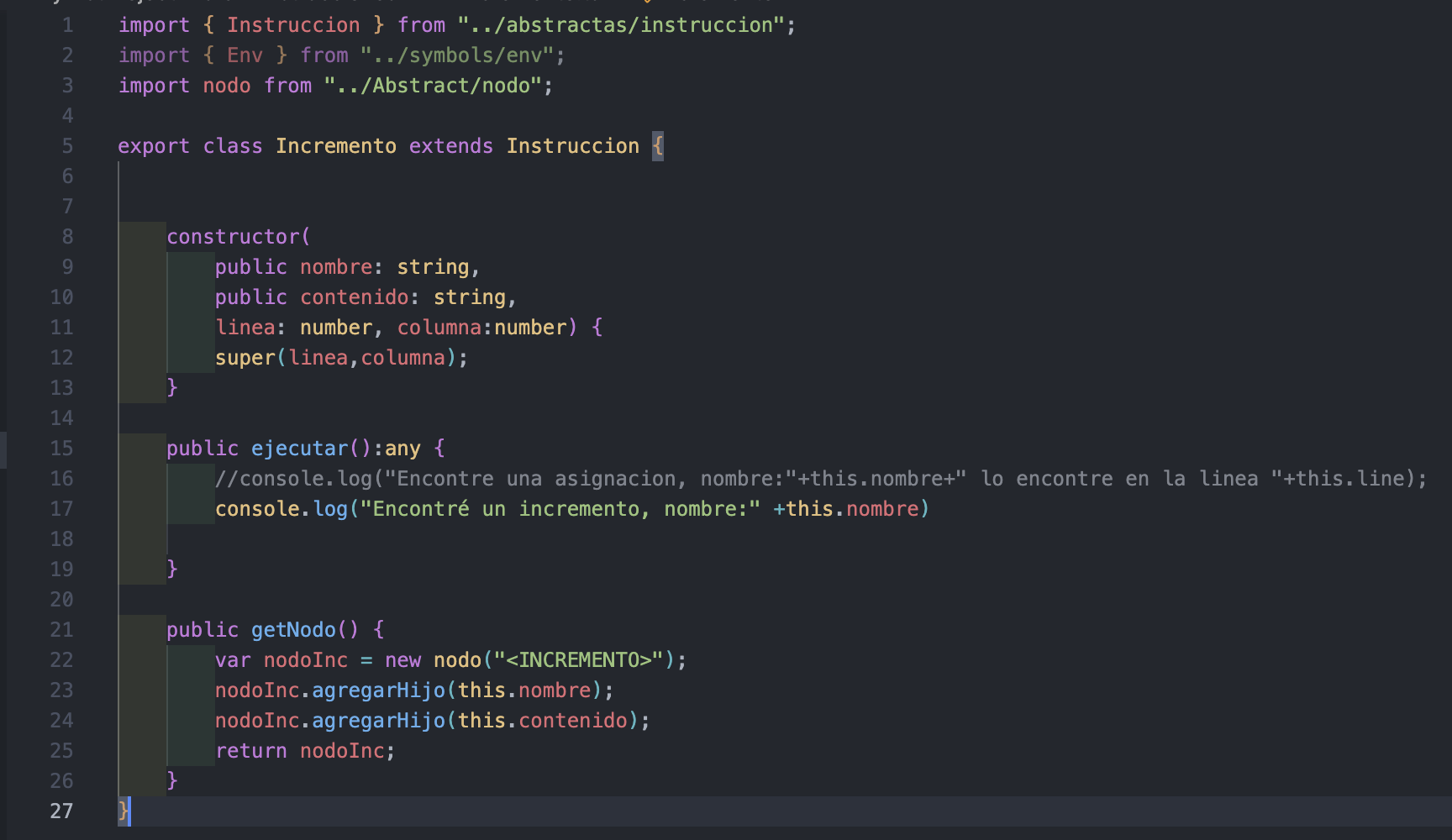
1. **Clases abstractas**



Clase para las instrucciones



Creando las instrucciones que recibe la clase del JISON.

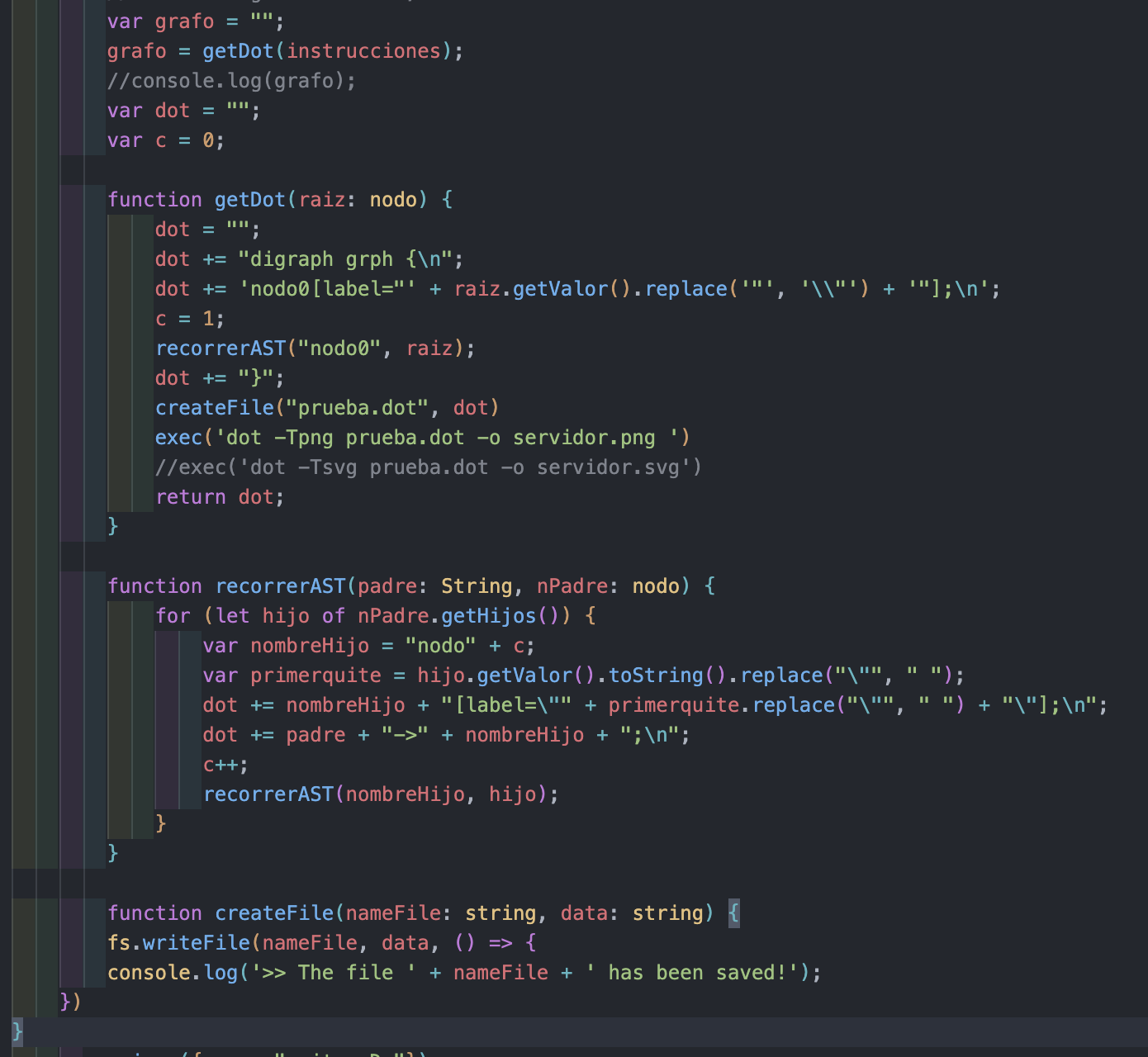


Sucesivamente con cada instrucción que nuestro programa acepta.

En INSTRUCCIÓN tenemos los siguientes no terminales

* Declaraciones
* Asignaciones
* Casteo
* Incremento
* Decremento
* Ciclo IF
* Ciclo Switch
* Ciclo For
* Ciclo While
* Ciclo Do While
* Retorno
* Metodo
* Funciones
* Ejecutar
* Print
* Println

1. **Gráfico del Árbol de sintaxis abstracta**

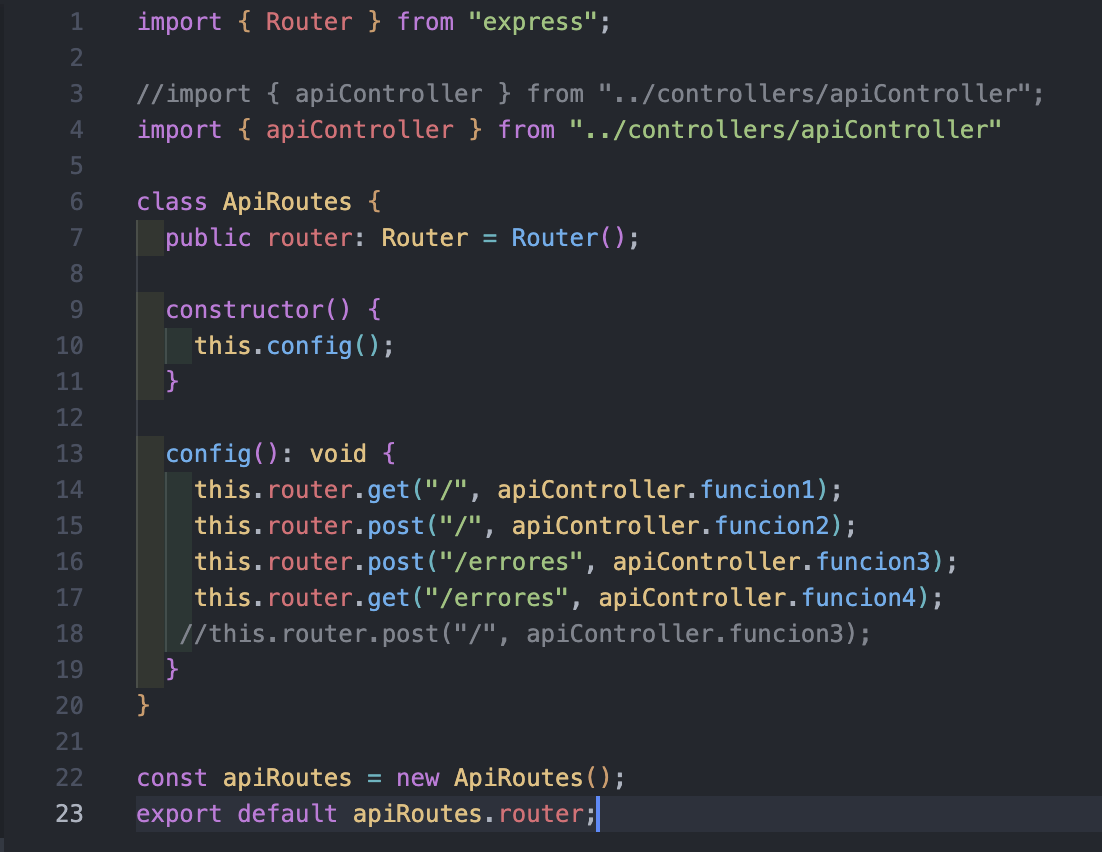


1. **Servidor**

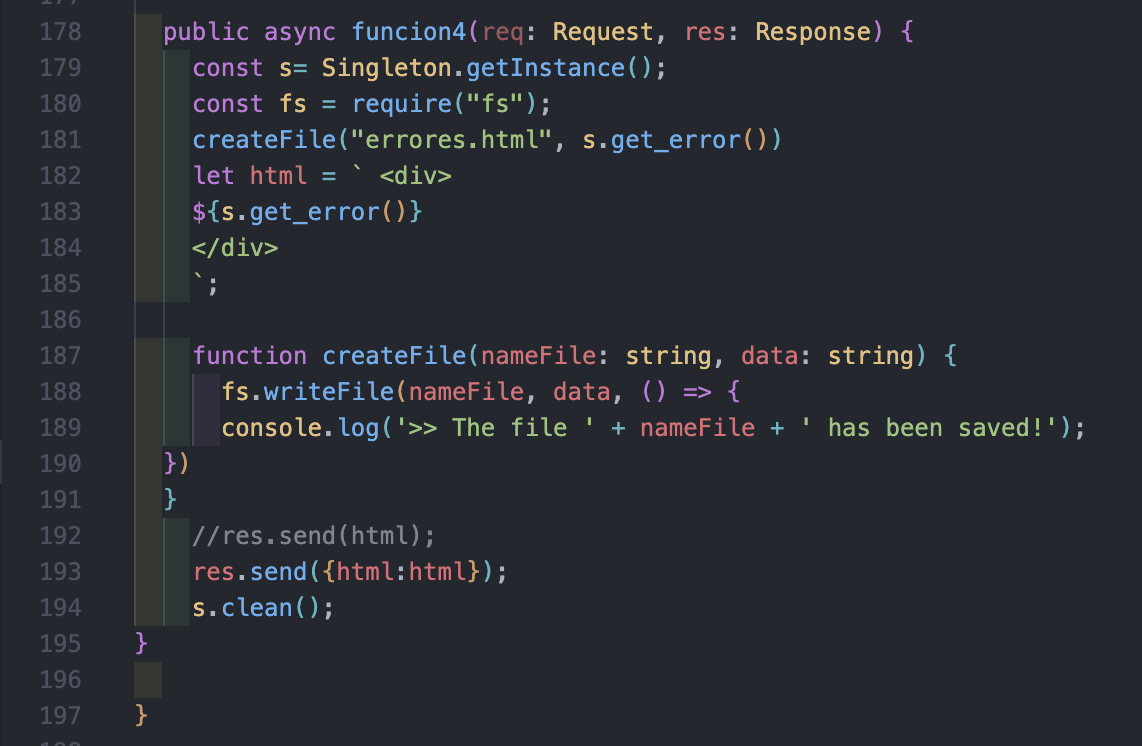
Se usó un servidor en typescript que conecta con angular.



Iniciando el servidor con las rutas para la conexión



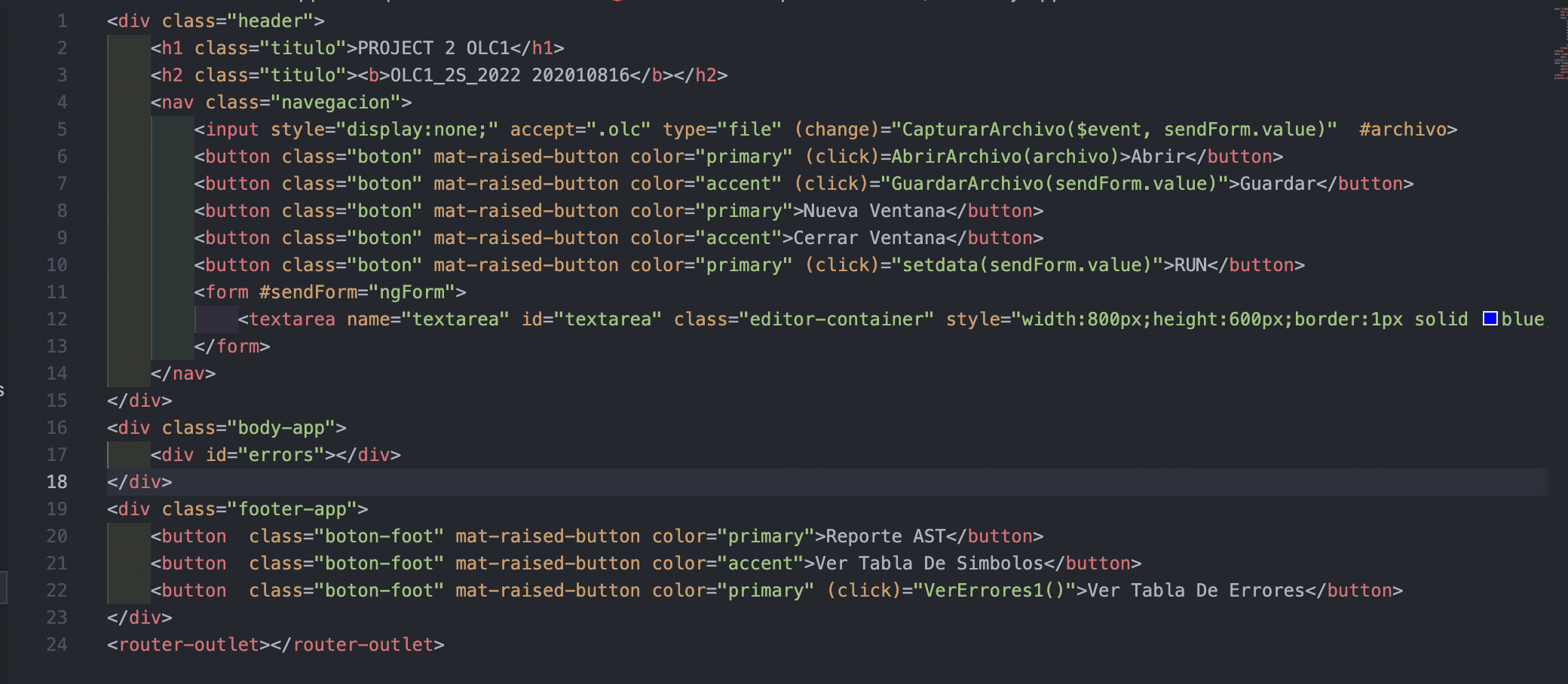
Las rutas que tendremos



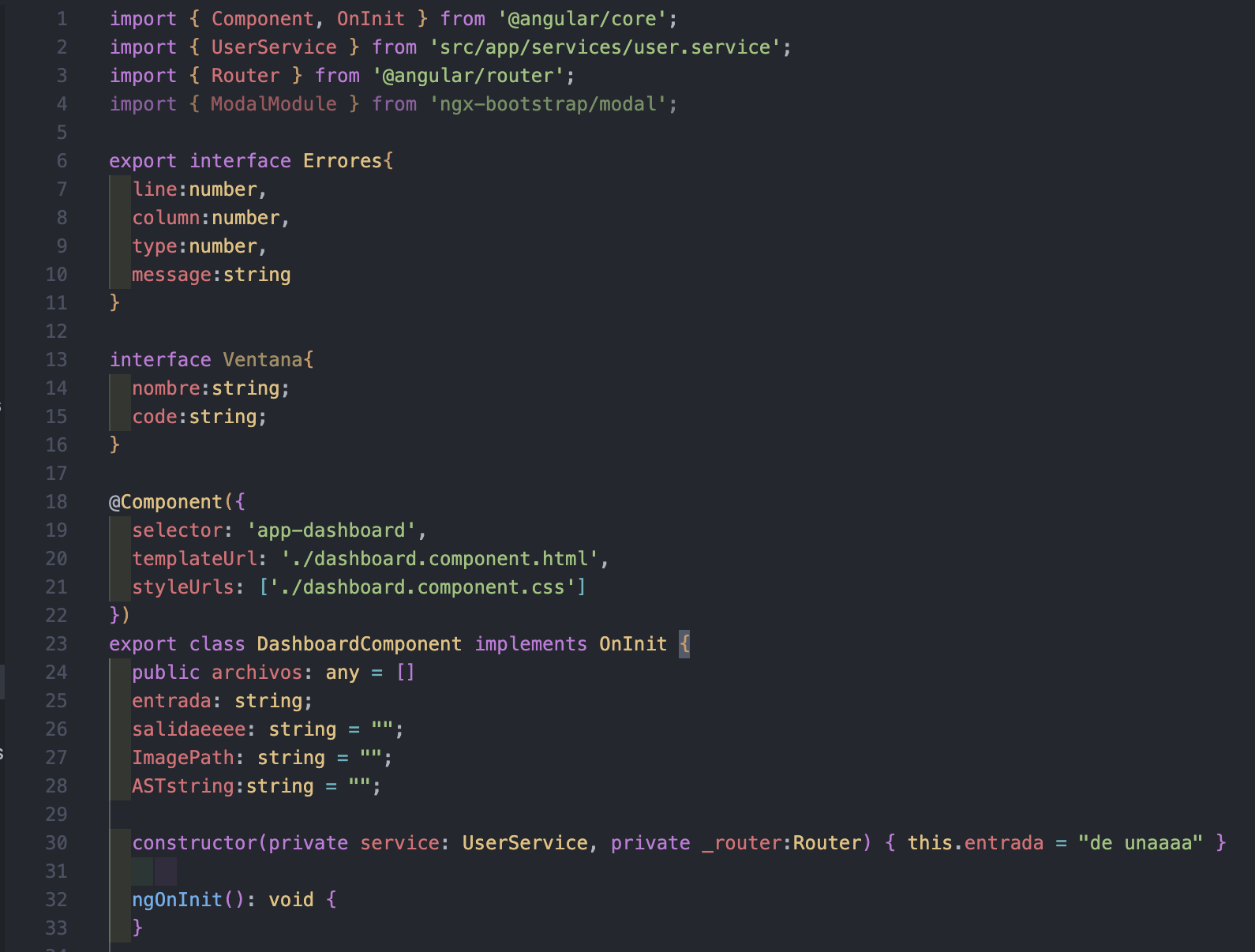
Ejemplo de un método get que da respuesta para el frontend.

1. **Frontend**

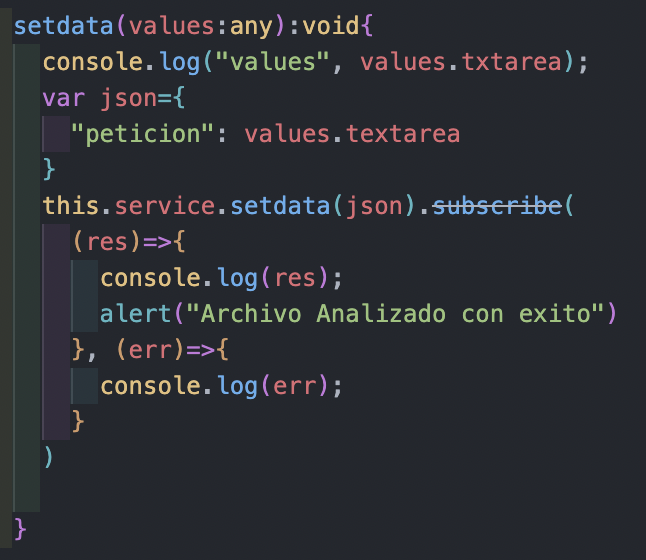
El frontend está desarrollado en angular CLI versión 14.2.5



Salida en html



Components.ts



Con este método podemos analizar el contenido del editor de texto en el backend.



Mandando petición al servidor.

Así sucesivamente con los demás métodos que tiene el servidor.

**LINK REPOSITORIO**

<https://github.com/AndreRD1026/OLC1-202010816>