**Universidad de San Carlos de Guatemala**

**Facultad de Ingeniería**

**Escuela de Ciencias y Sistemas**

**Laboratorio Organización de Lenguajes y Compiladores 1**

**Proyecto 2: CompiScript+**

**Manual Técnico**

**Nombre:** Josué Daniel Rojché García

**Carné:** 201901103

**Fecha:** 21/04/2024

**Sección:** N

**Auxiliar:** Walter Alexander Guerra Duque

**Requerimientos del Sistema**

En la realización del software el Editor de código utilizado fue VisualStudioCode, con las especificaciones que se observan en la siguiente imagen.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Para la realización del analizador se utilizó jison como herramienta, para la realización del backend y el manejo de rutas se utilizaron express y cors.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Para la realización del frontend, se utilizó react desde vite



La versión de nodejs utilizada es 

**Analizador Léxico y Sintáctico**

Para conseguir la solución del programa se realizó analizador léxico, para el reconocimiento de archivos de tipo .sc, por lo que se utiliza las librerías mencionadas anteriormente. En la siguiente imagen podrá observar el paquete para el analizador, con sus respectivos archivos.

Una captura de pantalla de un celular

Descripción generada automáticamente con confianza media

Dentro del archivo gramatica.jison se puede observar los caracteres y expresiones regulares y palabras reservadas que serán reconocidos por el lenguaje, así como el manejo de errores léxicos que puedan ocurrir.

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente con confianza media

Texto

Descripción generada automáticamente con confianza media

Dentro del archivo se tiene el manejo de errores sintácticos, la gramática que servirá para verificar la correcta lectura del archivo y la declaración de la precedencia de operaciones, también se realiza el manejo de la lógica para almacenar lo reconocido, como declaraciones, impresiones, operaciones, etc.

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente con confianza media

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Pantalla de computadora con fondo negro

Descripción generada automáticamente con confianza media

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

**Manejo de Errores**

Para el manejo de los errores, se utilizó un objeto que contendría los parámetros que se observan en la imagen y estas a su vez se almacenaron dentro de una lista y así poder recorrerla para los reportes.

Texto

Descripción generada automáticamente

**Manejo de Simbolos**

Para este se manejó un objeto y varias listas, el primero para almacenar datos con valores simples, y el segundo para manejar los que incluyen arreglos de valores y arreglos de 2 dimensiones, y todos estos objetos se almacenaron a su vez en una lista para posteriormente poder ser utilizado para obtener los datos tanto para manejo de funcionalidades y el reporte. Para ello se muestran los dos objetos en las imágenes siguientes:

Texto

Descripción generada automáticamente

Texto, Aplicación

Descripción generada automáticamente

Para tener un manejo correcto de los símbolos debido a los Entornos, se manejaron com diversas funciones, en las cuales se agrega el símbolo, se obtiene, se actualiza, esa misma lógica se utilizó para todas las declaraciones, declaraciones de arreglos, y para las funciones y métodos del lenguaje.

Texto

Descripción generada automáticamente

Captura de pantalla de computadora

Descripción generada automáticamente

**Manejo de Operaciones**

Para las operaciones aritméticas se realizó una clase donde se recibiera como parámetro en el constructor los valores de la operación, el tipo de operador a utilizar, el valor como resultado, la fila y la columna de dicha operación, primero se evalúa el tipo de operación a recibir, luego los tipos de datos de cada uno para así verificar si la operación se realizará como int, string, carácter, bool, double, y retorna el resultado. En la imagen solo se muestra el análisis de la suma, pero de la misma forma se analiza el resto de operaciones en el resto del archivo

Texto

Descripción generada automáticamente

Pantalla de computadora con letras

Descripción generada automáticamente con confianza media

**Manejo de Datos**

Dentro de la clase Dato, se retornan los datos encontrados en el análisis en jison, los cuales se convierten al valor correspondiente.

Texto

Descripción generada automáticamente

Cuando se tienen id, se busca en la tabla para obtener el valor y el tipo de dato del mismo y se retorna.

Texto

Descripción generada automáticamente