**UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

**FACULTAD DE INGENIERÍA**

**ESCUELA DE CIENCIAS Y SISTEMAS**

**RGANIZACIÓN DE LENGUAJES Y COMPILADORES 1**

**SEGUNDO SEMESTRE 2021**

**CATEDRATICO: ING. KEVIN LAJPOP**

**TUTOR ACADEMICO: SANDRA JIMÉNEZ**



**PROYECTO No.2**

**[MANUAL TÉCNICO]**

Lester Efrain Ajucum Santos 201504510

# PROYECTO SYSCOMPILER

Es un proyecto realizado para la Escuela de Ciencias y Sistemas de la Facultad de Ingeniería, que consiste en un lenguaje para los estudiantes del curso de Introducción a la programación y computación 1.

## ARQUITECTURA GENERAL DEL PROYECTO

El siguiente proyecto se manejo una arquitectura Cliente-Servidor, con el objetivo de que pueda separar los servicios administrados por el intérprete, de la aplicación cliente que se mostrará al usuario final.

Diagrama

Descripción generada automáticamente

# ENTORNO DE TRABAJO

## CLIENTE

Se desarrollo la siguiente web app, cuya finalidad es proporcionar ciertas funcionalidades, características y herramientas de utilidad al usuario.

La función principal es proveer de un editor para el ingreso de código fuente en el lenguaje antes mencionado y analizar cada una de sus instrucciones.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

## Sección de reportes

En la sección de reportes se muestra un resumen del análisis realizado al código fuente.

Reportes de errores

Se muestran los errores semánticos, léxicos y sintácticos detectados durante el análisis del código fuente.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Reportes de árbol de análisis sintáctico

El siguiente reporte despliega una imagen detallada del árbol de análisis sintáctico para el código fuente. Está es una representación abstracta de la estructura del código en el lenguaje syscompiler.

Esquemático

Descripción generada automáticamente con confianza media

Reportes de tabla de simbolos

El siguiente reporte es un resumen de todos los símbolos como variables, métodos y funciones que han sido declaradas dentro del flujo del programa.

Captura de pantalla de un videojuego

Descripción generada automáticamente

## Área de consola

Esta área sirve para controlar el flujo del programa, permite visualizar errores dentro del código fuente, mensajes y todo lo que sea indicado dentro del lenguaje.

Texto

Descripción generada automáticamente

## SERVIDOR

### NodeJs web server

Web server desarrollado con NodeJS en su versión 13.7.0



### Package manager NPM

Npm en su versión 6.13.6



### Typescript version

Typescript en su versión 4.0.3



## HERRAMIENTAS DE ANALISIS LEXICO Y SINTACTICO

### Jison

Generador de analizador sintáctico y léxico en lenguaje Javascript.

# LENGUAJE PARA PROYECTO 2 (SYSCOMPILER)

# PALABRAS RESERVADAS (KEYWORDS)

#### \*\*TIPOS DE DATOS\*\*

Int

Double

Boolean

Char

String

True

False

#### \*\*ESTRUCTURAS DE DATOS\*\*

\*Operador new\*  
new

\*Listas dinamicas\*

DynamicList

append

getValue

setValue

#### \*\*SENTENCIAS DE CONTROL\*\*

if

else

switch

case

default

#### \*\*SENTECIAS CICLICAS\*\*

while

for

Do

#### \*\*SENTENCIAS DE TRANSFERENCIA\*\*

\*sentencia break\*

break

continue

return

#### \*\*METODOS & FUNCIONES\*\*

void

WriteLine

toLower

toUpper

length

truncate

round

typeof

toString

toCharArray

start

with

# SIMBOLOS ACEPTADOS

. punto - dot

, coma - comma

; punto y coma - semicolon

: dos puntos - colon

[ corchete que abre - left square bracket

] corchete que cierra - right square bracket

( parentesis que abre - left parentheses

) parentesis que cierra - right parentheses

{ llave que abre - left brace - left bracket

} llave que cierra - right brace - right bracket

= igual - equal (Asignacion/Assignment)

## Operadores Relacionales

<= Menor o igual - less than or equal to

< Menor que - less than

>= Mayor o igual - greater than or equal to

> Mayor que - greater than

== Igualacion - Equal to

!= Diferencia - not equal

## Operador Ternario

?

## Operadores lógicos

&& And - and

|| Or - or

! not - not

## Operadores Aritméticos

-- menosmenos - decrement

++ masmas - increment

+ suma - plus

- resta - subtraction

\* multiplicacion - multiplication

/ division - division

^ potencia - exponentiation

% Modulo - modulus/remainder

- Unario - unary negation

|  |  |  |
| --- | --- | --- |
| PRECEDENCIA DE OPREADORES ( DE MAYOR A MENOR) | | |
| OPERADOR | | ASOCIATIVIDAD |
| masmas/incremento | ++ | n/a |
| menosmenos/decre | -- | n/a |
| negacion/not | ¡ | right to left |
| unario/unary | - |
| masmas/prefijo | ++ |
| menosmenos/prefijo | -- |
| potencia | ^ | righ to left (no asociativa?) |
| Multiplicacion | \* | left to right |
| division | / |
| modulo | % |
| suma | + | left to right |
| resta | - |
| menor que | < | left to right |
| menor igual que | <= |
| mayor que | > |
| mayor igual que | >= |
| igualdad | == | left to right |
| diferencia | != |
| y/and | && | left to right |
| o/or | || | left to right |