



MANUAL DE USUARIO

MATT

Juan Jose Lemus Vasquez
201404412

Contenido

Requerimientos Para el funcionamiento del Programa.....	2
Librerías Utilizadas:	2
Diagrama de Clases:	2
Diccionario de Clases:.....	3
Gramática:	4

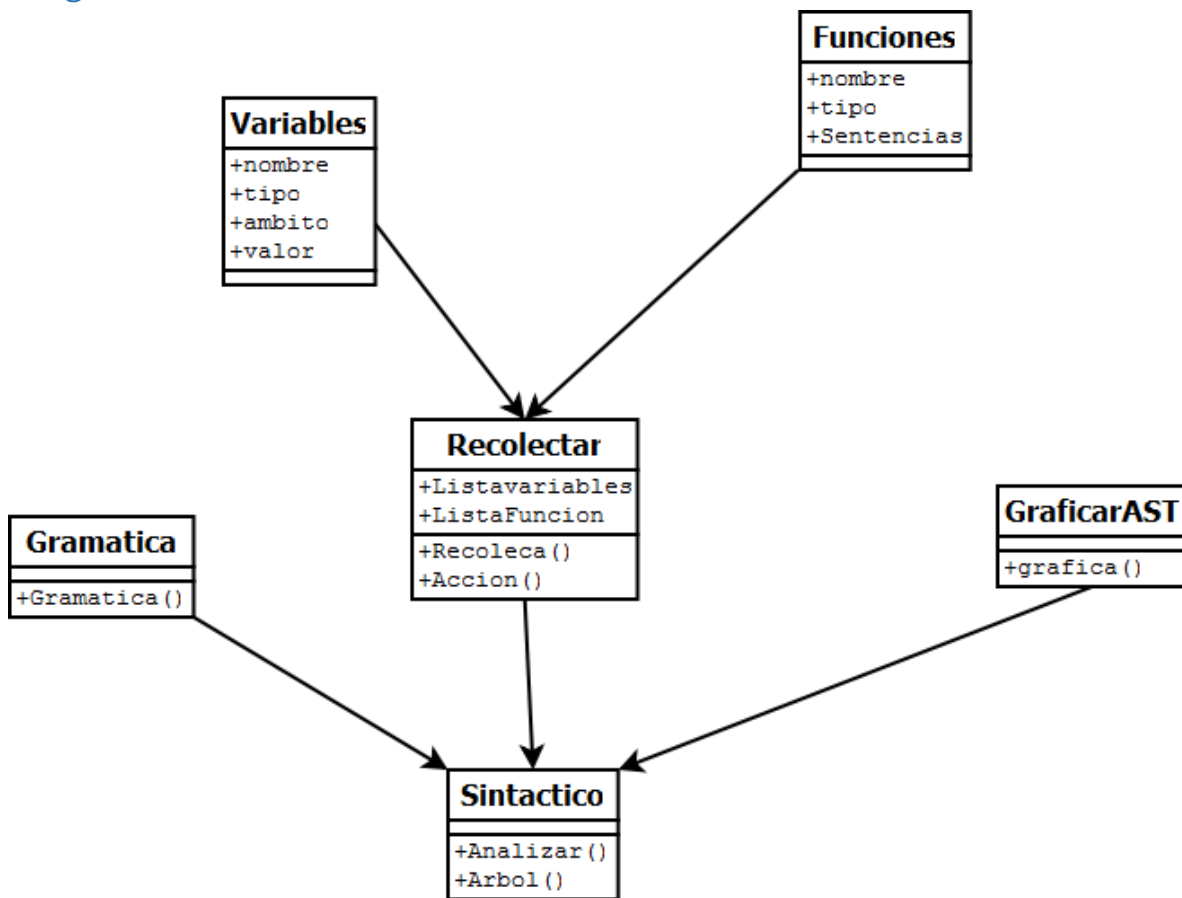
Requerimientos Para el funcionamiento del Programa

Para el correcto funcionamiento de la aplicación es necesario contar con un sistema operativo derivado de Windows, sea este Windows 7 o superior de preferencia. Dado que para la generación de algunos archivos de reporte se utiliza archivos HTML es necesario tener un navegador para poder verlos. Se recomienda de preferencia tener instalado explorador de Google Chrome.

Librerías Utilizadas:

- Irony

Diagrama de Clases:



Diccionario de Clases:

Clase: Sintactocp	
Interfas Grafica	
Atributos	
Métodos	
Analisar	Analiza las respuesta del servidor lexicamente y Sintacticamente
Arbol	Genera la imagen del Arbol AST

Clase: Graficar AST	
Interfas Grafica	
Atributos	
Métodos	
Graficar	Genera el código Dot del árbol ast.

Clase: Gramatica	
Interfas Grafica	
Atributos	
Métodos	
Gramatica	Contiene toda nuestra gramática.

Clase: Recolectar	
Interfas Grafica	
Atributos	
ListFunciones	Gurada todas las Funciones.
ListaVariales	Guardas todas las Variables.
Métodos	
Recolecta	Recolecta la información de váriales y métodos.
Accion	Realiza las acciones sobre la cadena de entrada.

Gramática:

```
//Inicio de la Gramatica
Inicio.Rule = programa + identificador + corcheteabre + Pcontp +
corchetecierra;
//Creamos Lista de Contenido de Programa
Pcontp.Rule = Pcontp + Pcont
| Pcont;
//Aqui ponemos todo lo que puede venir
Pcont.Rule = Dvariables
| Funciones
| Principal;
//Aqui definimos el Metodo Principal
Principal.Rule = principal + parentesisabre + parentisiscierra +
corcheteabre + Sent + corchetecierra
| principal + parentesisabre + parentisiscierra + corcheteabre +
corchetecierra;
//Aqui definimos las Funciones y Metodos
Funciones.Rule = Tipo + identificador + parentesisabre +
parentisiscierra + corcheteabre + Sent + corchetecierra
| Tipo + identificador + parentesisabre + parentisiscierra +
corcheteabre + corchetecierra
| Tipo + identificador + parentesisabre + Parametros +
parentisiscierra + corcheteabre + Sent + corchetecierra
| Tipo + identificador + parentesisabre + Parametros +
parentisiscierra + corcheteabre + corchetecierra;
//Aqui se definen los tipos de Funciones
Tipo.Rule = tkint
| tkdouble
| tkstring
| tkchar
| tkbool
| tkvoid;
//Aqui se definen los parametros que pueden llevar las funciones
Parametros.Rule = Parametros + coma + tkvar + identificador
| tkvar + identificador;
//Para la Declacion de Variables
Dvariables.Rule = tkvar + Ids + puntoycoma
| tkvar + identificador + asignacion + E + puntoycoma
| tkvar + identificador + asignacion + Log + puntoycoma
| identificador + asignacion + E + puntoycoma
| identificador + asignacion + Log + puntoycoma;

//multiple id...}
Ids.Rule = Ids + coma + identificador
| identificador;
//Se define E
E.Rule = E + mas + E
| E + menos + E
| E + por + E
| E + div + E
| numero_entero
| cadena
| identificador
| identificador + parentesisabre + parentisiscierra
| identificador + parentesisabre + ParValor + parentisiscierra
| caracter
| parentesisabre + E + parentisiscierra
```

```

        | raiz + parentesisabre + E + coma + E + parentisiscierra;
//Aqui definimos los valores que pueden ir dentro de una funcion
ParValor.Rule = ParValor + coma + ParValorp
        | ParValorp;
//Aqui definimos Todos los parametros que puede venir
ParValorp.Rule = numero_entero
        | cadena
        | tkchar
        | E
        | identificador
        | identificador + parentesisabre + parentisiscierra
        | identificador + parentesisabre + ParValor + parentisiscierra;
//Aqui definimos las Expresiones Logicas
Log.Rule = Log + or + Log
        | Log + and + Log
        | not + Log
        | Ex
        | tkfalse
        | tktrue
        | parentesisabre + Log + parentisiscierra;
//Aqui definimos las formas logicas
Ex.Rule = E + menor + E
        | E + mayor + E
        | E + menorigual + E
        | E + mayorigual + E
        | E + igual + E
        | E + diferente + E
        | parentesisabre + Ex + parentisiscierra
        | E;
//Aqui va la parte de las multiples sentencias que pueden ir dentro de
un metodo
Sent.Rule = Sent + Sentp
        | Sentp;
//| Empty;
//Lo que puede venir en las Sentencias
Sentp.Rule = Dvariables
        | imprimir + parentesisabre + E + parentisiscierra + puntoycoma
        | graficar + parentesisabre + cadena + coma + cadena + coma + NTvar
+ coma + NTvar + coma + cadena + parentisiscierra + puntoycoma
        | IF
        | Interrup
        | Mien
        | Hace
        | identificador + parentesisabre + parentisiscierra + puntoycoma
        | identificador + parentesisabre + ParValor + parentisiscierra +
puntoycoma;
//Definicion de NTvar
NTvar.Rule = numero_entero
        // | numero_decimal
        | menos + NTvar;
IF.Rule = SI + parentesisabre + Log + parentisiscierra + corcheteabre +
Sent2 + corchetecierra
        | SI + parentesisabre + Log + parentisiscierra + corcheteabre +
Sent2 + corchetecierra + SINO + corcheteabre + Sent2 + corchetecierra
        | SI + parentesisabre + Log + parentisiscierra + corcheteabre +
Sent2 + corchetecierra + Mulsinosi
        | SI + parentesisabre + Log + parentisiscierra + corcheteabre +
Sent2 + corchetecierra + Mulsinosi + SINO + corcheteabre + Sent2 + corchetecierra;

```

```

//Para los Multiples si no
Mulsinosi.Rule = Mulsinosi + Sinosi
| Sinosi;
//Definicon se Sinosi
Sinosi.Rule = SINO_SI + parentesisabre + Log + parentisiscierra +
corcheteabre + Sent2 + corchetecierra;
//Aqui definimos el Interrup
Interrup.Rule = INTERRUPTOR + parentesisabre + E + parentisiscierra +
corcheteabre + Casos + corchetecierra;
//Aqui definimos los Casos
Casos.Rule = Casos + Caso
| Caso;
//Aqui definimos lo que es un Caso
Caso.Rule = CASO + PorCaso + dospuntos + Sent2
| DEFECTO + dospuntos + Sent2;
//Aqui definimos esto del PorCaso
PorCaso.Rule = cadena
| numero_entero
| caracter
| tktrue
| tkfalse;
//Aqui definimos Mien
Mien.Rule = MIENTRAS + parentesisabre + Log + parentisiscierra +
corcheteabre + Sent2 + corchetecierra;
//Aqui hacemos lo Hace
Hace.Rule = HACER + corcheteabre + Sent2 + corchetecierra + MIENTRAS +
parentesisabre + Log + parentisiscierra + puntoycoma;
//Aqui definimos las Listas de Sentencis que puede ir en las estructuras
Sent2.Rule = Sent2 + Sent2p
| Sent2p;
//Aqui definimos las Sentencias
Sent2p.Rule = Dvariables
| imprimir + parentesisabre + E + parentisiscierra + puntoycoma
| graficar + parentesisabre + cadena + coma + cadena + coma + NTvar
+ coma + NTvar + coma + cadena + parentisiscierra + puntoycoma
| IF
| Interrup
| Mien
| Hace
| identificador + parentesisabre + parentisiscierra + puntoycoma
| identificador + parentesisabre + ParValor + parentisiscierra +
puntoycoma;

```