# MANUAL TÉCNICO:

### EMULADOR DB:

El presente programa consiste en un emulador de instrucciones de SQL, el cual tiene la funcionalidad de ejecutar acciones como lo haría un motor de base de datos real.

A continuación, se presenta el análisis léxico y la gramática en formato BNF.

**Análisis lexico:**

**Expresiones regulares:**

numero = [0-9]

entero = [0-9]+

decimal = [0-9]+(\.[0-9]+)?

letra = [A-Za-z]

simbolo =[\-\\_\@\+\\*\#]

nombreArchivo = ({numero}|{simbolo}|{letra})+

//([\-\\_\@\+\\*\#]|[a-zA-Z0-9])+([\-\\_\@\+\\*\#]|[a-zA-Z0-9])\*

identificador = [a-zA-Z]+([0-9a-zA-Z])\*

cadena=\"[^\"]\*\"

**Signos a identificar:** >, < , >= , <= , ( , ) , <>

**Palabras reservadas:** AND, OR, INSERTAR, SELECCIONAR, ACTUALIZAR, ELIMINAR, FILTRAR, VALORES, EN, ASIGNAR.

**Otros símbolos:** \* , coma (,), punto y coma (;)

**Análisis Sintáctico:**

**Gramática:** para la siguiente práctica se uso la siguiente gramática, incluye los terminales y no terminales. La gramática está definida según la sintaxis de JCUP.

terminal String SELECCIONAR, INSERTAR, ELIMINAR, ACTUALIZAR, FILTRAR, EN, AND, OR;

terminal String CADENA, ID, ARCHIVO, VALORES, ASIGNAR;

terminal int ENTERO;

terminal double DECIMAL;

terminal String SIG\_IGUAL, SIG\_MENOR, SIG\_MAYOR, SIG\_MENOR\_QUE, SIG\_MAYOR\_QUE;

terminal String SIG\_DIFERENTE, PCOMA, PUNTO, COMA, ASTERISCO, PARENA, PARENC;

/\* Non-terminals \*/

non terminal instruccion, select, insert, update, delete, lista\_columnas, ruta\_archivo;

non terminal condiciones, value\_list, next\_file, and\_or, numero, signos, and, or;

non terminal op\_select, lista\_valores, valores, end\_lista\_valores, update\_list, end\_up\_list;

/\* The grammar \*/

start with instruccion;

instruccion ::= select PCOMA

| insert PCOMA

| update PCOMA

| delete PCOMA

;

//--------> seleccionar

select ::= SELECCIONAR op\_select

;

op\_select ::= lista\_columnas EN ruta\_archivo condiciones

| ASTERISCO EN ruta\_archivo condiciones

;

//--------> insertar

insert ::= INSERTAR EN ruta\_archivo PARENA lista\_columnas PARENC VALORES PARENA lista\_valores PARENC

| INSERTAR EN ruta\_archivo VALORES PARENA lista\_valores PARENC

;

//--------> actualizar

update ::= ACTUALIZAR EN ruta\_archivo ASIGNAR update\_list condiciones;

update\_list ::= ID SIG\_IGUAL valores end\_up\_list;

end\_up\_list ::= COMA update\_list

|

;

//--------> eliminar

delete ::= ELIMINAR EN ruta\_archivo condiciones;

lista\_columnas ::= ID value\_list;

value\_list ::= COMA lista\_columnas

| /\* Nada \*/

;

ruta\_archivo ::= ID next\_file;

next\_file ::= PUNTO ruta\_archivo

| /\* Nada \*/

;

condiciones ::= FILTRAR ID SIG\_IGUAL CADENA and\_or

| FILTRAR ID signos numero and\_or

| /\* Nada \*/

;

and\_or ::= and

| or

;

and ::= AND ID SIG\_IGUAL CADENA and

| AND ID signos numero and

;

or ::= OR ID SIG\_IGUAL CADENA or

| OR ID signos numero or

;

signos ::= SIG\_MAYOR

| SIG\_MAYOR\_QUE

| SIG\_MENOR

| SIG\_MENOR\_QUE

| SIG\_DIFERENTE

| SIG\_IGUAL

;

lista\_valores ::= valores end\_lista\_valores

;

end\_lista\_valores ::= COMA lista\_valores

| /\* Nada \*/

;

valores ::= CADENA

| numero

;

numero ::= ENTERO

| DECIMAL

;

**Diagrama de clases:**

Al nivel de este documento en la carpeta “documentación” se incluye el diagrama de clases.