

**MANUAL TÉCNICO.**  
**Fredy José Gabriel Herrera Funes - 202130478**

- **Detalle de organización del proyecto:**

El proyecto se encuentra organizado por paquetes, el paquete principal es: com.example.aplicacioncompi1, que deriva de la carpeta main de la aplicación con Kotlin.

1. Luego está la subcarpeta: "uitheme", que dentro de ella se encuentra la carpeta "logic", en la cual se encuentran las clases principales de lógica de la aplicación, como clase principal está la clase de Kotlin, llamada: "Analizador", que es la que se encarga de llamar a las clases "Lexer" y "Parser" para realizar el análisis léxico y sintáctico de la entrada de texto que se tendrá.
2. La clase "Analizador" luego del análisis también se encarga de llamar a las clases Generadores(GeneradorDiagrama, GeneradorReportes, GeneradorReportesError) que tienen como tarea generar la información que necesita el usuario para la utilización de la aplicación.
3. También se tiene en cuenta que para la estructura de las clases "Lexer" y "Parser", se utilizó las gramáticas correspondientes y se generaron con las herramientas JFlex y CUP.
4. Luego en la subcarpeta "uitheme", se encuentra adentro la carpeta "ViewModel", la cual contiene una clase de Kotlin llamada "DiagramView" que se encarga de generar la vista del diagrama con la información y nodos que le manda el GeneradorDiagrama.
5. Como último tenemos la clase MainActivity, que se encuentra en el paquete principal, esta clase es la encargada de generar e inicializar toda la vista inicial, y de recolectar el archivo de entrada, para su posterior análisis, como generación de diagrama y reportes.

- **Gramática analizador léxico y sintáctico:**

$$G = \{V, \Sigma, R, S\}$$

- **SÍMBOLOS NO TERMINALES (V):**

V

= {programa, bloque\_codigo, lista\_sentencias, sentencia, condicion, expresion, termino, factor, bloque\_config, lista\_config, instrucion\_config, color, rgb, figura, letra}

- **SÍMBOLOS TERMINALES ( $\Sigma$ ):**

$\Sigma$  =

{INICIO, FIN, SI, ENTONCES, MIENTRAS, HACER, MOSTRAR, LEER, VAR, SEP  
ARADOR, DEFAULT, COLOR\_TEXTO\_SI, COLOR\_SI, FIGURA\_SI, LETRA\_SI,  
LETRA\_SIZE\_SI, COLOR\_TEXTO\_MIENTRAS, COLOR\_MIENTRAS, FIGUR  
A\_MIENTRAS, LETRA\_MIENTRAS, LETRA\_SIZE\_MIENTRAS, COLOR\_TEX

TO\_BLOQUE,COLOR\_BLOQUE,FIGURA\_BLOQUE,LETRA\_BLOQUE,LETRA\_SIZE\_BLOQUE,ELIPSE,CIRCULO,PARALELOGRAMO,RECTANGULO,ROMBO,RECTANGULO\_REDONDEADO,ARIAL,TIMES\_NEW\_ROMAN,COMIC\_SANS,VERDANA,PARENTESIS\_ABIERTO,PARENTESIS\_CERRADO,MAS,MENOS,MULTIPLICADO,DIVIDIDO,IGUAL,IGUAL\_IGUAL,DIFERENTE,MENOR,MAYOR,MENOR\_IGUAL,MAYOR\_IGUAL,AND,OR,NOT,COMA,PIPE,CADENA,ENTERO,DECIMAL,ID,COLOR\_HEX}

- REGLAS DE PRODUCCIÓN ( R ):

programa -> bloque\_codigo  
| bloque\_codigo SEPARADOR bloque\_config

bloque\_codigo -> INICIO lista\_sentencias FIN

lista\_sentencias -> lista\_sentencias sentencia  
| sentencia

sentencia -> VAR ID IGUAL expresion  
| ID IGUAL expresion  
| MOSTRAR CADENA  
| MOSTRAR ID  
| LEER ID  
| SI PARENTESIS\_ABIERTO condicion PARENTESIS\_CERRADO  
ENTONCES lista\_sentencias FIN SI  
| MIENTRAS PARENTESIS\_ABIERTO condicion  
PARENTESIS\_CERRADO HACER lista\_sentencias FIN MIENTRAS

condicion -> expresion IGUAL\_IGUAL expresion  
| expresion DIFERENTE expresion  
| expresion MENOR expresion  
| expresion MAYOR expresion  
| expresion MENOR\_IGUAL expresion  
| expresion MAYOR\_IGUAL expresion  
| condicion AND condicion  
| condicion OR condicion  
| NOT condicion  
| PARENTESIS\_ABIERTO condicion PARENTESIS\_CERRADO

expresion -> expresion MAS termino  
| expresion MENOS termino  
| termino

termino -> termino MULTIPLICADO factor  
| termino DIVIDIDO factor  
| factor

factor -> ENTERO

| DECIMAL  
| ID  
| PARENTESIS\_ABIERTO expresion PARENTESIS\_CERRADO

bloque\_config -> lista\_config

lista\_config -> lista\_config instrucion\_config  
| instrucion\_config

instrucion\_config -> DEFAULT IGUAL expresion  
| COLOR\_TEXTO\_SI IGUAL color PIPE expresion  
| COLOR\_SI IGUAL color PIPE expresion  
| FIGURA\_SI IGUAL figura PIPE expresion  
| LETRA\_SI IGUAL letra PIPE expresion  
| LETRA\_SIZE\_SI IGUAL expresion PIPE expresion  
| COLOR\_TEXTO\_MIENTRAS IGUAL color PIPE expresion  
| COLOR\_MIENTRAS IGUAL color PIPE expresion  
| FIGURA\_MIENTRAS IGUAL figura PIPE expresion  
| LETRA\_MIENTRAS IGUAL letra PIPE expresion  
| LETRA\_SIZE\_MIENTRAS IGUAL expresion PIPE expresion  
| COLOR\_TEXTO\_BLOQUE IGUAL color PIPE expresion  
| COLOR\_BLOQUE IGUAL color PIPE expresion  
| FIGURA\_BLOQUE IGUAL figura PIPE expresion  
| LETRA\_BLOQUE IGUAL letra PIPE expresion  
| LETRA\_SIZE\_BLOQUE IGUAL expresion PIPE expresion

color -> COLOR\_HEX  
| rgb

rgb -> expresion COMA expresion COMA expresion

figura -> ELIPSE  
| CIRCULO  
| PARALELOGRAMO  
| RECTANGULO  
| ROMBO  
| RECTANGULO\_REDONDEADO

letra -> ARIAL  
| TIMES\_NEW\_ROMAN  
| COMIC\_SANS  
| VERDANA

- SÍMBOLO INICIAL (S):

S= {programa}

- **Diagrama de Clases:**

