

Grupo 18

Manual Tecnico

Tecnologías Usadas

Codemirror :

<https://codemirror.net/>

Un editor de texto construido en javascript, que se utilizó para mostrar el código en la interfaz.

Viz js:

<https://github.com/mdaines/viz.js/>

Librería que se utilizó para graficar los árboles en el proyecto.

Angular:

<https://angular.io/>

Framework que se utilizó para realizar el proyecto

Gh pages:

<https://tytusdb.github.io/tytusx/20211SVAC/G18/>

tecnología de github para mostrar páginas estáticas, se utilizo para mostrar el proyecto en la web.

Bootstrap:

<https://getbootstrap.com/>

Librería que se utilizó para agregar estilo a la interfaz del proyecto.

Font Awesome:

<https://fontawesome.com/>

Librería que se utilizó para agregar iconos a la interfaz.

Gramatica ASC XML

```
S: tk_xmldec I EOF
| I EOF
| EOF
;

I: OTAG CONTENIDO CTAG
| OTAG CTAG
;

OTAG: tk_starttag tk_id tk_endtag
| tk_starttag tk_id ARGUMENTOS tk_endtag
```

```

;

ARGUMENTOS: ARGUMENTOS tk_id tk_igual tk_tagval
| tk_id tk_igual tk_tagval
;

CONTENIDO: CONTENIDO OTAG CONTENIDO CTAG
| CONTENIDO OTAG CTAG
| CONTENIDO SIMPLETAG2
| CONTENIDO tk_valin
| OTAG CONTENIDO CTAG
| OTAG CTAG
| SIMPLETAG2
| tk_valin
;

SIMPLETAG2: tk_starttag tk_id ARGUMENTOS tk_closetag tk_endtag
| tk_starttag tk_id tk_closetag tk_endtag
;

CTAG: tk_starttag tk_closetag tk_id tk_endtag;

```

Gramatica DESC XML

```

S: tk_xmldec I EOF
| I EOF
| EOF
;

I: OTAG LELEMENT CTAG;

LELEMENT: OTAG LELEMENT CTAG LELEMENT
| tk_valin LELEMENT
| epsilon
;

OTAG: tk_starttag tk_id tk_endtag
| tk_starttag tk_id ARGUMENTOS tk_endtag
;

ARGUMENTOS: tk_id tk_igual tk_tagval ARGUMENTOSP ;

```

```

ARGUMENTOSP: tk_id tk_igual tk_tagval ARGUMENTOSP
    | epsilon
    ;

CTAG: tk_closetag tk_id tk_endtag;

```

Gramatica ASC XPATH

```

%start Init
%%
Init : Exp EOF
    {
        var init = new NodoXML("Init","Init",@1.first_line+1,@1.first_column+1);
        init.addHijo($1);
        return init;
    } ;

Exp : DIVSIGN Lexp
    {
        var exp = new NodoXML("Exp","Exp",@1.first_line+1,@1.first_column+1);
        var val = new NodoXML($1,"Exp",@1.first_line+1,@1.first_column+1);
        exp.addHijo(val);
        exp.addHijo($2);
        $$ = exp;
    }
    | Lexp
    {
        var exp = new NodoXML("Exp","Exp",@1.first_line+1,@1.first_column+1);
        exp.addHijo($1);
        $$ = exp;
    } ;

Lexp : Lexp ORSIGN DIVSIGN Syntfin
    {
        var lexp = new NodoXML("Lexp","Lexp",@1.first_line+1,@1.first_column+1);
        var val1 = new NodoXML($2,"Lexp",@1.first_line+1,@1.first_column+1);
        var val2 = new NodoXML($3,"Lexp",@1.first_line+1,@1.first_column+1);
        lexp.addHijo($1);
        lexp.addHijo(val1);
        lexp.addHijo(val2);
        lexp.addHijo($4);
        $$ = lexp;
    }
    | Lexp DIVSIGN Syntfin
    {
        var lexp = new NodoXML("Lexp","Lexp",@1.first_line+1,@1.first_column+1);
        var val1 = new NodoXML($2,"Lexp",@1.first_line+1,@1.first_column+1);
        lexp.addHijo($1);
        lexp.addHijo(val1);
        lexp.addHijo($3);
        $$ = lexp;
    } ;

Syntfin
    {
        var lexp = new NodoXML("Lexp","Lexp",@1.first_line+1,@1.first_column+1);
        lexp.addHijo($1);
        $$ = lexp;
    } ;

Syntfin : Fin
    {
        var syntfin = new NodoXML("Syntfin","Syntfin",@1.first_line+1,@1.first_column+1);
        syntfin.addHijo($1);
        $$ = syntfin;
    }
    | 'e' Valor Opc
    {
        var syntfin = new NodoXML("Syntfin","Syntfin",@1.first_line+1,@1.first_column+1);
        var val1 = new NodoXML($1,"Syntfin",@1.first_line+1,@1.first_column+1);
        syntfin.addHijo(val1);
        syntfin.addHijo($2);
        syntfin.addHijo($3);
        $$ = syntfin;
    }
    | Preservada '::' Fin
    {
        var syntfin = new NodoXML("Syntfin","Syntfin",@1.first_line+1,@1.first_column+1);
        var val1 = new NodoXML($2,"Syntfin",@1.first_line+1,@1.first_column+1);
        syntfin.addHijo($1);
        syntfin.addHijo(val1);
        syntfin.addHijo($3);
        $$ = syntfin;
    }
    | 'e' Preservada Opc
    {
        var syntfin = new NodoXML("Syntfin","Syntfin",@1.first_line+1,@1.first_column+1);
        var val1 = new NodoXML($1,"Syntfin",@1.first_line+1,@1.first_column+1);
        syntfin.addHijo(val1);
        syntfin.addHijo($2);
        syntfin.addHijo($3);
        $$ = syntfin;
    }
    | 'e' '*'
    {
        var syntfin = new NodoXML("Syntfin","Syntfin",@1.first_line+1,@1.first_column+1);
        var val1 = new NodoXML($1,"Syntfin",@1.first_line+1,@1.first_column+1);
        var val2 = new NodoXML($2,"Syntfin",@1.first_line+1,@1.first_column+1);
        syntfin.addHijo(val1);
        syntfin.addHijo(val2);
        $$ = syntfin;
    } ;

```

```

Fin : Valor Opc
{
    var fin = new NodoXML("Fin","Fin",@1.first_line+1,@1.first_column+1);
    fin.addHijo($1);
    fin.addHijo($2);
    $$ = fin;
}

DIR Opc
{
    var fin = new NodoXML("Fin","Fin",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($1,"Fin",@1.first_line+1,@1.first_column+1);
    fin.addHijo(val1);
    fin.addHijo($2);
    $$ = fin;
}

TEXT '(' ')'
{
    var fin = new NodoXML("Fin","Fin",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($1,"Function",@1.first_line+1,@1.first_column+1);
    fin.addHijo(val1);
    $$ = fin;
}

NODE '(' ')'
{
    var fin = new NodoXML("Fin","Fin",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($1,"Function",@1.first_line+1,@1.first_column+1);
    fin.addHijo(val1);
    $$ = fin;
}

POSITION '(' ')'
{
    var fin = new NodoXML("Fin","Fin",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($1,"Function",@1.first_line+1,@1.first_column+1);
    fin.addHijo(val1);
    $$ = fin;
}

LAST '(' ')'
{
    var fin = new NodoXML("Fin","Fin",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($1,"Function",@1.first_line+1,@1.first_column+1);
    fin.addHijo(val1);
    $$ = fin;
}

```

```

Preservada Opc
{
    var fin = new NodoXML("Fin","Fin",@1.first_line+1,@1.first_column+1);
    fin.addHijo($1);
    fin.addHijo($2);
    $$ = fin;
}

'*' Opc
{
    var fin = new NodoXML("Fin","Fin",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($1,"Fin",@1.first_line+1,@1.first_column+1);
    fin.addHijo(val1);
    fin.addHijo($2);
    $$ = fin;
};

Valor : ID
{
    var val = new NodoXML($1,"Valor",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

NUMBER
{
    var val = new NodoXML($1,"Valor",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

STRING
{
    var val = new NodoXML($1,"Valor",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

STRING2
{
    var val = new NodoXML($1,"Valor",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

DECIMAL
{
    var val = new NodoXML($1,"Valor",@1.first_line+1,@1.first_column+1);
    $$ = val;
};

```

```

Preservada: CHILD
{
    var val = new NodoXML($1,"Axes",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

DESCENDANT
{
    var val = new NodoXML($1,"Axes",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

ANCESTOR
{
    var val = new NodoXML($1,"Axes",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

PRECEDING
{
    var val = new NodoXML($1,"Axes",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

FOLLOWING
{
    var val = new NodoXML($1,"Axes",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

NAMESPACE
{
    var val = new NodoXML($1,"Axes",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

SELF
{
    var val = new NodoXML($1,"Axes",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

PARENT
{
    var val = new NodoXML($1,"Axes",@1.first_line+1,@1.first_column+1);
    $$ = val;
}

ATTR
{
    var val = new NodoXML($1,"Axes",@1.first_line+1,@1.first_column+1);
    $$ = val;
};

```

```

Opc : LPredicado
{
    var opc = new NodoXML("Opc","Opc",@1.first_line+1,@1.first_column+1);
    opc.addHijo($1);
    $$ = opc;
}

var opc = new NodoXML("Opc","Opc",@1.first_line+1,@1.first_column+1);
$$ = opc;
};

LPredicado : LPredicado Predicado
{
    var lpredicado = new NodoXML("LPredicado","LPredicado",@1.first_line+1,@1.first_column+1);
    lpredicado.addHijo($1);
    lpredicado.addHijo($2);
    $$ = lpredicado;
}

Predicado
{
    var lpredicado = new NodoXML("LPredicado","LPredicado",@1.first_line+1,@1.first_column+1);
    lpredicado.addHijo($1);
    $$ = lpredicado;
};

Predicado: '[' ExprLogica ']'
{
    var predicado = new NodoXML("Predicado","Predicado",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($1,"Predicado",@1.first_line+1,@1.first_column+1);
    var val2 = new NodoXML($2,"Predicado",@1.first_line+1,@1.first_column+1);
    predicado.addHijo(val1);
    predicado.addHijo($2);
    predicado.addHijo(val2);
    $$ = predicado;
};

ExprLogica
: Expr '<=' Expr
{
    var exprLogica = new NodoXML("ExprLogica","ExprLogica",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    exprLogica.addHijo($1);
    exprLogica.addHijo(val1);
    exprLogica.addHijo($3);
    $$ = exprLogica;
}

```

```

Expr '>=' Expr
{
    var exprLogica = new NodoXML("ExprLogica","ExprLogica",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    exprLogica.addHijo($1);
    exprLogica.addHijo(val1);
    exprLogica.addHijo($3);
    $$ = exprLogica;
}

Expr '=' Expr
{
    var exprLogica = new NodoXML("ExprLogica","ExprLogica",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    exprLogica.addHijo($1);
    exprLogica.addHijo(val1);
    exprLogica.addHijo($3);
    $$ = exprLogica;
}

Expr '!=' Expr
{
    var exprLogica = new NodoXML("ExprLogica","ExprLogica",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    exprLogica.addHijo($1);
    exprLogica.addHijo(val1);
    exprLogica.addHijo($3);
    $$ = exprLogica;
}

Expr '>' Expr
{
    var exprLogica = new NodoXML("ExprLogica","ExprLogica",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    exprLogica.addHijo($1);
    exprLogica.addHijo(val1);
    exprLogica.addHijo($3);
    $$ = exprLogica;
}

Expr '<' Expr
{
    var exprLogica = new NodoXML("ExprLogica","ExprLogica",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    exprLogica.addHijo($1);
    exprLogica.addHijo(val1);
    exprLogica.addHijo($3);
    $$ = exprLogica;
}

```

```

Expr
{
    var exprLogica = new NodoXML("ExprLogica","ExprLogica",@1.first_line+1,@1.first_column+1);
    exprLogica.addHijo($1);
    $$ = exprLogica;
};

Expr : Expr '+' Expr
{
    var expr = new NodoXML("Expr","Expr",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    expr.addHijo($1);
    expr.addHijo(val1);
    expr.addHijo($3);
    $$ = expr;
}

Expr '-' Expr
{
    var expr = new NodoXML("Expr","Expr",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    expr.addHijo($1);
    expr.addHijo(val1);
    expr.addHijo($3);
    $$ = expr;
}

Expr '*' Expr
{
    var expr = new NodoXML("Expr","Expr",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    expr.addHijo($1);
    expr.addHijo(val1);
    expr.addHijo($3);
    $$ = expr;
}

Expr DIV Expr
{
    var expr = new NodoXML("Expr","Expr",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    expr.addHijo($1);
    expr.addHijo(val1);
    expr.addHijo($3);
    $$ = expr;
}

Expr MOD Expr
{
    var expr = new NodoXML("Expr","Expr",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"Operador",@1.first_line+1,@1.first_column+1);
    expr.addHijo($1);
}

```

```

        expr.addHijo($1);
        expr.addHijo(val1);
        expr.addHijo($3);
        $$ = expr;
    }
| Expr MOD Expr
{
    var expr = new NodoXML("Expr","Expr",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"operator",@1.first_line+1,@1.first_column+1);
    expr.addHijo($1);
    expr.addHijo(val1);
    expr.addHijo($3);
    $$ = expr;
}
| Expr OR Expr
{
    var expr = new NodoXML("Expr","Expr",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"operator",@1.first_line+1,@1.first_column+1);
    expr.addHijo($1);
    expr.addHijo(val1);
    expr.addHijo($3);
    $$ = expr;
}
| Expr AND Expr
{
    var expr = new NodoXML("Expr","Expr",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($2,"operator",@1.first_line+1,@1.first_column+1);
    expr.addHijo($1);
    expr.addHijo(val1);
    expr.addHijo($3);
    $$ = expr;
}
| '(' Expr ')'
{
    var expr = new NodoXML("Expr","Expr",@1.first_line+1,@1.first_column+1);
    var val1 = new NodoXML($1,"operator",@1.first_line+1,@1.first_column+1);
    var val2 = new NodoXML($3,"operator",@1.first_line+1,@1.first_column+1);
    expr.addHijo(val1);
    expr.addHijo($2);
    expr.addHijo(val2);
    $$ = expr;
}
| Exp
{
    var expr = new NodoXML("Expr","Expr",@1.first_line+1,@1.first_column+1);
    expr.addHijo($1);
    $$ = expr;
};

```

Gramatica DESC XPATH

```
Init : Exp EOF ;

Exp : DIVSIGN Lexp
    | Lexp;

Lexp : Syntfin Lexp_prima;

Lexp_prima : '|' DIVSIGN Syntfin Lexp_prima
            | DIVSIGN Syntfin Lexp_prima
            | ;

Syntfin : Fin
        | '@' Valor Opc
        | Preservada '::' Fin
        | '@' Preservada Opc;

Fin : Valor Opc
    | DIR Opc
    | TEXT '(' ')'
    | NODE '(' ')'
    | POSITION '(' ')'
    | Preservada Opc
    | '*' Opc ;

Valor : ID
      | NUMBER
      | STRING
      | STRING2
      | DECIMAL;

Preservada: CHILD
          | DESCENDANT
          | ANCESTOR
          | PRECEDING
          | FOLLOWING
          | NAMESPACE
          | SELF
          | PARENT;

Opc : LPredicado
    | ;
```



```

LPredicado : Predicado LPredicado_prima;

LPredicado_prima : Predicado LPredicado_prima
|
|;

Predicado: '[' ExprLogica']';

ExprLogica
|
| : Expr '<=' Expr
| Expr '>=' Expr
| Expr '==' Expr
| Expr '!=' Expr
| Expr '>' Expr
| Expr '<' Expr
| Expr;

Expr : Exp Expr_prima;

Expr_prima: '+' Exp Expr_prima
| '-' Exp Expr_prima
| '*' Exp Expr_prima
| DIV Exp Expr_prima
| MOD Exp Expr_prima
| OR Exp Expr_prima
| AND Exp Expr_prima
|;

```