

Laboratorio 7

Dayana Marín Mayorga

B64096

Métodos:

Se crea un mapa con todas las posibles etiquetas.

```
using namespace std;

map<string,int> agregar(){

map<string,int> map;|
map["!-...-"]=0;
map["!DOCTYPE"]=0;
map["a"]=0;
map["abbr"]=0;
map["address"]=0;
map["area"]=0;
map["article"]=0;
map["aside"]=0;
map["audio"]=0;
map["b"]=0;
map["base"]=0;
map["bdi"]=0;
map["bdo"]=0;
map["blockquote"]=0;
map["body"]=0;
map["br"]=0;
map["button"]=0;
map["canvas"]=0;
map["caption"]=0;
map["cite"]=0;
map["code"]=0;
map["col"]=0;
map["colgroup"]=0;
map["data-bbox=" data-cs="2" data-kind="parent">
```

Método que se encarga básicamente de comparar la palabra que entra con las etiquetas que se encuentran guardadas.

```
map<string,int> comparar(char * token, map<string,int> etiqueta) {
    map<string,int>::iterator ite = etiqueta.begin();
    string llave;
    while(ite != etiqueta.end()){
        llave = ite->first;
        if((llave.compare(token)) == 0){
            etiqueta[llave]++;
        }
        ite++;
    }
    return etiqueta;
}
```

Método encargado de procesar la línea del archivo.

```
map<string, int > processLine( const char * line, map <string,int> etiqueta ) {
    char * token;
    regex texto( ">[^<]*<" );
    string ecci;

    ecci = regex_replace( line, texto, "> <" );
    // std::cout << "eccí = " << ecci << std::endl;

    token = strtok( (char * ) line, "< >\t\n" );

    while ( NULL != token ) {
        etiqueta = comparar(token,etiqueta);

        token = strtok( NULL, "< >\t\n" );
    }
    return etiqueta;
}
```