

**UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

**CENTRO UNIVERSITARIO DE OCCIDENTE**

**DIVISIÓN DE CIENCIAS DE LA INGENIERÍA**



**MANUAL TÉCNICO DE PROYECTO FINAL**

**LABORATORIO DE  
ORGANIZACIÓN DE LENGUAJES Y COMPILADORES 1**

**PRESENTADO POR:**

**UMAÑA DE LEÓN, WILLIAM RODRIGO**

**(201931448)**

**DOCENTE:**

**Ing. JOSE MOISES GRANADOS GUEVARA**

**AUXILIAR**

**Bach. BRAYN MISAEL MONZON FUENTES**

**QUETZALTENANGO – QUETZALTENANGO – GUATEMALA**

**02-04-2021**

## Índice

<b>Índice</b>	<b>2</b>
<b>Contenido</b>	<b>3</b>
Información de la aplicación	3
Detalle proyecto	3
Analizadores	3
Léxico GCIC	3
Sintáctico GCIC	4

## Contenido

### Información de la aplicación

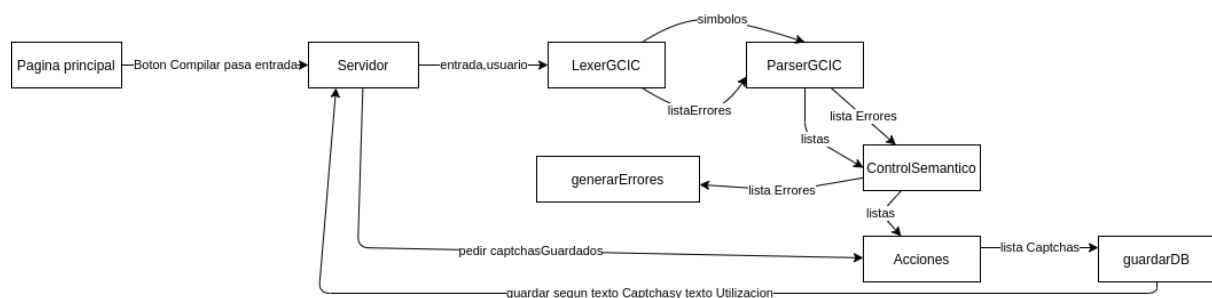
El proyecto fue escrito en el IDE Netbeans versión 12.0 y contiene las siguientes especificaciones.

Lenguajes utilizados: JAVA WEB, jFlex, java Cup, javascript, html, bootstrap y css.

Nombre del proyecto: serverProy2

Versión del proyecto: 1.0-SNAPSHOT

### Detalle proyecto



### Analizadores

#### Léxico GCIC

Este analizador es el encargado de leer la entrada por el usuario en la página principal

Se utilizaron las siguientes reglas léxicas:

comentario1 = "!"[a-zA-Z0-9 ]\*

comentario2 = "<!--"[a-zA-Z0-9 \r\t\b\f\n]\*"-->"

palabra = [a-zA-ZÀ-ÿ0-9\u00f1\u00d1]

idEt = [{palabra}\$\_-]+

referencia = https?:\V[a-zA-Z0-9\-.]+(\.[a-zA-Z0-9\-.]+)([/#?]?)

decim = [0-9]+.[0-9]+

valp = "\"([ {palabra} #(\)\n]+| {idEt}| {referencia})+\""

caracter = "\"({palabra}| {idEt}| [ <>\- \[ \] \= \])+\""

proceso = PROCESS\_{palabra}+

palabras = ({palabra})({palabra}[ ])\*

Las palabras reservadas son las siguientes:

C\_GCIC,C\_HEAD,C\_TITLE,C\_LINK,C\_BODY,C\_SPAM,C\_INPUT,C\_TEXTAREA,C\_SELECT,C\_OPTION,C\_DIV,C\_IMG,C\_BR,C\_BUTTON,C\_H1,C\_P,C\_SCRIPTING,href,background,color,font-size,font-family,text-align,type,id,name,cols,rows,class,src,width,height,alt,onclick,integer , decimal, boolean, char, string, ASC, DESC, LETPAR\_NUM, LETIMPAR\_NUM, REVERSE, CHARACTER\_ALEATORIO, NUM\_ALEATORIO,

ALERT\_INFO, EXIT, IF, THENWHILE, THEN, INIT, END, ELSE IF, ELSE, REPEAT, HUNTIL, INSERT, WHILE, ON\_LOAD, true, false, @global, getElementById.

Los simbolos reservados son los siguientes: :, !, <, >, {, }, [, ] \_, -, \$, ', ', (, ), =

### Sintáctico GCIC

$G = \{N, T, P, L\}$

$N = \{\text{inicio}\}$

$T = \{\text{Son todos los anteriormente descritos en el lexer indigo como palabras reservadas y reglas lexicas}\}$

$L = \{\text{inicio}\}$

inicio	< C_GCIC par0 > sig </C_GCIC>
--------	-------------------------------

sig	<C_HEAD> hed </C_HEAD> sig
	<C_BODY par2>sig</C_BODY> sig
	<C_SPAM par3>pals </C_SPAM> sig
	<C_INPUT par4 ></C_INPUT> sig
	<C_TEXTAREA par5 >pals</C_TEXTAREA> sig
	<C_SELECT par6 >opt</C_SELECT> sig
	<C_DIV par7>sig</C_DIV> sig
	<C_IMG par8></C_IMG> sig
	<C_BR>sig
	<C_BUTTON par9></C_BUTTON> sig
	<C_H1 par10>pals</C_H1> sig
	<C_P par11>pals</C_P> sig
	<C_SCRIPTING>scrip</C_SCRIPTING>
	e

hed	<C_TITLE> pals </C_TITLE> hed
	<C_LINK par1> pals </C_LINK> hed
	e

opt	<C_OPTION>pals</C_OPTION> opt
	e

scrip	PROCESO () [instrucciones] scrip
	ONLOAD ()[instrucciones] scrip
	e

pals	palabra
	palabra+
	e

par0	[id valp] par0
	[name valp] par0
	e
par1	href
par2	[background valp]
	e
par3	[color valp] par3
	parGlob par3
	e
par4	[type valp] par4
	parGlob par4
	e
par5	[cols valp] par5
	[rows valp] par5
	parGlob par5
	e
par6	[color valp] par6
	parGlob par6
	e
par7	[class valp] par7

	[color valp] par7
	[background valp] par7
	parGlob par7
	e
par8	[src valp] par8
	[width valp] par8
	[height valp] par8
	[alt valp] par8
	[id valp] par8
	e
par9	[onClick valp] par7
	[color valp] par7
	[background valp] par7
	parGlob par7
par10	[color valp] par10
	parGlob par10
	e
par11	[color valp] par11
	parGlob par11
	e

valp	= " pals "
parGlob	[font-size valp]
	[font-family valp]
	[text-align valp]
	[id valp]

instrucciones	declaracion instrucciones
	asignacion instrucciones

	IF ( condicion ) THEN init ifp instrucciones
	REPEAT (rep) HUNTIL (hun) init instrucciones
	WHILE (condicion) THENWHILE init instrucciones
	INSERT(token) instrucciones
	ALERT(parS) instrucciones
	EXIT()
	e
inite	INIT { : instrucciones : } END
ifp	ELSE init
	ELSE IF (condicion) init ifp
	e
rep	declaracion
	asignacion
hun	exprNum
	variable
token	variable tokenp
	caracter tokenp
tokenp	, token
	e
condicion	opNot asg2 opRel opNot asg2 opLog
	opNot asg2
opNot	!
	e

tipo	integer
	string
	decimal
	boolean

	char
variable	palabras
	idEtiqueta
modo	@global
	e
declaracion	tipo modo variable fin1
asignacion	variable = asg
asg2	valp fin3
	true fin3
	false fin3
	variable fin3
	decim fin3
	caracter fin3
	getElement ( caracter ) fin3
	ASC(parS) fin3
	DESC(parS) fin3
	LETPARNUM(parS) fin3
	LETIMPARNUM(parS) fin3
	REVERSE(parS) fin3
	CARACTERALEAT() fin3
	NUMALEATORIO() fin3
	(asg2) fin3
fin3	+ asg
	- asg
	* asg
	/ asg
	e
opRel	<=
	>=
	<



	>
	==
	!=
opLog	condicion
	&& condicion
	e

fin1	;
	, variable fin1
	= asg
asg	valp fin2
	true fin2
	false fin2
	variable fin2
	decim fin2
	caracter fin2
	getElement ( caracter ) fin2
	ASC(parS) fin2
	DESC(parS) fin2
	LETPARNUM(parS) fin2
	LETIMPARNUM(parS) fin2
	REVERSE(parS) fin2
	CARACTERALEAT() fin2
	NUMALEATORIO() fin2
	(asg) fin2
fin2	+ asg
	- asg
	* asg
	/ asg
	;
parS	valp
	caracter
	variable

