

Manual Técnico

-Proyecto 2 Lenguajes Formales De Programación -

Elaborado por: Sergio Guix

Carné: 201906468

Fecha: 28 / 04 / 2022

Objetivos

Como principal objetivo crear un software capaz de brindar un servicio optimo al usuario mediante uso de interfaces gráficas, donde el usuario interactúa de manera sencilla e intuitiva mediante del uso del mouse, de su computador siendo así que también pueda realizar una búsqueda de una forma rápida de información sobre los partidos de la La Liga.

Conocimientos Previos

Los conocimientos que deberán tener las personas que manejan el programa son:

- Conocimiento de lenguaje Python
- Manejo de librería Tkinter
- Manejo de listas
- Manejo de Frame
- Manejo de archivos CSV

Requerimientos del Software

-Sistema operativo:

- 32 o 64 bits (dependiendo de esto se da seguimiento a los requerimientos de software)

-versión de Python:

- Se recomienda la versión utilizar la versión más reciente.
- Este es el lenguaje en el cual fue elaborado el software.

-Requerimientos mínimos de hardware (Windows):

- Procesador: Intel Pentium III o equivalente a 800 MHz
- Memoria: : 2 GB (recomendado)
- Espacio en disco: 20 GB (recomendado).
- Mouse y monitor funcional.

Descripción de Clases

Clase	Descripción
Analizado Léxico	En esta clase esta el objeto de tipo analizador léxico que se utilizara para realizar el análisis léxico del comando ingresado por el usuario.
Sintáctico	En esta clase se instancia un objeto de tipo Sintáctico para realizar el análisis sintáctico del comando ingresado por el usuario.
Token	Clase donde se creara un objeto de tipo Token que almacenara los tokens obtenidos por el analizador léxico
TypeToken	En esta clase se importa la clase Enum para crear constantes enumeradas de los tokens que se obtendrán con el analizado léxico
LigaBot	Clase LigaBot que servirá para crear un objeto de tipo LigaBot, donde se almacenaran los datos proporcionados por el CSV
TablaLiga	Clase TablaLiga que servirá para crear un objeto de tipo TablaLiga , que tiene como atributos: equipo, total_pts, temporada.
ReporteErroresSintacticos	Clase ReporteErroresSintacticos que servirá para crear un objeto de tipo ReporteErroresSintacticos , para realizar los reportes html.
ReporteErroresLexicos	Clase ReporteErroresLexicos que servirá para crear un objeto de tipo ReporteErroresLexicos , para realizar los reportes html.
main	Clase principal donde se ejecuta el programa.

Descripción de Métodos

Clase	Descripción
inicializarGUI	Método que se encargara de inicializar los objetos de la interfaz grafica.
obtenerTextoDeUsuario	Método que tiene la lógica para obtener el comando ingresado por el usuario.
seleccionarArchivo	Método que se encargara de extraer el contenido del archivo CSV seleccionado y asignarlo a una lista de tipo LigaBot.
resultadoDeUnPartido	Método que tiene la lógica para obtener el resultado de un partido.
resultadoDeUnaJornada	Método que tiene la lógica para obtener el resultado de una jornada.
totalDeGolesEnUnaTemporada	Método que tiene la lógica para obtener el total de goles anotado por un equipo durante una temporada especificada.
tablaGeneralDeTemporada	Método que tiene la lógica para ordenar los equipos por los puntos obtenidos en una temporada.
temporadaDeUnEquipo	Método que muestra todos los resultados de un equipo durante una temporada de La Liga
TopDeEquipos	Método que realiza los algoritmos para mostrar el top (superior o inferior) de los equipos clasificados según los puntos conseguidos.

Método del Árbol y Autómata de Finito Determinista:

https://app.diagrams.net/#G1qTzw_QptR8K-IRGCMq-Atv9Jqmo25g86

Gramática Libre de Contexto Utilizada para el Análisis Sintáctico:

$\langle \text{Inicio} \rangle ::= \langle \text{RESULTADO} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{JORNADA} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{GOLES} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{TABLA} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{PARTIDOS} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{TOP} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{ADIOS} \rangle \langle \text{Repetir} \rangle$

$\langle \text{Repetir} \rangle ::= \langle \text{RESULTADO} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{JORNADA} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{GOLES} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{TABLA} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{PARTIDOS} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{TOP} \rangle \langle \text{Repetir} \rangle$

| $\langle \text{ADIOS} \rangle \langle \text{Repetir} \rangle$

| Epsilon

<RESULTADO>::= Tk_resultado Tk_cadena tk_vs tk_cadena tk_temporada tk_año
<JORNADA>::= Tk_jornada Tk_numero tk_temporada tk_año <JORNADA_SIGUIENTE>
<GOLES>::= Tk_goles Tk_condicion_gol tk_cadena tk_temporada tk_año
<TABLA>::= Tk_tabla tk_temporada tk_año <TABLA_SIGUIENTE>
<PARTIDOS>::= Tk_partidos tk_cadena tk_temporada tk_año <PARTIDOS_SIGUIENTE>
<TOP>::= Tk_top Tk_condicion_top tk_temporada tk_año <TOP_SIGUIENTE>
<ADIOS>::= Tk_adios

<JORNADA_SIGUIENTE>::= tk_guion_f tk_nombre_archivo
| Epsilon

<TABLA_SIGUIENTE>::= tk_guion_f tk_nombre_archivo
| Epsilon

<PARTIDOS_SIGUIENTE>::= tk_guion_ji tk_numero tk_guion_jf tk_numero
| tk_guion_f tk_nombre_archivo

<TOP_SIGUIENTE>::= tk_guion_n tk_numero
| Epsilon