



USAC

TRICENTENARIA

Universidad de San Carlos de Guatemala

INGENIERIA

CUNOC

Lenguajes Formales y de Programación

Práctica1: Manual Técnico

Sección: A

Nombre:

Registro académico:

Mariano Francisco Camposeco Camposeco

202030987

Quetzaltenango, 05 de octubre de 2021.

INTRODUCCIÓN

En este manual iremos directo al grano en que consiste cada clase, con palabras y un con UML, ya que para entender que se hizo por alguien que intente hacerlo nuevamente ya tenga una base y pueda guiarse con ello, por lo cual vendrá siendo una parte de lenguaje de bajo nivel comparado al manual de usuario, sin más espero le preste la suficiente atención.

OBJETIVOS


Objetivo general:

- Explicar las diferentes clases que se hicieron

Objetivos específicos:

1. Identificar paquetes
2. Usar técnicas de programación
3. Entender el porqué de la clase


PAQUETE ALMACENADOR REPORTES

 AlmacenadorReportes

Documentamos código y estructuramos posibles errores al no ingresar c...

En este paquete incluimos la creación de error, token y lexema para pasarle parámetros y así mismo las clases de estos para poder rellenar en el array list según se vayan creando.


PAQUETE ARCHIVOS

 Archivos

Documentamos código y estructuramos posibles errores al no ingresar c...

En este paquete realizamos lo que viene siendo la carga de archivos y también lo que es la exportación del propio archivo.


PAQUETE AUTOMATAS

 Automatas

Documentamos código y estructuramos posibles errores al no ingresar c...

En este paquete establecemos los automatistas que serán definidos, identificador, numero, decimal, puntuacion, agrupacion, operador, donde se encontrará el respectivo código para la detección de estos en su propia clase.


PAQUETE CARGADOR DE OBJETOS

 CargadorObjetos

Documentamos código y estructuramos posibles errores al no ingresar c...

Para el paquete cargador de objetos realizamos estructura de array list de los objetos, error, lexema, token para poder crearlos según vayan surgiendo al leer automatistas.


PAQUETE CONTROLADORES

 Controladores

Documentamos código y estructuramos posibles errores al no ingresar c...

Para este contamos con las clases de constructores que no servirán para movilizar datos de un lugar a otro con la utilización de set y get como lo son movilizadores, datos instanciadores y condicionales para codigos repetidos que este es un método de repitencia de lectura de automatistas.


PAQUETE REGISTRO TABLAS

 RegistroTablas

Mejoramos deteccion de automatas, optimizamos código para dividir en ...

Para el paquete de registro tabla, contamos con 3 clases y cada una para crear la tabla y rellenarla cuando sea el caso con el array list según corresponda que fuimos estableciendo al leer los automatas.


PAQUETE FRAMES

 frames

Documentamos código y estructuramos posibles errores al no ingresar c...

En este paquete encontramos el frame inicial que nos muestra todas las opciones que tendremos y la parte esencial del programa, como también la de búsqueda de cadena que solo emergerá cuando quedrámos localizar algo en específico.

CLASE PRINCIPAL

 PracticaLenguajes.java

Documentamos código y estructuramos posibles errores al no ingresar c...

Para la clase principal lo que hacemos es inicialización de datos y constructores, así mismo el llamado al frame inicial.

UML:

