

TECNOLÓGICO DE COSTA RICA
Escuela de Ingeniería en Computación
Proyecto de Ingeniería de Software

Profesora:
Maria Estrada

Entrega 1:
Resumen técnico
de Lenguaje Triángulo

Estudiantes:
Christian Leon Guevara - 2013371982
Gabriel Ramírez Ramírez - 201020244

Fecha de entrega:
22-12-2018

Información del proyecto	3
Resumen ejecutivo	4
Alcance de las pruebas	4
Elementos de pruebas	4
Pruebas de regresión	5
Funcionalidades a no probar	5
Enfoque de pruebas (estrategia)	6
Pruebas funcionales	6
Pruebas no funcionales	6
Criterios de aceptación o rechazo	6
Entregables	6
Recursos	7
Requerimientos de entornos – Hardware	7
Requerimientos de entornos – Software	7
Personal	7
Herramientas de pruebas requeridas	7
Entrenamiento	7
Planificación y organización	8
Matriz de responsabilidades	8
Dependencias y Riesgos	8

Información del proyecto

Proyecto	Analizador Contextual del Lenguaje Triangulo
Fecha de preparación	
Cliente	Ignacio Trejos
Gerente / Líder de proyecto	Gabriel Ramirez Ramirez
Gerente / Líder de pruebas de software	Christian Leon Guevara

Resumen ejecutivo

Este documento tratará de definir ciertos aspectos que tengan relación a proceso de aplicación de pruebas del proyecto. No están completamente definidas las pruebas pero dará una visión general de la intención del proceso. Este proyecto al ser de fin educativo no va a requerir hardware muy potente o específico, el compilador completo debe poder ser ejecutado en una computadora no muy potente.

Alcance de las pruebas

Elementos de pruebas

Teniendo en cuenta el diagrama hecho por el equipo que implementó la totalidad del compilador Triángulo, podemos determinar 3 componentes a los cuales se le realizarán las pruebas en ese proyecto.

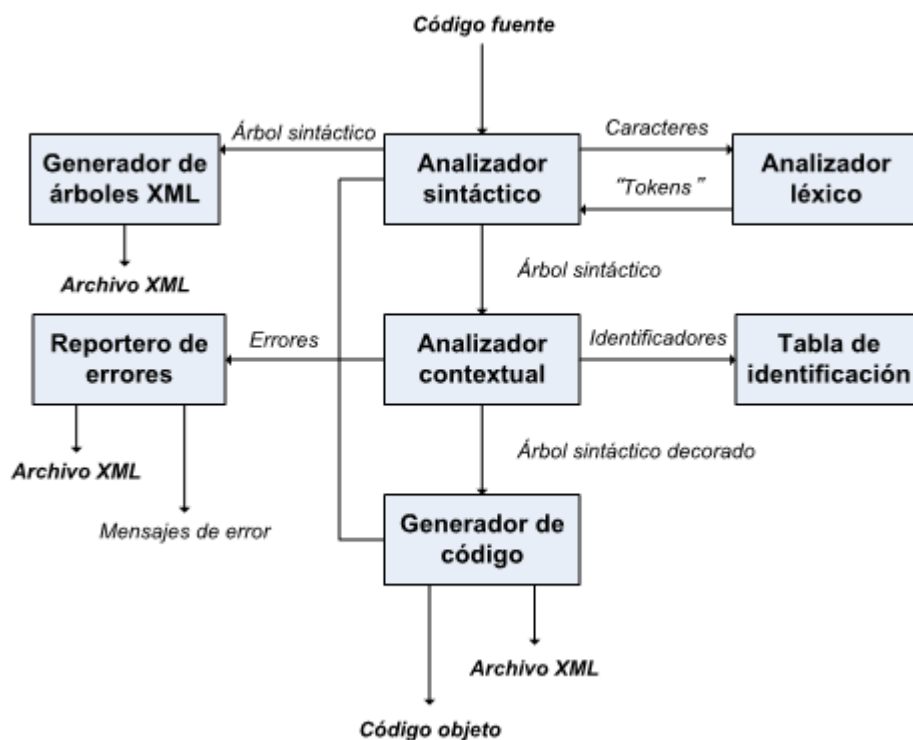


Diagrama 1

A continuación se presentan los componentes que se probarán en este proyecto:

- Analizador contextual
- Tabla de Identificación
- Reportero de errores

Se verificará que el componente de Analizador Léxico recorra el árbol sintáctico y este interactúe con la tabla de identificación para validar posibles errores de tipo o declaración.

Al final se corroborará que el Analizador contextual cuando encuentre un error lo reporte correctamente al dicho reportero.

Pruebas de regresión

Este proyecto se está realizando en paralelo con el analizador lexico y sintactico del lenguaje triángulo. El fin del cliente es poder integrar el analizador contextual con el proyecto de ellos, para ello se definió una interfaz del árbol sintáctico que es el elemento que une a estos dos módulos. Tiene que poder ser integrado a ese proyecto.

Funcionalidades a no probar

A continuación se listan las funciones que no serán probadas durante el proyecto:

- La correcta generación y consistencia del árbol sintáctico, existe actualmente una implementación de una “interfaz” del árbol sintáctico en oCaml

Enfoque de pruebas (estrategia)

Pruebas funcionales

Debido a las limitaciones con las herramientas del lenguaje Ocaml, se optara por realizar las pruebas usando OUnit para las pruebas unitarias. Estas comprenderán las pruebas de elementos como el manejo correcto de la tabla de Identificación

Pruebas no funcionales

El código al ser usado con fines educativos es de alta importancia que siga los criterios de Clean Code y algún estándar de programación.

Criterios de aceptación o rechazo

El código deberá cumplir con los siguientes estándares para ser aceptado:

- El 100% de las pruebas unitarias deben de salir exitosas
- Deberá cumplir con el estándar de codificación de
- Deberá cumplir con al menos un 85% de las reglas de declaración de variables.
- El código deberá poder correr un código de 100 líneas en menos de 3 min

Entregables

Los entregables de las pruebas serán:

- Un reporte de pruebas especificando las pruebas realizadas, su resultado y descripción en caso de que sea fallida.
- Reporte de herramienta de prueba.

Recursos

Requerimientos de entornos – Hardware

El entorno del proyecto va ser el de una computadora que tenga como mínimo las siguientes características:

- 512 MB de memoria RAM
- 500 MB de espacio de disco duro

Requerimientos de entornos – Software

La computadora donde se realizarán las pruebas debe tener lo siguiente instalado:

- El lenguaje Ocaml
- Algún editor de archivos (Sublime Text, Notepad++, ...)

Personal

Se requerirá solamente de 2 personas para realizar las pruebas:

- 1 Tester: El que realizará las pruebas en el software.
- 1 Analista: El que va analizar los datos obtenidos de las pruebas y determinar si la prueba fue exitosa o fallida.

Herramientas de pruebas requeridas

Las herramientas de prueba que se planea usar son:

- Ounit
- Ocamllint

Entrenamiento

Para poder realizar las pruebas se necesitará de entrenamiento en el uso de:

- Ounit
- Ocamllint

Planificación y organización

Matriz de responsabilidades

Actividad / Recurso	Christian Leon	Gabriel Ramirez
Ejecutar pruebas unitarias	I	R
Ejecutar pruebas de Ocamllint	I	R
Realizar reporte de los resultados	R	C

- R: Responsable
- A: Aprobador
- C: Consultado,
- I: Informado

Dependencias y Riesgos

El tiempo de prueba en cada iteración puede llegar a ser menor a 1 día debido a la cantidad de tiempo para estudiar e implementar el código es de 1 semana aproximadamente.