



# CALLCENTERLOG

— TI ASSISTANT —

**DOCUMENTACIÓN TÉCNICA.**  
**VERSIÓN: 1.0**

**CARTAGO, COSTA RICA.**

**Septiembre, 2019.**

---

# Tabla de Contenidos

<b>Participantes</b>	<b>2</b>
<b>Introducción</b>	<b>2</b>
<b>Descripción de los hechos y reglas implementadas</b>	<b>2</b>
Descripción de los hechos	2
Descripción de las reglas	4
<b>Descripción de las estructuras de datos desarrolladas</b>	<b>4</b>
<b>Descripción detallada de los algoritmos desarrollados</b>	<b>4</b>
Algoritmo para validar la oración	4
Algoritmo para decisión de soluciones	5
<b>Problemas conocidos</b>	<b>5</b>
<b>Problemas encontrados</b>	<b>5</b>
<b>Flujo del Proyecto</b>	<b>6</b>
<b>Conclusiones</b>	<b>6</b>
<b>Recomendaciones</b>	<b>7</b>
<b>Bibliografía</b>	<b>7</b>
<b>Anexos</b>	<b>7</b>

# Documentación Técnica CallCenterLog

## Participantes

Nombre	Rol
Angelo Ortiz Vega	Desarrollador, Líder de Equipo.
Jonathan Esquivel Sánchez	Desarrollador.
Iván Solís Ávila	Desarrollador.

## Introducción

CallCenterLog corresponde al Proyecto II para el curso de Lenguajes, Compiladores e Intérpretes. (CE3104), Módulo Lenguajes. El mismo consiste en la implementación de una aplicación que permita reafirmar el conocimiento del paradigma de programación lógico utilizando Prolog.

El presente tiene como objetivo el desarrollo de una aplicación que se comporte como un experto en la solución de problemas comunes de un Call Center de TI utilizando Prolog. Los Sistemas expertos, de ahora en adelante SE, son aplicaciones de cómputo que involucran experiencia no algorítmica, para resolver cierto tipo de problema. La interfaz debe ser completamente natural utilizando el lenguaje español. El usuario que presenta el problema, ingresa e informa al SE de todos los inconvenientes que tiene (hardware y software) que le impiden realizar sus tareas normalmente y finalmente puede consultar.

## Descripción de los hechos y reglas implementadas

### Descripción de los hechos

main.pl

- oracion : recibe una lista de palabras y una lista vacía y verifica si es una oración gramaticalmente correcta según la estructura establecida.
- sintagma\_nominal : recibe una lista de palabras y una lista vacía; elimina el primer sintagma nominal encontrado y devuelve el resto de las palabras.
- sintagma\_verbal: recibe una lista de palabras y una lista vacía; elimina el primer sintagma verbal encontrado y devuelve el resto de las palabras.
- sintagma\_saludo: recibe una lista de palabras y una lista vacía y verifica si estas palabras componen un saludo al programa (Ej. “hola log”).
- inicio(): inicia el programa, responde al saludo del usuario, le pregunta el nombre, lo saluda de vuelta con dicho nombre y le pregunta por el problema.
- validacion\_gramatical: valida si la oración digitada por el usuario está gramaticalmente correcta según el BNF establecido.
- input\_to\_list: obtiene lo escrito por el usuario y lo devuelve como una lista de palabras.
- input\_to\_string: obtiene lo escrito por el usuario y lo devuelve como un string.
- obtener\_causas: recibe un string con el problema digitado por el usuario y devuelve las posibles causas.
- obtener\_referencias: recibe un string con el problema digitado por el usuario y devuelve links de referencias.
- eliminar\_primeros: elimina partes innecesarias (para el reconocimiento de un problema) de una oración.
- causas: devuelve posibles causas para un problema dado.
- referencias: devuelve referencias para un problema dado.
- conversacion: comienza conversación.

#### application\_db.pl

- raiz: realiza un árbol de decisión, utilizando los hechos correspondientes a las soluciones, las causas y las preguntas para esas causas.

## Descripción de las reglas

- **saludo, despedida:** representan las palabras claves para el saludo y la despedida de una conversación.
- **nombre\_programa:** nombres del programa, utilizados en sintagma\_saludo
- **determinante\_m, determinante\_f, determinante\_n:** Determinantes principales utilizados en el lenguaje del SE.
- **sustantivo\_m, sustantivo\_f:** Principales sustantivos, que relacionan objetos, utilizado en sintagma\_nominal.
- **verbo:** contiene los principales verbos utilizados por el SE, utilizado en sintagma\_verbal
- **problema:** casos bases, son los problemas que el usuario inserta.
- **causa:** relaciona las causas con problemas.
- **solucion:** contiene un listado de soluciones para problemas, relaciona solución con problemas en el parámetro.
- **pregunta:** contiene un listado de preguntas para cada subproblema, recibe como parámetros la pregunta y la causa a la que pertenece.
- **referencia:** contiene un listado de referencias correspondientes a un determinado problema, tiene como parámetros el link de internet y el problema al que pertenece

## Descripción de las estructuras de datos desarrolladas

- **Listas:** todas las oraciones utilizadas durante el programa se manejan como listas.

## Descripción detallada de los algoritmos desarrollados

### Algoritmo para validar la oración

Este algoritmo detecta la entrada de texto del usuario y la convierte en una lista de palabras utilizando `input_to_list`. Luego de esto, se pasa esta lista a la regla oración, la cual verifica que la oración esté bien estructurada según el BNF establecido, de ser así devuelve `true` y sigue con la siguiente regla, de no ser así se envía un mensaje que le indica al usuario que la oración que escribió está gramaticalmente incorrecta y espera por la siguiente oración. Este algoritmo se utiliza cada vez que el usuario escribe alguna pregunta o consulta. Este no revisa el saludo ni el nombre.

## Algoritmo para decisión de soluciones

El SE le pide al usuario que únicamente responda con “si.” o “no.”, esto con el fin de simplificar el árbol de decisión, el SE realiza una pregunta al usuario, cuando recibe “si.”, pasa a la siguiente, caso contrario toma decisión y escribe la posible solución al problema determinado, relacionando las causas con las solución.

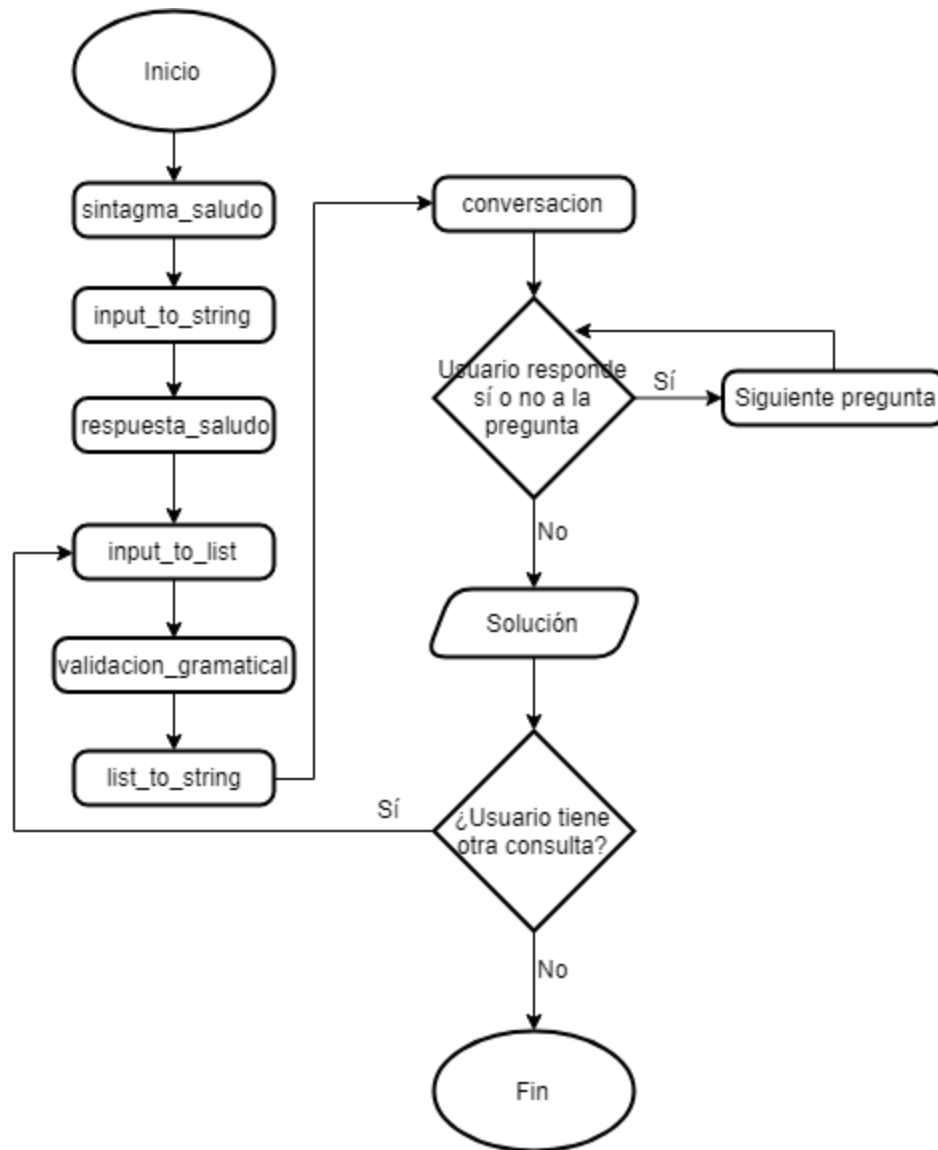
## Problemas conocidos

- Cuando se piden las referencias o las causas solo aparece una y al final de la ejecución pueden salir las que faltaron.
- Al final del programa el programa devuelve un true. o un false.

## Problemas encontrados

- Validación gramatical: Se intentó ponerle una validación para ver si le llegaba una lista vacía pero al inicio siempre le llegaba una lista vac; luego se intentó poner el input\_to\_list en diferentes lugares pero no funcionó ; se solucionó sacando completamente la función de validación\_gramatical y metiendo la oración como input.

## Flujo del Proyecto



## Conclusiones

- Para leer y recorrer con mayor facilidad cada oración se lee el texto que digita el usuario como una lista.
- Para validar la gramática de cada oración se separó cada oración en “sintagmas”, así se evalúan las estructuras posibles de una oración.

- Utilizar árboles de decisión para análisis de resultados tienden a ser precisos independientemente de si viola las suposiciones de los datos de origen.
- Los lenguajes de Programación lógica, ven el problema como una teoría, e intentan alcanzar una solución como una demostración de un teorema a partir de dicha teoría.

## Recomendaciones

- Para leer y recorrer las oraciones escritas por el usuario, es más fácil y rápido convertirlo en listas y recorrerlo como tales.
- Como las oraciones tienen diferentes estructuras gramaticales, se pueden crear hechos para recorrer cada oración como lista.
- Crear hechos genéricos que puedan servir para cualquier oración.

## Bibliografía

Monferrer, T. E., Toledo, F., & Pacheco, J. (2001). El lenguaje de programacion Prolog. Valencia.

## Anexos

Árbol de decisión para problemas.

1. Los componentes que acepta son:
  - a. Computadora
    - i. La computadora no enciende
      1. ¿Está conectada la computadora?
        - a. Si: Siguiente.
        - b. No: Conéctela. Inténtalo y pruebe de nuevo
      2. ¿Ha intentado probar en otro enchufe?
        - a. Si: Siguiente.
        - b. No: Inténtalo y pruebe de nuevo.



3. ¿Los cables están bien conectados?
  - a. Si: Lleve su computador donde un técnico.
  - b. No: Inténtalo y pruebe de nuevo
- ii. No puedo ver mis archivos.
  1. ¿Ya se aseguró de que el archivo que está buscando existe?
  2. ¿Ya se aseguró de que la unidad de almacenamiento esté bien conectada?
  3. ¿Ya instaló los controladores necesarios?
- b. Impresora
  - i. La impresora no imprime.
    1. ¿Ya revisó que tenga papel?
    2. ¿Ya revisó que tenga tinta?
    3. ¿Ya se aseguró de que esté bien configurada?
  - ii. La impresora se le atasca el papel
    1. ¿El tamaño de papel es el correcto?
      - a. Si: Siguiente
      - b. No: Utilice el tamaño correcto e intente otra vez
    2. ¿El papel está bien alineado?
      - a. Si: Siguiente.
      - b. No: Alinielo e intente otra vez
    3. ¿El número de hojas cargada en la impresora es menor a la capacidad?
      - a. Si: Siguiente
      - b. No: Disminúyala e intente de nuevo
- c. Internet
  - i. No hay conexión a internet
    1. ¿El módem está conectado?
      - a. Si: Siguiente
      - b. No: Conéctelo e intente de nuevo

2. ¿La computadora tiene habilitada la tarjeta WI-FI o está conectada por cable?
  - a. Si: Siguiente
  - b. No: habilítela o conéctela e intente otra vez
3. ¿El router está conectada al modem?
  - a. Si: Consulte con su proveedor de internet
  - b. No: Conéctelo e intente de nuevo
- ii. El internet está lento
  1. ¿El router está cerca del dispositivo que está usando?
    - a. Si: Siguiente
    - b. No: Acerque el router y el dispositivo e intente de nuevo
  2. ¿El dispositivo no tiene muchas aplicaciones que están utilizando internet simultáneamente?
    - a. Si: Siguiente
    - b. No: Cierre unas cuantas aplicaciones e intente otra vez
  3. ¿El internet es mayor a 5 MB?
    - a. Si:
    - b. No: Aumente si bando
- d. Televisor
  - i. El Televisor no enciende.
    1. ¿El televisor está conectado?
      - a. Si: Siguiente
      - b. No: conéctelo e intentelo de nuevo
    2. ¿El control remoto tiene batería?
      - a. Si: Siguiente
      - b. No: Cambielas e intentelo de nuevo
    3. ¿El interruptor de ahorro de energía del televisor está encendido?
      - a. Si: Siguiente
      - b. No: Enciendalo e intente otra vez

4. ¿Ha intentado cambiar el toma al que está conectado el televisor?
  - a. Sí: Lleve su televisor donde un técnico.
  - b. No: conéctelo e intentelo de nuevo
- ii. El televisor tiene mala imagen
  1. ¿El televisor está bien conectado ya sea a cable o a antenna?
    - a. Sí: Siguiente
    - b. No: Conéctelo bien e intentelo de nuevo
  2. ¿Su televisor tiene la actualización más reciente?
    - a. Sí: Siguiente
    - b. No: Instalela e intentelo de nuevo
  3. ¿Hay dispositivos cerca del televisor?
    - a. Sí: Alejelos e intentelo de nuevo
    - b. No: Consulte a un técnico
- e. Celular
  - i. El celular está lento.
    1. ¿Ya probó apagándolo y volviéndolo a encender?
      - a. Sí: Siguiente
      - b. No: pruebe reiniciar el celular
    2. ¿Verificó que aún tenga suficiente espacio de almacenamiento?
      - a. Sí: Siguiente
      - b. No: Borre algunas aplicaciones, fotos, videos o archivos y verifique si se arregló el problema.
    3. ¿Su celular tiene la actualización más reciente?
      - a. Sí: Consulte a un técnico
      - b. No: Busque en internet la última versión y revise si es compatible con su celular. De ser así pruebe instalarla y verifique si se arregló el problema.

Definición y Relación entre hechos.

- problema (A).
- causa(B,A).
- raiz(B,A).
- solucion(C,B).
- pregunta(D,B)
- referencia(E,A).