

# **Instituto Tecnológico de Costa Rica**

Administración de Tecnologías de Información

Lenguajes de Programación

Profesor: Andréi Fuentes Leiva



## **TAREA PROGRAMADA 2**

*Documentación Técnica*

***Víctor Vargas Ramírez***

***José Pablo Matamoros Moya***

**Tabla de Contenido**

Descripción Corta del Programa.....3

Funcionamiento del Programa.....3

Diseño del Programa.....4

Análisis de Resultados.....4

    Objetivos alcanzados.....4

    Objetivos no alcanzados.....4

Conclusiones Personales.....5

## Descripción Corta del Programa

La tarea programada consta de un sistema automatizado para administrar la distribución de regalos que debe hacer Santa para la navidad ya que es una tarea complicada, por el simple hecho de que son demasiados niños alrededor del mundo y no todos se comportan como deben por lo que no se les puede enviar los regalos que ellos solicitan.

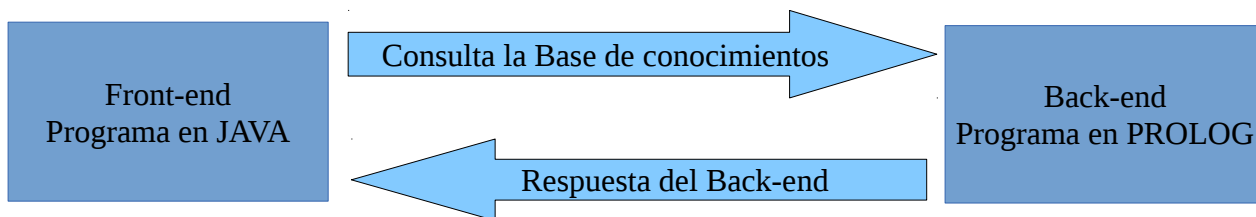
El programa debe ser lo suficiente funcional como para administrar cuales son los niños que deben y los que no obtener regalos entre otras funciones, para así simplificar las tareas que santa debe llevar acabo, las funciones serán especificadas a continuación:

- 1. Lista de niños malos y niños buenos:** buenos = más acciones buenas que malas, malos = más acciones malas que buenas.
- 2. Lista de niños que tienen una acción buena en particular:** El usuario puede escribir una acción y el sistema lo busca en la lista de acciones buenas de los niños.
- 3. Lista de niños que tienen una acción mala en particular:** El usuario puede escribir una acción y el sistema lo busca en la lista de acciones malas de los niños.
- 4. Lista de regalos más solicitados por los niños:** Ordenados por popularidad.
- 5. Consulta niño-regalos:** El usuario digita el nombre de un niño, y el sistema despliega la información de regalos que quiere ese niño, con sus respectivos datos (precio, marca). Si el niño tiene más acciones malas que buenas, el sistema deberá indicar que al niño se le tienen que dar medias de golfista .
- 6. Consulta niño-regalos posibles:** Igual que la consulta anterior, pero en este caso, se filtran solo los regalos que sean válidos para la edad del niño (usando la edad del niño y la edad mínima para el juguete), y filtrando los juguetes que están fuera del presupuesto para ese niño (en lo que el precio del regalo es mayor al presupuesto asignado por Santa para ese niño). Si el niño tiene más acciones malas que buenas, el sistema deberá indicar que al niño se le tienen que dar medias de golfista .
- 7. Presupuesto total:** Deberán mostrar una lista de todos los regalos que se les deben dar a los niños buenos, tomando en cuenta las consideraciones de edad y de precio especificadas en el punto anterior. Además, deberán mostrar el monto total que deberá invertir Santa para poder comprar todos los regalos.

## Funcionamiento del Programa

El programa estará dividido en dos partes:

- 1. Front-End:** Programa escrito en Java el cual brindará la interfaz gráfica necesaria para hacer las consultas pertinentes al sistema.
- 2. Back-End:** Programa escrito en PROLOG el cual almacenará la base de conocimientos para así brindar los datos solicitados por el Front-End (Programa en Java).



## Diseño del Programa

La base de conocimientos está ubicada en un archivo de tipo .pl de PROLOG, el cual consta de los predicados que son definidos anteriormente y contienen las reglas para así generar las consultas necesarias, los algoritmos fueron creados y escritos bajo métodos de recursividad de PROLOG para así retornar las respuestas correctas a las consultas realizadas.

Para la comunicación de JAVA con PROLOG utilizamos la librería **tuprolog**, la cual puede ser ubicada en la siguiente dirección: <http://apice.unibo.it/xwiki/bin/view/Tuprolog/> esta se encarga de crear un motor de inferencia de Prolog, únicamente que lo hace desde JAVA, simulando así el comportamiento de Prolog.

La librería nos permite mediante cierto tipo de métodos ya establecidos, hacer las consultas y los cálculos correspondientes sin necesidad de agregar o instalar programas extras al nuestro para poder ser ejecutado.



Decidimos realizarlo de esta forma ya que muchos métodos estudiados implicaban instalar paquetes extra y modificar variables de entorno de Java por lo que se complicaba de una manera significativa el trabajo y el uso del programa en los demás dispositivos donde sea ejecutado.

El programa cuenta con dos modalidades principales:

**1. Mantenimiento de Datos:** Este este módulo los encargados del taller de santa podrán ingresar los datos de los niños, edad, país, lista de buenas acciones, lista de malas acciones, lista de regalos deseados (wishlist) y presupuesto de santa para el niño. Además se contará con la información de regalos o juguetes, para los cuales se contará con el nombre, marca, precio, edad necesaria para usarlo (Edad mínima de uso para el juguete).

**2. Modalidad de Consulta:** Los encargados del taller podrán hacer consultas con bases de ciertos atributos de niños y juguetes, el sistema deberá mostrar la información que cumpla con los atributos ingresados.

## Análisis de Resultados

### Objetivos alcanzados

La Base de conocimientos de Prolog fue programada en su totalidad, cuenta con todas las funciones especificadas y solicitadas para el programa, esta base de conocimientos o Back-end del programa cuenta con las reglas y hechos que definen el funcionamiento de la aplicación, al ser ejecutada la base de conocimientos directamente desde Prolog podemos realizar las consultas que se necesiten y además obtener lo resultados necesarios que se le solicitan a las aplicación.

Esta base de conocimientos puede ser chequeada desde la consola de linux, ejecutando prolog y cargando la base de conocimientos en el programa para realizar las pruebas que sean necesarias para comprobar su correcto funcionamiento.

### Objetivos no alcanzados

Se tienen problemas a la hora de hacer ciertas consultas al motor de inferencia de Prolog creado en java con las respectivas librerías, ya que algunas de estas no pueden ser resueltas, aunque directamente desde Prolog si es posible, se hace la investigaron para determinar que causa dichos errores de ejecución pero no se tiene éxito a la hora de encontrar con que tipo de problema estamos enfrentando ya que esto depende mucho del código de las librerías los cuales en su caso se cree deben ser modificados para lograr el funcionamiento del programa, por ejemplo al hacer la consulta desde Prolog directamente de la lista record de regalos solicitados a Santa esta tiene éxito y nos

devuelve la lista completamente ordenada y con cada regalo y la cantidad que se han solicitado, pero al hacerlo desde Java la consulta no tiene éxito, esto de igual forma no es con todas las consultas pero si con algunas necesarias para el correcto funcionamiento de la aplicación.

## **Conclusiones Personales**

Podemos comprender mediante esta investigación cuales formas existen de conectar un lenguaje de programación con otro, aunque estos tengan diferente paradigma y trabajen de forma distinta internamente.

Existen varios métodos para poder unificar estos lenguajes trabajando de forma conjunta, utilizando métodos y características especiales de cada uno de ellos, los métodos convencionales son muy útiles para cuando se necesita una programación específica de funciones que son mejor tratadas con estos lenguajes de programación y paradigmas.

Con esta investigación podemos comprender como funcionan estos métodos y como se puede lograr crear sistemas basados en múltiples lenguajes de programación, aumentando así los niveles funcionales de las aplicaciones y creando programas con características al gusto de los clientes que lo soliciten.

Podemos ver la importancia de conocer estos métodos en la funcionalidad que nos brindan a la hora de crear aplicaciones para las organizaciones o clientes específicos, encontramos una forma de miles que existen de conectar distintos paradigmas y con esto logramos comprender la importancia de conocer estas características y posibilidades que existen en el mundo de la programación de aplicaciones.