

**UNIVERSIDAD DE SANTIAGO DE CHILE  
FACULTAD DE INGENIERÍA  
DEPARTAMENTO DE INGENIERÍA INFORMÁTICA**



**LABORATORIO 2  
PARADIGMAS DE LA PROGRAMACIÓN**

**BASTIÁN GONZALO VERA PALACIOS**

Profesor: Roberto González

Santiago - Chile  
1-2018



## Tabla de Contenido

CAPÍTULO 1.	5
CAPÍTULO 2.	6
2.1 CLÁUSULAS	6
2.2 HECHOS	6
2.3 REGLAS	6
2.4 RECURSIVIDAD	6
2.5 TDA	6
CAPÍTULO 3. DESCRIPCIÓN DEL PROBLEMA	7
CAPÍTULO 4. DESCRIPCIÓN DE LA SOLUCIÓN	8
CAPÍTULO 5. RESULTADOS	10
CAPÍTULO 6. CONCLUSIONES	11

## Índice de Figuras

Figura 1:Estructura del Chatbot	8
Figura 2: Constructor TDA	<b>¡Error! Marcador no definido.</b>
Figura 3 Salida del Programa	10

# **CAPÍTULO 1. INTRODUCCIÓN**

Un chatbot es un mecanismo el cual permite generar una conversación con un usuario con un propósito, el cual puede ser desde ofrecer un servicio hasta resolver consultas sobre algún tema en particular.

## **1.1 DESCRIPCIÓN DEL PROBLEMA**

Para poder resolver este problema se tendrá que desarrollar un método el cual pueda analizar el mensaje entregado por el usuario, encontrar los principales conceptos dentro de él, y posteriormente entregar la respuesta respectiva a la consulta realizada, todo esto hasta que el usuario decida terminar la conversación y entregar un puntaje respecto a la calidad de ésta.

## **1.2 DESCRIPCIÓN DE LA SOLUCIÓN**

Dentro de los principales paradigmas para los distintos tipos de problemas que se puede encontrar, en este informe se hará referencia al pensamiento y la resolución del problema señalado en el laboratorio 2, el cual se debe realizar a través del paradigma Lógico con el lenguaje de programación Prolog.

Dado lo mencionado anteriormente, se plantea como objetivo de este laboratorio, poder aplicar e implementar todas las técnicas y métodos aprendidos en clases, junto a la investigación de posibles recursos que presente el lenguaje y el paradigma para facilitar así el desarrollo y el trabajo en este.

## **CAPÍTULO 2. MARCO TEÓRICO**

### **2.1 CLÁUSULAS**

Representaciones de todo el conocimiento necesario para resolver el problema, tales como reglas, hechos o consultas.

### **2.2 HECHOS**

Mecanismo básico para la representación de objetos, propiedades o relaciones de este.

### **2.3 REGLAS**

Permiten establecer relaciones más elaboradas entre objetos, por ejemplo, relaciones generalizadas o particularizadas o relaciones causa-efecto.

### **2.4 RECURSIVIDAD**

La recursividad es una herramienta que consiste en un proceso basado en su propia definición, realizando un autollamado a sí mismo. Es un recurso fuertemente empleado en este lenguaje de programación, ya que genera eficiencia a la hora de realizar un mismo llamado a una función.

### **2.5 TDA**

Los tipos de datos abstractos son una herramienta que facilita la administración de ciertos valores los cuales podemos trabajar con características propias a nuestra implementación, facilitando el uso de estas durante el desarrollo de el trabajo.

## CAPÍTULO 3. DESCRIPCIÓN DEL PROBLEMA

Se solicita la creación de un Chatbot, el cual corresponde a un sistema de respuestas automáticas según la lectura de un mensaje ingresado por el usuario, logrando así una conversación fluida en la cual el usuario pueda lograr desde recibir información hasta solicitar algún servicio, entre otras cosas.

Dentro de los principales requisitos solicitados para la creación y el funcionamiento de este programa se encuentran:

- **Log**: Registro el cual contendrá la conversación que se desarrolle entre el usuario y el chatbot.
- **beginDialog** : Predicado que indica el comienzo de la conversación y genera un mensaje de bienvenida hacia el usuario.
- **endDialog** : Predicado que indica el término de la conversación entregando un mensaje de término hacia el usuario.
- **sendMessage**: Predicado en la cual el usuario entrega un mensaje junto con ciertos valores establecidos.
- **test**: Predicado encargado de realizar una prueba respecto al funcionamiento del chatbot y sus respuestas.

Estas funcionalidades son la base del planteamiento del chatbot. Dentro de los otros requisitos para el desarrollo de este chatbot, se encuentran:

- el uso de TDAs, los cuales nos facilitará el manejo de nuestra información y su implementación dentro de los predicados de nuestra solución.
- La implementación de una personalidad propia para el chatbot, con distintos tipos de respuestas según se plantee la situación.

Toda la ejecución de el programa se realizará desde la consola del programa, por ende no se necesita ningún elemento externo para la ejecución del código.

## CAPÍTULO 4. DESCRIPCIÓN DE LA SOLUCIÓN

La primera etapa para el desarrollo del chatbot, fue establecer las características propias, tales como sus respuestas y definirlas dentro de un hecho llamado estructura, el cual contiene el nombre del chatbot y una lista de listas con sus respuestas.

```

estructura(chatbot,[
  ["Hola! Bienvenido a su asistente de salud virtual, ¿Me podrias decir tu nombre? ",
    "Bienvenido! me llaman asistente de salud virtual y tu? ¿como te llamas? ",
    "Estas hablando con su asistente de salud virtual ¿Cual es tu nombre?"],
  ["¿presenta usted alguna molestia o malestar?",
    "¿con que puedo ayudarle esta vez respecto a su salud? ",
    "¿que problema ha sentido ultimamente?"],
  ["Hasta luego, Cuídense y ante cualquier duda puede volver a consultar :)",
    "Gracias por consultar y espero que te mejores :), hasta pronto",
    "Hasta pronto! espero haber sido de gran ayuda :)!"],
  ["calor", "cansado", "mocos", "estomago", "no"],
  ["es probable que tengas fiebre, te recomiendo usar paños frios para reducir la temperatura",
    "por lo que me cuentas puede que te falten vitaminas y minerales en tu cuerpo",
    "por lo que me cuentas puedo decir que te encuentras con un resfriado comun, por lo que necesitas descansar, hidratarte y un tapsin en la noche",
    "es probable que tengas gastroenteritis, así que tendras que cuidar tu alimentacion",
    "Me alegra haber resuelto tus dudas :D"]
]).

```

*Figura 1:Estructura del Chatbot*

Posterior a la estructura, se genera un TDA para poder facilitar la implementación y el uso de ciertos datos. En este caso, se plantea como TDA un modelo que consiste en una lista de tres elementos, los cuales representan la fecha/hora, el usuario perteneciente al diálogo y el mensaje, respectivamente. Junto con el Constructor, se generan el resto de funciones necesarias para una correcta implementación de los TDA, tales como la función de pertenencia, selectores y modificadores.

```

datos(Usuario, Mensaje, TDA) :-
    dateTimeFormat(Value),
    TDA = [Value,Usuario, Mensaje].

```

Una vez creadas y definidas las estructuras sobre las cuales se va a desarrollar el chatbot, se implementan los predicados *beginDialog*, *endDialog* y *sendMessage* para empezar a generar las instancias de diálogo.

*BeginDialog* consiste en una predicado que recibe un InputLog vacío, correspondiente a una lista, y sobre esta genera un mensaje que contiene una etiqueta la cual indica el inicio de la conversación, los valores actuales de tiempo, el emisor del mensaje y una bienvenida hacia el usuario, los cuales son ingresados a un OuputLog y retornados al final de la función.

De manera semejante, el predicado *endDialog* recibe como InputLog al Log modificado de el último predicado que lo utiliza y le agrega un mensaje de despedida al usuario, junto con una etiqueta al final, que indica el fin de la conversación.



El predicado *sendMessage* es la que realiza la mayor cantidad de predicados extra, debido a que recibe el mensaje de el usuario, realiza una primera evaluación para así poder obtener el nombre del usuario y realizar así respuestas más personalizadas.

Posterior a esto recurre a otro predicado llamado *intersection*, la que gracias a la técnica de recursión, realiza una comparación entre el mensaje de el usuario separado por el predicado *split\_string* y la lista de posibles respuestas en la información que posee el chatbot con el fin de poder señalar si se puede realizar la respuesta o no. En caso de que si halle una respuesta válida para el usuario, se obtiene la posición en la lista de el elemento y se utiliza para entregar la respuesta al elemento la cual se encuentra en otra lista pero con la misma posición. Esto se agrega a nuestra variable *OutputLog* la cual posee el Log con las respectivas modificaciones.

Durante toda esta selección de qué mensaje se le entrega al usuario, aparece el uso del valor *seed*, que posterior a ser entregado por el usuario, se modifica con el fin de obtener uno de los valores dentro de las posibles soluciones planteadas.

*SendMessage* puede ser llamada cuantas veces el usuario quiera realizar consultas respecto a la temática del chatbot.

Una vez que el usuario decida terminar con la conversación, se emplea el predicado *endDialog* que genera el mensaje de despedida y retorna el *Outputlog* con toda la conversación generada.

Para finalizar se emplea la función *test*, la cual consiste en una función recursiva y anidada que se utiliza como herramienta para realizar una prueba del funcionamiento e implementación de todas las funciones anteriores. Para su funcionamiento se entrega como parámetro un usuario de prueba (*user1*, *user2*, ...) los cuales simulan respuestas de usuarios, evaluadas con el chatbot.

## CAPÍTULO 5. RESULTADOS

Como resultado de la implementación mencionada anteriormente, se obtuvo un chatbot capaz de responder correctamente a cada una de las inquietudes solicitadas por el usuario y a la vez manteniendo la lógica de las conversaciones. Todas las funciones cumplen su funcionamiento de manera óptima respecto al paradigma.

Se adjunta un ejemplo de la salida del programa posterior a la aplicación del predicado *logToStr*.

```
BEGIN_DIALOG
0
21/5/2018 21:37
chatbot
Bienvenido! me llaman asistente de salud virtual y tu? ¿como te llamas?
21/5/2018 21:37
usuario
user2
21/5/2018 21:37
chatbot
Hola user2 ¿con que puedo ayudarle esta vez respecto a su salud?
21/5/2018 21:37
usuario
tengo demasiada calor
21/5/2018 21:37
chatbot
es probable que tengas fiebre, te recomiendo usar paños frios para reducir la temperatura
21/5/2018 21:37
chatbot
¿Tiene alguna otra duda?
21/5/2018 21:37
usuario
llego en la noche a mi casa muy cansado
21/5/2018 21:37
chatbot
por lo que me cuentas puede que te falten vitaminas y minerales en tu cuerpo
21/5/2018 21:37
chatbot
¿Tiene alguna otra duda?
21/5/2018 21:37
usuario
no
21/5/2018 21:37
chatbot
Me alegra haber resuelto tus dudas :D
21/5/2018 21:37
chatbot
¿Tiene alguna otra duda?
21/5/2018 21:37
chatbot
Gracias por consultar y espero que te mejores :), hasta pronto
21/5/2018 21:37
END_DIALOG
0
```

*Figura 3 Salida del Programa*

Debido a inconvenientes con el tiempo durante el desarrollo del laboratorio, no se logró la implementación de predicados como *possibleResponses* ni *chatToXml*, entre otros.

## **CAPÍTULO 6. CONCLUSIONES**

El paradigma Lógico es una potente herramienta gracias a el uso de la lógica para el planteamiento de problemas y control sobre reglas de inferencia para encontrar una solución automática.

Respecto a la resolución del laboratorio, se torno interesante el hecho de tener que pensar en cada uno de los predicados generados según la veracidad que posea respecto a los hechos que hayamos desarrollado, ya que no se está acostumbrado a realizar ese ejercicio. Este laboratorio fomenta el análisis de cuándo utilizar este pensamiento se vuelve más útil que pensar las cosas de manera lineal, como en búsqueda de valores válidos para una entrada establecida, y cuando se vuelve más tedioso con respecto a otros paradigmas donde pequeñas cosas van mucho más a lo que uno necesita realizar.

Finalmente, como se ha podido realizar de manera satisfactoria la base del chatbot y lograr su funcionamiento de manera correcta, es posible concluir que se logró cumplir en gran parte el objetivo principal del laboratorio.