

*Instituto Tecnológico de Costa Rica*

*Unidad de Computación*

Tarea Programada lll

*Danny Rojas Arguedas*

*Sede San Carlos*

*17/05/2019*

# Introducción

En el siguiente trabajo se analiza la creación de un juego de culebra (Snake) de estilo laberinto, en el cual dicha culebra tendrá que comer algunas manzanas puestas aleatoriamente en el campo de juego, esto la hará crecer, y una vez comidas todas las manzanas esta acabará su juego. La culebra tendrá que desplazarse desde su lugar actual hasta encontrarse con cada manzana, la ruta que esta tomará será calculada en el lenguaje de programación Prolog, en el cual se hará llamada y este devolverá la ruta más corta para llegar a dicha manzana.

La parte gráfica de desarrolla en el lenguaje de programación Java, el cual se encargará de realizar el desplazo de la culebra en la ruta indicada por Prolog. El usuario tendrá la posibilidad de escoger entre tres mapas, los cuales fueron creados por defecto por el programa al iniciar la sesión.

# Análisis del problema

El problema por resolver es la confección de un juego de culebra, en este caso este contará con obstáculos tipo paredes que le impedirán desplazarse por todo el mapa de manera libre. El juego deberá constar de al menos tres mapas con obstáculos ya creados, estos para ser escogidos por el usuario para que de inicio el juego de la culebra, también se le deberá solicitar al usuario la cantidad de manzanas que este desea colocar en el mapa, estas posteriormente serán comidas por la culebra para poder finalizar el juego.

El mapa debe ser creado inteligentemente, al usuario seleccionar un mapa, este deberá crear las conexiones necesarias entre los nodos posibles para que la culebra pueda realizar movimiento entre el mapa. Cuando se cree el mapa este deberá construir las conexiones del mismo en Prolog, por lo que deberá llamar una regla en Prolog que realice dicha acción.

Cuando se inicie el juego, la culebra deberá guiarse o dirigirse hacia las diferentes manzanas que estarán en el mapa, esto buscando mediante Prolog la ruta más corta para llegar a cada X manzana que haya elegido.

# Solución del problema

La interfaz gráfico del juego fue desarrollada en el lenguaje de programación Java, el cual mediante los diferentes componentes que contiene facilita dicha acción. El juego contiene un apartado que pedirá al usuario la escogencia entre tres mapas ya creados, en el momento que el usuario seleccionó alguno de los tres mapas este creará sus respectivas conexiones lógicas en Prolog, cada unos de los mapas son generados mediante una matriz 6x6 de botones. Estos botones son instancias de la clase llamada “Cuadro” la cuál tiene los atributos de un “JButton”.

Para la solución del problema expuesto se utilizó el lenguaje de programación funcional “Clojure” en conjunto con la librería Seesaw para la implementación gráfica, esto con el fin de facilitar el uso del sistema por el usuario final.

Para la creación de encuestas se optó por mantener una lista que es trasladada por parámetros para mantener las encuestas creadas, ya que pueden ser muchas encuestas. Estas encuestas contienen preguntas, de las cuales como se mencionó anteriormente son de dos tipos: Selección única, para este tipo se generó la siguiente estructura de lista (cuerpoPregunta (opcionesPregunta) “tipo1”), la cual almacena la pregunta introducida por el usuario, una respectiva lista con las posibles opciones y también el tipo de pregunta para estar ser reconocida posteriormente, el siguiente tipo de pregunta es por Escala, la cual es un poco más estructurada a la hora de ser respondida, esta consta de la siguiente estructura (cuerpoPregunta (itemsPregunta) “tipo2”), la cual al igual que la anterior contiene una pregunta general, pero esta vez contiene ítems que al ser respondidos serán evaluados con la escala (Muy bien, Bien, Regular, Malo Muy malo); la estructura de la encuesta es (nombreEncuesta (listaPreguntas)), esta lista de preguntas es la que contiene todas las creadas de los tipos mencionados anteriormente.

A la hora de responder las preguntas se pregunta cuántas veces se desea contestar la encuesta que anteriormente ya seleccionó el usuario, esto provocará que recorra cada pregunta de la encuesta y vuelva a iniciar la cantidad de veces que se seleccionó. También las preguntas serán mostradas de dos formas de acuerdo a su tipo, si es de tipo1 (selección única) solo se muestra el cuerpo de la pregunta y las opciones en un combobox, el usuario solamente tendrá que seleccionar la opción que le parezca y dará siguiente, si es de tipo 2 (escala) se hará un recorrido por los ítems de esa pregunta y se preguntará con la escala mencionada anteriormente. Las respuestas de estas preguntas serán almacenadas en una lista de respuestas que después será consultada para realizar las estadísticas.

# Análisis de resultados

|  |  |  |
| --- | --- | --- |
| ***Tarea/Requerimiento*** | ***Estado*** | ***Observaciones*** |
| *Crear y guardado de encuestas* | *100%* |  |
| *Creación y guardado de Preguntas de Selección Única* | *100%* |  |
| *Creación y guardado de Preguntas de Escala* | *100%* |  |
| *Responder las Preguntas de Selección Única* | *100%* |  |
| *Responder las Preguntas de Escala* | *100%* |  |
| *Almacenar las respuestas de las preguntas* | *100%* |  |
| *Uso de función lambda y map* | *100%* |  |
| *Mostrar las estadísticas* | *50%* | *Guarda todas las respuestas en una lista que se puede imprimir al final pero no muestras las estadísticas* |

Manual de Usuario

Al iniciar la aplicación esta mostrará la pantalla de inicio en la cual se le pide al usuario la acción que desea realizar, entre estas se encuentran: crear encuestas, responder encuestas y una opción opcional que permite visualizar la lista creada de encuestas en la consola del IDE, esta se puede ver en la figura 1.

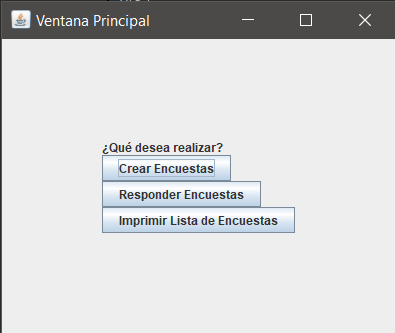


figura 1

Seguidamente si el usuario selecciona crear encuestas ahí se le mostrará otra ventana en la cuál se le solicita el nombre de la encuesta que desea crear, un botón para proceder y el botón volver para regresar a la ventana principal, esto se muestra en la figura 2.

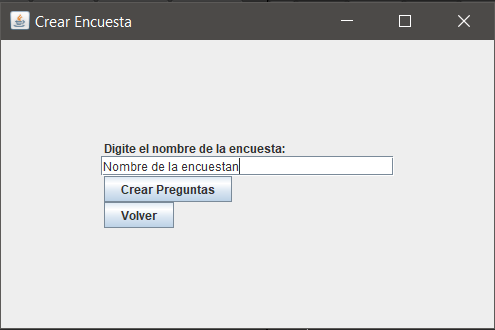


figura 2

Al usuario proceder a crear la encuesta se le mostrará otra ventana (figura 3) en la cual se le muestran los dos diferentes tipos de pregunta que se le pueden añadir a la encuesta, el botón listo para terminar de crear la encuesta y un botón adicional que imprime en consola la lista de preguntas que se van creando.

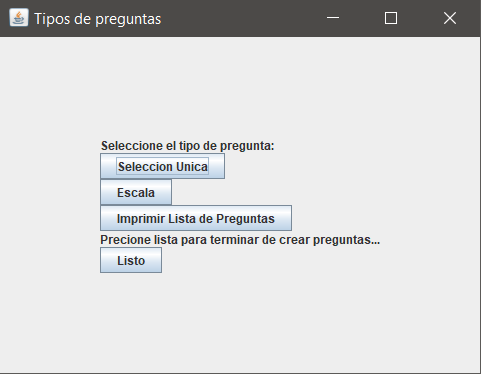
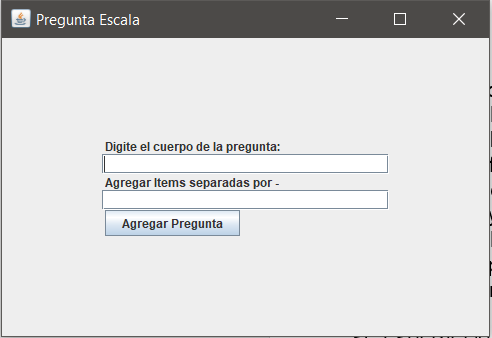
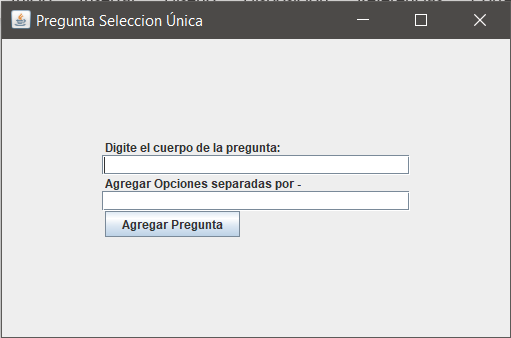


figura 3

Al seleccionar una de las dos opciones de pregunta se dirigirá hacia las ventanas que se encuentran en la figura 4 y 5. Estas reciben el cuerpo de la pregunta y las respectivas opciones separadas por un - para Selección Única y los respectivos ítems separados igualmente por un guión para la de Escala.

figura 4 figura 5

Al finalizar de escribir la pregunta y los respectivos ítems o opciones, el usuario deberá presionar “Agregar pregunta”, lo cual lo dirigirá la la pantalla de la figura 3, en la que puede nuevamente crear preguntas las veces que quiera y del tipo que quiera.

Al terminar con la creación de preguntas y haber terminado con la encuesta presionando “Listo”. Esta volverá a la pantalla inicial (figura 1), en la cual puedes responder encuestas, al realizar esta acción aparecerá una ventana que pregunta en un combobox cuál encuesta desea contestar, y también solicita el número de personas o cantidad de veces que desea responder la encuesta, se muestra en la figura 6

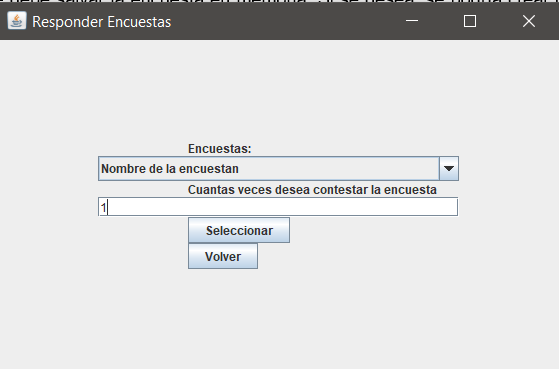
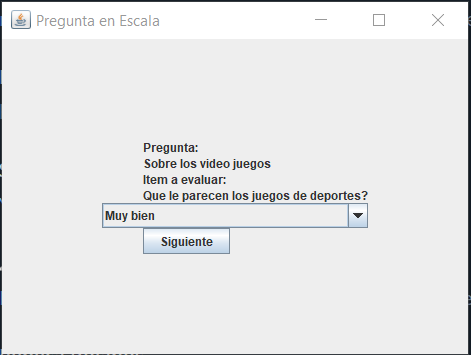
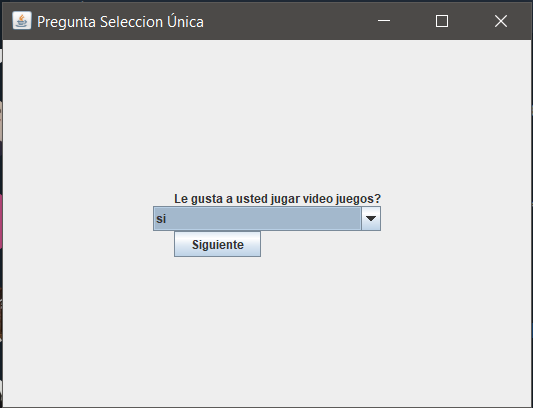


figura 6

figura 7 figura 8

Para continuar al presionar seleccionar, se le mostrarán todas las preguntas de cada encuesta, sean el tipo que sean figura 7 y 8. Y esto las veces que se describió al inicio que se quería contestar la encuesta.

Acabando en la ventana que debería ser de estadísticas en la cual se muestra un botón que imprime todas las respuestas ordenadas en una lista, lamentablemente no se muestran las estadísticas en pantalla figura 9.

# figura 9

# 

# 

# 

# 

# Conclusiones

En conclusión, el uso del paradigma funcional facilita de gran manera el uso de estructuras como lo son las listas, ya que la información que almacenan estas al inicio se puede recuperar ya que esta es inmutable. La implementación de las características de la programación funcional no solo se pueden aplicar en el lenguaje de programación “Clojure”, existen gran variedad de lenguajes que permiten el uso de las mismas.

# Referencias

Conectar prolog con Java paso a paso - YouTube. (n.d.). Retrieved October 27, 2018, from <https://www.youtube.com/watch?v=pJ42DK7FMOo&t=>