



Área de ingeniería en computadores

Lenguajes, compiladores e intérpretes

Manual de Usuario Wazelog

PROFESOR

Marco Rivera Menénes

ESTUDIANTES

Sebastián Mora Godínez 2019227554 sebastianmg@estudiantec.cr

Emanuel Marín Gutiérrez 2019067500 emarin702740530@estudiantec.cr

FECHA

9 de abril, 2021

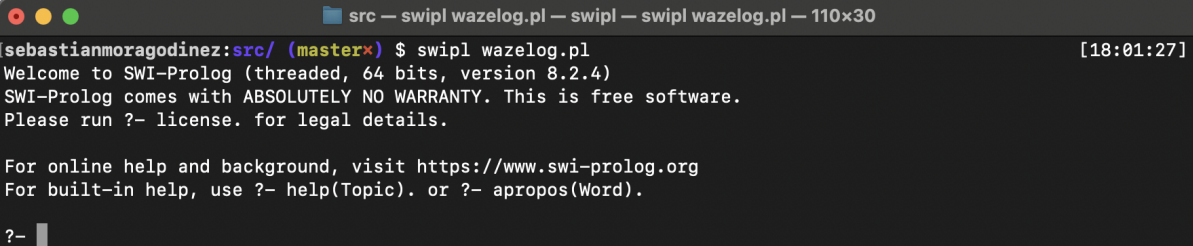
Manual de usuario:

Para la correcta utilización del juego se necesita cumplir con una serie de requerimientos de Hardware y Software que se detallan a continuación:

Se recomienda tener como mínimo la versión 8.2.4 de Prolog instalada para la correcta ejecución de la aplicación. Además, se recomienda utilizar un editor que contenga su propia terminal, o bien, utilizar la propia consola del sistema operativo. Por otra parte, se aclara que el desarrollo del juego se realizó en la plataforma de Linux, específicamente la distribución de Ubuntu en su versión 20.04 LTS, por lo que al correr la aplicación en otra plataforma u distribución podría causar problemas en la ejecución.

Instalación y flujo de Wazelog:

Una vez descargado el proyecto “Wazelog”, se debe descomprimir la carpeta y abrirla en algún editor, o bien, desde la consola del sistema operativo. Ahora se debe buscar un archivo con el nombre wazelog.pl y en la ventana de comandos del editor o desde la propia consola, se debe ejecutar el siguiente comando:



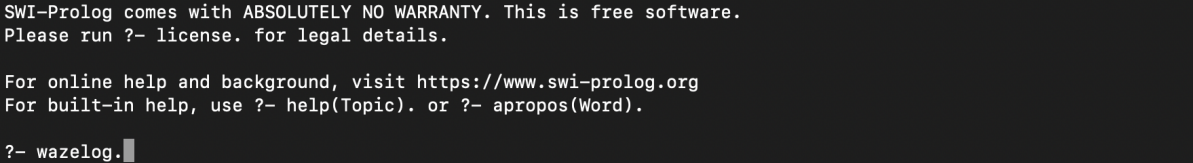
```
src — swipl wazelog.pl — swipl — swipl wazelog.pl — 110x30
sebastianmoragodinez:src/ (master*) $ swipl wazelog.pl [18:01:27]
Welcome to SWI-Prolog (threaded, 64 bits, version 8.2.4)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- 
```

Figura 1. Ejecución del archivo principal de la aplicación.

Una vez ejecutado este comando, se procede a llamar la cláusula con el nombre **wazelog**, como se muestra a continuación:



```
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- wazelog.
```

Figura 2. Ejecución de la cláusula principal de la aplicación.

La ejecución de la cláusula de la figura 2. comenzará la aplicación. Se mostrará una interfaz, donde se recibe al usuario con un saludo y se le pregunta cuál es su lugar de origen.

```
?- wazelog.  
Wazelog: Bienvenido a Wazelog, la mejor lógica para llegar a su destino  
Wazelog: Por favor indique donde se encuentra  
Usuario: █
```

Figura 3. Comienza con la interfaz de wazelog.

En este momento el usuario deberá ingresar una respuesta que cuente con la gramática establecida en el programa. A continuación se detalla la estructura de posibles oraciones permitidas:

- **me encuentro en <lugar>**
- **estoy en <lugar>**
- **estamos en <lugar>**
- **estoy ubicado en <lugar>**
- **estamos ubicados en <lugar>**
- **ellos estan en <lugar>**
- **el esta en <lugar>**

```
?- wazelog.  
Wazelog: Bienvenido a Wazelog, la mejor lógica para llegar a su destino  
Wazelog: Por favor indique donde se encuentra  
Usuario: estoy ubicado en corralillo █
```

Figura 4. Ejemplo de una oración dada por el usuario con la correcta gramática.

Una vez ingresado el lugar de origen, Wazelog pregunta por los destinos intermedios. El usuario primeramente debe ingresar una respuesta donde se incluya el nombre del establecimiento al que desea pasar. A continuación se detallan posibles oraciones permitidas:

- **quiero pasar al <establecimiento>**
- **quiero pasar a el/la <establecimiento>**
- **tengo que pasar al <establecimiento>**
- **tengo que pasar a el/la <establecimiento>**
- **quiero ir al <establecimiento>**
- **quiero ir a el/la <establecimiento>**
- **yo tengo que pasar al <establecimiento>**
- **yo tengo que pasar a el/la <establecimiento>**

- ella/el tiene que pasar al <establecimiento>
- ella/el tiene que pasar a el <establecimiento>
- ellos tienen que pasar al/el <establecimiento>
- ellas tienen que pasar al/el <establecimiento>
- nosotros tenemos que pasar al/el <establecimiento>
- nosotros tenemos que pasar a el/la <establecimiento>

```
?- wazelog.
Wazelog: Bienvenido a Wazelog, la mejor lógica para llegar a su destino
Wazelog: Por favor indique donde se encuentra
Usuario: estoy ubicado en corralillo
Wazelog: ¿Tiene algún destino intermedio?
Usuario: quiero pasar al supermercado
```

Figura 5. Ejemplo de una oración dada por el usuario para establecer que tiene un destino intermedio.

Después de ingresar alguna de las respuestas anteriormente mencionadas, Wazelog le preguntará por la ubicación de ese establecimiento. A continuación se detallan oraciones permitidas para esta parte:

- **se ubica en <lugar>**
- **esta ubicado en <lugar>**
- **se encuentra en <lugar>**
- **esta en <lugar>**

En el momento en que el usuario ingresa su respuesta, Wazelog le informará que el destino intermedio fue agregado, siempre y cuando la respuesta se encuentre dentro de la gramática establecida y el lugar efectivamente exista. Posteriormente, el programa le volverá a preguntar si tiene algún destino intermedio nuevamente. El usuario puede ingresar tantos destinos intermedios como quiera. En el momento en que ya no desee agregar más destinos intermedios deberá responder con un **no** a la pregunta para continuar con la aplicación.

```
Wazelog: ¿Tiene algún destino intermedio?
Usuario: quiero pasar al supermercado
Wazelog: ¿Dónde se encuentra el/la supermercado?
Usuario: se encuentra en dulce_nombre
Wazelog: Destino intermedio agregado: dulce_nombre
Wazelog: ¿Tiene algún destino intermedio?
Usuario: no
```

Figura 6. Ejemplo de agregar un destino intermedio y cómo salirse del loop para continuar con la aplicación.

Finalmente el programa le preguntará por su lugar de destino final. El usuario deberá responder con una oración que cumpla la gramática establecida. A continuación se detalla oraciones permitidas para esta parte:

- nos dirigimos a <lugar>
- ellos se dirigen a <lugar>
- ellas se dirigen a <lugar>
- el se dirige a <lugar>
- ella se dirige a <lugar>
- nosotros vamos a <lugar>
- voy para <lugar>
- voy a <lugar>
- yo voy a <lugar>
- yo voy para <lugar>
- el va a <lugar>
- el va para <lugar>
- ella va a <lugar>
- ella va para <lugar>
- ellos van a <lugar>
- ellos van para <lugar>
- ellos van a <lugar>
- ellos van para <lugar>

Si la respuesta dada por el usuario es correcta y el lugar existe, entonces el programa procede a calcular la ruta más corta entre los lugares dados. Una vez con la ruta calculada, se imprime la ruta, la distancia estimada, el tiempo estimado y el tiempo en presa estimado. Posteriormente, Wazelog comienza a ejecutarse desde el inicio.

```
Wazelog: ¿Cuál es su destino?  
Usuario: voy para guadalupe  
Wazelog: Su ruta sería: corralillo, dulce_nombre, guadalupe , Distancia estimada: 29.0 KM , Tiempo estimado: 59 min ,  
Tiempo en presa estimado: 79 min  
Wazelog: Muchas gracias por utilizar Wazelog, lo esperamos pronto  
Wazelog: Bienvenido a Wazelog, la mejor lógica para llegar a su destino  
Wazelog: Por favor indique donde se encuentra  
Usuario:
```

Figura 7. Ejemplo de una para respuesta para el lugar de destino y la interfaz donde muestra la ruta, distancia estimada, tiempo estimado y tiempo estimado en presa.

En caso de que no exista una ruta para los lugares dados. Wazelog imprime un mensaje y vuelve a empezar la aplicación.

```
Wazelog: No existe una ruta, por favor intentelo de nuevo
Wazelog: Bienvenido a Wazelog, la mejor lógica para llegar a su destino
Wazelog: Por favor indique donde se encuentra
Usuario: █
```

Figura 8. Ejemplo del programa en caso de que la ruta no exista.

En caso de que las respuestas dadas por el usuario no cumplan con la gramática establecida o el lugar o establecimiento ingresado que no exista, Wazelog le comunicará al usuario imprimiendo un mensaje y preguntando nuevamente.

```
Disculpa, no te he entendido o el lugar que ingresaste no existe
Wazelog: Por favor indique donde se encuentra
Usuario: █
```

Figura 9. Ejemplo en caso de que la respuesta dada no cuente con la estructura establecida en la gramática.

Notas extras

- Las oraciones se deben escribir sin tildes, y en minúscula, incluyendo los nombres de lugares. Las palabras que lleven una ñ se deben cambiar con una n.
- Aquellos lugares donde su nombre esté formado por dos o más palabras, se deberá unir mediante un “_”. Ejemplo: san_jose, tres_rios.
- Para salir de la aplicación ingrese **q** como respuesta.