



Casos de uso

Fco. Javier Bohórquez Ogalla

Índice

1. Visión general	4
2. Actores	4
3. Intérprete	5
3.1. Interpretar entrada estándar	5
3.2. Interpretar fichero	6
3.3. Interpretar línea	7
3.4. Interpretar	7
3.5. Ver ayuda	8
3.6. Cargar extensión	9
3.7. Listar extensiones	9
3.8. Iniciar interpretación red	10
3.9. Obtener pasos interpretación red	10
4. runTree	12
4.1. Enviar código fuente	12
4.2. Siguiente paso	13
4.3. Siguiente sentencia	14
4.4. Activar ejecución automática	15
4.5. Desactivar ejecución automática	15
4.6. Limpiar salida	16
4.7. Ver información de nodo	17

4.8. Ver contenido de la tabla de símbolos	17
4.9. Guardar código fuente	18
4.10. Abrir código fuente	18

1. Visión general

En esta sección se detallan los casos de usos que recogen los sistemas que conforman el proyecto OMI. Un caso de uso describe la secuencia de interacciones entre el sistema y sus actores como consecuencia de un evento.

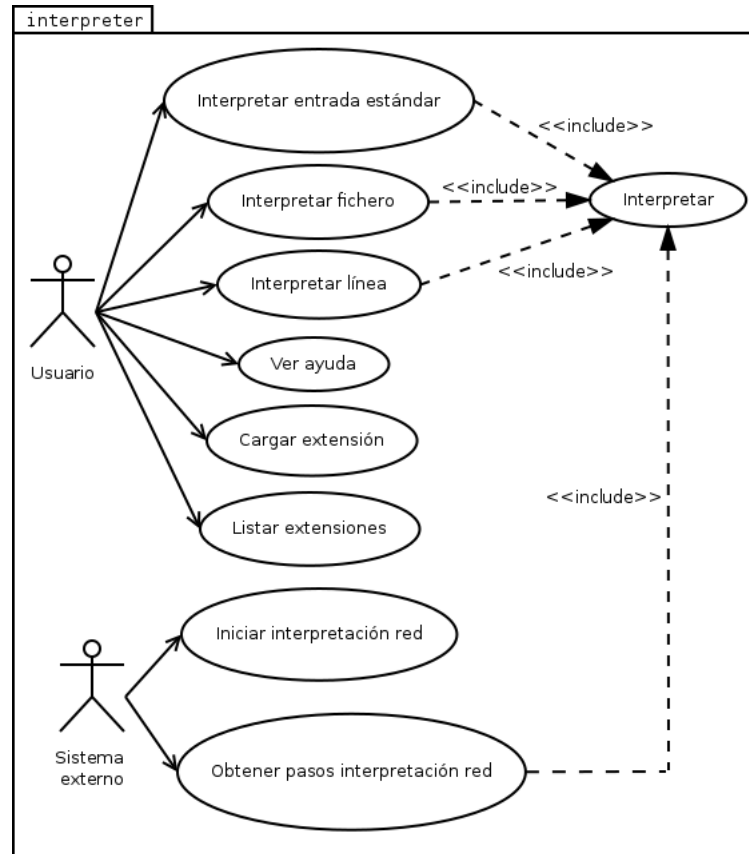
En primer lugar se describen los actores que hacen uso del sistema, llegándose a distinguir dos que serán nominados usuario y sistema externo. Luego se describen los casos de uso del sistema intérprete y del cliente runTree.

2. Actores

Los actores humanos que interactúan con los sistemas OMI presentan todos un mismo rol. Así el único actor definido será llamado usuario. Los usuarios que hacen uso del intérprete pueden ser desarrolladores, estudiantes u otros perfiles técnicos, pero todos usarán el sistema de la misma forma.

Por otro lado el intérprete OMI puede ser usado por otros sistemas, viéndose estos actores de determinados casos de uso. Algunos sistemas del proyecto OMI hacen uso del intérprete para llevar a cabo su propósito.

3. Intérprete



3.1. Interpretar entrada estándar

Caso de Uso: Interpretar entrada estándar.

Tipo: General.

Descripción: El usuario introduce un bloque de código en forma de cadena de caracteres mediante la entrada estándar. El sistema lo interpreta y ejecuta.

Actores: Usuario.

Precondiciones: El sistema debe estar esperando un bloque de código mediante la entrada estándar.

Postcondiciones: El código introducido es interpretado y ejecutado.

Escenario principal:

1. El usuario inicia el sistema facilitando un listado de argumentos.
2. El sistema asigna como variables cada argumento y solicita contenido fuente al usuario.
3. El usuario introduce un bloque de código en la entrada estándar.
4. El sistema obtiene el bloque de código a interpretar mediante la entrada estándar.
5. Incluir (Interpretar).

3.2. Interpretar fichero

Caso de Uso: Interpretar fichero.

Tipo: General.

Descripción: El usuario indica un fichero que contiene código y que será interpretado y ejecutado por el sistema.

Actores: Usuario.

Precondiciones: El sistema espera que se le indique un fichero.

Postcondiciones: El fichero es leído y el código en el mismo es interpretado y ejecutado.

Escenario principal:

1. El usuario indica la ruta a un fichero y una serie de argumentos
2. El sistema lee el fichero y obtiene el código en el mismo, además asigna cada argumento a variables.
3. Incluir (Interpretar).

Flujo alternativo:

- 2a. El fichero indicado no se encuentra.
 1. El sistema informa del error y finaliza.

3.3. Interpretar línea

Caso de Uso: Interpretar línea.

Tipo: General.

Descripción: El usuario introduce bloques de códigos denominados líneas de una forma interactiva. El sistema solicita por la entrada estándar las líneas de código, que serán interpretadas y ejecutadas.

Actores: Usuario.

Precondiciones: El sistema se encuentra en modo interactivo.

Postcondiciones: Se interpreta cada línea introducida por el usuario

Escenario principal:

1. El usuario inicia el sistema facilitando un listado de argumentos y la opción de interprete de línea.
2. El sistema asigna como variables cada argumento y muestra un prompt que indica que espera una línea de código.
3. El usuario introduce una línea de código.
4. El sistema lee de la entrada estándar la línea introducida.
5. Include (Interpretar).

El sistema repite el caso de uso hasta que se interpreta una sentencia que produzca la salida.

3.4. Interpretar

Caso de Uso: Interpretar.

Tipo: General.

Nivel: Subfunción.

Descripción: El sistema analiza, interpreta y ejecuta un bloque de código facilitado por el usuario. Para ello comprueba que este cumple con el léxico y la gramática

del lenguaje que define, dividiéndolo a su vez en sentencias que serán interpretadas.

Precondiciones: Se dispone de un bloque de código.

Postcondiciones: El bloque de código es interpretado.

Escenario principal:

1. El sistema procesa y comprueba el bloque de código, aplicando la gramática y léxico que define.
2. El sistema obtiene e interpreta cada sentencia en el código, produciéndose el significado semántico que estas encierran.

Flujo alternativo:

- 1a. El código no respeta el léxico del lenguaje.
 1. El sistema informa del error y finaliza.
- 1b. El código no respeta la gramática del lenguaje.
 1. El sistema informa del error y finaliza.
- 2a. El código no contiene ninguna sentencia.
 1. El sistema finaliza.

3.5. Ver ayuda

Caso de Uso: Ver ayuda.

Tipo: General.

Descripción: Se muestra una ayuda que detalla cada opción disponible en el sistema.

Actores: Usuario.

Precondiciones: Sin precondiciones.

Postcondiciones: El sistema muestra un listado que presenta las distintas opciones.

Escenario principal:

1. El usuario indica que quiere visualizar la ayuda.
2. El sistema muestra un listado completo de las opciones que presenta.

3.6. Cargar extensión

Caso de Uso: Cargar extensión.

Tipo: General.

Descripción: El usuario indica que una extensión que será cargada por el sistema.

Actores: Usuario.

Precondiciones: Sin precondiciones.

Postcondiciones: El sistema cargar la extensión facilitada.

Escenario principal:

1. El usuario indica la ruta a la extensión que desea cargar.
2. El sistema carga la extensión para disponer de las distintas opciones que ofrece.

Flujo alternativo:

- 2a. La extensión indicada no se encuentra.
 1. El sistema informa del error.
- 2b. La extensión indicada no es una extensión válida.
 1. El sistema informa del error.

3.7. Listar extensiones

Caso de Uso: Listar extensiones.

Tipo: General.

Descripción: Lista las extensiones que serán cargadas en cada ejecución del sistema.

Actores: Usuario.

Precondiciones: Sin precondiciones.

Postcondiciones: El sistema lista las extensiones que serán cargadas.

Escenario principal:

1. El usuario indica que desea listar las extensiones cargadas.
2. El sistema lista las extensiones cargadas por defecto.

3.8. Iniciar interpretación red

Caso de Uso: Iniciar interpretación red

Tipo: Red

Descripción: Un sistema exteno inicia una interpretación por red, estableciendo el contenido fuente que será interpretado. El sistema procesa la petición y abre un nuevo proceso de interpretación por pasos.

Precondiciones: No tiene.

Postcondiciones: Se establece el código fuente a interpretar y se inicia una interpretación por pasos.

Escenario principal:

1. El sistema espera una petición de iterpretación por red.
2. El sistema externo inicia la comunicación y facilita el código fuente.
3. El sistema establece el código fuente y abre un nuevo proceso de interpretación por pasos.

Flujo alternativo:

- 1a. El servicio no se encuentra disponible.
 1. Se devuelve un estado de error.

3.9. Obtener pasos interpretación red

Caso de Uso: Obtener pasos interpretación red

Tipo: Red

Descripción: Un sistema exteno obtiene un nuevo estado dentro de los pasos en el proceso de interpretación del código fuente establecido.

Precondiciones: Se ha establecido código fuente para una interpretación por pasos.

Postcondiciones: Se lleva a cabo un nuevo paso dentro del proceso de interpretación. Obteniéndose el siguiente estado.

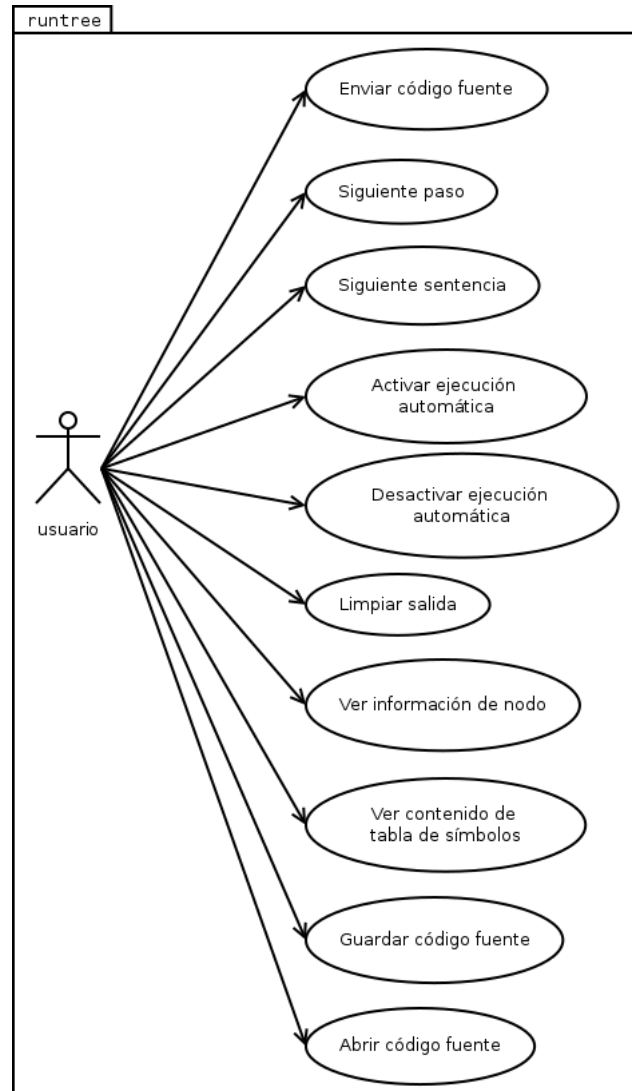
Escenario principal:

1. El sistema espera la petición de un nuevo paso en una iterpretación por red.
2. El sistema externo realiza la petición de un nuevo paso.
3. Incluir (Interpretar).
4. El sistema devuelve una estructura de datos que representa el estado actual del proceso de interpretación.

Flujo alternativo:

- 1a. El servicio no se encuentra disponible.
 1. Se devuelve un estado de error.
- 3a. El código fuente presenta errores sintáctico o léxicos.
 1. Se devuelve un estado de error.

4. runTree



4.1. Enviar código fuente

Caso de Uso: Enviar código fuente

Tipo: runTree

Descripción: El usuario envía texto correspondiente a código fuente para su interpretación y análisis. El sistema cliente envía el código al servidor que lo establecerá

como fuente a interpretar y enviará datos relativos al proceso. El cliente imprime el árbol sintáctico y espera acciones del usuario.

Precondiciones: No tiene.

Postcondiciones: Se ha establecido el código fuente para el análisis del proceso de interpretación y se ha imprimido el árbol sintáctico.

Escenario principal:

1. El sistema solicita código fuente escrito en OMI.
2. El usuario introduce código fuente escrito en el lenguaje OMI.
3. El sistema cliente runtree envia el código fuente al servidor para su interpretación.
4. El sistema servidor devuelve una representacion de los datos que describen el árbol sintáctico relativo al código fuente enviado.
5. El sistema cliente imprime el árbol sintáctico.

Flujo alternativo:

- 4a. El código fuente no presenta una sintaxis OMI correcta.
 1. Se devuelve un estado de error.
- 4b. El servicio no se encuentra disponible.
 1. Se devuelve un estado de error.

4.2. Siguiendo paso

Caso de Uso: Siguiendo paso

Tipo: runTree

Descripción: El usuario usuario solicita un nuevo paso en el proceso de interpretación. El sistema cliente obtiene el nuevo paso del servidor y lo representa en pantalla.

Precondiciones:

- Existe un código fuente establecido para estudio.
- El estado actual del proceso no es final.

- La ejecución automática se encuentra desactivada.

Postcondiciones: Se representa en nuevo paso.

Escenario principal:

1. El usuario solicita un nuevo paso en el proceso de interpretación
2. El sistema cliente solicita la resolución de un nuevo paso y representa el nuevo estado en pantalla.

Flujo alternativo:

- 2a. El servicio no se encuentra disponible.
 1. Se devuelve un estado de error.

4.3. Siguiendo sentencia

Caso de Uso: Siguiendo sentencia

Tipo: runTree

Descripción: El usuario solicita la resolución de la siguiente sentencia. El sistema cliente obtiene el estado correspondiente a la interpretación de la siguiente sentencia dentro del proceso de interpretación y lo representa en pantalla.

Precondiciones:

- Existe un código fuente establecido para estudio.
- El estado actual del proceso no es final.
- La ejecución automática se encuentra desactivada.

Postcondiciones: Se representa en nuevo estado correspondiente a la interpretación de una sentencia completa.

Escenario principal:

1. El usuario solicita la resolución de una nueva sentencia en el proceso de interpretación.
2. El sistema cliente solicita la resolución de la sentencia completa y representa el nuevo estado en pantalla.

Flujo alternativo:

- 2a. El servicio no se encuentra disponible.
 1. Se devuelve un estado de error.

4.4. Activar ejecución automática

Caso de Uso: Activar ejecución automática

Tipo: runTree

Descripción: El usuario solicita la ejecución automática. El sistema cliente obtiene y representa cada paso dentro del proceso de interpretación hasta obtenerse un estado final.

Precondiciones:

- Existe un código fuente establecido para estudio.
- El estado actual del proceso no es final.
- La ejecución automática se encuentra desactivada.

Postcondiciones: Se obtiene un estado final.

Escenario principal:

1. El usuario activa la ejecución automática.
2. El sistema obtiene y representa cada paso en el proceso de interpretación hasta que se obtiene un estado final.

Flujo alternativo:

- 2a. El servicio no se encuentra disponible.
 1. Se devuelve un estado de error.

4.5. Desactivar ejecución automática

Caso de Uso: Desactivar ejecución automática

Tipo: runTree

Descripción: El usuario solicita la detención de la ejecución automática. El sistema cliente detiene la ejecución automática y representa el estado actual.

Precondiciones:

- Existe un código fuente establecido para estudio.
- El estado actual del proceso no es final.
- La ejecución automática se encuentra activada.

Postcondiciones: Se obtiene el estado actual del proceso.

Escenario principal:

1. El usuario desactiva la ejecución automática.
2. El sistema cliente para la ejecución automática y representa el estado actual

4.6. Limpiar salida

Caso de Uso: Limpiar salida

Tipo: runTree

Descripción: El usuario solicita la limpieza de los datos de salida. El sistema cliente elimina la información disponible en la consola de salida.

Precondiciones: No tiene.

Postcondiciones: La consola de salida queda vacía.

Escenario principal:

1. El usuario solicita la limpieza de la salida.
2. El sistema cliente limpia la información disponible en la consola de salida.

4.7. Ver información de nodo

Caso de Uso: Ver información de nodo.

Tipo: runTree

Descripción: El usuario marca un nodo para ver su información. El sistema cliente muestra la información relativa al nodo.

Precondiciones: Existe un código fuente establecido para estudio.

Postcondiciones: Se muestra información relativa al nodo.

Escenario principal:

1. El usuario marca un nodo para ver su información.
2. El sistema muestra información relativa al nodo tal como su posición de memoria interna, su tamaño, su tipo y su nombre.

4.8. Ver contenido de la tabla de símbolos

Caso de Uso: Ver contenido de la tabla de símbolos.

Tipo: runTree

Descripción: El usuario marca una tabla de símbolos para ver su contenido. El sistema cliente muestra la información relativa a la tabla de símbolos.

Precondiciones: Existe un código fuente establecido para estudio.

Postcondiciones: Se muestra información relativa a la tabla de símbolos.

Escenario principal:

1. El usuario indica una tabla de símbolos para ver su contenido.
2. El sistema cliente muestra los nodos referenciados desde la tabla de símbolos dada.

4.9. Guardar código fuente

Caso de Uso: Guardar código fuente.

Tipo: runTree

Descripción: El usuario solicita guardar el código fuente escrito en un fichero local y el cliente abre el cuadro de diálogo correspondiente.

Precondiciones: Existe un código escrito en el cliente

Postcondiciones: Se guarda el código fuente en un fichero local.

Escenario principal:

1. El usuario solicita guardar el código fuente en un fichero local.
2. El sistema abre el cuadro de diálogo correspondiente.

4.10. Abrir código fuente

Caso de Uso: Abrir código fuente.

Tipo: runTree

Descripción: El usuario solicita cargar código fuente desde un fichero local.

Precondiciones: No tiene.

Postcondiciones: Se carga el código fuente contenido en el fichero.

Escenario principal:

1. El usuario solicita abrir un fichero de código fuente.
2. El sistema abre el cuadro de diálogo correspondiente.
3. El sistema carga el contenido del código fuente.

Flujo alternativo:

- 2c. El fichero no contiene código en texto plano.
 1. Se devuelve un estado de error.