



Modelo de comportamiento

Fco. Javier Bohórquez Ogalla

Índice

1. Visión general	3
2. Diagramas de secuencia del sistema	3
2.1. Interpretar entrada estándar	3
2.2. Interpretar línea	4
2.3. Interpretar fichero	4
2.4. Ver ayuda	5
2.5. Cargar extensión	5
2.6. Listar extensiones	6
3. Contratos de operaciones del sistema	6
3.1. Operación inicio_interprete_stdin	6
3.2. Operación inicio_interprete_linea	8
3.3. Operación interpretar_cadena	9
3.4. Operación interpretar_fichero	10

1. Visión general

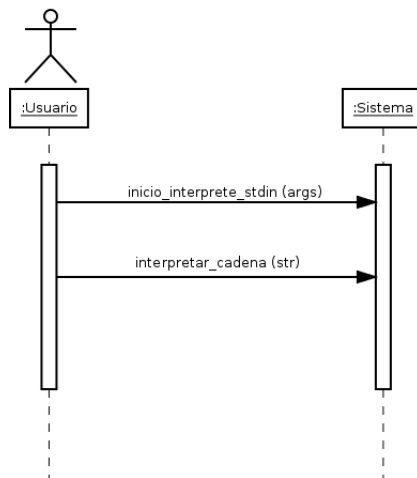
El presente documento constituye el modelo de comportamiento del sistema.

Inicialmente se presentan los diagramas de secuencias que modelan los eventos que el sistema puede recibir del usuario y los valores de retorno que produce como consecuencia de estos.

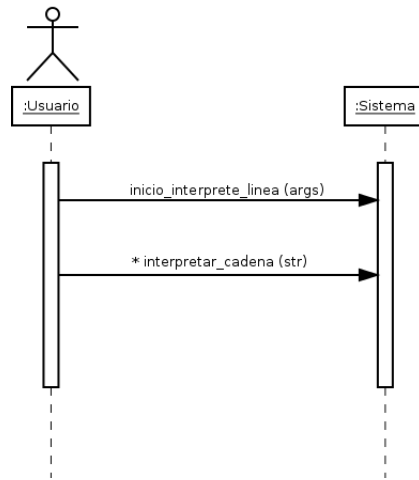
A partir de los diagramas de secuencia del sistema se obtienen las operaciones que este presenta. Luego se describen los contratos de cada una de las operaciones.

2. Diagramas de secuencia del sistema

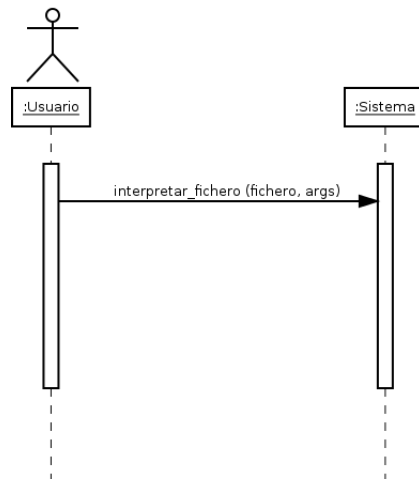
2.1. Interpretar entrada estándar



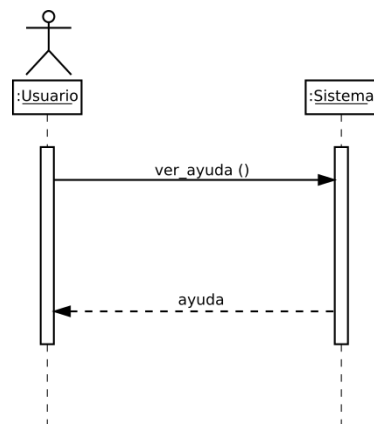
2.2. Interpretar línea



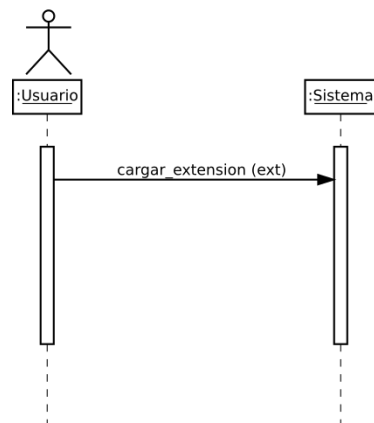
2.3. Interpretar fichero



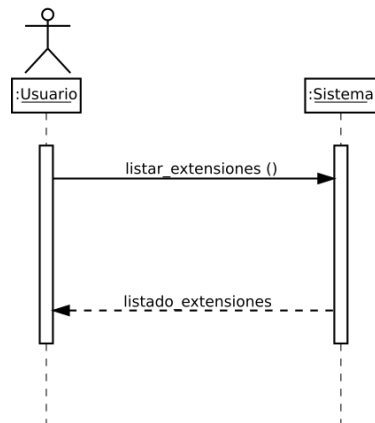
2.4. Ver ayuda



2.5. Cargar extensión



2.6. Listar extensiones



3. Contratos de operaciones del sistema

Sistema
+inicio_interprete_stdin(args) +inicio_interprete_linea(args) +interpretar_cadena(str) +interpretar_fichero(fichero,args) +ver_ayuda() +cargar_extension(ext) +listar_extensiones()

3.1. Operación inicio_interprete_stdin

Nombre: inicio_interprete_stdin(args)

Responsabilidades: Iniciar la interpretación de la entrada estándar.

Referencias Cruzadas: Caso de Uso: Interpretar entrada estándar

Precondiciones: No tiene.

Postcondiciones:

- Se creó una instancia “i” de “interpreter” (creación de objeto).
- Se inicializaron los atributos de “i”.
- Se crearon las instancias “ $a_0 \dots a_n$ ” de “arg” por cada argumento en “args” (creación de objeto).
- Se asignó “ args_i ” a “ $a_i.\text{value}$ ” ($a_i.\text{value} = \text{args}_i$) (modificación de atributos).
- Se asoció los “arg” “ $a_0 \dots a_n$ ” al objeto “i” de “interpreter” (creación de enlace).
- Se creó una instancia “p” de “parser” (creación de objeto).
- Se inicializaron los atributos de “p”.
- Se creó una instancia “s” de “scanner” (creación de objeto).
- Se inicializaron los atributos de “s”.
- Se asoció el “scanner” “s” al objeto “p” de “parser” (creación de enlace).
- Se asoció el “parser” “p” al objeto “i” de “interpreter” (creación de enlace).
- Se creó una instancia “c” de “context” (creación de objeto).
- Se inicializaron los atributos de “c”.
- Se creó una instancia “var” de “varSymbols” (creación de objeto).
- Se inicializaron los atributos de “var”.
- Se asoció el “varSymbols” “var” al objeto “c” de “context” (creación de enlace).
- Se creó una instancia “func” de “funcSymbols” (creación de objeto).
- Se inicializaron los atributos de “func”.
- Se asoció el “funcSymbols” “func” al objeto “c” de “context” (creación de enlace).
- Se creó una instancia “class” de “classSymbols” (creación de objeto).
- Se inicializaron los atributos de “class”.
- Se asoció el “classSymbols” “class” al objeto “c” de “context” (creación de enlace).
- Se asoció el “context” “c” al objeto “i” de “interpreter” (creación de enlace).

3.2. Operación inicio_interprete_linea

Nombre: inicio_interprete_linea(args)

Responsabilidades: Iniciar la interpretación interactiva línea a línea.

Referencias Cruzadas: Caso de Uso: Interpretar línea

Precondiciones: No tiene.

Postcondiciones:

- Se creó una instancia “i” de “interpreter” (creación de objeto).
- Se inicializaron los atributos de “i”.
- Se crearon las instancias “ $a_0 \dots a_n$ ” de “arg” por cada argumento en “args” (creación de objeto).
- Se asignó “args_i” a “ $a_i.value$ ” ($a_i.value = args_i$) (modificación de atributos).
- Se asoció los “arg” “ $a_0 \dots a_n$ ” al objeto “i” de “interpreter” (creación de enlace).
- Se creó una instancia “p” de “parser” (creación de objeto).
- Se inicializaron los atributos de “p”.
- Se creó una instancia “s” de “scanner” (creación de objeto).
- Se inicializaron los atributos de “s”.
- Se asoció el “scanner” “s” al objeto “p” de “parser” (creación de enlace).
- Se asoció el “parser” “p” al objeto “i” de “interpreter” (creación de enlace).
- Se creó una instancia “c” de “context” (creación de objeto).
- Se inicializaron los atributos de “c”.
- Se creó una instancia “var” de “varSymbols” (creación de objeto).
- Se inicializaron los atributos de “var”.
- Se asoció el “varSymbols” “var” al objeto “c” de “context” (creación de enlace).
- Se creó una instancia “func” de “funcSymbols” (creación de objeto).
- Se inicializaron los atributos de “func”.
- Se asoció el “funcSymbols” “func” al objeto “c” de “context” (creación de enlace).
- Se creó una instancia “class” de “classSymbols” (creación de objeto).

- Se inicializaron los atributos de “class”.
- Se asoció el “classSymbols” “class” al objeto “c” de “context” (creación de enlace).
- Se asoció el “context” “c” al objeto “i” de “interpreter” (creación de enlace).

3.3. Operación interpretar_cadena

Nombre: interpretar_cadena(str)

Responsabilidades: Interpreta el contenido fuente almacenado en la cadena “str”

Referencias Cruzadas:

- Caso de Uso: Interpretar entrada estándar
- Caso de Uso: Interpretar línea

Precondiciones:

- Se creó un “interpreter” “i”.
- Se creó y asoció una instancia de “parser” y “scanner” a “i”.
- Se creó y asoció una instancia de “varSymbols”, “funcSymbols” y “classSymbols” a “i”.

Postcondiciones:

- Se creó una instancia “s” de “source” (creación de objeto).
- Se asignó “str” a “s.src” (s.src = str) (modificación de atributos)
- Se asoció “s” al “scanner” componente del “interpreter” “i” (creación de enlace).
- Se creó un conjunto “ $t_0...t_n$ ” de “token” a partir del análisis léxico (creación de objeto).
- Se creó un conjunto “ $n_0...n_n$ ” de “runNode” a partir del análisis sintáctico (creación de objetos).
- Se asoció “ $n_i \in n_0...n_n$ ” a “ $n_k \in n_0...n_n$ ” para construir el árbol sintáctico (creación de enlace).
- Se asoció “ $n_r \in n_0...n_n$ ”, raíz del árbol sintáctico, al “interpreter” “i” (creación de enlace).

- Se creó un conjunto “ $var_0...var_n$ ” de “refNode” correspondientes a las variables definidas en el contenido fuente (creación de objetos).
- Se creó un conjunto “ $val_0...val_n$ ” de “runNode” correspondientes a los valores asignados a las variables definidas en el contenido fuente (creación de objetos).
- Se asoció “ $val_i \in val_0...val_n$ ” a “ $var_i \in var_0...var_n$ ” donde val_i es el valor de la variable var_i (creación de enlace).
- Se asoció todo “refNode” “ $var_0...var_n$ ” al componente “varSymbols” de “i” (creación de enlace).
- Se creó un conjunto “ $func_0...func_n$ ” de “refNode” correspondientes a las funciones con identificador definidas en el contenido fuente (creación de objetos).
- Se asoció “ $func_i \in func_0...func_n$ ” a “ $n_i \in n_0...n_n$ ” donde n_i es un “funcNode” correspondiente a la definición de la función $func_i$ (creación de enlaces).
- Se asoció todo “refNode” “ $func_0...func_n$ ” al componente “funcSymbols” de “i” (creación de enlace).
- Se creó un conjunto “ $class_0...class_n$ ” de “refNode” correspondientes a las clases definidas en el contenido fuente (creación de objetos).
- Se asoció “ $class_i \in class_0...class_n$ ” a “ $n_i \in n_0...n_n$ ” donde n_i es un “classNode” correspondiente a la definición de la clase $class_i$ (creación de enlaces).
- Se asoció todo “refNode” “ $class_0...class_n$ ” al componente “classSymbols” de “i” (creación de enlace).

3.4. Operación interpretar_fichero

Nombre: interpretar_fichero(fichero, args)

Responsabilidades: Interpreta el contenido fuente almacenado en “fichero”

Referencias Cruzadas: Caso de Uso: Interpretar fichero

Precondiciones: No tiene.

Postcondiciones:

- Se creó una instancia “i” de “interpreter” (creación de objeto).

- Se inicializaron los atributos de “i”.
- Se creó el conjunto “ $a_0...a_n$ ” de “arg” según el número de argumentos en “args” (creación de objeto).
- Se asignó “ $args_i$ ” a “ $a_i.value$ ” ($a_i.value = args_i$) (modificación de atributos).
- Se asoció los “arg” “ $a_0...a_n$ ” al objeto “i” de “interpreter” (creación de enlace).
- Se creó una instancia “p” de “parser” (creación de objeto).
- Se inicializaron los atributos de “p”.
- Se creó una instancia “s” de “scanner” (creación de objeto).
- Se inicializaron los atributos de “s”.
- Se asoció el “scanner” “s” al objeto “p” de “parser” (creación de enlace).
- Se asoció el “parser” “p” al objeto “i” de “interpreter” (creación de enlace).
- Se creó una instancia “c” de “context” (creación de objeto).
- Se inicializaron los atributos de “c”.
- Se creó una instancia “var” de “varSymbols” (creación de objeto).
- Se inicializaron los atributos de “var”.
- Se asoció el “varSymbols” “var” al objeto “c” de “context” (creación de enlace).
- Se creó una instancia “func” de “funcSymbols” (creación de objeto).
- Se inicializaron los atributos de “func”.
- Se asoció el “funcSymbols” “func” al objeto “c” de “context” (creación de enlace).
- Se creó una instancia “class” de “classSymbols” (creación de objeto).
- Se inicializaron los atributos de “class”.
- Se asoció el “classSymbols” “class” al objeto “c” de “context” (creación de enlace).
- Se asoció el “context” “c” al objeto “i” de “interpreter” (creación de enlace).
- Se creó una instancia “src” de “source” (creación de objeto).
- Se asignó el contenido de “fichero” a “src.src” ($src.src = fichero$) (modificación de atributos)
- Se asoció “src” al “scanner” “s” componente del “interpreter” “i” (creación de enlace).
- Se creó un conjunto “ $t_0...t_n$ ” de “token” a partir del análisis léxico (creación de objeto).

- Se creó un conjunto " $n_0...n_n$ " de "runNode" a partir del análisis sintáctico (creación de objetos).
- Se asoció " $n_i \in n_0...n_n$ " a " $n_k \in n_0...n_n$ " para construir el árbol sintáctico (creación de enlace).
- Se asoció " $n_r \in n_0...n_n$ ", raíz del árbol sintáctico, al "interpreter" "i" (creación de enlace).
- Se creó un conjunto " $var_0...var_n$ " de "refNode" correspondientes a las variables definidas en el contenido fuente (creación de objetos).
- Se creó un conjunto " $val_0...val_n$ " de "runNode" correspondientes a los valores asignados a las variables definidas en el contenido fuente (creación de objetos).
- Se asoció " $val_i \in val_0...val_n$ " a " $var_i \in var_0...var_n$ " donde val_i es el valor de la variable var_i (creación de enlace).
- Se asoció todo "refNode" " $var_0...var_n$ " al componente "varSymbols" de "i" (creación de enlace).
- Se creó un conjunto " $func_0...func_n$ " de "refNode" correspondientes a las funciones con identificador definidas en el contenido fuente (creación de objetos).
- Se asoció " $func_i \in func_0...func_n$ " a " $n_i \in n_0...n_n$ " donde n_i es un "funcNode" correspondiente a la definición de la función $func_i$ (creación de enlaces).
- Se asoció todo "refNode" " $func_0...func_n$ " al componente "funcSymbols" de "i" (creación de enlace).
- Se creó un conjunto " $class_0...class_n$ " de "refNode" correspondientes a las clases definidas en el contenido fuente (creación de objetos).
- Se asoció " $class_i \in class_0...class_n$ " a " $n_i \in n_0...n_n$ " donde n_i es un "classNode" correspondiente a la definición de la clase $class_i$ (creación de enlaces).
- Se asoció todo "refNode" " $class_0...class_n$ " al componente "classSymbols" de "i" (creación de enlace).