



MANUAL DE USO RAGNAROK



VERSION 1.0 - 2012

TABLA DE CONTENIDO

1.	Ragnarok, la aplicación	8
1.1	Qué es Ragnarok?	8
1.2	Cómo se instala?	8
1.3	En qué sistema operativo funciona?	9
1.4	Cuáles son los requerimientos mínimos?	9
1.5	Quienes pueden usar Ragnarok?	10
2.	Dando los primeros pasos en Ragnarok	12
2.1	Cómo empiezo?	12
2.2	Qué es un Ejercicio?	15
2.3	Qué es una Resolución de Ejercicio?	15
2.3	Cómo guardar los archivos?	16
2.4	Cómo abrir los archivos? Qué extensiones se manejan?	16
3.	Configuración	17
3.1	Cómo definir los directorios de trabajo por defecto?	17
3.2	Directorio de Ejercicios Creados	17
3.3	Directorio de Resoluciones	17
3.4	Directorio Ejercicios Descargados	17
3.5	Directorio Temporales	17
3.6	Directorio Abrir Default	18
3.7	Activar Tutorial/Desactivar Tutorial	18
4.	Sincronización	19
4.1	Qué es la Sincronización? Para qué sirve?	19
4.2	Tengo que tener internet para usarla?	19

4.3	Propiedades	19
4.4	Sincro General.....	19
4.5	Sincro Por Curso.....	20
4.6	Sincro Por Ejercicio	20
5.	Modo Gráfico, Modo Texto	21
5.1	Modo Gráfico	21
5.1.1	Qué son los Tabs?	21
5.1.2	La Barra de Actividades.....	21
5.1.3	El Área de Trabajo.....	21
5.1.4	Cómo se agrega un nuevo Procedimiento?	23
5.1.5	Cómo se agrega una nueva Función?	24
5.1.6	Qué es la “P” que se muestra al lado del procedimiento Salida? Por qué Principal, Variables y Constantes no la tienen?	25
5.1.7	Arrastrando actividades al área de trabajo	26
5.1.8	Por qué no se pueden arrastrar cualquier tipo de actividades a cualquier área de trabajo?	27
5.1.9	Puedo copiar y pegar figuras? Las puedo cambiar de orden?	27
5.1.10	Exportando actividades como imágenes (para que las compartas por mail con tus compañeros... o para que hagas trampa, da igual).....	28
5.1.11	Los errores de compilación	28
5.1.12	Las actividades de Ragnarok	30
5.1.12.1	DeclaracionArreglo	30
5.1.12.2	DeclaracionVariable	31
5.1.12.3	DeclaracionConstante	32

5.1.12.4	Si	32
5.1.12.5	Mostrar	34
5.1.12.6	Mientras.....	35
5.1.12.7	Asignación.....	35
5.1.12.8	LlamarProcedimiento	36
5.1.12.9	Leer	37
5.2	Modo Texto	38
5.2.1	El editor de texto (te preguntas si tenés que escribir acá? En el modo gráfico también)	38
5.2.2	Las palabras coloreadas.....	38
5.2.3	Buscar	40
5.2.4	Buscar y Reemplazar.....	42
5.2.5	Identar (Iden-qué?).....	43
5.2.6	Copiar/Cortar/Pegar	45
5.2.7	Rehacer/Deshacer (si cualquier editor lo tiene, por qué no?)	46
5.2.8	Los errores de compilación.....	47
5.3	Puedo pasar de uno al otro? (no te cases con ninguno, usá el que quieras, después cambiás)	48
6.	¿Está bien, hice un nuevo ejercicio, y ahora qué hago? (que no cunda el pánico)	49
6.1	Nuevo Ejercicio	49
6.2	Escribir un programa	50
6.3	Compilar un programa.....	50
6.4	Ejecutar un programa	51
6.5	Fin de la ejecución	52

7.	Las funciones predefinidas del lenguaje GarGar	55
8.	Muy lindo, codifiqué el programa, ¿cómo sé si está bien?	57
8.1	Tests de Prueba:	57
8.1.1	¿Qué son?	57
8.1.2	Crear un test de prueba	58
8.1.2.1	Paso 0: Cómo funcionan los test.....	59
8.1.2.2	Paso 1: Elegir las variables de entrada.....	60
8.1.2.3	Paso 2: Descripción de las variables de entrada.....	61
8.1.2.4	Paso 3: Valores iniciales de las variables de entrada.....	62
8.1.2.5	Paso 4: Elegir las variables de salida	64
8.1.2.6	Paso 5: Descripción de las variables de salida	65
8.1.2.7	Paso 6: Ejecutar.....	66
8.1.2.8	Paso 7: Chequear variables.....	68
8.1.3	Ejecutar test de prueba	68
8.1.3.1	Paso 0: Cómo funcionan los test.....	69
8.1.3.2	Paso 1: Elegir el test de prueba	70
8.1.3.3	Paso 2: Elegir las variables de entrada que corresponden a las del test	71
8.1.3.4	Paso 3: Valores iniciales de las variables de entrada.....	72
8.1.3.5	Paso 4: Elegir las variables de salida que corresponden a las del test	74
8.1.3.6	Paso 5: Ejecutar.....	75
8.1.3.7	Paso 6: Resultados	77
8.1.4	Ver los tests de prueba creados en el ejercicio	77
9.	Detalles de un Ejercicio.....	80

9.1	Enunciado (todo ejercicio tiene uno, verdad?)	80
9.2	Solución Explicada (No entendiste el enunciado? Lee esto con atención entonces)	80
9.3	Dificultad (algunos son más fáciles que otros)	80
10.	Exportar un ejercicio para subir a la Web	82
10.1	Yo también quiero que mis ejercicios sean descargados	82
10.2	Condiciones mínimas para la exportación a la web	83
11.	RepoWeb	84
11.1	Que es? Para que sirve?	84
El RepoWeb es un “navegador” de Cursos y Ejercicios que te va a permitir obtener todos los ejercicios que quieras/necesites a sólo 2 clicks de distancia. Lo podés encontrar en la solapa de “Sincronización”, en el botón que dice “Sincro por Repositorio”		84
11.2	Ejercicios	84
11.2.1	Buscador	84
11.2.2	Detalle.....	86
11.2.3	Descarga	88
11.3	Cursos	88
11.3.1	Buscador	89
11.3.2	Detalle.....	89
11.3.3	Descarga	89

1. Ragnarok, la aplicación

1.1 *Qué es Ragnarok?*

Ragnarok es el nombre de la aplicación desarrollada por Wild4Morgado. Esta herramienta es una interfaz de usuario cuyo objetivo es que vos aprendas a programar, a dar tus primeros pasos. Para que entiendas que la programación no es solo “11001010” o “010110100” sino que existen varios niveles de programación. Y el básico está muy lejos aún de los “1” y los “0”.

Ragnarok trabaja con el lenguaje GarGar. Este lenguaje tipo pseudocódigo y está escrito en castellano, a propósito, para que ayude a familiarizarte, porque vos sabés español seguro.

El nombre Ragnarok fue elegido porque todos los programas llevan un nombre código. Y nosotros elegimos este, no tiene nada que ver con el apocalipsis nórdico, sólo nos gustó el nombre. Si alguna vez jugaste Ragnarok Online, no te ilusiones, esto no tiene nada que ver con ese juego.

1.2 *Cómo se instala?*

Es muy sencillo. Lo primero que tenés que hacer es descargarlo de internet. No te preocupes, es totalmente gratuito, así que no vas a necesitar conseguir ningún crack o serial. Debés ingresar a la página www.program-ar.com.ar y dirigirte a la solapa que dice “Ragnarok”. Ahí encontrarás el link correspondiente para descargártelo a tu computadora.

Una vez que lo hayas descargado, sólo tienes que darle doble click al ejecutable y seguir paso a paso lo que aparece en el instalador.

1.3 *En qué sistema operativo funciona?*

Ragnarok sólo funciona en computadoras con sistema operativo Microsoft Windows. Aquellas versiones donde fue probada y sabemos que funciona son:

- Windows XP SP2/SP3
- Windows Vista SP1/SP2
- Windows 7 SP1

1.4 *Cuáles son los requerimientos mínimos?*

Bien, los requerimientos mínimos que debe tener tu PC para utilizar Ragnarok son:

Procesador	Dual Core 2.2 GHZ
Memoria RAM	2 GB
Disco Rígido	400 MB
PreRequisitos	Instalación .NET Framework 4.0

Si sos un poco más exigente y te gusta el rendimiento óptimo, entonces tu PC debería tener:

Procesador	Triple/Quad Core 2.0 GHZ o superior
Memoria RAM	4 GB o superior
Disco Rígido	400 MB
PreRequisitos	Instalación .NET Framework 4.0

1.5 Quienes pueden usar Ragnarok?

Todos. Todos. Todos. Cualquiera. El que tenga ganas. El que sepa leer y escribir y le guste pensar. El que tenga ganas de aprender. No hay límites para utilizar Ragnarok, sólo saber leer y escribir.

2. Dando los primeros pasos en Ragnarok

2.1 *Cómo empiezo?*

Una vez que ya tengas instalada la aplicación, vas a ver un ícono en el escritorio de tu PC que se llama “Ragnarok”. Sólo tenés que hacer doble click para ejecutarlo. Una vez que lo ejecutes, te vas a encontrar con la siguiente pantalla:



Fig.1 Pantalla de carga de Ragnarok

La Figura 1 tardará algunos segundos en cargar, y cuando la carga se haya finalizado, te vas a encontrar con la siguiente pantalla:



Listo

Fig.2 Pantalla inicial de Ragnarok

Para comenzar, debemos ir al menú que se encuentra a la izquierda de la pantalla (en Azul), seleccionar “Nuevo” y después “Nuevo Ejercicio”:

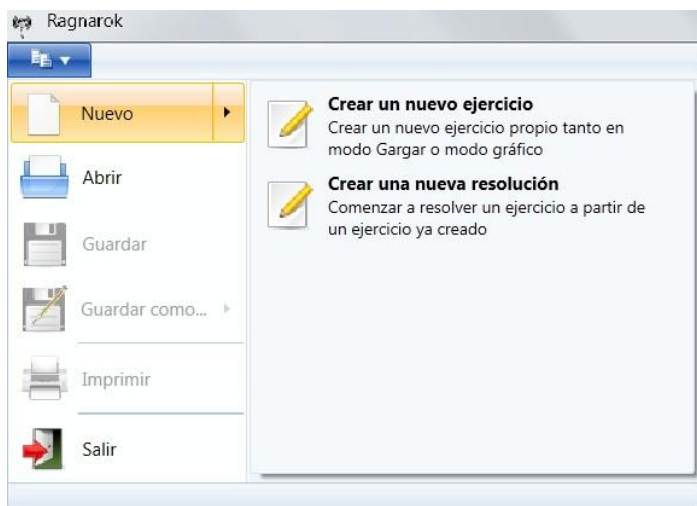


Fig.3 Menú principal de Ragnarok

Al hacer click sobre la opción “Crear un nuevo ejercicio”, aparecerá una pantalla que nos pedirá que elijamos un nombre para nuestro nuevo ejercicio. Ingresamos algún nombre (la aplicación se encarga de agregarle la extensión automáticamente) y clickeamos el botón “Aceptar”.

A continuación, Ragnarok desplegará una pantalla donde nos pregunta en qué modo vamos a querer trabajar (Modo Gráfico o Modo Texto). Qué son estos modos? Lo veremos más adelante. Por el momento vamos a seleccionar cualquiera de los 2:

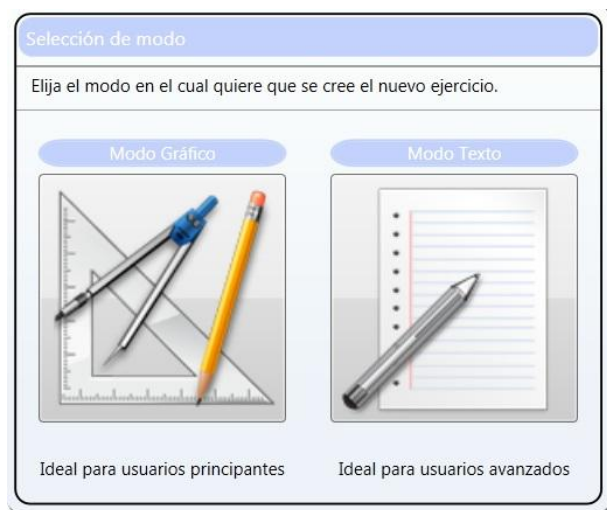


Fig.4 Selección de Modo (Gráfico o Texto)

De esta forma, habiendo elegido alguno de los 2 modos, ya estaremos en condiciones de comenzar a trabajar con Ragnarok. Todavía te parece medio raro todo esto no? Tranquilo, las respuestas están por llegar.

2.2 *Qué es un Ejercicio?*

Un ejercicio es el archivo principal de Ragnarok. Un ejercicio contiene un Enunciado (problema a resolver), una solución explicada (Explicación más detallada del problema) y un nivel de dificultad. Este es el archivo que va a crear una persona para que otra la resuelva. Supongamos el siguiente caso: Un aula con 20 alumnos y 1 profesor. El creador del ejercicio será el Profesor.

2.3 *Qué es una Resolución de Ejercicio?*

Siguiendo con el ejemplo anterior, los 20 alumnos deberán resolver este ejercicio a través de una Resolución de Ejercicio. Para crear una nueva resolución, se debe ir al menú principal, seleccionar la opción “Nuevo”, y luego “Crear una resolución”. Lo primero que hará Ragnarok es solicitar

que se elija el Ejercicio a resolver. Se deberá elegir el ejercicio creado por el profesor. Una vez hecho esto, se debe elegir un nombre para la resolución.

La resolución difiere del ejercicio en el hecho de que el menú de “Detalles Ejercicio” será de sólo lectura, sin posibilidad de modificar ni el enunciado, ni la solución explicada ni la dificultad. Tampoco se podrán crear Tests de Prueba, sólo ejecutarlos. Además incorpora una nueva opción llamada “Solución GarGar”. Aquí es donde se puede visualizar el código escrito por el profesor que sirve como guía, para visualizar cómo sería la solución correcta.

2.3 Cómo guardar los archivos?

Para guardar los archivos (sean ejercicios o resoluciones), sólo debemos dirigirnos al menú principal y presionar el botón “Guardar”. Si se desea guardar en otra ubicación, debemos elegir “Guardar Como...” y elegir la nueva ubicación para el archivo.

2.4 Cómo abrir los archivos? Qué extensiones se manejan?

Para abrir un archivo, debemos ir al menú principal y seleccionar “Abrir”. Ragnarok solicitará que se elija el archivo a abrir. Para ello, dará 2 opciones de extensión: “.gej” o “.gres”. Veamos qué es cada una:

.gej: Es la extensión de un Ejercicio creado en Ragnarok.

.gres: Es la extensión de una Resolución creada en Ragnarok.

Los archivos .gres, como vimos anteriormente, son creados a partir de los .gej (es decir, es necesario tener un Ejercicio previo a la creación de una Resolución del mismo).

3. Configuración

3.1 Cómo definir los directorios de trabajo por defecto?

La configuración ayuda a Ragnarok a ubicar directamente los directorios más utilizados, para que vos no tengas que andar buscándolos cada vez que quieras guardar o abrir un archivo. No es obligatorio que los definas, pero te ayudará a hacer las cosas más rápidamente. Para ello, es necesario ir a la solapa “Configuración” de Ragnarok. Allí vamos a encontrar 5 opciones:

3.2 Directorio de Ejercicios Creados

Este es el directorio donde Ragnarok se posicionará cada vez que se quiera crear un nuevo Ejercicio.

3.3 Directorio de Resoluciones

Este es el directorio donde Ragnarok se posicionará cada vez que se quiera crear una nueva Resolución de Ejercicio.

3.4 Directorio Ejercicios Descargados

Este es el directorio donde Ragnarok descargará los ejercicios que se soliciten a través de la Sincronización de manera automática.

3.5 Directorio Temporales

Este es el directorio donde Ragnarok alojará sus archivos temporales necesarios para la ejecución. Ragnarok generará archivos por cada ejecución de un programa. Luego, los eliminará automáticamente sin tu intervención.

3.6 Directorio Abrir Default

Este es el directorio donde Ragnarok se posicionará cada vez que quieras abrir un archivo (tanto un Ejercicio como una Resolución de Ejercicio).

3.7 Activar Tutorial/Desactivar Tutorial

Cuando abrís la aplicación por primera vez, te aparecerá un tutorial, que consiste en mostrarte a grandes rasgos como comenzar a utilizar Ragnarok. En cualquier momento que quieras, puedes desactivarlo. La próxima vez que abras la aplicación, ya no se mostrará el tutorial (en caso de que lo hayas desactivado, caso contrario, volverá a aparecer).

4. Sincronización

4.1 *Qué es la Sincronización? Para qué sirve?*

La sincronización permite la descarga de Ejercicios a través de la Web. La idea es que vos como usuario de Ragnarok, puedas descargar los Ejercicios que están en la Web para que los vayas resolviendo. De esta forma, el Profesor de tu curso puede tener una determinada cantidad de Ejercicios, e indicarte a vos que te descargues sus ejercicios.

Los Ejercicios son de libre acceso para todos los usuarios. Es decir, no hay ejercicio al cual no puedas acceder. Podes descargar los cursos/ejercicios que vos quieras.

4.2 *Tengo que tener internet para usarla?*

Sí, es requisito fundamental tener internet para poder sincronizar tus ejercicios, ya que están en un servidor que se accede únicamente a través de internet. No es necesario que tengas una conexión muy veloz, la sincronización no requiere de super-conexiones para poder funcionar.

4.3 *Propiedades*

Para poder comenzar a sincronizar, es necesario que definas el servidor dónde Ragnarok se conectará para descargar los Ejercicios. El valor por defecto es:

www.program-ar.com.ar/WS/SvcProgramar.asmx

4.4 *Sincro General*

Existen ejercicios denominados “Globales”. Esto quiere decir que son independientes de curso/ejercicio. Los ejercicios globales son los que puedes descargar para comenzar a usar la aplicación, ya que son de carácter introductorio. Para descargar los ejercicios globales, tenés que

ir a la solapa de “Sincronización” y seleccionar la opción “Sincro General”. Te aparecerá un cuadro de diálogo, donde tendrás que presionar el botón “Aceptar”. Ragnarok se conectará al servidor y descargará los Ejercicios a la carpeta que hayas definido en la “Configuración” -> “Directorio Ejercicios Descargados”.

4.5 Sincro Por Curso

En este tipo de sincronización podrás descargar los ejercicios que estén asociados a un curso determinado. Para descargarte los ejercicios de un curso, tenés que ir a la solapa de “Sincronización” y seleccionar la opción “Sincro Por Curso”. Te aparecerá un cuadro de diálogo, donde tendrás que ingresar el Identificador de curso (será provisto por tu Profesor, o podrás visualizarlo en la página web de Program.AR). Recordá que debés ingresar un número, sin letras ni símbolos.

4.6 Sincro Por Ejercicio

Si no querés descargarte los Ejercicios de un curso, podés descargar un Ejercicio solo. En la solapa “Sincronización”, en la opción “Sincro Por Ejercicio”. En el cuadro de diálogo deberás ingresar el identificador de Ejercicio (ídem que el Curso, deberá ser numérico).

5. Modo Gráfico, Modo Texto

5.1 Modo Gráfico

5.1.1 *Qué son los Tabs?*

Los Tabs (o solapas) son los que puedes visualizar ni bien ingresas al modo gráfico. Cada solapa representa un procedimiento/función o lugar especial de trabajo del lenguaje GarGar. A través de ellas, puedes acceder a cada una, y son independientes entre sí.

5.1.2 *La Barra de Actividades*

La barra de actividades se encuentra a la izquierda de tu pantalla. Estas son las actividades que podrás arrastrar al área de trabajo. Recordá que no todas las actividades pueden ser arrastradas a todas las áreas de trabajo. Cada una de estas actividades representa una instrucción del lenguaje GarGar. La barra de actividades irá cambiando a medida que te posiciones en las diferentes solapas.

En la solapa de “VARIABLES”, sólo podrás visualizar 2 actividades: DeclaracionVariable y DeclaracionArreglo, ya que son las únicas 2 opciones que podrás arrastrar al área de variables.

En la solapa de “CONSTANTES”, sólo podrás visualizar 1 actividad: DeclaracionConstante. No podrás declarar un arreglo como constante porque carece de sentido. Ya entenderás por qué.

En la solapa de “PRINCIPAL”, “SALIDA” y en los Procedimientos/Funciones definidos por vos podrás visualizar todas las actividades.

5.1.3 *El Área de Trabajo*

El área de trabajo es el lugar donde podrás arrastrar las actividades desde la barra izquierda. Existen 2 tipos de áreas de trabajo: El área de trabajo

de declaración de variables y el área de trabajo del cuerpo principal del procedimiento o la función. En el área de trabajo de declaración, sólo podrás arrastrar las actividades DeclaracionVariable y DeclaracionArreglo. En el área de trabajo del cuerpo principal, podrás arrastrar todas MENOS DeclaracionVariable y DeclaracionArreglo.

En el área de trabajo del cuerpo principal, podrás arrastrar las actividades en forma de secuencia, es decir, una sucesión de actividades que se ejecutan una tras otra.

5.1.4 *Cómo se agrega un nuevo Procedimiento?*

Para agregar un nuevo procedimiento, existe un botón especial dentro de las solapas que se llama “PROC +”. Con sólo presionar ese botón, se te desplegará una ventana donde deberás definir el nombre de tu procedimiento. Además, podrás configurar los parámetros que tendrá (en caso de no tener parámetros, no deberás colocar nada):

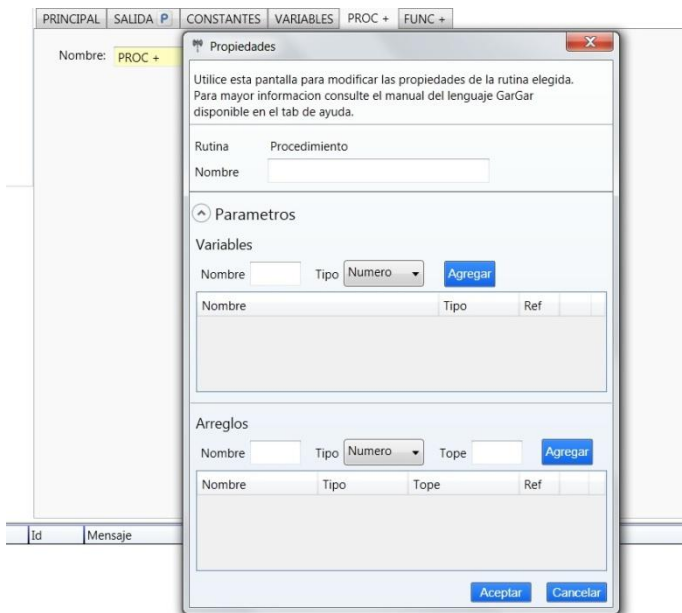


Fig.5 Creación de Procedimiento con sus parámetros

5.1.5 *Cómo se agrega una nueva Función?*

Para agregar una nueva función, existe un botón especial dentro de las solapas que se llama “FUNC +”. Con sólo presionar ese botón, se te desplegará una ventana donde deberás definir el nombre de tu función, pero a diferencia de los procedimientos, las funciones tienen un valor de retorno, así que además del nombre deberás definir el Retorno (qué variable tiene el valor que devolverá la función, y su tipo). Además, podrás configurar los parámetros que tendrá (en caso de no tener parámetros, no deberás colocar nada):

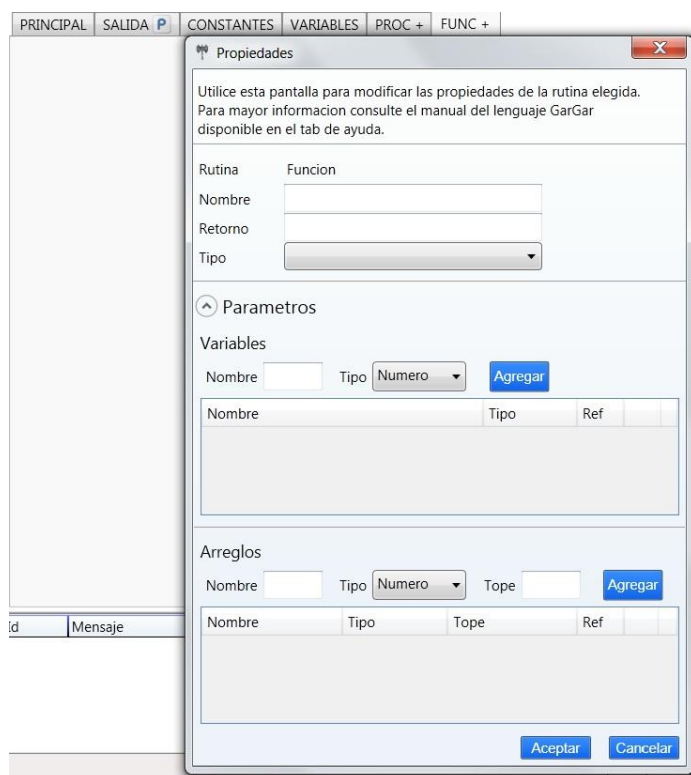


Fig.6 Creación de una Función con sus parámetros

5.1.6 Qué es la “P” que se muestra al lado del procedimiento Salida? Por qué Principal, Variables y Constantes no la tienen?

La “P” viene de “Propiedades”. Aquí podrás visualizar los parámetros de la función/procedimiento, así como también modificarlos. Además podrás modificar el nombre (el nombre también se puede modificar haciendo doble click en el encabezado de la solapa, no es necesario ingresar a las propiedades para modificarlo).

La razón por la cual el procedimiento Principal no la tiene es porque el procedimiento Principal no lleva parámetros. Variables y Constantes no son procedimientos o funciones, por lo que tampoco tienen habilitada la posibilidad de editar.

5.1.7 Arrastrando actividades al área de trabajo

Para arrastrar una actividad al área de trabajo, sólo deberás clicar con el botón izquierdo del mouse la actividad, y sin soltar el botón izquierdo del mouse deberás llevarla al área de trabajo. Ragnarok te avisará cuando el arrastre sea válido, dándote a entender que la operación que estás realizando es la correcta. En el caso de que estés haciendo un arrastre incorrecto, Ragnarok te notificará con un mensaje.



Fig.7 Arrastre de la actividad Mostrar al área de trabajo

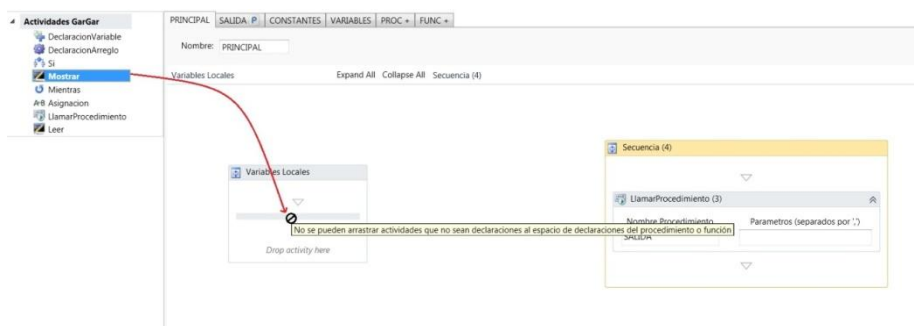


Fig.8 Arrastre de la actividad Mostrar al área de trabajo, pero en este caso Ragnarok indica que no se puede arrastrar porque allí sólo se pueden arrastrar las actividades DeclaracionVariable o DeclaracionArreglo.

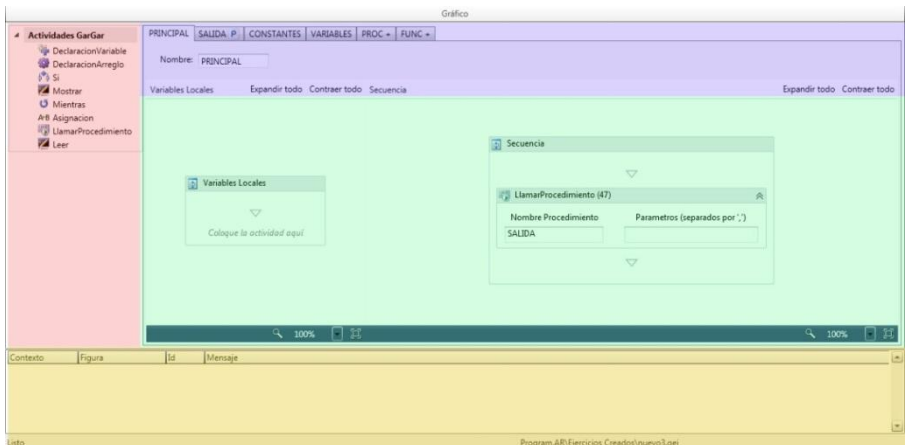


Fig.9 El modo gráfico visto en su totalidad: En el sector rojo encontramos las actividades. En el violeta encontramos las solapas (o tabs). En el verde clarito encontramos el área de trabajo. En el marrón encontramos la barra de mensajes de error de compilación.

5.1.8 Por qué no se pueden arrastrar cualquier tipo de actividades a cualquier área de trabajo?

La razón por la cual no se pueden arrastrar cualquier tipo de actividades a cualquier área de trabajo es porque GarGar como lenguaje limita los lugares donde se pueden utilizar las sentencias. Como cualquier lenguaje estructurado, las declaraciones de variables globales, constantes y variables locales sólo pueden hacerse en un lugar determinado (ver el Manual de GarGar para mayor información). Por ejemplo, GarGar no admite que se declaren variables en el medio del procedimiento. Para ello existe una sección especial entre el Procedimiento y el Comenzar.

5.1.9 Puedo copiar y pegar figuras? Las puedo cambiar de orden?

Claro que sí. Ragnarok te provee de un menú contextual, al cual podés acceder con el botón derecho del mouse. También existen atajos que

podes utilizar sin la necesidad del mouse y son los más conocidos (Ctrl+X para Cortar, Ctrl+C para copiar, Ctrl+V para pegar y Suprimir para borrar).

Para cambiar las figuras de posición, basta con arrastrarlas de un lugar a otro. Recordá que siempre deben hacerse dentro de las posibilidades de la actividad (no podrás arrastrar una actividad del estilo “Mostrar” hacia el área de declaración de variables).

5.1.10 Exportando actividades como imágenes (para que las compartas por mail con tus compañeros... o para que hagas trampa, da igual)

Para exportar una actividad como imagen, deberás posicionarte sobre la actividad que querés exportar, hacer click con el botón derecho del mouse y seleccionar la opción “Salvar como imagen...”. Ragnarok te preguntará dónde querés almacenar la imagen, sólo deberás indicarle un nombre y una ubicación y listo.

5.1.11 Los errores de compilación

Cuando compiles tu programa, Ragnarok te informará (en el caso de que los haya) los errores que cometiste haciendo tu programa. Los errores los podés visualizar en la barra inferior, que se divide en 4 columnas:

- Contexto: El nombre del procedimiento donde se encuentra el error.
- Figura: Nombre de la figura en la que se encuentra el error.
- Id: Identificador de la figura en la que se encuentra el error (este identificador es único para cada actividad del programa)
- Mensaje: Mensaje descriptivo del error.

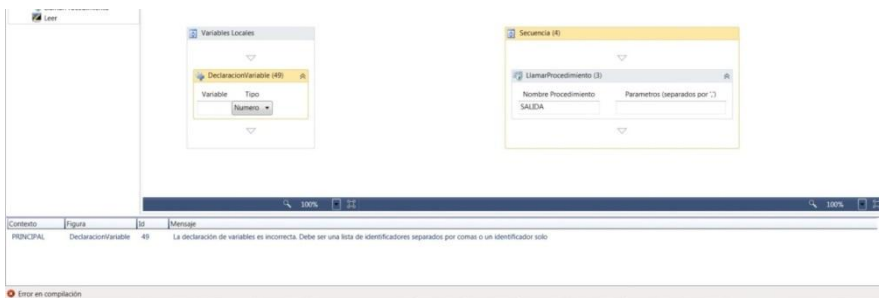


Fig.10 Error de compilación emitido por Ragnarok. Indica que la declaración de la variable es incorrecta

5.1.12 Las actividades de Ragnarok

Las actividades son pequeñas cajas con contenido que vamos a encontrar a lo largo de la aplicación. Cada actividad representa algún tipo de instrucción del lenguaje GarGar. A medida que vayas trabajando con las actividades descubrirás cómo se traducen a GarGar. Comencemos a enumerarlas.

5.1.12.1 DeclaracionArreglo

Esta actividad, como su nombre lo indica, permite declarar un arreglo. Como habrás leído en el manual de GarGar, un arreglo es un montón de variables juntas del mismo tipo. Sólo podrás arrastrar esta actividad en aquellas áreas de trabajo que sean de declaraciones. Si intentas arrastrarla fuera de ellas, Ragnarok te informará que no es posible.

Veamos qué campos tenés que completar:

- Arreglo**: Aquí va el nombre
- Tipo**: De qué tipo es el arreglo (Numero, Texto o Booleano)
- Tope**: Cantidad máxima de elementos que tendrá el arreglo. Recordá que acá podés usar el nombre de una constante declarada en el lugar de constantes, o algún número entero (no decimal).



Arreglo	Tipo	Tope
<input type="text"/>	Numero ▼	<input type="text"/>

Fig.11 La actividad DeclaracionArreglo.

5.1.12.2 *DeclaracionVariable*

Esta actividad, también como su nombre lo indica, sirve para declarar variables (sí, te voy a explicar cada una, por más que te parezca obvio). Una variable es un recipiente donde se colocan valores, con un tipo de dato determinado.

Los campos obligatorios para completar son:

-**Variable**: Nombre de la variable

-**Tipo**: Tipo de dato de la variable (Numero, Texto o Booleano)



Variable	Tipo
<input type="text"/>	Numero ▼

Fig.12 La actividad DeclaracionVariable.

5.1.12.3 DeclaracionConstante

Esta actividad sirve para declarar constantes. Las constantes son parecidas a las variables, con la diferencia de que su valor no puede ser modificado (es decir, es de sólo lectura).

Los campos obligatorios son:

- Constante**: Nombre de la constante
- Tipo**: Tipo de dato de la constante (Numero, Texto o Booleano)
- Valor**: El valor que tendrá la constante (debe coincidir con el tipo obviamente)

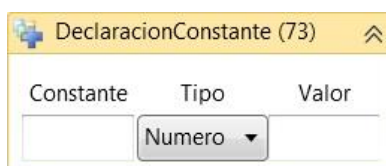


Fig.13 La actividad DeclaracionConstante.

5.1.12.4 Si

Esta actividad representa un condicional. Un condicional se caracteriza por evaluar una condición determinada y tomar acciones en caso de que esa condición sea verdadera o falsa.

Por ejemplo, tomemos el caso de la siguiente condición

$$\text{numero} < 10$$

Si el contenido de la variable “numero” es menor que 10, entonces se ejecutarán las actividades que arrastres a la izquierda (En el lugar donde dice “Entonces”). Por el contrario, si el valor de “numero” es mayor o igual a 10, entonces se ejecutarán las actividades que arrastres a la derecha (En el lugar donde dice “Sino”). Como te habrás dado cuenta,

sólo se ejecutan las actividades de la izquierda o de la derecha, pero nunca las 2 al mismo tiempo (si sucede una cosa, no sucede la otra).

Esto es como el ABC de lógica: *“Si llueve voy al cine, sino me quedo en casa.”*

Solo pasa una de las cosas, o voy al cine, o me quedo en casa (todo depende de si llueve o no), pero no hago las 2 cosas al mismo tiempo.

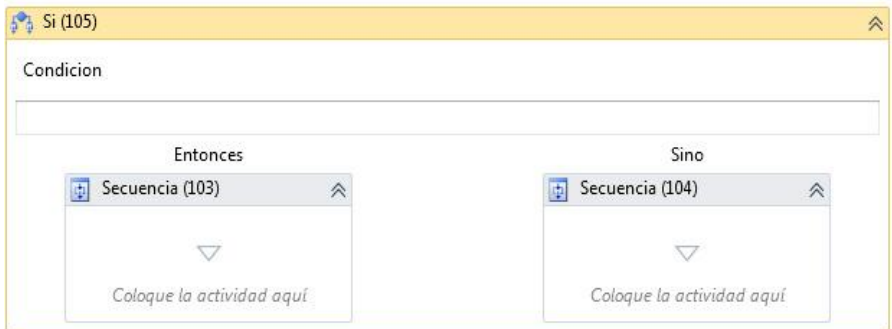


Fig.14 La actividad Si. Debe definirse la Condición del Si, y las acciones en caso de que la condición definida sea Verdadera(en el área de la izquierda) o Falsa(en el área de la derecha)

5.1.12.5 *Mostrar*

Esta actividad se utiliza para imprimir por pantalla. Aquí se pueden mostrar todos los tipos de dato soportados por GarGar (te los recuerdo por las dudas, son Numero, Texto y Booleano).

Existe además la opción “Con Pausa”, esto significa que cada impresión por pantalla esperará a que vos presiones una tecla para continuar.

Si lo que querés es mostrar texto, debés completar el campo entre comillas simples (ejemplo: ‘Hola Mundo’). Si son números o booleanos, deben ir sin comillas. Existen casos más complejos, donde se pueden concatenar los tipos de dato. Te lo explico más fácil con un ejemplo:

Supongamos que tenemos declarada una variable que se llama “numero” (sí, otra vez con el mismo ejemplo). Y queremos mostrar un texto al lado de la variable, algo así como “La variable numero equivale a <valor>”. Estas 2 cosas se concatenarían con el carácter “,” de esta manera:

‘La variable numero equivale a ‘ , numero



Fig.15 La actividad Mostrar. Debe definirse qué mostrar. Si es un texto, debe ir entre comillas simples. Se pueden concatenar texto y variables, a través de “,”. La opción de “Pausa” permite emitir el valor por pantalla y que la aplicación espere hasta que se presione una tecla para continuar.

5.1.12.6 Mientras

Esta actividad se comporta como una actividad que se repite. Cuántas veces? Las que quieras. Cómo definís eso? En la condición! Veámoslo con un ejemplo:

Supongamos que tenemos una variable declarada que se llama “contador” de tipo *Numero* y queremos que el Mientras se repita 10 veces. Entonces debemos poner en la condición

$$\text{Numero} < 10$$

Luego, en el cuerpo del Mientras (donde dice “Hacer”) debemos incrementar la variable *numero* en 1 unidad, a través de la actividad Asignación (no te preocupes, está inmediatamente abajo) y ponemos en el Lado Izquierdo a la variable *numero* y en el Lado Derecho ponemos *numero + 1*. De esta forma, la variable *numero* se irá incrementando hasta que ya no se cumpla la condición de que sea menor a 10. Ahí es cuando se termina el Mientras.

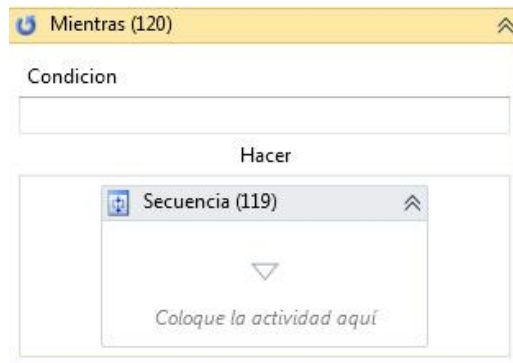


Fig.16 La actividad Mientras. Debe definirse la condición de la iteración y el cuerpo (es decir, que debe repetirse si se satisface la condición).

5.1.12.7 Asignación

Lo prometido es deuda. Arriba te mencioné esta actividad pero no te la expliqué. Bueno, acá vamos.

La asignación tiene 2 partes: El Lado Izquierdo y el Lado Derecho. En el Lado Derecho vamos a colocar QUÉ queremos asignar, y en el Lado Izquierdo vamos a colocar a QUIÉN se lo queremos asignar.

Siguiendo el caso del ejemplo anterior, donde dijimos que íbamos a incrementar la variable *número* en 1 unidad, lo que deberíamos poner es:

-**Lado Izquierdo:** *numero*

-**Lado Derecho:** *numero + 1*

Fig.17 La actividad Asignacion. Debe definirse el lado izquierdo (asignado) y el lado derecho (asignante).

5.1.12.8 LlamarProcedimiento

Esta actividad permite llamar (o Invocar como se le dice en la jerga) un procedimiento. Qué procedimiento? Cualquiera que hayas declarado o que sea propio de GarGar, como ser el Procedimiento SALIDA.

-**Nombre Procedimiento:** Nombre del procedimiento que se desea llamar

-**Parámetros:** Todos los parámetros que recibe ese procedimiento, separados por “,”

Fig.18 La actividad LlamarProcedimiento. Debe definirse el nombre del procedimiento al

cual se desea invocar y sus parámetros separados por “,”(en caso de que los tenga, sino debería dejarse en blanco).

5.1.12.9 Leer

Esta actividad te permite leer un valor por pantalla. Se utiliza cuando se espera que el usuario de tu aplicación ingrese un valor para que vos hagas algo. Lo que debés colocar en la casilla es el nombre de la variable a la que se le asignará el valor ingresado por el usuario en la pantalla.

LEER *numero*

Supongamos que el usuario de tu programa ingresa el valor 7. La variable *numero* tendrá ese valor.



Fig.19 La actividad Leer. Debe definirse la variable a la cuál se asignará lo ingresado por pantalla.

5.2 Modo Texto

5.2.1 *El editor de texto (te preguntas si tenés que escribir acá? En el modo gráfico también)*

El editor de texto de Ragnarok es como un procesador de texto, a diferencia de que acá no existen los errores de ortografía, sino que existen los errores de compilación (ojo, no estoy diciendo que vayas a cometer un error, te cuento cómo es el panorama).

Aquí es donde Ragnarok te permite escribir tú código. Te preguntarás qué podés escribir? La respuesta es cualquier cosa que sea válido para GarGar. En realidad, podés escribir lo que quieras, el problema aquí será compilar.

5.2.2 *Las palabras coloreadas*

Este editor de texto cuenta con una particularidad: Colorea palabras. Qué palabras colorea? Aquellas palabras que son RESERVADAS, esto significa, palabras que GarGar interpreta como palabras especiales.

No solo eso, además las palabras se colorean de diferente manera. Veamos qué significa cada color:

Las palabras coloreadas en **AZUL** son aquellas identificadas por GarGar como estructurales (por ejemplo, **PROCEDIMIENTO**, **COMENZAR**, **CONSTANTES**, etc [para ver un listado completo, tenés que referirte al manual de GarGar]).

Las palabras coloreadas en **TERRACOTA** son aquellas identificadas como tipos de datos de GarGar(**NUMERO**, **TEXTO**, **BOOLEANO**, **ARREGLO**).

Las palabras coloreadas en **GRIS** son aquellas que tienen que ver con la entrada/salida de datos (**MOSTRAR**, **LEER**).

Las palabras coloradas en **NARANJA** son aquellas que representan cadenas literales de texto (Por ejemplo, 'Esta es una cadena de texto').

Las palabras coloreadas en **MARRON** son aquellas que representan números (Por ejemplo, 32 o 9437 o 542.393).

Las palabras coloreadas en **ROJO** son aquellas que representan funciones integradas de GarGar (Por ejemplo, **COSENO**, **ESPAR**, **TRUNCAR**, etc).

5.2.3 Buscar

Como todo editor de texto, Ragnarok te provee la posibilidad de buscar texto. La búsqueda podés hacerla de diferentes maneras: Distinguiendo mayúsculas de minúsculas, por expresiones regulares (si no sabés qué son, ni te compliques), hacia arriba, etc.

Para acceder al panel de búsqueda, deberás clickear en el menú superior la opción “Buscar” (aquella que tiene una lupa):

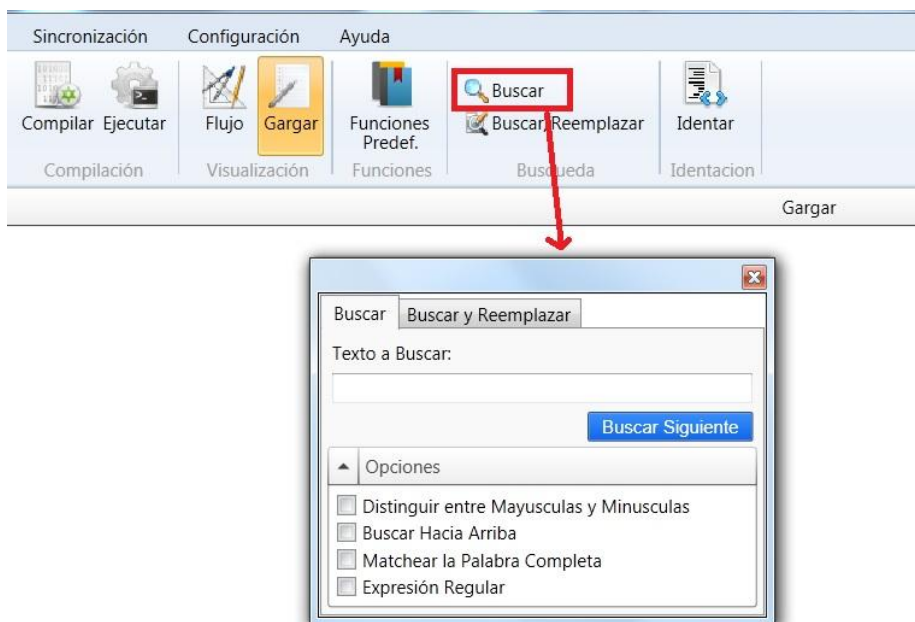


Fig.20 Desde el menú superior (estando en modo Texto) accedemos mediante “Buscar”

Como podemos visualizar en la Figura 20, la ventana de diálogo nos pide que completemos la información para poder realizar la búsqueda. Además de “Texto a Buscar” (que es en esencia lo que queremos buscar),

nos permite elegir 4 opciones (las selecciones pueden combinarse, no son excluyentes):

- *Distinguir entre Mayúsculas y Minúsculas*: Realiza una búsqueda donde mayúsculas y minúsculas se discriminan (no es lo mismo buscar “Procedimiento” que “PROCEDIMIENTO” con esta opción activada).
- *Buscar Hacia Arriba*: Con esta opción activada, la búsqueda se realiza por encima de la posición actual del cursor. Supongamos que tenemos el cursor en la línea 33 de nuestro código, con esta opción se buscará desde la línea 33 hasta la línea 1, todo lo que esté por debajo de la línea 33 no se toma en cuenta en la búsqueda.
- *Matchear la Palabra Completa*: Esta opción permite realizar una búsqueda por palabra. Por ejemplo, si buscamos “Pro”, si no tenemos esta opción activada encontraremos la palabra “Procedimiento”. Pero si sólo queremos buscar la cadena exacta “Pro”? Bueno, esta es la opción que hace eso.
- *Expresión Regular*: Con esta opción activada, lo que se busca son patrones dentro del texto. La idea de este manual no es explicar Expresiones Regulares ya que tomaría un montón de tiempo además de que no es el objetivo. Por ejemplo, si tenemos esta opción activada y colocamos en “Texto a Buscar” lo siguiente: “\d”, Ragnarok buscará todos los caracteres que sean dígitos (Para una mayor referencia de expresiones regulares, visitar http://es.wikipedia.org/wiki/Expresi%C3%B3n_regular)

Una vez que se estableció qué se desea buscar y con qué opciones, procederemos a apretar el botón “Buscar Siguiente”. Si existe alguna coincidencia, nos pintará la palabra encontrada, caso contrario nos informará que no existe ninguna cadena que coincida con los parámetros de búsqueda.

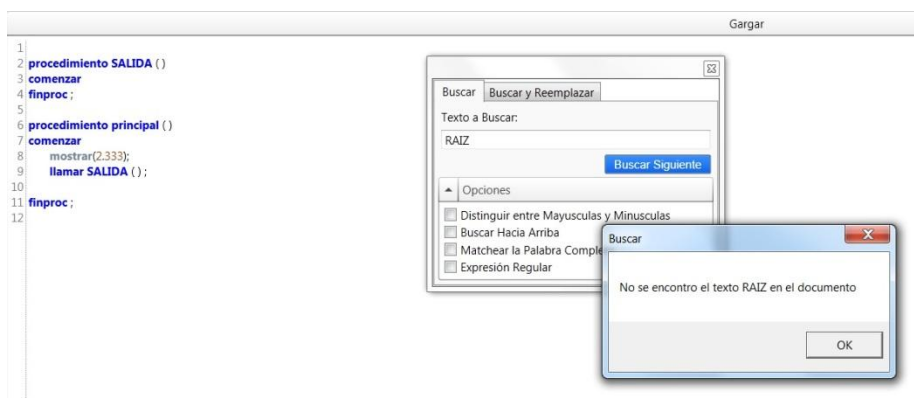


Fig.21 Una búsqueda que no produce resultados.

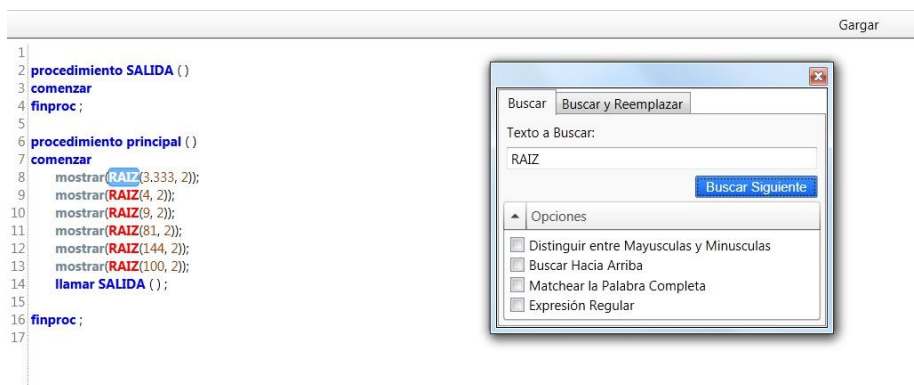


Fig.22 Una búsqueda que sí produce resultados, vemos la palabra RAIZ pintada.

A medida que vayamos presionando el botón “Buscar Siguiente”, Ragnarok irá pintando las coincidencias. Cuando se llega al final del documento, se vuelve a comenzar desde el principio del mismo.

5.2.4 Buscar y Reemplazar

Además de buscar, Ragnarok permite Buscar y Remplazar. Esto nos permite buscar una palabra o palabras y remplazarlas por otra. Las opciones provistas son las mismas que para búsqueda.

Podemos acceder desde el cuadro de búsqueda, en la solapa superior que dice “Buscar y Reemplazar” o directamente desde el menú superior.

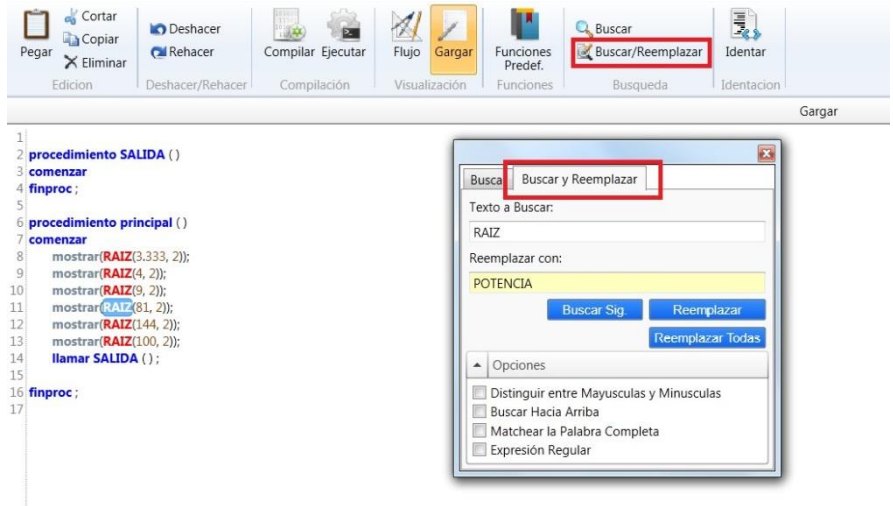


Fig.23 Reemplazar la palabra RAIZ por POTENCIA.

Los botones disponibles en este cuadro de diálogo son:

- Buscar Sig.: Busca la siguiente coincidencia (sin reemplazar nada).
- Reemplazar: Reemplaza la primer coincidencia encontrada.
- Reemplazar Todas: Reemplaza todas las coincidencias encontradas.

5.2.5 Identificar (Iden-qué?)

La indentación es una buena práctica dentro de la programación. Lo que quiere decir es que el código respete los espacios y tabulaciones necesarias para la correcta lectura del código.

```
1 PROCEDIMIENTO PRINCIPAL()
2 VAR num : NUMERO;
3 COMENZAR
4 LEER num;
5 SI (num < 10) ENTONCES
6 MOSTRAR('El número es menor que diez');
7 SINO
8 MOSTRAR('El número es mayor o igual que diez');
9 FINSI;
10 FINPROC;
```

Fig.24 Ejemplo de un código sin indentación.

En el ejemplo de la Figura 24, vemos un código no indentado. Para que el código quede indentado, debemos colocar tabulaciones al comienzo de cada bloque de código. Estas tabulaciones se irán anidando a medida que aparezcan bloques dentro de otros.

Pero no tenés que preocuparte por esto, ya que Ragnarok tiene una herramienta que lo hace automáticamente por vos. La idea es que la utilices en todos los programas que construyas, y que vayas aprendiendo cómo se indenta, así a medida que vas avanzando con tu aprendizaje vas a poder indentar el código vos solo.

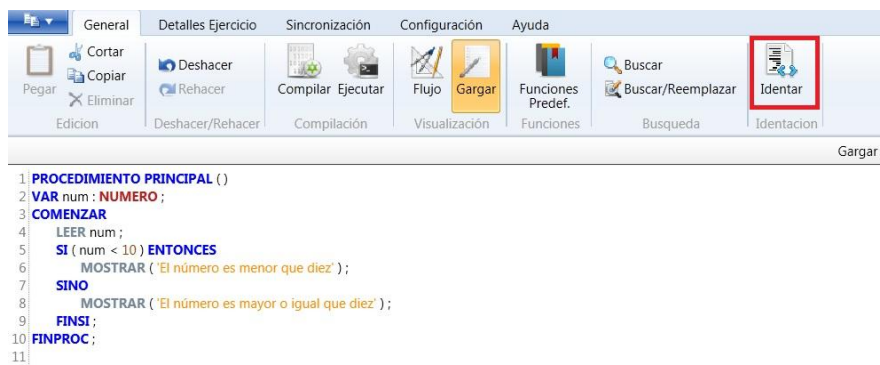


Fig.25 Ejemplo del mismo código que el de la Figura 24 pero indentado a través de la opción “Indentar”.

5.2.6 Copiar/Cortar/Pegar

Ragnarok también tiene las funcionalidades básicas de Copiar, Cortar y Pegar texto.

Podés acceder a estas funcionalidades a través del menú superior, en la categoría de “Edición”. O también podés hacerlo mediante los atajos de teclado mundialmente conocidos:

- Ctrl+C : Copiar
- Ctrl+V: Pegar
- Ctrl+X: Cortar

5.2.7 Rehacer/Deshacer (si cualquier editor lo tiene, por qué no?)

La posibilidad de poder Deshacer o Rehacer es fundamental cuando se trabaja con texto.

Podés acceder a estas funcionalidad a través del menú superior, en la categoría de “Deshacer/Rehacer”. También podés hacerlo mediante atajos:

- Ctrl+Z: Deshacer
- Ctrl+Y: Rehacer

5.2.8 Los errores de compilación

Los errores de compilación se dan cuando el compilador detecta que cometiste algún error cuando escribiste el programa. No te preocupes, al principio vas a recibir muchos errores de compilación. Con el tiempo vas a ir mejorando.

Estos errores se marcan subrayando la línea conflictiva en rojo. En la barra inferior, vas a visualizar el error, en qué línea se produjo y el motivo. Haciendo click en cada uno de los errores de la barra inferior, te va a posicionar sobre el error.



Fig.26 Un programa con un error de compilación. Abajo visualizamos el motivo, subrayado en rojo visualizamos la línea causante del error.

5.3 Puedo pasar de uno al otro? (no te cases con ninguno, usá el que quieras, después cambiás)

Claro que sí! Podés pasar del modo texto al gráfico cuando quieras, pero tené en cuenta que el programa debe COMPILAR antes de hacerlo. Esto te resultará un poco incómodo al principio, y no es un capricho de la aplicación, creeme. La razón por la cual debe compilar antes de cambiar el modo es que al compilar, Ragnarok arma una estructura de cómo es tu programa. Y sin esta estructura Ragnarok no puede ni escribir ni dibujar tu código.

Al pasar de un modo a otro, vas a notar que Ragnarok tarda un poco. Lo que está haciendo es armar la estructura para que vos puedas trabajar.

Para pasar de un modo a otro, tenés que apretar los botones del menú superior que dicen “Flujo” o “GarGar”. El “Flujo” es el modo gráfico, donde arrastras actividades. “GarGar” es el modo texto, en el cuál escribís todo el código.

La idea de Ragnarok es que vos al principio siempre trabajes en modo gráfico, y que cada tanto te pases al modo texto. De esta forma podés ir familiarizándote con el lenguaje. Hasta que llegue un día en el cuál ni tengas que abrir el modo gráfico, en donde sólo vas a trabajar en modo texto.

Y más adelante, la idea es que empieces a trabajar en algún lenguaje de programación más conocido (Pascal, C, C++...)

6. ¿Está bien, hice un nuevo ejercicio, y ahora qué hago? (que no cunda el pánico)

6.1 *Nuevo Ejercicio*

Bueno, vamos a explicarte como armar un programa sencillo que muestre por pantalla el texto “Hola Mundo”. Vamos a verlo de la forma gráfica y de la forma texto.

Abrí la aplicación Ragnarok, dirigitte al menú superior izquierdo, y presiona el botón que se encuentra a la izquierda en azul. Se te va a desplegar un menú, seleccioná “Nuevo” -> “Crear un nuevo ejercicio”.

Se te va a desplegar una ventana (ver Figura 4). Vamos a elegir el “Modo Gráfico”. Luego te va a aparecer el área de trabajo. Vamos a ir a la solapa que dice “SALIDA” y hacemos un click para seleccionarla.

Ahora vamos a tomar la actividad “Mostrar” desde la barra izquierda, presionamos con el botón izquierdo del mouse, y sin soltar, la arrastramos hasta el área donde dice “Secuencia”. Vas a ver que la aplicación va a ejecutar una animación que te va a indicar que podés arrastrar la actividad ahí. Ahora soltá el botón izquierdo, y la actividad Mostrar queda ahí.

Dentro del campo de texto, colocá entre comillas simples ‘Hola Mundo’. Debería quedarte algo así:

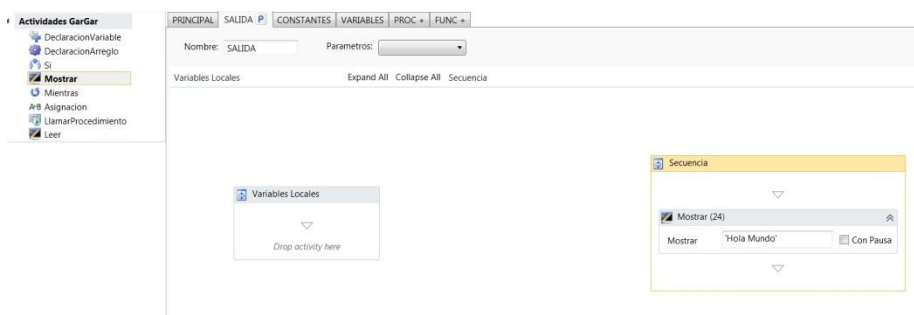


Fig.27 Ejemplo del programa que muestra por pantalla “Hola Mundo” en modo gráfico.

6.2 Escribir un programa

Ahora, vamos a compilarlo. Acá tenés 2 opciones:

- Ir al menú superior, y en la categoría que dice “Compilación” presionar el botón que dice “Compilar”.
- Presionar la tecla F3 (atajo para “Compilar”)

Elegí algún método para compilar. El que más cómodo te resulte. Una vez que elegiste, Ragnarok va a compilar tu programa. En caso de que haya salido todo bien, vas a ver un tilde verde con una leyenda que dice “Compilación Correcta” debajo de todo a la izquierda. Si no sucede esto, significa que hiciste algo mal, volvé a empezar desde el principio.

6.3 Compilar un programa

Una vez que compilamos nuestro programa, lo vamos a ejecutar. Para ejecutarlo, tenés 2 opciones nuevamente:

- Ir al menú superior, en la categoría que dice “Compilación” presionar el botón que dice “Ejecutar”
- Presionar la tecla F4 (atajo para “Ejecutar”)

6.4 Ejecutar un programa

Cuando ejecutes, te va a aparecer una pantalla en negro que simplemente dice “Hola Mundo”. Debajo figura una leyenda que te indica que presiones cualquier tecla para finalizar la ejecución. Ragnarok hace esto para que puedas ver tus resultados, de otra forma, no los podrías visualizar (la computadora hace las cosas MUY rápido).

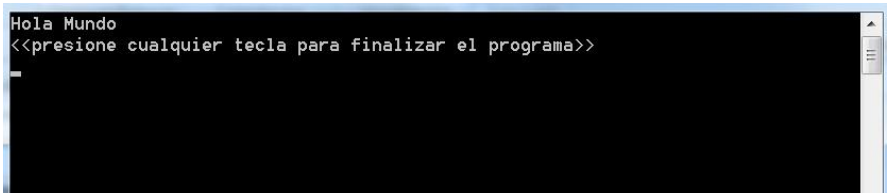


Fig.28 Fin de la ejecución del programa “Hola Mundo”.

6.5 Fin de la ejecución

Cuando presiones cualquier tecla, esa pantalla negra se va a cerrar y te va a aparecer esta otra ventana en Ragnarok:

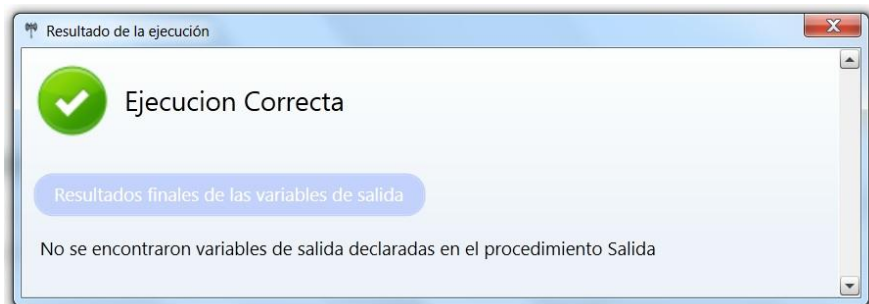


Fig.29 Ragnarok informa que la ejecución del programa finalizó correctamente.

Ahora, volvé al ejercicio y fijate que pasa si pasas al Modo Texto. Vas a ver que tu programa ahora va a estar escrito. Probá agregarle otra cosa, y cambiar al Modo Gráfico. Apareció una nueva actividad, verdad? Así funciona esto, la idea es ir de un modo al otro para ver cómo queda lo que hicimos (acordate que siempre tiene que compilar el programa para pasar de modo).

Si por alguna razón no pudiste pasar de un modo a otro, revisá que errores te está informando la aplicación. Al principio te va a costar identificar el error cometido, pero a medida que lo uses te va a costar cada vez menos.

En la Figura 29, aparece la leyenda “No se encontraron variables de salida declaradas en el procedimiento Salida”. Justamente porque no se declararon variables en el procedimiento Salida, sólo mostramos un “Hola Mundo”. Cuando declares variables en el procedimiento Salida, te van a aparecer ahí.



Fig.30 La finalización de una ejecución con una variable “num” pasada por parámetro al procedimiento Salida cuyo valor es 54.

7. Las funciones predefinidas del lenguaje GarGar

Algunas funciones a veces resultan esenciales para hacer un programa. GarGar incorpora funciones que te van a ayudar a resolver tus problemas (o al menos, facilitarte el trabajo).

En el manual de GarGar podés encontrar todas las funciones que se incorporan dentro del lenguaje (ESPAR, RAIZ, POTENCIA, REDONDEAR, etc).

En caso de que no te acuerdes cómo usarlas o cuáles eran, podés consultarlas dentro de Ragnarok. En el menú superior, vas a ver un botón que dice “Funciones Predef.”. Te va a aparecer una pantalla con todas las funciones y procedimientos que GarGar incorpora, junto con su modo de utilización y sus parámetros.

Para usar estas funciones, sólo tenés que escribirlas. Si estás en Modo Texto, te aparecerá la palabra coloreada en Rojo. En modo gráfico no se pinta de ningún color.

Acordate que estas funciones siempre devuelven un valor, no podés usarlas solas (es decir, sin asignarlas, o sin pasarlas como parámetro a algún procedimiento o función).

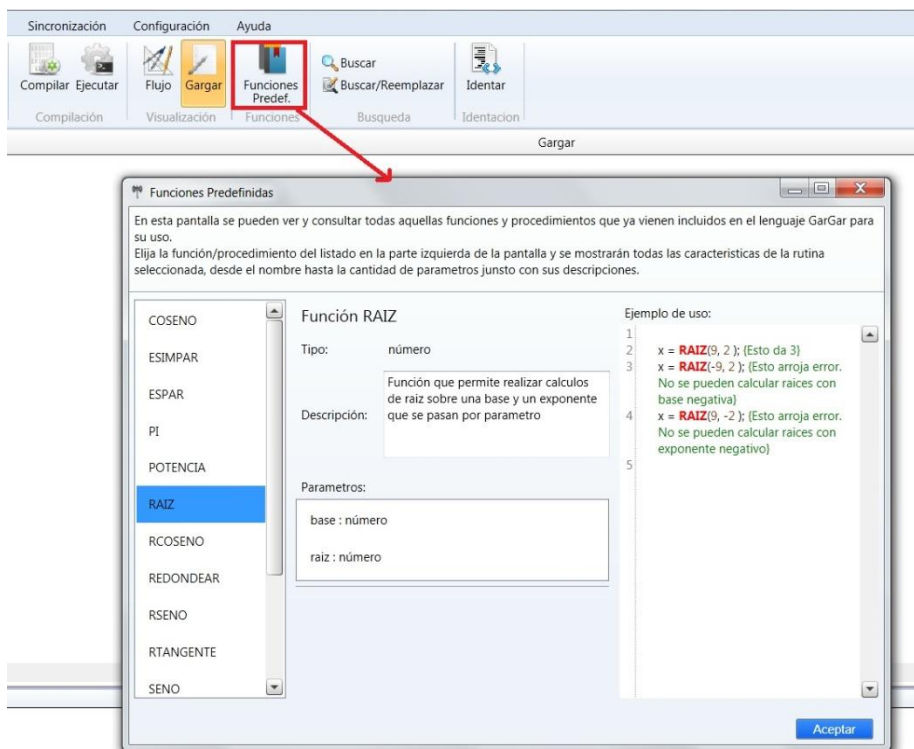


Fig.31 El diálogo de Funciones Predefinidas donde se visualiza la función RAIZ.

Este cuadro de diálogo es sólo informativo y no tiene otra funcionalidad. Básicamente es para que no tengas que ir al manual a revisarlas.

8. Muy lindo, codifiqué el programa, ¿cómo sé si está bien?

8.1 Tests de Prueba:

8.1.1 *¿Qué son?*

La aplicación Ragnarok, en conjunto con el compilador GarGar te dan la posibilidad de saber si el código GarGar que estas armando es correcto en su sintaxis (o sea, en como lo escribiste) y en su semántica (o sea, que tiene sentido lo que escribiste).

Lo único que el compilador no puede hacer, es darse cuenta si lo que vos escribiste cumple con el fin con el que lo estas escribiendo. ¡Pero a no temer! Ragnarok cuenta con los llamados Test de Prueba, que posibilitan probar tu código con valores predefinidos de entrada y de salida. Todo esto con el fin de corroborar que el código que vos escribiste cumple con su cometido original.

La idea de todo esto, es que quien crea un ejercicio, cree además algunos tests de prueba, para que quien lo resuelva, pueda fijarse si lo que hizo esta bien.

Por ejemplo, si tuviésemos un ejercicio que fuera “duplicar un número”, podríamos crear un test de prueba que tome como valor de entrada 3 y como valor de salida 6. De esta manera, quien resuelva el ejercicio sabe que su código GarGar para “duplicar un número” tiene que ser 6, cuando el número ingresado es 3.

NOTA: Los tests de prueba deben ser correctos si se quiere guardar el ejercicio en formato para subir a la web (correctos significa que todas sus variables de entrada y salida existan en el programa actual).

8.1.2 Crear un test de prueba

Bueno, vamos a mostrar cómo podemos crear un test de prueba. Esta funcionalidad está representada en una serie de 7 pantallas.

La funcionalidad de creación de test de prueba está disponible únicamente en los ejercicios.

NOTA: Ragnarok siempre hace la sugerencia de que el código este correctamente indentado para realizar esta función. Siempre conviene hacerlo para garantizar el correcto funcionamiento de los mismos.



Fig.32 En que espacio se encuentra el acceso a los tests.

8.1.2.1 Paso 0: Cómo funcionan los test

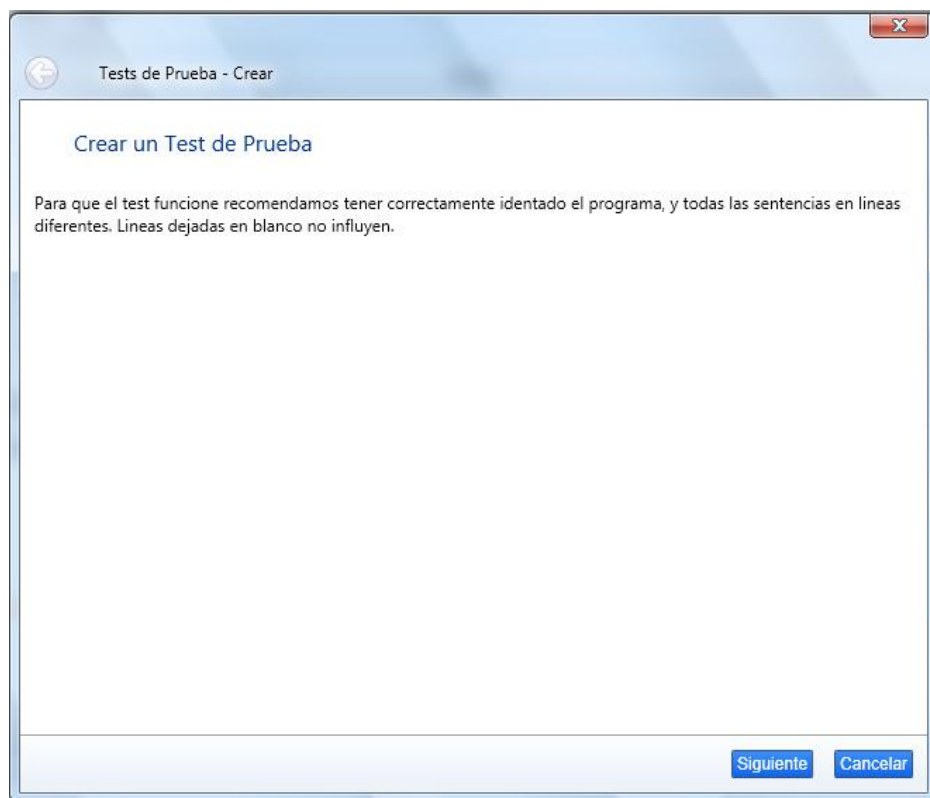


Fig. 33 Dialogo Principal de Creación de Tests

Esta pantalla es meramente explicativa. Aclara el tema de que el código debe estar correctamente indentado y que las líneas vacías no son tenidas en cuenta.

8.1.2.2 Paso 1: Elegir las variables de entrada

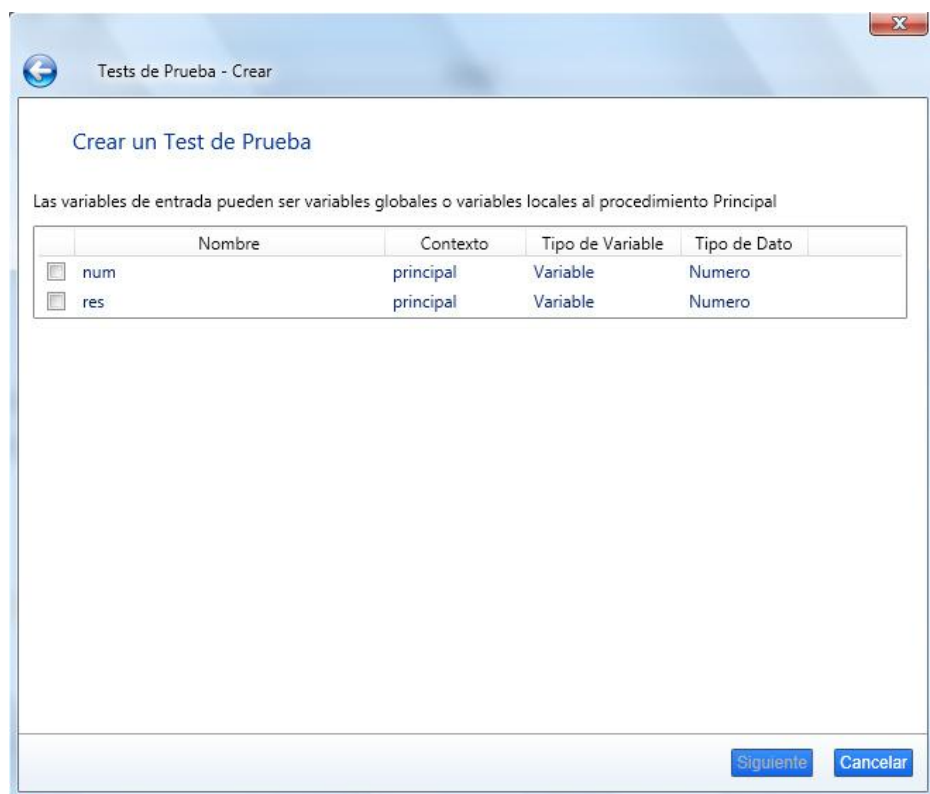


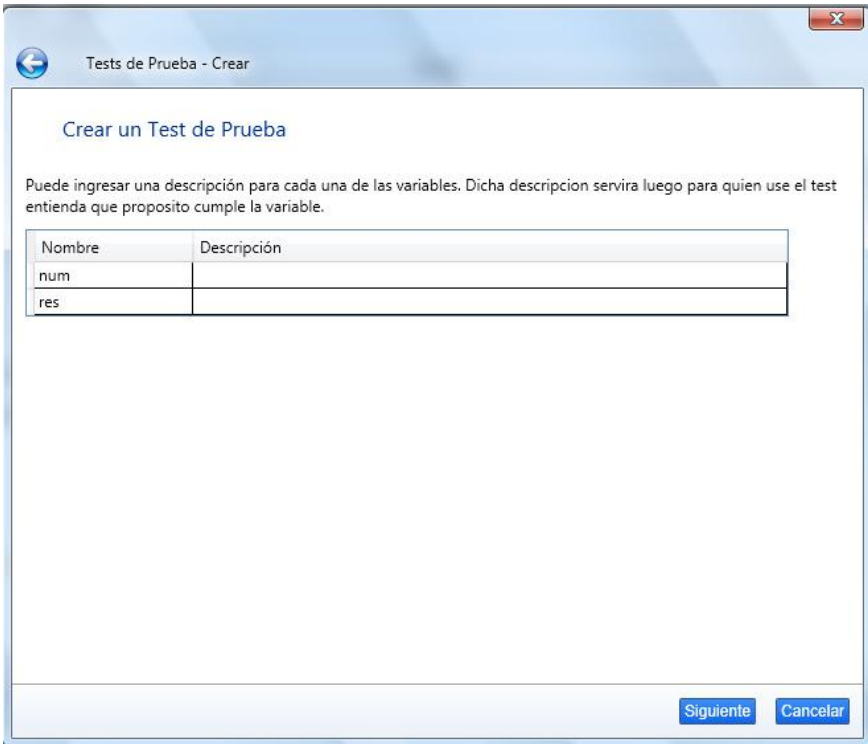
Fig. 34 Dialogo de Selección de Variables de Entrada

En esta pantalla ya empiezan las decisiones. Lo que ves acá es el listado de todas las variables que en tu programa califican para ser tomadas como variables de entrada. El texto es claro: solo aquellas que sean globales o variables locales del procedimiento principal pueden serlo.

Las variables que elegís en este paso son las que tendrán sentido luego para las variables de salida. En otras palabras, estas eligiendo las variables de entrada que condicionan el resultado final de una variable de salida.

Tenés que elegir al menos una variable de entrada para poder pasar al siguiente paso.

8.1.2.3 Paso 2: Descripción de las variables de entrada



Tests de Prueba - Crear

Crear un Test de Prueba

Puede ingresar una descripción para cada una de las variables. Dicha descripción servirá luego para quien use el test entienda que propósito cumple la variable.

Nombre	Descripción
num	
res	

Siguiete Cancelar

Fig. 35 Dialogo de Ingreso de Descripción de Variables Seleccionadas en el paso anterior

Esta pantalla es opcional. Lo que puedes hacer es colocarle una descripción informal a cada variable, con el fin de que quien ejecute el test, sepa que significa esa variable, y que es lo que contiene como valor.

Si bien es opcional, se recomienda aunque sea poner unas palabras a modo informativo. ¡Quien ejecute el test te lo va a agradecer!

8.1.2.4 Paso 3: Valores iniciales de las variables de entrada

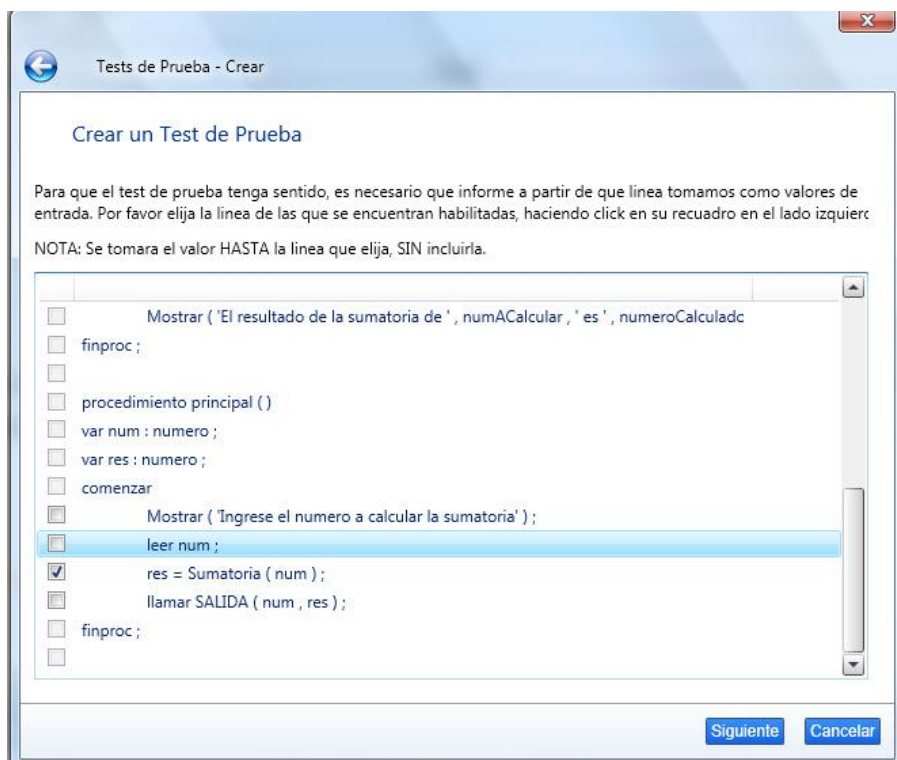


Fig. 36 Dialogo de Selección de Valores iniciales de Variables de entrada

A prestar atención que esta pantalla es un poco más compleja que las anteriores. El fin de la misma, es decir, “bueno programa, en esta línea que yo te voy a decir es cuando yo ya tengo los valores cargados en mis variables de entrada, que hacen que después las variables de salida tengan el valor que tienen”.

La pantalla muestra todo el código, y cada línea tiene una casilla para ser tildada en el lado izquierdo. Solo unas pocas casillas están habilitadas, y son aquellas que corresponden a líneas del procedimiento principal. Si

bien esto puede sonar un poco restrictivo (y quizás lo sea), esta estandarización es la única manera de resolver esta problemática.

Solo una línea puede ser elegida, y se tomará el valor que tengan las variables de entrada, HASTA la línea que se elija SIN incluirla.

Este paso es crítico, ya que si se elige mal la línea, las variables pueden que no contengan los valores aun, o que estos hayan sido modificados ya por el programa, así que te recomendamos que pongas especial atención en este punto.

No se puede avanzar a la siguiente pantalla hasta que no se elija una línea.

8.1.2.5 Paso 4: Elegir las variables de salida

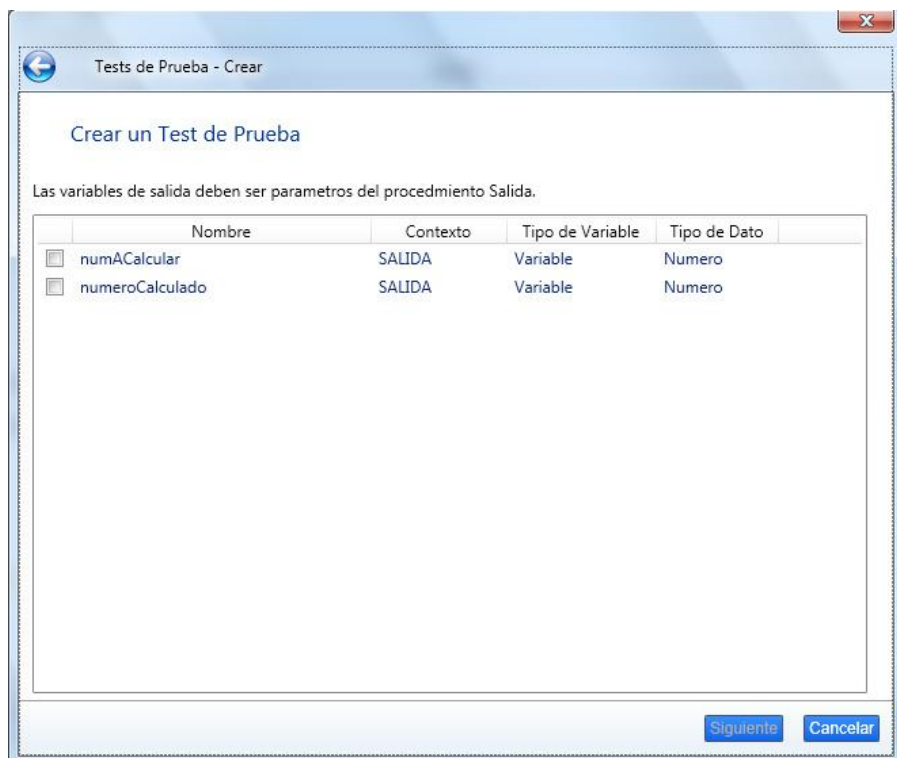


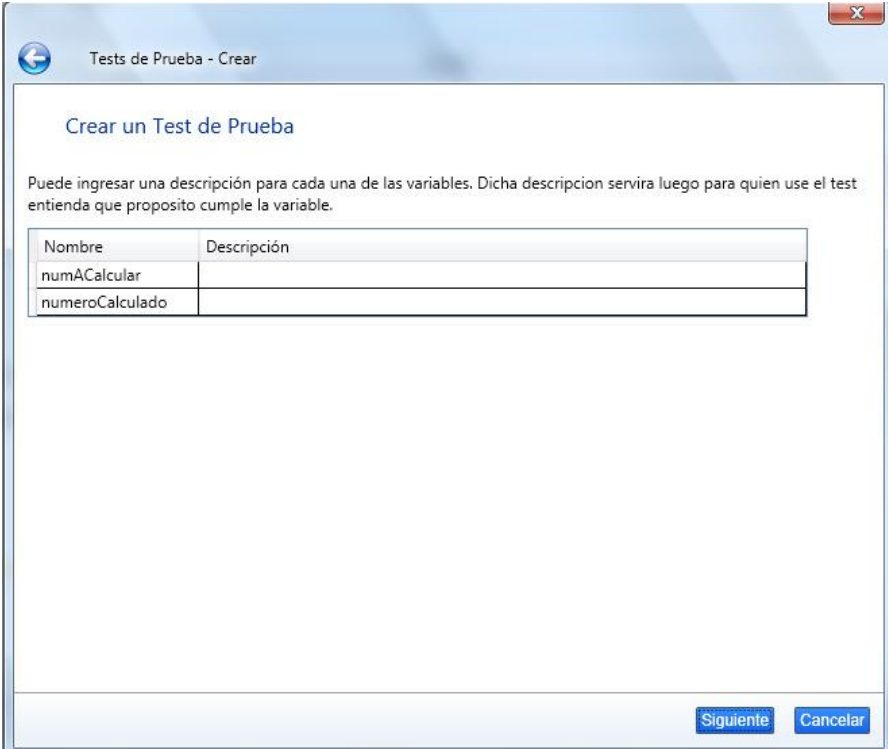
Fig. 37 Dialogo de Selección de Variables de Salida

Tal como hicimos con las variables de entrada, ahora es el turno de las variables de salida. Lo que ves acá es el listado de todas las variables que en tu programa califican para ser tomadas como variables de salida, o sea, aquellas que están como parámetro del procedimiento salida.

Las variables que elegís en este paso son las que tendrán son condicionadas por las variables de entrada que elegiste. En otras palabras, aquellos resultados que varían en base a las variables que elegiste en el paso 1.

Tenés que elegir al menos una variable de entrada para poder pasar al siguiente paso.

8.1.2.6 Paso 5: Descripción de las variables de salida



Tests de Prueba - Crear

Crear un Test de Prueba

Puede ingresar una descripción para cada una de las variables. Dicha descripción servirá luego para quien use el test entienda que propósito cumple la variable.

Nombre	Descripción
numACalcular	
numeroCalculado	

Siguiente **Cancelar**

Fig.38 Dialogo de Ingreso de descripción de variables de seleccionadas en el paso anterior

Esta pantalla es opcional también. Se recomienda siempre colocar descripciones a las variables tanto de entrada como de salida.

8.1.2.7 Paso 6: Ejecutar

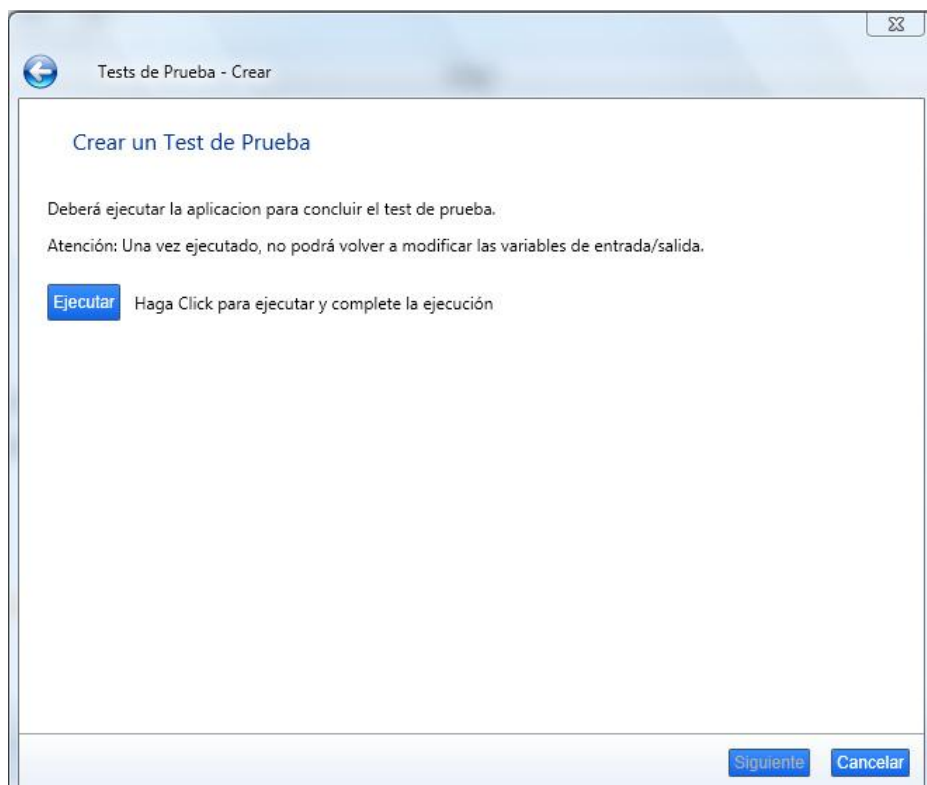


Fig. 39 Dialogo de confirmación de ejecución de test de prueba.

¡Ya casi estamos! Después de tanto definir, tanta variable de entrada y salida, ya estamos listos para ejecutar el programa, y hacer que los resultados se guarden.

Basta con presionar el botón de ejecutar, que hará que el programa se ejecute (si esperabas otra cosa, por favor volvé a leer el manual), finalizar la ejecución y listo. Una vez ejecutado el programa, los resultados se

guardan para todas las variables, por lo que ya no vas a poder volver atrás.

Generalmente, vas a ver esto, lo que significa que todo salió bien:



Fig. 40 Dialogo con resultado satisfactorio de ejecución.

Sin embargo, si había algún error en el programa (tranquilo, a todos nos puede pasar), lo que vas a terminar viendo es esto:

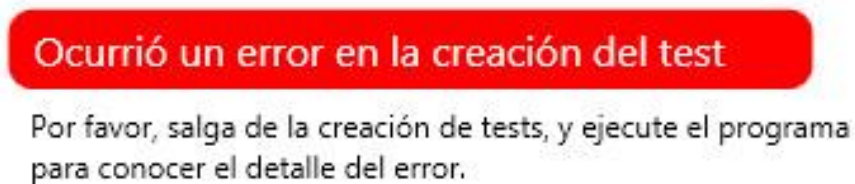


Fig. 41 Dialogo con resultado pesimista de ejecución.

Si existen errores en ejecución, no se puede guardar el test de prueba. Lo que tenés que hacer en este caso, es cerrar la ventana, corregir tu programa, y después volver a intentar crear los test.

8.1.2.8 Paso 7: Chequear variables

Tests de Prueba - Crear

Crear un Test de Prueba

Elija un nombre y descripción para el test que acaba de crear:

Nombre:

Descripcion:

El test quedará guardado de la siguiente manera:

Variables Entrada

Nombre	Valor	Contexto	Tipo de Variable	Tipo de Dato
num	12	principal	Variable	Numero
res	0	principal	Variable	Numero

Variables Salida

Nombre	Valor	Contexto	Tipo de Variable	Tipo de Dato
numACalcular	12	SALIDA	Variable	Numero
numeroCalculado	78	SALIDA	Variable	Numero

Finalizar Cancelar

Fig. 42 Dialogo de chequeo de variables luego de ejecución

Este es el último paso del test. Además de poder ver todas las variables de entrada y de salida, con sus respectivos valores puestos por la ejecución, también tenés un apartado para colocarle el nombre y descripción al test. Para poder finalizar la creación del test, tenés que colocarle ambas cosas al test.

8.1.3 Ejecutar test de prueba

Bueno, ya vimos como se crean, ahora nos falta ver cómo hacemos para ejecutarlo. Esta funcionalidad, como la de creación, esta representada en

una serie de X pantallas. Muchas de las pantallas son muy similares a las de creación, lo cual es una ventaja al momento de aprender a usarlas.

NOTA: Ragnarok siempre hace la sugerencia de que el código este correctamente indentado para realizar esta función. Siempre conviene hacerlo para garantizar el correcto funcionamiento de los mismos.

8.1.3.1 Paso 0: Cómo funcionan los test

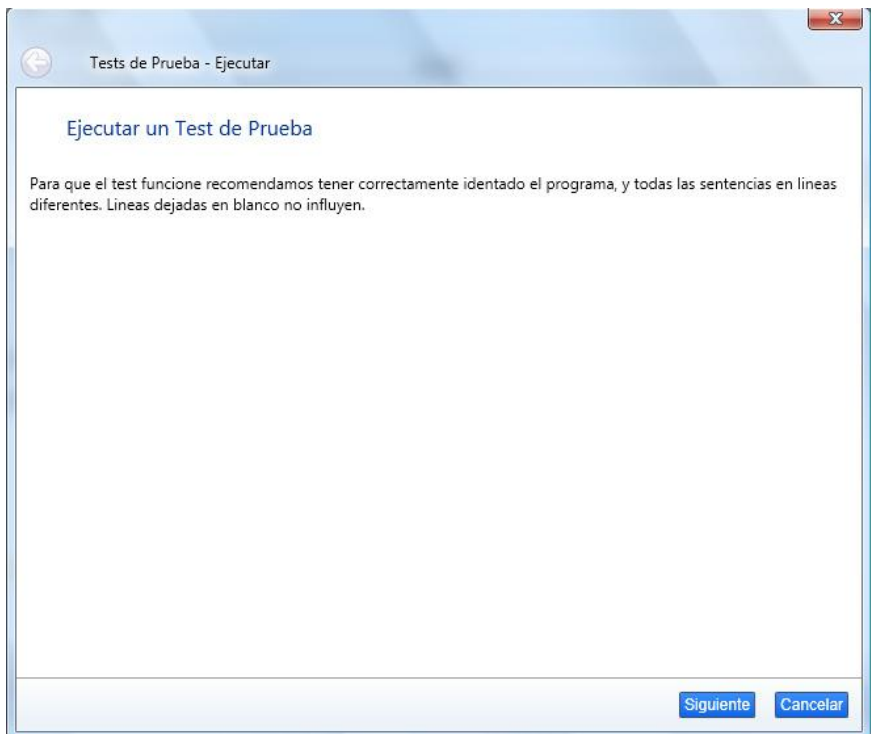


Fig. 43 Dialogo inicial de ejecución de test de prueba.

Esta pantalla es meramente explicativa. Aclara el tema de que el código debe estar correctamente indentado y que las líneas vacías no son tenidas en cuenta.

8.1.3.2 Paso 1: Elegir el test de prueba

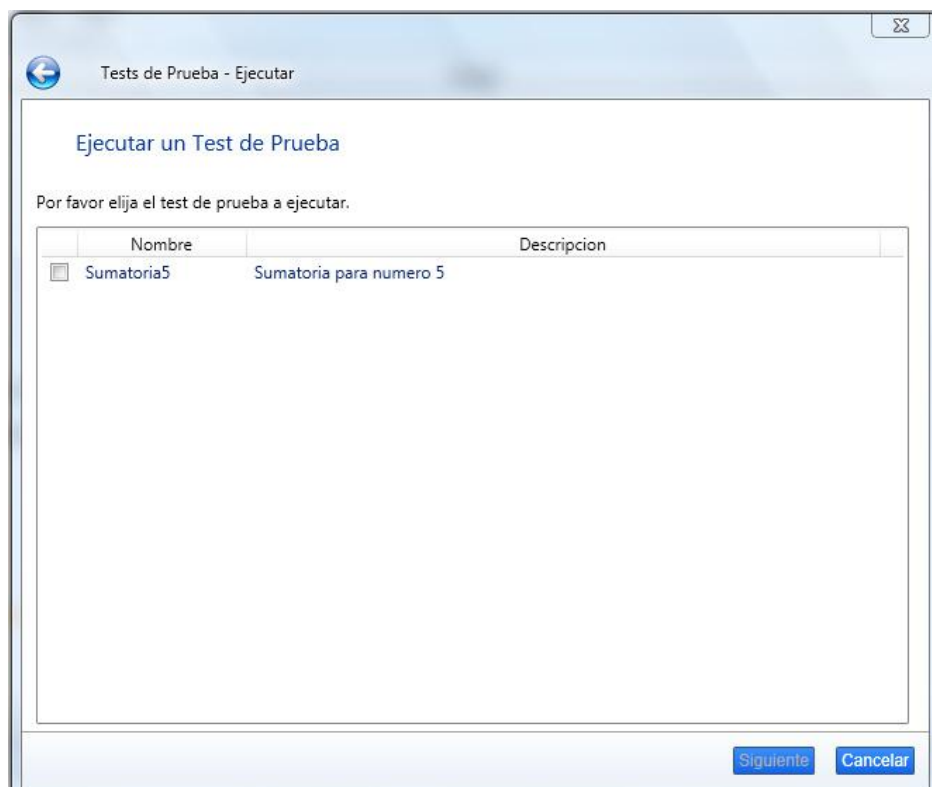


Fig. 44 Dialogo donde se elige el test de prueba que se desea ejecutar.

En esta pantalla podemos ver los tests de prueba que tiene disponible el ejercicio para ejecutar. La aplicación solo permite ejecutar de uno a la vez. (Por si no te diste cuenta, este es el mismo nombre y descripción del test que pusiste en el paso 7 de la creación del test).

8.1.3.3 Paso 2: Elegir las variables de entrada que corresponden a las del test

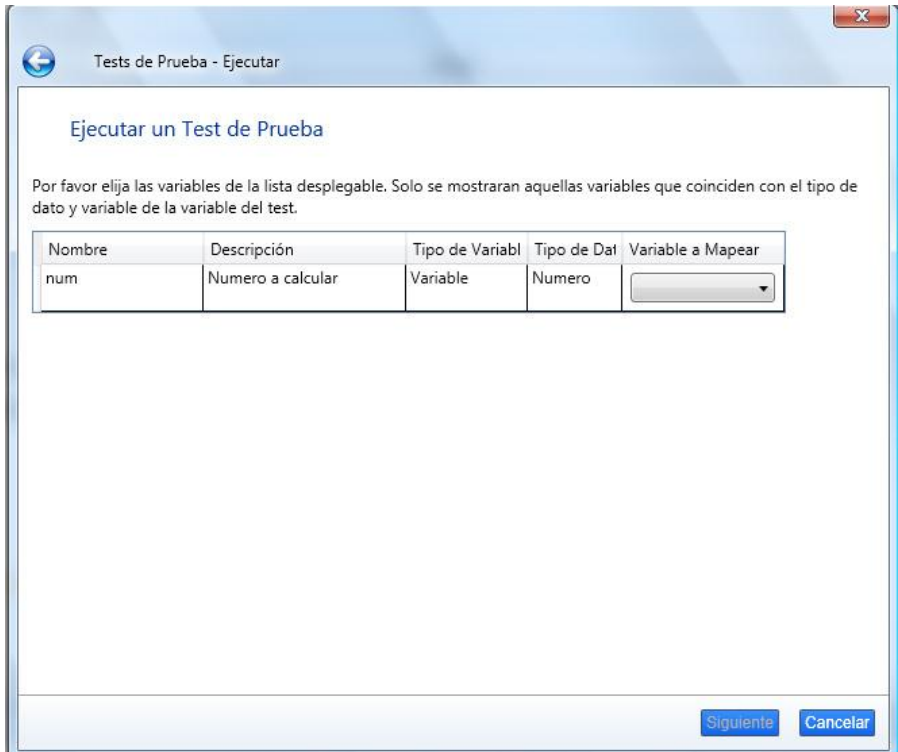


Fig. 45 Dialogo de elección de variables de entrada

Bueno, como hicimos cuando creamos un test, es el momento de definir las variables de entrada. El listado que ves acá sin embargo, son las variables de entrada que tiene el test que estas por ejecutar. Pero como vos en tu programa no necesariamente usaste el mismo nombre de variable que quien creo el test, tenés que indicarle (mediante la lista desplegable), cual de tus variables es la que corresponde. Esto se llama “mapear” la variable.

Tené en cuenta que la lista de variables que te ofrece son variables globales o locales al procedimiento salida, y que ya están filtradas por

tipo de dato y variable, para evitar que vos cometas un error (¡no es que no te tengamos fe, es que no queremos que te frustres!) En caso de estar vacía la lista desplegable, quiere decir que ninguna de tus variables globales o del procedimiento principal sirve, y no vas a poder ejecutar el test.

Para avanzar a la próxima pantalla, hay que mapear todas las variables de entrada.

8.1.3.4 Paso 3: Valores iniciales de las variables de entrada

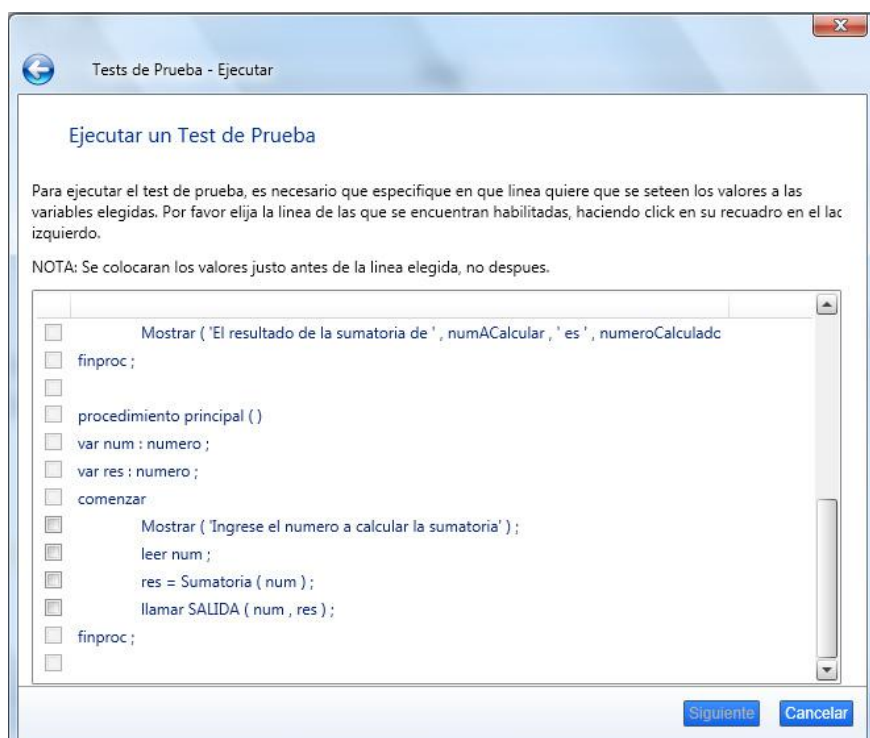


Fig.46 Dialogo de selección de valor inicial de variable de entrada

¿Te parece familiar esta pantalla? Esta pantalla es exactamente igual a la que vimos en el modo creación (paso 3) La única diferencia que en este

momento, en lugar de decirle al programa “en este punto, toma el valor de la variable X”, le estás diciendo “en este punto, asigne a mi variable X, el valor que tiene la variable de entrada del test”.

Solo una línea puede ser elegida, y se tomará el valor que tengan las variables de entrada, HASTA la línea que se elija SIN incluirla.

Este paso es crítico, ya que si se elige mal la línea, las variables pueden que no contengan los valores aun, o que estos hayan sido modificados ya por el programa, así que te recomendamos que pongas especial atención en este punto.

No se puede avanzar a la siguiente pantalla hasta que no se elija una línea.

8.1.3.5 Paso 4: Elegir las variables de salida que corresponden a las del test

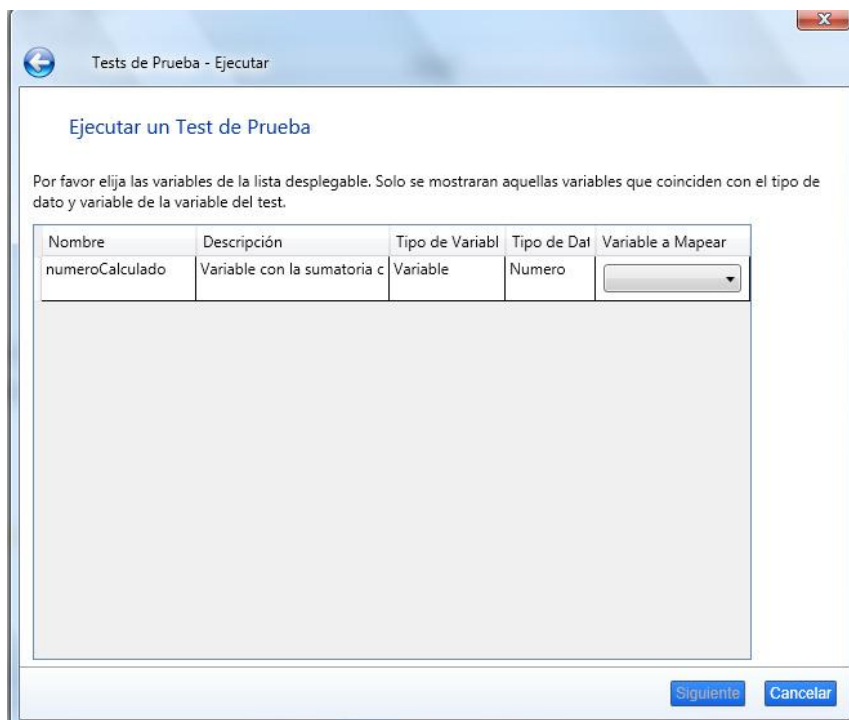


Fig. 47 Dialogo de selección de variables de salida

Tal como hicimos con las variables de entrada, ahora es el turno de las variables de salida. En este caso, toca mapear las variables de salida del test, a las variables del procedimiento salida que tiene tu programa.

Tal como en el paso 2, el programa filtra para cada variable, aquellas variables que coinciden en tipo de dato y variable. Si no aparece ninguna variable en el listado, es porque ninguna variable coincide, y no vas a poder ejecutar el test.

Tenés que elegir al menos una variable de entrada para poder pasar al siguiente paso.

8.1.3.6 Paso 5: Ejecutar

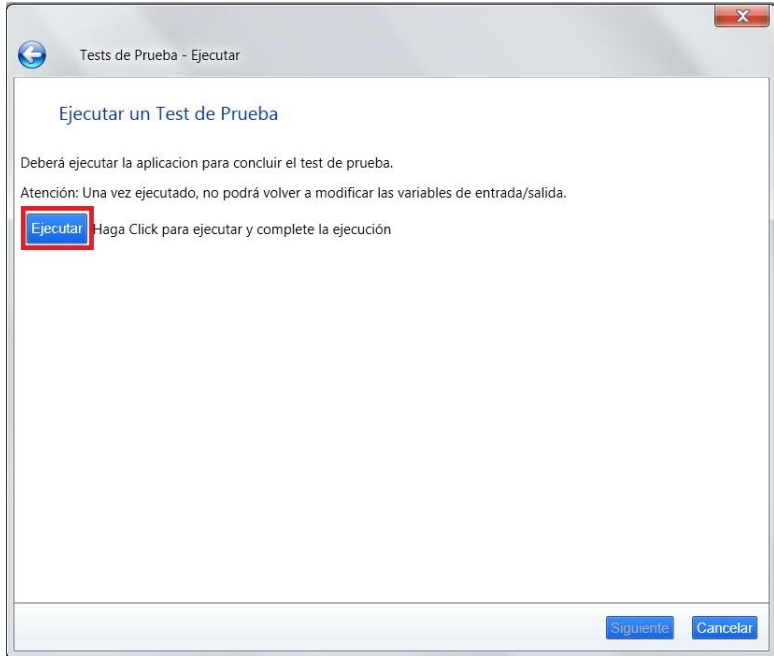


Fig. 48 Dialogo de confirmación de ejecución de test de prueba.

¡Nuevamente estamos por terminar! Estamos a un paso de ejecutar el programa, y ver si con los valores de entrada del test, terminas con el mismo valor de salida.

Tal como en la creación, una vez ejecutado el programa, los resultados se guardan para todas las variables, por lo que ya no vas a poder volver atrás.

Nuevamente, vas a ver esta pantalla si la ejecución es correcta:

Ejecucion satisfactoria

La ejecución del test de prueba se completo correctamente y se registraron los valores de las variables de entrada y salida.

Fig. 49 Dialogo con resultado satisfactorio de ejecución.

Y esta pantalla si no lo fue:

Ocurrió un error en la creación del test

Por favor, salga de la creación de tests, y ejecute el programa para conocer el detalle del error.

Fig. 50 Dialogo con resultado pesimista de ejecución.

Si existen errores en ejecución, no se podrá saber si el test salió bien o mal. Lo que tenés que hacer en este caso, es cerrar la ventana, corregir tu programa, y después volver a intentar ejecutar el test.

8.1.3.7 Paso 6: Resultados

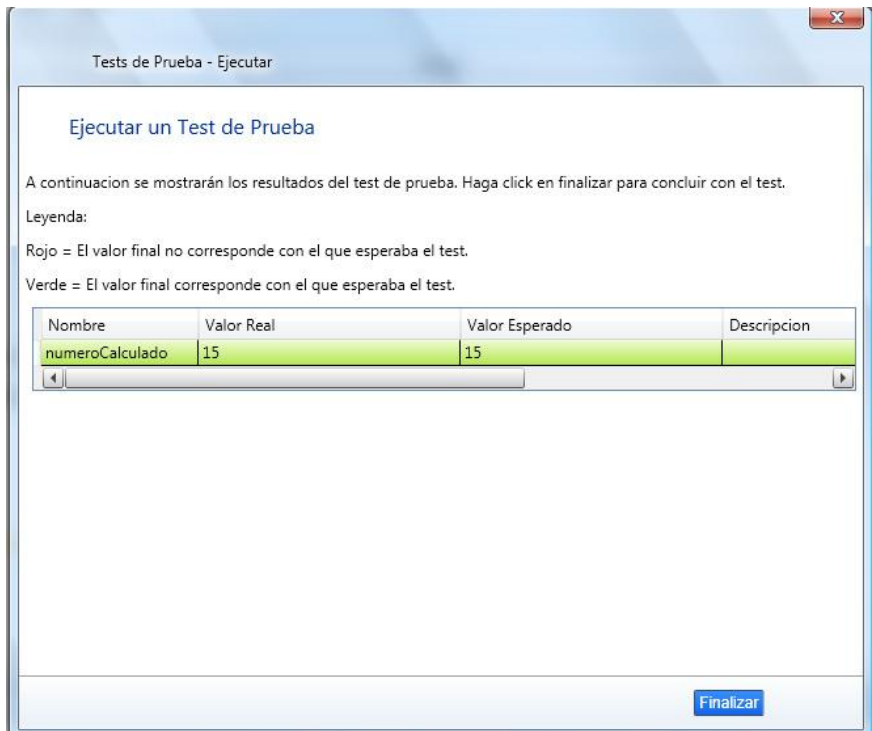


Fig. 51 Dialogo con resultados de ejecución.

Bueno, finalmente terminamos con la ejecución del test. Esta pantalla lo único que tiene es el listado de las variables de salida, con la información de que valor debía tener, y que valor termino teniendo.

Además, vas a ver cada fila resaltada de color rojo si los valores no coinciden, y verde si los valores si coinciden.

8.1.4 Ver los tests de prueba creados en el ejercicio

Uno no necesariamente tiene tan buena memoria como para saber que tests de prueba creo para cada ejercicio. Por esta razón, Ragnarok nos

provee la posibilidad (solo en archivos de tipo ejercicio), de consultar y eliminar aquellos tests de prueba asociados.

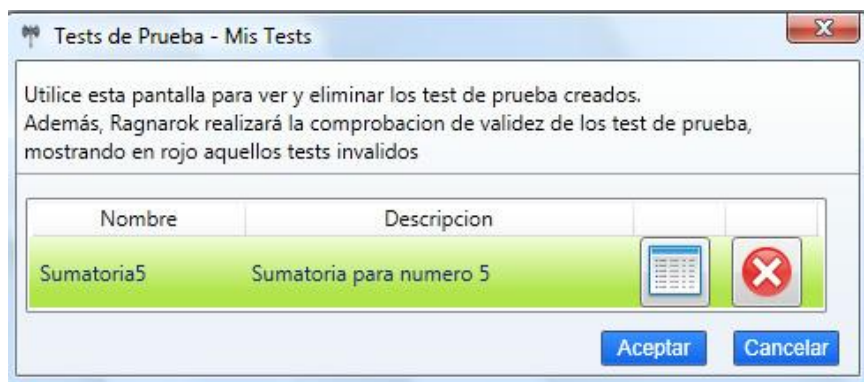


Fig. 52 Dialogo con listado de test de pruebas creados.

En esta pantalla, se muestran listados, todos los tests de prueba creados, con su nombre y descripción, y 2 botones, correspondientes a las funcionalidades de “ver detalles” y de eliminar.

Además, cada fila tiene un color verde o rojo. ¿Qué quiere decir esto? Muy simple: como existe la posibilidad que uno cree el test de prueba, y luego siga haciendo modificaciones al programa, esta característica cumple con la función de indicar a simple vista cual test de prueba sigue siendo correcto (porque está verde) y cuales son incorrectos (porque esta rojo).

El criterio que toma para ver si un test sigue siendo correcto, es si las variables tanto de entrada como de salida, siguen existiendo en el programa. Si ya no existen más, el test será invalido.

La funcionalidad de ver detalles abre una nueva ventana en donde podemos consultar las variables de entrada y de salida, y los valores que tiene cada una de ellas. Acá también las filas pueden ser rojas o verdes, con el mismo criterio que usan los tests de prueba: si la variable dejo de existir, aparecerá roja

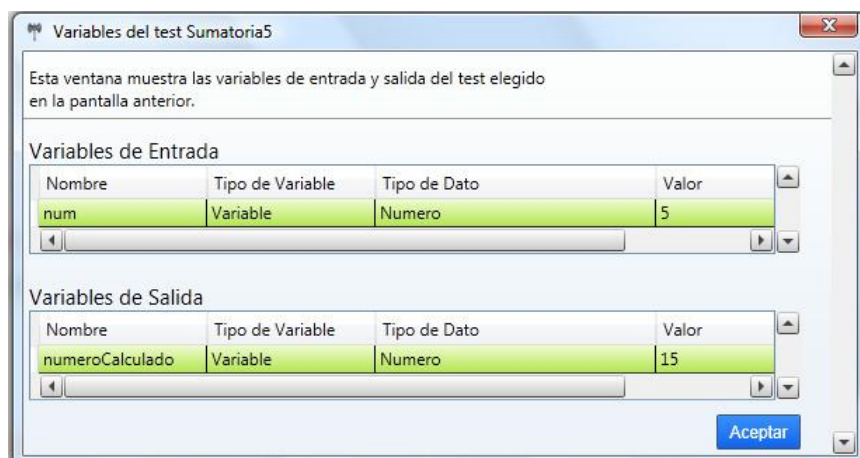


Fig.53 Dialogo con detalle de Test de prueba

La funcionalidad de eliminar, quita un test de prueba del ejercicio. Para confirmar el eliminado, es necesario hacer click en aceptar. Por el contrario, si uno se arrepiente, y siente que cometió un error gravísimo al borrar el test de prueba, el botón cancelar descarta todas las eliminaciones hechas.

9. Detalles de un Ejercicio

9.1 Enunciado (todo ejercicio tiene uno, verdad?)

El enunciado del ejercicio representa la consigna. Acá vas a encontrar lo que tenés que hacer. Tené en cuenta que es de sólo lectura y por lo tanto no se puede modificar.

9.2 Solución Explicada (No entendiste el enunciado? Lee esto con atención entonces)

La solución explicada representa el Enunciado pero explicado con mayor detalle. También, así como en el Enunciado, es de sólo lectura.

9.3 Dificultad (algunos son más fáciles que otros)

La dificultad la coloca el creador de ejercicio, y es sólo ilustrativa. El número de dificultad va de 1 a 10, siendo 1 el nivel más fácil y 10 el más difícil. No te asustes si recién empezás y te encontrás con un ejercicio cuyo nivel de dificultad es mayor que 5, animáte a equivocarte!

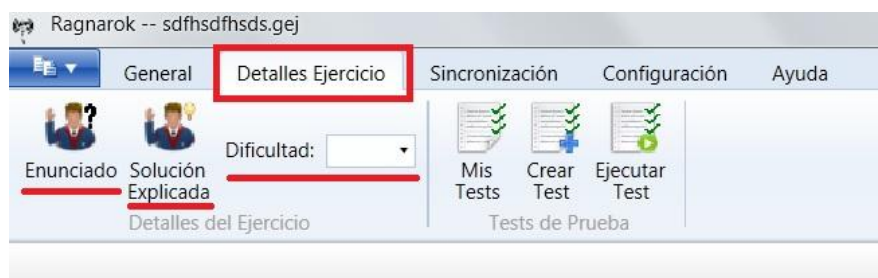


Fig.54 La solapa de “Detalle de Ejercicio” con su contenido de Enunciado, Detalle y Nivel de Dificultad.

10. Exportar un ejercicio para subir a la Web

10.1 Yo también quiero que mis ejercicios sean descargados

Vos también podés publicar ejercicios. Es más, cualquiera puede hacerlo (incluso los no estudiantes). Para exportar un ejercicio, primero tenés que crearlo (Menú -> Nuevo Ejercicio). Lo primero que te conviene hacer es completar el enunciado (de esta forma sabés a qué apunta tu ejercicio). Luego cargar el Detalle del Enunciado explicando a los usuarios que se vayan a descargar tu ejercicio lo que tienen que hacer. Y luego la dificultad, del 1 al 10.

Una vez que tenés esto definido, podés empezar a escribir el programa (es lo que los usuarios van a ver como un nuevo botón que se llama “Solución GarGar” cuando hagan la resolución del ejercicio). Luego, cuando tengas todo listo, tenés que ir al menú superior, seleccionar “Guardar Como...” -> “Formato Web”. Lo primero que va a hacer Ragnarok es verificar si completaste todo lo necesario (ver 10.2 Condiciones mínimas...). Si tenés todo lo necesario, Ragnarok te va a solicitar que ejecutes el ejercicio una vez, para poder verificar que al menos termina de ejecutar una vez correctamente. No sirve que se suban ejercicios si no ejecutan.

Tu ejercicio va a quedar supeditado a moderación (pendiente de aprobación). Si algún moderador considera que tu ejercicio está en condiciones de formar parte del repositorio de Program.AR, aprobará tu ejercicio y quedará disponible para que se lo descarguen otros usuarios. Caso contrario, te desaprobarán el ejercicio y se te enviará un mail con la justificación de por qué no se aprobó tu ejercicio.

10.2 Condiciones mínimas para la exportación a la web

Para poder exportarlo a la web, hay 3 cosas que tenés que tener completas sí o sí: El Enunciado, la Solución Explicada y la Dificultad. Porque esto es lo que va a ayudar a los demás a resolver tu ejercicio.

11. RepoWeb

11.1 Que es? Para que sirve?

El RepoWeb es un “navegador” de Cursos y Ejercicios que te va a permitir obtener todos los ejercicios que quieras/necesites a sólo 2 clicks de distancia. Lo podés encontrar en la solapa de “Sincronización”, en el botón que dice “Sincro por Repositorio”



Fig.55 El punto de acceso al RepoWeb (no se llama así, es su nombre coloquial).

11.2 Ejercicios

11.2.1 Buscador

Mediante el buscador de Ejercicios, podemos encontrar los ejercicios que se encuentran en internet. Para ello, es necesario disponer de conexión a internet, caso contrario no podremos buscar Ejercicios (sucede lo mismo con Cursos, sin internet no podremos buscar).

Existen 4 filtros posibles para encontrar ejercicios: Podés buscar por nombre, por número de ejercicio, por usuario creador y por dificultad. Podés usar estos filtros y combinarlos como quieras (Por ejemplo, puedes buscar ejercicios que empiecen con “Ejer” y cuya dificultad sea “8”).

Buscador de Ejercicios

Utilice esta pantalla para buscar los ejercicios a bajar. La pantalla consulta la web de Program.Ar para obtener los ejercicios y cursos.

Dirección de ejercicios:

Url de ejercicios:

Buscar Ejercicios Buscar Cursos

Número de ejercicio: Nombre:

Usuario: Dificultad:

Buscar

Id	Nombre	Usuario	Dificultad			
----	--------	---------	------------	--	--	--

Listo

Fig.56 La búsqueda por Ejercicios.

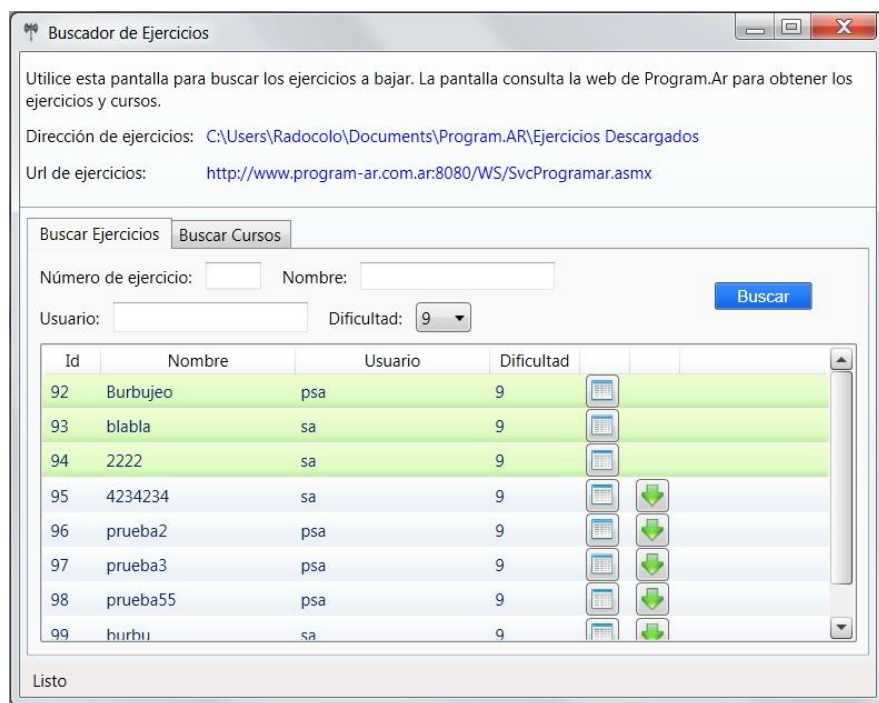


Fig.57 Una búsqueda de Ejercicios cuya dificultad es "9". Los que figuran en verde son aquellos ejercicios que ya se encuentran en tu PC.

11.2.2 Detalle

Si presionás el botón que figura al lado de la dificultad, podés visualizar con mayor detalle al ejercicio (Enunciado, Solución, Dificultad, Fecha de Creación, etc).

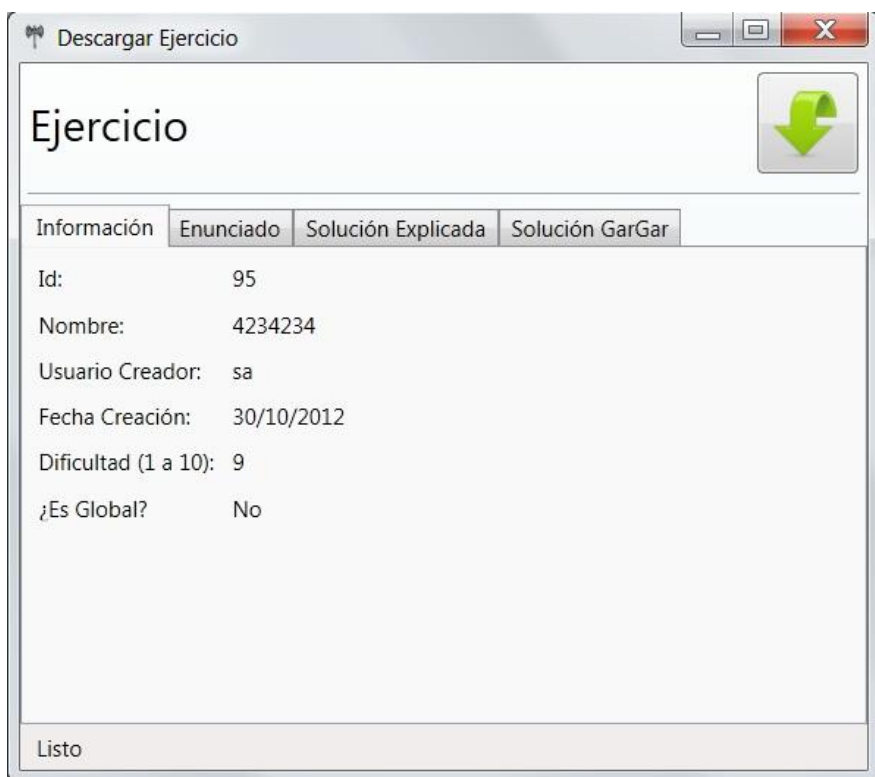


Fig.58 Detalle de un ejercicio, con sus solapas de “Enunciado”, “Solución Explicada” y “Solución GarGar”. El botón superior derecho permite descargar el ejercicio. Este botón no aparecerá si ya lo tenés en tu PC.

11.2.3 Descarga

Para descargar el Ejercicio, tenés que apretar el botón que representa una flecha verde hacia abajo. Acordate que si el Ejercicio aparece pintado en verde, significa que el Ejercicio ya está almacenado localmente en tu computadora.

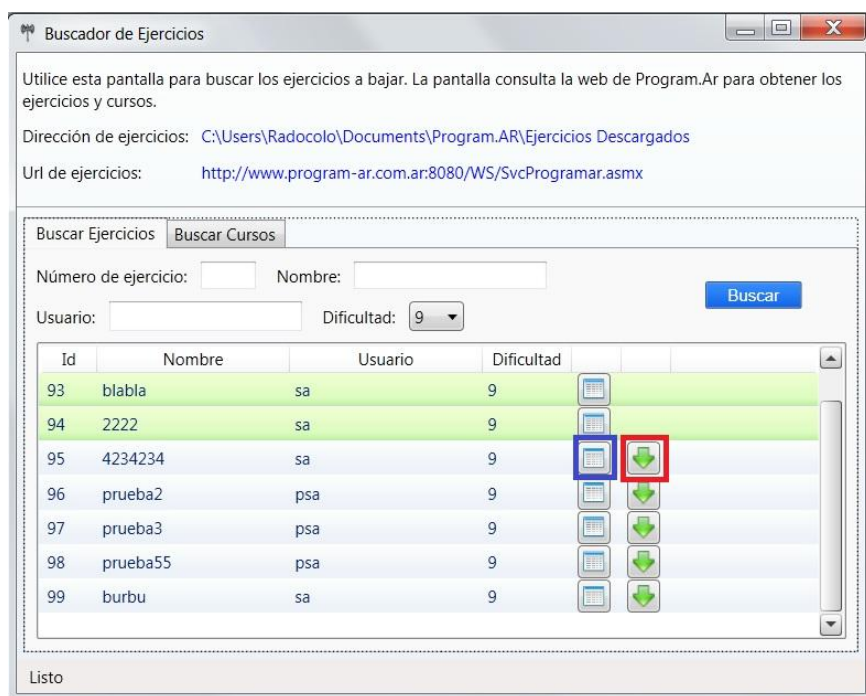


Fig.59 El cuadrado azul contiene el botón que permite acceder al detalle del Ejercicio. El cuadrado rojo contiene el botón que permite descargar el Ejercicio. La ubicación del archivo descargado está especificado en la Configuración de la aplicación, en el directorio de "Ejercicios Descargados".

11.3 Cursos

11.3.1 Buscador

El buscador de Cursos funciona exactamente igual que el de Ejercicios. Un Curso puede contener Ejercicios (puede no tener ninguno, como tener N). Igual que con el buscador de Ejercicios, se necesita una conexión a internet para poder buscar.

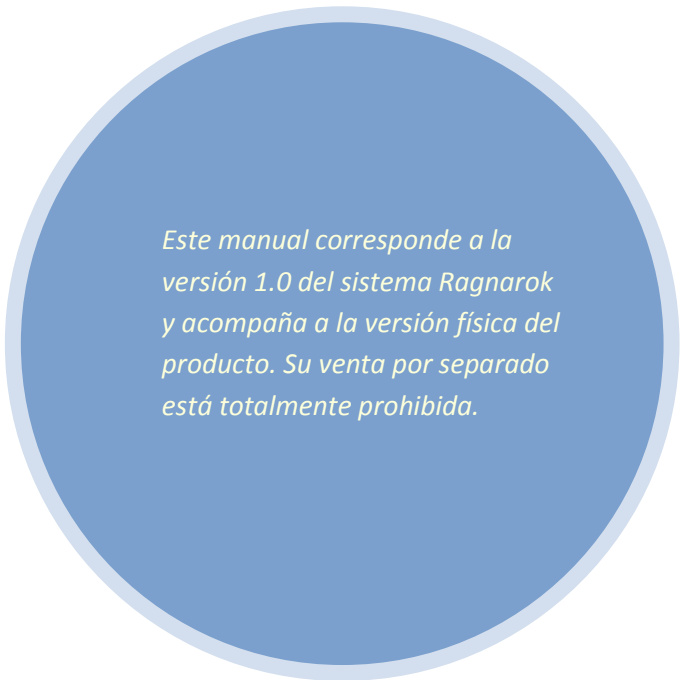
Existen 3 filtros posibles para encontrar Cursos: Podés buscar por Número de Curso, por Nombre y por Usuario Creador del Curso. Igual que con los filtros de Ejercicios, podés combinarlos. Si no ingresás ningún filtro Ragnarok buscará todos los cursos existentes.

11.3.2 Detalle

En el detalle del Curso podés encontrar todos los ejercicios que están asociados a él.

11.3.3 Descarga

Descargar un Curso significa que vas a descargar TODOS los Ejercicios que están asociados. Aquellos que ya tenés no los vas a volver a descargar. En caso de que ya te hayas descargado todos los ejercicios de un curso, te aparecerá una leyenda que dice “Este curso ya fue descargado”.



Este manual corresponde a la versión 1.0 del sistema Ragnarok y acompaña a la versión física del producto. Su venta por separado está totalmente prohibida.