TESIS
II 2004
Q8
V. 2

UNIVERSIDAD CATÓLICA ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA INFORMÁTICA

DESARROLLO DE UN SOFTWARE EDUCATIVO QUE ESTIMULE EL PENSAMIENTO LÓGICO-MATEMÁTICO A TRAVÉS DE LA PROGRAMACIÓN, SEGÚN EL ENFOQUE CONSTRUCTIVISTA DE LAS ESCUELAS DE FE Y ALEGRÍA
(ANEXOS)

REALIZADO POR	Quezada Da Silva, Sandra María Quijada Naranjo, Dojelin Enny
PROFESOR GUIA	Lic. Alejandro del Mar
FECHA	Caracas, 27 de Julio del 2004

ÍNDICE GENERAL

Índice General I

Índice de Gráficas III

Índice de Cuadros V

Índice de Diagramas VII

Índice de Tablas VIII

Anexo II-1: Evolución de los ambientes Logo 1

Anexo II-2: Definición y Características de los PPA 4

Anexo II-3: El modelo Constructivista 7

Anexo II-4: Entrevista con el padre del software libre 10

Anexo II-5: Con el software libre los países generan desarrollo..... 14

Anexo II-6: Licencia Pública GNU 20

Anexo IV-1: Encuesta Nº 1 27

Anexo IV-2: Resultados arrojados por las encuestas Nº 1..... 28

Anexo IV-3: Comentarios de profesores encuestados 31

Anexo IV-4: Comparación de primitivas de MSWLogo y MicroMundos..... 33

Anexo IV-5: Diagrama de Clases de MSWLogo 90

Anexo IV-6: Cuadro de Atributos y Métodos de las Clases de MSWLogo..... 97

Anexo IV-7: Código fuente de la Caja de Herramientas "Mover" 104

Anexo IV-8: Diagrama de Clases de MundoLogo 108

Anexo IV-9: Cuadro de Atributos y Métodos de las Clases de MundoLogo..... 115

Anexo IV-10: Casos de uso de MundoLogo 117

Anexo IV-11: Cuadros descriptivos de los Casos de Uso de MundoLogo..... 121

Anexo IV-12: Diagramas de Actividades de MundoLogo 168
Anexo IV-13: Traducción al español de las primitivas de MSWLogo 185
Anexo IV-14: Traducción al español de las librerías de Logo 199
Anexo IV-15: Traducción al español de los mensajes de error de MSWLogo 201
Anexo IV-16: Traducción al español de los Menús de MSWLogo..... 204
Anexo IV-17: Código fuente de las nuevas primitivas incorporadas a MundoLogo..... 207
Anexo IV-18: Encuesta de Evaluación del Software para profesores 218
Anexo IV-19: Resultados arrojados por las encuestas aplicadas a los profesores..... 221
Anexo IV-20: Actividad de Think Aloud 223
Anexo IV-21: Encuesta de Evaluación del Software para niños 224
Anexo IV-22: Resultados arrojados por las encuestas aplicadas a los niños..... 226
Anexo IV-23: Análisis de las encuestas aplicadas a los niños 234
Anexo IV-24: Tríptico presentado durante la IV Feria de Tecnología Educativa 2004..... 238

ÍNDICE DE GRÁFICAS

Gráfica 1: Uso de MicroMundos en las escuelas de Fe y Alegría 28
Gráfica 2: Versión de MicroMundos utilizada por los profesores 28
Gráfica 3: Nivel de conocimiento de MicroMundos por los profesores.... 29
Gráfica 4: Utilización de MicroMundos dentro de las escuelas de Fe y Alegría 29
Gráfica 5: Versión de MicroMundos utilizada en Fe y Alegría..... 30
Gráfica 6: Valor esperado y obtenido para la variable Diseño Conceptual 221
Gráfica 7: Valor esperado y obtenido para la variable Aspectos Pedagógicos 222
Gráfica 8: Valor esperado y obtenido para la variable Aspectos Técnicos y de Interfaz 222
Gráfica 9: Valor esperado y obtenido para la variable Aspectos de Diseño Gráfico..... 222
Gráfica 10: Pregunta 1 de la encuestas aplicadas a los niños..... 226
Gráfica 11: Pregunta 2 de la encuestas aplicadas a los niños..... 226
Gráfica 12: Pregunta 3 de la encuestas aplicadas a los niños..... 226
Gráfica 13: Pregunta 4 de la encuestas aplicadas a los niños..... 227
Gráfica 14: Pregunta 5 de la encuestas aplicadas a los niños..... 227
Gráfica 15: Pregunta 6 de la encuestas aplicadas a los niños..... 227
Gráfica 16: Pregunta 7 de la encuestas aplicadas a los niños..... 228
Gráfica 17: Pregunta 8 de la encuestas aplicadas a los niños..... 228
Gráfica 18: Pregunta 9 de la encuestas aplicadas a los niños..... 228
Gráfica 19: Pregunta 10 de la encuestas aplicadas a los niños..... 229

Gráfica 20: Pregunta 11 de la encuestas aplicadas a los niños..... 229
Gráfica 21: Pregunta 12 de la encuestas aplicadas a los niños..... 229
Gráfica 22: Pregunta 13 de la encuestas aplicadas a los niños..... 230
Gráfica 23: Pregunta 14 de la encuestas aplicadas a los niños..... 230
Gráfica 24: Pregunta 15 de la encuestas aplicadas a los niños..... 230
Gráfica 25: Pregunta 16 de la encuestas aplicadas a los niños..... 231
Gráfica 26: Pregunta 17 de la encuestas aplicadas a los niños..... 231
Gráfica 27: Pregunta 18 de la encuestas aplicadas a los niños..... 231
Gráfica 28: Pregunta 19 de la encuestas aplicadas a los niños..... 232
Gráfica 29: Pregunta 20 de la encuestas aplicadas a los niños..... 232
Gráfica 30: Pregunta 21 de la encuestas aplicadas a los niños..... 232
Gráfica 31: Pregunta 22 de la encuestas aplicadas a los niños..... 233
Grafica 32: Valor esperado y obtenido para la variable Motivación..... 235
Grafica 33: Valor esperado y obtenido para la variable Evaluación..... 235
Grafica 34: Valor esperado y obtenido para la variable Ritmo 236
Grafica 35: Valor esperado y obtenido para la variable Contenido 236
Grafica 36: Valor esperado y obtenido para la variable Aprendizaje..... 236
Grafica 37: Valor esperado y obtenido para la variable Ejercitación Práctica 237
Grafica 38: Valor esperado y obtenido para la variable Actitud ante los MEC 237
Grafica 39: Valor esperado y obtenido para la variable Interfaz 237

ÍNDICE DE CUADROS

Cuadro 1: Caso de uso A1: Usar Primitivas..... 121
Cuadro 2: Casos de Uso A2:Administrar Procedimientos..... 122

Cuadro 3: Casos de Uso A2.1:Crear Procedimientos..... 123
Cuadro 4: Casos de Uso A2.2:Guardar Procedimientos..... 124
Cuadro 5: Casos de Uso A2.3:Imprimir Procedimientos..... 125
Cuadro 6: Casos de Uso A2.4:Editar Procedimientos..... 126
Cuadro 7: Casos de Uso A2.4.1:Deshacer..... 127
Cuadro 8: Casos de Uso A2.4.2:Cortar..... 128
Cuadro 9: Casos de Uso A2.4.3:Copiar..... 129
Cuadro 10: Casos de Uso A2.4.4:Pegar..... 130
Cuadro 11: Casos de Uso A2.4.5:Borrar..... 131
Cuadro 12: Casos de Uso M2.5:Seleccionar fuente..... 132
Cuadro 13: Casos de Uso M2.6:Realizar búsqueda..... 133
Cuadro 14: Casos de Uso M2.6.1:Buscar por..... 134
Cuadro 15: Casos de Uso M2.6.2:Reemplazar..... 135
Cuadro 16: Casos de Uso M2.6.3:Buscar siguiente..... 136
Cuadro 17: Casos de Uso M2.7:Realizar prueba..... 136
Cuadro 18: Casos de Uso M2.8:Consultar ayuda..... 137
Cuadro 19: Casos de Uso A3:Administrar proyecto..... 138
Cuadro 20: Casos de Uso A3.1:Crear nuevo proyecto..... 139
Cuadro 21: Casos de Uso A3.2:Abrir proyecto..... 140
Cuadro 22: Casos de Uso A3.3:Guardar proyecto..... 141
Cuadro 23: Casos de Uso A3.4:Editar proyecto..... 142
Cuadro 24: Casos de Uso A4:Administrar imagen..... 143
Cuadro 25: Casos de Uso A4.1:Crear nueva imagen..... 143
Cuadro 26: Casos de Uso A4.2:Abrir imagen existente..... 144
Cuadro 27: Casos de Uso A4.3:Guardar imagen..... 145
Cuadro 28: Casos de Uso A4.4:Seleccionar área activa..... 146
Cuadro 29: Casos de Uso A4.5:Imprimir imagen..... 146
Cuadro 30: Casos de Uso A4.5.1:Configurar página..... 147
Cuadro 31: Casos de Uso A5: Administrar formato..... 148
Cuadro 32: Casos de Uso A5.1:Seleccionar tamaño de la pluma..... 148

Cuadro 33: Casos de Uso A5.2:Seleccionar fuente de etiquetas..... 149
Cuadro 34: Casos de Uso A5.3:Seleccionar fuente de comandos..... 149
Cuadro 35: Casos de Uso A5.4:Seleccionar color de la pluma..... 150
Cuadro 36: Casos de Uso A5.5:Seleccionar color de relleno..... 151
Cuadro 37: Casos de Uso A5.6:Seleccionar color de fondo..... 152
Cuadro 38: Casos de Uso M6:Insertar máscara a la tortuga..... 153
Cuadro 39: Casos de Uso B7:Seleccionar vista..... 154
Cuadro 40: Casos de Uso A8:Consultar ayuda..... 155
Cuadro 41: Casos de Uso A9:Trabajar en modo gráfico..... 156
Cuadro 42: Casos de Uso A9.1:Seleccionar copiar..... 157
Cuadro 43: Casos de Uso A9.2:Seleccionar pegar..... 157
Cuadro 44: Casos de Uso A9.3:Seleccionar pluma..... 158
Cuadro 45: Casos de Uso A9.4:Seleccionar rectángulo..... 158
Cuadro 46: Casos de Uso A9.5:Seleccionar rectángulo relleno..... 159
Cuadro 47: Casos de Uso A9.6:Seleccionar elipse..... 159
Cuadro 48: Casos de Uso A9.7:Seleccionar elipse rellena..... 160
Cuadro 49: Casos de Uso A9.8:Seleccionar línea recta..... 160
Cuadro 50: Casos de Uso A9.9:Seleccionar píxel..... 161
Cuadro 51: Casos de Uso A9.10:Seleccionar borrador..... 161
Cuadro 52: Casos de Uso A9.11:Seleccionar relleno..... 162
Cuadro 53: Casos de Uso A9.12:Seleccionar grosor de la pluma..... 162
Cuadro 54: Casos de Uso A9.13:Seleccionar color..... 163
Cuadro 55: Casos de Uso A9.14:Seleccionar modo no gráfico..... 164
Cuadro 56: Casos de Uso A10:Agregar tortuga..... 164
Cuadro 57: Casos de Uso A11:Borrar pantalla 165
Cuadro 58: Casos de Uso M12:Detener proceso..... 165
Cuadro 59: Casos de Uso M13:Pausar proceso..... 166
Cuadro 60: Casos de Uso M14:Realizar traza..... 166
Cuadro 61: Casos de Uso M15:Correr por pasos..... 167
Cuadro 62: Casos de Uso A16:Salir del sistema..... 167

Cuadro 63: Estructura de la encuesta aplicada a los niños..... 234

ÍNDICE DE DIAGRAMAS

Diagrama 1: Diagrama de Clase de MSWLogo. Parte 1..... 91
Diagrama 2: Diagrama de Clase de MSWLogo. Parte 2..... 92
Diagrama 3: Diagrama de Clase de MSWLogo. Parte 3..... 93
Diagrama 4: Diagrama de Clase de MSWLogo. Parte 4..... 94
Diagrama 5: Diagrama de Clase de MSWLogo. Parte 5..... 95
Diagrama 6: Diagrama de Clase de MSWLogo. Parte 6..... 96
Diagrama 7: Diagrama de Clase de MundoLogo. Parte 1..... 110
Diagrama 8: Diagrama de Clase de MundoLogo. Parte 2..... 111
Diagrama 9: Diagrama de Clase de MundoLogo. Parte 3..... 112
Diagrama 10: Diagrama de Clase de MundoLogo. Parte 5..... 113
Diagrama 11: Diagrama de Clase de MundoLogo. Parte 6..... 114
Diagrama 12: Casos de Uso, Administrar procedimiento. Nivel 2..... 117
Diagrama 13: Casos de Uso, Editar procedimiento. Nivel 2..... 118
Diagrama 14: Casos de Uso, Realizar búsqueda. Nivel 3..... 118
Diagrama 15: Casos de Uso, Administrar proyecto. Nivel 2..... 118
Diagrama 16: Casos de Uso, Administrar imagen. Nivel 2..... 119
Diagrama 17: Casos de Uso, Imprimir imagen. Nivel 3..... 119
Diagrama 18: Casos de Uso, Administrar formato. Nivel 2..... 119
Diagrama 19: Casos de Uso, Trabajar en modo gráfico. Nivel 2..... 120
Diagrama 20: Diagrama de Actividad Insertar Máscara a la tortuga..... 168
Diagrama 21: Diagrama de Actividad Activar el Modo Gráfico..... 169
Diagrama 22: Diagrama de Actividad Seleccionar Copiar..... 170
Diagrama 23: Diagrama de Actividad Seleccionar Pegar..... 171

Diagrama 24: Diagrama de Actividad Seleccionar Pluma..... 172
Diagrama 25: Diagrama de Actividad Seleccionar Rectángulo..... 173
Diagrama 26: Diagrama de Actividad Seleccionar Rectángulo Relleno..... 174
Diagrama 27: Diagrama de Actividad Seleccionar Elipse..... 175
Diagrama 28: Diagrama de Actividad Seleccionar Elipse Rellena..... 176
Diagrama 29: Diagrama de Actividad Seleccionar Línea Recta..... 177
Diagrama 30: Diagrama de Actividad Seleccionar Píxel..... 178
Diagrama 31: Diagrama de Actividad Seleccionar Borrador..... 179
Diagrama 32: Diagrama de Actividad Seleccionar Relleno..... 180
Diagrama 33: Diagrama de Actividad Seleccionar Grosor de la Pluma... 181
Diagrama 34: Diagrama de Actividad Seleccionar Color..... 182
Diagrama 35: Diagrama de Actividad Seleccionar Modo no Gráfico..... 183
Diagrama 36: Diagrama de Actividad Insertar Nueva Tortuga..... 184

ÍNDICE DE TABLAS

Tabla 1: Listado de primitivas de MSWLogo y sus funciones..... 33
Tabla 2:Listado de primitivas de MicroMundos y sus funciones..... 76
Tabla 3: Atributos y métodos de MSWLogo..... 97
Tabla 4: Atributos y métodos de MundoLogo..... 115
Tabla 5: Traducción al español de Primitivas de MSWLogo..... 185
Tabla 6: Traducción al español de mensajes de error de MSWLogo..... 201
Tabla 7: Traducción al español de menús de MSWLogo..... 204
Tabla 8: Traducción al español de menús de MSWLogo..... 205
Tabla 9: Traducción al español de menús de MSWLogo..... 206

Anexo II-1

Evolución de los ambientes Logo

Con el tiempo, esos productos evolucionaron en productos mas bien comerciales. Inicialmente La compañía Texas Instrument lanzó al mercado TILOGO. Algunos años después, en 1977, Terrapin Software, una compañía creada para comercializar tortugas robóticas, licenció la versión para la Apple II de MIT Logo, la cual ha sido mejorada con el transcurso de los años, primero con Terrapin Logo y actualmente con Logo PLUS.

En el año de 1980, fue creada la compañía Logo Computer Systems, Inc. (LCSI), por un grupo importante de investigadores, maestros y programadores que se han dedicado al desarrollo de importantes avances en Logo.

Luego fueron desarrolladas e implementadas nuevas versiones de Logo, en más de una docena de idiomas, sobre un amplia variedad de máquinas, muchas con gráficos y con una gran capacidad de sonido. Estas versiones se hicieron populares en Europa, Sudamérica y Japón.

Apple Computers comercializó el Apple Logo y durante algún tiempo lo entregaba empaquetado junto a las computadoras que regaló a algunas escuelas de California. Mas adelante la compañía IBM, comercializó IBM Logo de la LCSI, así como también lo hizo con Logo Leaner. En el año 1985 fue introducido al mercado LogoWriter por la LCSI, el cual fue novedoso, debido a que incluyó capacidades de procesador de palabras. Además esta versión incluía una interfaz de usuario mucho más simplificada e intuitiva y el uso de múltiples tortugas que podían asumir diferentes figuras. LogoWriter fue desarrollada en una gran variedad de idiomas y se hizo bastante popular en el mundo.

Mas adelante, Mitchel Resnick y Steve Ocko, quienes trabajaban para el "Media Lab" del MIT desarrollaron un sistema, LEGO Logo, que implementaba una interfaz entre Logo y motores, luces y sensores que fueron incorporados en máquinas creadas a partir de objetos LEGO.

Algunos años después con el frenesí de nuevos desarrollo de Logo, además de los desarrollos en Europa, América Latina, Canadá y Estados Unidos, se han dedicado a la introducción de MicroWorlds. Este software fue lanzado en el año 1993 por la LCSI y presenta grandes cambios tanto en el ambiente Logo como en el lenguaje. MicroWorlds, incluye algunas características no típicas de Logo, como herramientas de dibujo, un editor de figuras, un constructor de melodías, la habilidad de importar gráficos y sonido, que trabajan junto con el soporte de Logo para la creación de proyectos multimedia, juegos y simulaciones. Además, permite la inclusión de multi-tarea y procesamiento paralelo.

Otra versión paralela de Logo y de uso muy extendido es StarLogo, y fue desarrollada por el MIT, bajo la conducción de Mitchel Resnick. Una de sus virtudes, es que miles de tortugas pueden realizar procesos simultáneamente e interactuar entre ellas. El sistema está diseñado específicamente para facilitar la exploración de sistemas descentralizados, fenómenos emergentes, y comportamiento auto-organizable.[1]

En los últimos años se han desarrollado muy variadas versiones comerciales de Logo, como Logo Gráfico en Argentina y Mach Turtle Logo en Canadá.

---

[1] Ajoy, Daniel. Documento electrónico: *¿Qué es Logo?*

Dentro del área de software libre, se han desarrollado también, importantes ambientes Logo, como es el caso de UCBLogo, versión compatible para Macintosh, MSDOS, Windows y Unix, creada por Brian Hervey, autor del clásico Computer Science Logo Style.

Mas adelante, George Mills, usó como base de su desarrollo el núcleo de UCBLogo para crear su MSWLogo, el cual corre bajo Windows, incluyendo mejoras que son posibles en ese sistema operativo.

Durante las últimas tres décadas, los cambios que ha experimentado Logo, han ido unidos al veloz desarrollo de la tecnología de las computadoras. Ahora es cuando mas adeptos posee la familia de ambientes Logo y se ha convertido en un movimiento mundial de la gente que comparte el inmenso interés por la filosofía educativa constructivista.

Anexo II-2

Definición y Características de los PPA

Los PPA se definen de la siguiente forma[2]:

Objetivos de los PPA

Los P.P.A tienen como objetivos generales los siguientes:

- Objetivos relacionados con:

El Educando:

Propiciar aprendizajes significativos e integrales, al relacionar "nuevos conocimientos" con los que ya existen en su estructura cognitiva y con sus experiencias, además le permite construir estrategias para aprender a aprender, describir sus propias potencialidades, ser más participativos, reflexivos y autónomos.

En la búsqueda del sentido de su propia existencia que formulen proyectos de vida, que sepan tomar decisiones y resolver problemas; que prevean las consecuencias de sus acciones, que en su vida activa sean felices, participativos, honestos, responsables y comprometidos con la transformación cultural, económica y social.

El Docente:

Favorecer el crecimiento profesional, al ofrecer posibilidades para su actualización, compartir experiencias con otros colegas y desarrollar su creatividad personal en la planificación y ejecución de la labor educativa.

La Institución Escolar:

---

[2] Extraído del documento electrónico *Proyecto Pedagógico de Aula.*

Elevar la eficiencia y la calidad de la escuela, promover la detección y atención de necesidades e intereses de los alumnos y de la comunidad escolar (especialmente los pedagógicos) contextualizados en el marco del Currículo Básico Nacional y del Currículo Estadal.

Principios y Características:

Los Principios:

- GLOBALIZACIÓN:

En los PPA se organizan todos los componentes del currículo con especial énfasis en los ejes transversales.

- CARÁCTER INVESTIGATIVO:

Definen la investigación acción en el aula como parte esencial del proceso de construcción del aprendizaje.

- CARÁCTER SOCIAL:

Contribuyen a dar respuesta desde el plano educativo en la crisis ética y moral del país.

Características:

- INTEGRADORES:

En ellos se integran los diferentes elementos del currículo ¿Qué, cómo y cuándo enseñar y evaluar?

- COOPERATIVOS:

Son construidos y desarrollados conjuntamente por docentes, alumnos, familia y miembros de la comunidad e implican altos niveles de participación y compromiso grupal.

-PERTINENTES:

Permiten contextualizar los contenidos y las actividades en atención a las características de los educandos y del contexto escolar.

-DIDÁCTICOS:

Orientan la acción formativa mediante una interacción constructiva entre los individuos y el ambiente, evaluando permanentemente la forma de intervenir en ella.

Anexo II-3

El modelo Constructivista

Según el artículo "El Constructivismo" publicado en www.espacioenciclopedico.com, el modelo constructivista está centrado en la persona y en sus experiencias previas, que le permite la realización de nuevas construcciones mentales. Dentro de este modelo se considera que la construcción se produce:

a. Cuando el sujeto interactúa con el objeto del conocimiento (Piaget).

b. Cuando esto lo realiza en interacción con otros (Vigotsky).

c. Cuando es significativo para el sujeto (Ausubel).

En este Modelo el rol del docente cambia. Es moderador, coordinador, facilitador, mediador y también un participante más. El constructivismo supone también un clima afectivo, armónico, de mutua confianza, ayudando a que los alumnos y alumnas se vinculen positivamente con el conocimiento y por sobre todo con su proceso de adquisición.

El profesor como mediador del aprendizaje debe:

- Conocer los intereses de alumnos y alumnas y sus diferencias individuales (Inteligencias Múltiples)
- Conocer las necesidades evolutivas de cada uno de ellos.
- Conocer los estímulos de sus contextos: familiares, comunitarios, educativos y otros.
- Contextualizar las actividades.

Los tres tipos de constructivismo

Actualmente, de acuerdo a lo expuesto por diferentes autores, se puede afirmar de la existencia de tres tipos de constructivismo. Que se mencionará a continuación:

***1)** El aprendizaje es una actividad solitaria.* De acuerdo a la visión de Piaget, Ausubel y la Psicología Cognitiva se basa en la idea de un individuo que aprende al margen de su contexto social. Por supuesto, a nivel teórico se concede un papel a la cultura y a la interacción social, pero no se especifica cómo interactúa con el desarrollo cognitivo y el aprendizaje. En general los autores nos transmiten la imagen de un ser que aprende básicamente en solitario y de manera un tanto autodidacta.

***2)** Con amigos se aprende mejor.* Esta posición ha sido mantenida por investigadores constructivistas que pueden considerarse a medio camino entre las aportaciones piagetianas y cognitivas y las vygotskianas. Los que mantienen esta posición afirman que la interacción social produce un favorecimiento del aprendizaje mediante, el intercambio de información entre compañeros que tienen diferentes niveles de conocimiento, provocando así modificación de los esquemas del individuo y que termina produciendo un aprendizaje, además de mejorar las condiciones motivacionales de la instrucción. En concreto: en este enfoque se estudia el efecto de la interacción y el contexto social sobre el mecanismo de cambio y aprendizaje individual.

***3)** Sin amigos no se puede aprender.* Esta sería la posición vygotskiana radical. Desde esta posición se mantiene que el conocimiento no es un producto individual sino social. Así pues, cuando el alumno está adquiriendo información, lo que establece es un proceso de negociación entre los contenidos establecidos por la sociedad y el individuo. Por tanto, aunque el alumno realice también una actividad individual, el énfasis debe ponerse en

el intercambio social. El peligro que puede tener un enfoque como éste es el riesgo de la desaparición del alumno individual.

Cada una de estas posiciones ofrece una opción dentro del constructivismo, donde los tres son enfoques complementarios y los aportes de cada uno de ellos son mutuamente enriquecedores.

Anexo II-4

Entrevista con el padre del software libre

Extraido de http://www.hispamp3.com/noticias/noticia.php?noticia=20031205090543

Sekhem, 05/12/2003 (09:05).

PuertoRock.com.ar ha llevado a cabo una interesante entrevista a Richard Stallman, el padre del software libre.

- El 27 de Septiembre de 1983 pusiste en marcha el Proyecto GNU y dos años después diste origen a La Fundación para el Software Libre. Hoy, 20 años después, más de 20 millones de personas en todo el mundo utilizan GNU/Linux y diferentes aplicaciones Libres a diario sin tener que ser rehenes del Software Propietario. Teniendo en cuenta esto, que objetivos se plantea la FSF en el mediano y largo plazo? Imaginaste aquel 27 de septiembre todo lo que hoy representa el Software Libre en general y GNU/Linux en particular?.

El trabajo de hoy contiene tres partes: llenar las lagunas en el sistema, desarrollar aplicaciones libres para todas las tareas, y resistir a las empresas globales que tratan de obstaculizar y prohibir el software libre. Debemos oponernos a las patentes informáticas, a las leyes que prohíben software libre para leer DVDs...

Nunca hubiera imaginado cómo las empresas han "utilizado" el movimiento, usando el sistema libre GNU sin mencionar ni "GNU" ni "libre". Ahora no basta promover el uso del software libre, sino también debemos trabajar explícitamente para difundir la filosofía de la libertad....

- Vos trabajabas en el laboratorio de Inteligencia Artificial del MIT, lugar donde muchos sueñan con estar algún día, hasta que una empresa contrató a todos los hackers que trabajaban con vos. Podrías detallarnos cómo fue que tomaste la decisión de renunciar, y empezar con el Proyecto GNU? Cuanto tiempo te llevó eso?

Renuncié al laboratorio, para que la universidad no tuviera derecho propietario sobre mi trabajo, que a partir de entonces, sería el sistema GNU. Para asegurar que GNU sea libre, debía negar a la universidad la opción de restringirlo.

Reflexioné durante unos meses en 1983 para decidir como hacer para construir una nueva comunidad parecido a la que había muerto. Elegí desarrollar un sistema compatible con Unix, porque lograr el desarrollo sería más fácil y el uso sería cómodo para usuarios ya acostumbrado a él.

- Hay mucha gente que opina que si una empresa invierte dinero en desarrollo, ésta está en todo su derecho de no liberar el código de su software, y mantenerlo como software propietario. Que podes decir al respecto?

¡Que guarden su dinero, y no traten de imponer su control a los usuarios! El software privativo es un mal. No se puede obtener por pago el derecho de sujetar a otros.

Pero la mayoría de software no es privativo. Es software privado, usándose dentro de una sola organización. Eso no proporciona ningún problema moral, porque el usuario único no cede su libertad. El asunto moral se plantea al momento de ofrecer el programa a otros exigiendo que cedan su libertad, cuando el desarrollador intenta privar la libertad a otros. Con este acto, el software se vuelve privativo.

- Pero podría beneficiar, económicamente hablando, a una empresa el hecho de liberar el código de sus programas bajo la GNU General Public Licence?

A veces sí, a veces no, depende de que modelo de negocio usa la empresa. También a veces una empresa puede ganar publicando programas libres bajo otras licencias libres.

Pero la ganancia es tema secundario, porque la libertad es más importante que los intereses de cualquiera empresa. Las empresas deben en todos campos adaptarse a respetar la libertad de los individuos.

- El concepto de Software Libre sin dudas ha provocado una revolución. Crees posible que esta filosofía, con variantes claro, se pueda aplicar a otros ámbitos de la vida?

La filosofía de software libre se puede aplicar casi completamente a otras obras de uso funcional, por ejemplo libros de texto y de referencia. Diccionarios y enciclopedias deben ser libres; ya tenemos una gran enciclopedia libre. También se puede aplicar, pero no completamente, a otras obras, como ficción y ensayos.
No puede ir más allá que las obras de información. El asunto de la libertad de copiar sólo se puede aplicar en campos donde hay copiadores. No tiende sentido para objetos físicos. No hay copiadores de coches, ni de mesas, ni de casas, ni de computadoras, ni aún de drogas. Esos se deben producir en fábricas especializadas, y no por un proceso de copia. Si un día existiesen copiadores de objetos físicas, tal vez tengamos la opción de copiarlos y distribuir versiones modificadas

- Hoy en día se habla mucho de Open Source. Podrías sintetizarnos cuales son las diferencias básicas con el Software Libre?

"Open source" es otro término para el software libre, construido específicamente para promover el software mismo, pero ocultando la filosofía de libertad. En el movimiento de open source, no dicen (bajo ningún término) que los usuarios merecen el derecho de copiar, cambiar y distribuir el software que usen. Presentan el asunto en términos superficiales económicos. Sólo hablan de un "modelo de desarrollo", diciendo que usualmente puede proporcionar programas superiores. Cuando dicen "superiores", quieren decir sólo técnicamente.
Para nosotros, es asunto de libertad, no de modelos de desarrollo. Poco importa cómo se desarrolla el programa, siempre que podamos usarlo en libertad. Un programa libre, que sea superior o inferior técnicamente, siempre es superior éticamente. Si un programa es inferior técnicamente pero libre, podemos mejorarlo. Si no es libre, no podemos liberarlo, debemos rechazarlo.

En efecto, "open source" es una manera de utilizar nuestro trabajo, de separarlo de nuestra filosofía. Para empresas que distribuyen copias del sistema GNU/Linux, con añadición de programas privativos ofrecidos como primas, decir "open source" es cómodo. Así pueden invitar a clientes posibles a usar el software y a valorizar la prima privativa, sin enseñarles a reconocer la trampa ética dentro de aquella.

-Bill Gates, Linus Torvalds. Podrías hacernos un comentario sobre cada uno de ellos?

Gates es tonto; no reconoce que ya tiene bastante dinero para su vida, y la vida de sus hijos, y sus nietos, y sus bisnietos, y sus tataranietos, y aún más. Le falta imaginación de qué más puede hacer, en su vida, a parte ganar más dinero.
Pero no piense que nuestra meta es oponernos a Microsoft. El problema que queremos corregir es más grande: el software privativo, que le quita la libertad al usuario. Microsoft es el desarrollador más grande de software privativo, pero todo software privativo impone el mismo problema social. Si piensa que el problema es Microsoft, fácilmente puede suponer que otras empresas de software privativo son buenos por hacer competencia a Microsoft.
Linus Torvalds desarrolló un programa libre importante, el núcleo Linux. No lo hizo para el proyecto GNU, ni para la visión de la libertad, pero su programa completó el sistema GNU, que aún no tenía núcleo. Tener un sistema completo nos permitía de usar una computadora en libertad, que era la meta de GNU desde el principio.

- Hace unos años no cualquier usuario de computadoras tenía la capacidad de instalar GNU/Linux, debido a que generalmente los procesos de instalación no eran todo los "amigables" que un usuario común deseaba. Hoy todas las distribuciones de GNU/Linux han evolucionado en ese sentido, llegando al punto de que cualquier persona con mínimos conocimientos de computación puede instalar el sistema operativo. Que crees que debe mejorarse para que GNU/Linux siga creciendo en el desktop?

Necesitamos más aplicaciones, pero generalmente el sistema ya está listo para usar. Difundamos la noticia.!!

Anexo II-5

Con el software Libre los países generan desarrollo

Extraido de http://www.hispamp3.com/noticias/noticia.php?noticia=20030429105417

Golem, 29/04/2003 (10:54).
A continuación una interesante entrevista realizada al gurú Richard Stallman y en el da a conocer algunas de sus atrevidas propuestas:

La Entrevista fué concedida por Richard Stallman a Alex Barnet de "Muy Interesante" y publicada íntegramente por Rebelion.org.

Richard Stallman, de 49 años, estudió Física en Harvard y trabajó en el Instituto Tecnológico de Massachusetts (MIT) antes de convertirse en el gran libertario de la informática. En 1984 fundó la Free Software Foundation y hoy viaja por el mundo predicando el libre uso de los programas de ordenador. Su ideario inspiró la creación de GNU/Linux, un sistema operativo de acceso gratuito.
Contrario a las patentes de software, Stallman propugna eliminar las restricciones al uso, copiado, redistribución y modificación de los programas informáticos. Su filosofía choca de frente con empresas como Microsoft, cuyo poder sobre el mercado considera ilegítimo.

El movimiento del software libre apunta directamente al corazón del mercado informático, una sabrosa tarta monopolizada por grandes marcas cuyos programas están protegidos por un severo copyright. En cambio, los productos de software libre, desarrollados por programadores voluntarios, pueden ser alterados libremente y distribuirse sin trabas a mucho menor precio o incluso gratuitamente, como el sistema operativo GNU/Linux, que ya utilizan numerosas empresas y administraciones públicas. El físico y ex programador informático norteamericano George Stallman es el principal impulsor y apóstol de esta doctrina.

– ¿Por qué el software libre?

– ¿Por qué el software libre? Porque es conocimiento que necesita la sociedad digital. El software es acumulativo y no se puede desarrollar bien si hay piezas que se bloquean. El software en propiedad –también llamado software con copyright o con licencia– implica un sistema depredador que coloca a los usuarios en una situación precaria. Un programa informático es como una receta de cocina. Cuando vas a preparar un plato, lo normal es que la consultes y luego le des tu toque personal y la compartas con otros. No has de seguirla al pie de la letra, la puedes cambiar y sigue siendo de todos. Imagínese que las recetas estuviesen en un lenguaje secreto y que nadie pudiese cambiarlas o usarlas sin permiso. Sería horrible. Pues así funciona el mundo del software en propiedad. Si haces una copia de la receta para un amigo, te pueden meter en la cárcel.

–Free en inglés significa "libre" y también "gratuito". ¿Free software incluye ambos conceptos?

–No necesariamente. Ante todo, implica libertad en cuatro aspectos: usar el programa; estudiarlo y adaptarlo a tus necesidades; ayudar al prójimo a usarlo, y publicar una versión mejorada en beneficio de todos. El software libre puede ser gratuito o no, aunque la experiencia dice que tiende a ser mucho más barato que el de propiedad. Una vez desarrollado, es muy barato de distribuir, porque reproducirlo prácticamente no tiene coste. No hay justificación para los precios que cobran algunas compañías por programas cuyo coste han amortizado miles de veces.

–¿Propone usted acabar con la industria del software?

–La industria tal y como la conocemos puede desaparecer o adaptarse, y me da igual, porque no la necesitamos. Su modelo es antisocial y anacrónico. Sólo se preocupa de acumular dinero. Los programas que son propiedad de las compañías impiden el progreso y significan un robo a la sociedad.

–¿Sin industria, habrá trabajo para los informáticos? ¿No cree que se estancarían los programas?

–¡Vaya tontería! ¿No existe el GNU/Linux, que es más seguro y estable que Windows y que se ha elaborado con la colaboración voluntaria de casi medio millón de programadores de todo el mundo? La demanda de software seguirá existiendo, y

como la mayoría de los programas se elaboran a medida seguirá habiendo mucho trabajo para los informáticos. No es ningún cataclismo, simplemente se haría un trabajo informático con más sentido social.

–Ustedes se oponen radicalmente a las patentes de programas...

–Porque son muy negativas. Los programas informáticos necesitan muchos elementos y las patentes blindan ideas y sistemas necesarios para seguir mejorando los productos. Las grandes compañías quieren que estos conocimientos se usen sólo con licencias y previo pago. Esto es un freno para los programadores independientes, que se mueven sobre un campo de minas. Pero los programas los escribimos todos usándolos cada día, igual que la lengua inglesa o la española la construimos millones de personas al hablarla cada día.

–Cuesta imaginar una industria hecha por trabajadores voluntarios, sin patentes, que cubra un mercado tan vasto y genere dinero.

–Pensar así es caer en la trampa que nos tienden las grandes empresas: quieren que creamos que los humanos sólo nos movemos por el provecho económico, porque eso es lo que hacen ellos. GNU/ Linux es una prueba de que existe una alternativa.

–¿Si el software libre es tan atractivo, por qué aún es desconocido para muchos usuarios?

–Inercia social. La gente usa Windows porque la mayoría lo utiliza. Esto es un escollo para nosotros, pero la situación está variando. Se decía que GNU/Linux fracasaría, pero ya está instalado en 20 millones de ordenadores y funciona muy bien. Cada vez hay más empresas y gobiernos interesados y pronto se incorporarán los usuarios particulares, pues ya disponen de procesadores de texto, programas de imagen, multimedia...

En 1992, el finlandés Linus Torvalds, un seguidor de la doctrina de Stallman, colocó en Internet el embrión de GNU/Linux, un sistema operativo gratuito. Hoy, Google, el mayor buscador de la red, funciona con él. También lo usan productoras de cine y de efectos digitales como Disney, Pixar, Light and Magic y Digital Domain. Intel y Amazon lo usan en algunas parcelas de sus complejos sistemas, y Panasonic y

Sony lo incluirán en una próxima generación de electrodomésticos inteligentes. Alemania, Noruega, Nigeria, Sudáfrica, Francia, India, China y Brasil son algunos de los países que tienen en marcha proyectos institucionales para usar Linux y ahorrarse el pago de copyrights. Por su parte, la Junta de Extremadura ha lanzado el proyecto LinEx, por el cual ha repartido 80.000 copias gratuitas de este sistema en su comunidad y con el que se ha ahorrado 13 millones de euros en licencias.

–¿El principal combate entre el software en propiedad y el libre se dará en los países en desarrollo?

–Con el software libre, además de ser independientes, los países pueden ahorrarse mucho dinero que ahora va a parar a manos de las compañías. En zonas con poco presupuesto para tecnología, distribuir libremente los programas es una vía para generar desarrollo. Por ejemplo, en Sao Paulo hay centros públicos que tienen instalado software libre en los ordenadores y que se dedican a la educación digital de usuarios que no sabían nada de informática.

–El auge del software libre afecta especialmente a Microsoft, que controla el 95 por 100 del mercado de PCs y buena parte de la red. ¿Cómo se llevan con esta empresa?

–IBM, Oracle o Sun no son mejores, pero Microsoft es la firma que más tajada saca y la que usa más medios para desprestigiarnos y bloquearnos. El juicio antimonopolio no sirvió para nada y la resolución de dividir la empresa está diseñada para que continúe con su modelo y prácticas. La llegada de Bush fue decisiva. Como es sabido, Microsoft contribuyó con mucho dinero a su campaña y el Presidente le ha devuelto el favor. –Microsoft va a dejar consultar el código fuente de Windows a gobiernos y organismos internacionales para que hagan adaptaciones personalizadas. ¿Le sorprende?

– –Microsoft va a dejar consultar el código fuente de Windows a gobiernos y organismos internacionales para que hagan adaptaciones personalizadas. ¿Le sorprende? –No. Los analistas internacionales han visto, con razón, que es una medida para frenar la ascensión de GNU/Linux entre las administraciones públicas. Además, es una medida limitada, que sólo aplicarán a clientes muy importantes.

Microsoft seguirá controlando Windows y cobrando las licencias al precio que quiera. Tener acceso al código fuente no es todo.

–¿La idea del software libre se puede extrapolar a otros ámbitos como los libros o la música?

–Se puede aplicar a herramientas de trabajo o consulta, como diccionarios, enciclopedias o manuales, pues las obras de utilidad social deben ser libres. Pero las obras de entretenimiento, una novela o una canción, no tienen por qué serlo. De todas maneras, siempre debe existir la libertad para difundirla sin fines lucrativos

–¿Cómo ve la piratería digital?

–Hacer copias de un producto ajeno y venderlas para ganar dinero es un delito. Pero la copia que uno comparte sin fines comerciales no tiene por qué ser un delito. Muchas de las medidas que se están tomando indican que la batalla legal y política se plantea no contra las organizaciones que piratean, sino contra la idea y el derecho individual de compartir.

–¿Qué piensa sobre las patentes en general?

–Es un tema complejo. Creo que las tecnologías necesarias para garantizar la vida de la gente, como ciertos tratamientos médicos, deben fabricarse gratuitamente o a bajo coste. Las patentes tal vez tengan sentido en entornos ricos como Europa y Estados Unidos, pero no en países pobres.

–¿Le gusta cuando al movimiento del software libre le tachan de revolucionario e idealista?

–Esto no es una revolución. Sólo buscamos un cambio en un aspecto de la vida. Las cosas cambian constantemente, ¿por qué no va a hacerlo el mercado informático? Respecto al idealismo, hay que recordar que la única manera de transformar las cosas es creer que se pueden transformar.

–¿Cuál es su pronóstico sobre el futuro de su propuesta?

–Pienso que si muchas personas entienden el mensaje y adoptan nuestra idea, el software libre irá hacia delante. Desde que empezamos hemos avanzado mucho.

–¿David contra Goliat o Don Quijote contra los molinos de viento?

–Me quedo con David. Don Quijote es interesante, pero nosotros sabemos que nuestro enemigo existe y no tenemos ninguna confusión entre realidad y fantasía. Además, David logró tumbar a Goliat.

Anexo II-6

Licencia Pública GNU

Esta es la conocida GNU Public License (GPL), versión 2 (de junio de 1.991), que cubre la mayor parte del software de la Free Software Foundation, y muchos más programas.

Los autores de esta traducción son:
Jesús González Barahona
Pedro de las Heras Quirós

NOTA IMPORTANTE:

Esta es una traducción no oficial al español de la GNU General Public License. No ha sido publicada por la Free Software Foundation, y no establece legalmente las condiciones de distribución para el software que usa la GNU GPL. Estas condiciones se establecen solamente por el texto original, en inglés, de la GNU GPL. Sin embargo, esperamos que esta traducción ayude a los hispanohablantes a entender mejor la GNU GPL.

IMPORTANT NOTICE:

This is an unofficial translation of the GNU General Public License into Spanish. It was not published by the Free Software Foundation, and does not legally state the distribution terms for software that uses the GNU GPL--only the original English text of the GNU GPL does that. However, we hope that this translation will help Spanish speakers understand the GNU GPL better.

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, EEUU
Se permite la copia y distribución de copias literales de este documento, pero no se permite su modificación.

Preámbulo

Las licencias que cubren la mayor parte del software están diseñadas para quitarle a usted la libertad de compartirlo y modificarlo. Por el contrario, la Licencia Pública General de GNU pretende garantizarle la libertad de compartir y modificar software libre, para asegurar que el software es libre para todos sus usuarios. Esta Licencia Pública General se aplica a la mayor parte del software del la Free Software Foundation y a cualquier otro programa si sus autores se comprometen a utilizarla. (Existe otro software de la Free Software Foundation que está cubierto por la Licencia Pública General de GNU para Bibliotecas). Si quiere, también puede aplicarla a sus propios programas.

Cuando hablamos de software libre, estamos refiriéndonos a libertad, no a precio. Nuestras Licencias Públicas Generales están diseñadas para asegurarnos de que tenga la libertad de distribuir copias de software libre (y cobrar por ese servicio si quiere), de que reciba el código fuente o que pueda conseguirlo si lo quiere, de que pueda modificar el software o usar

fragmentos de él en nuevos programas libres, y de que sepa que puede hacer todas estas cosas.

Para proteger sus derechos necesitamos algunas restricciones que prohiban a cualquiera negarle a usted estos derechos o pedirle que renuncie a ellos. Estas restricciones se traducen en ciertas obligaciones que le afectan si distribuye copias del software, o si lo modifica.

Por ejemplo, si distribuye copias de uno de estos programas, sea gratuitamente, o a cambio de una contraprestación, debe dar a los receptores todos los derechos que tiene. Debe asegurarse de que ellos también reciben, o pueden conseguir, el código fuente. Y debe mostrarles estas condiciones de forma que conozcan sus derechos.

Protegemos sus derechos con la combinación de dos medidas:

Ponemos el software bajo copyright
le ofrecemos esta licencia, que le da permiso legal para copiar, distribuir y/o modificar el software.

También, para la protección de cada autor y la nuestra propia, queremos asegurarnos de que todo el mundo comprende que no se proporciona ninguna garantía para este software libre. Si el software se modifica por cualquiera y éste a su vez lo distribuye, queremos que sus receptores sepan que lo que tienen no es el original, de forma que cualquier problema introducido por otros no afecte a la reputación de los autores originales.

Por último, cualquier programa libre está constantemente amenazado por patentes sobre el software. Queremos evitar el peligro de que los redistribuidores de un programa libre obtengan patentes por su cuenta, convirtiendo de facto el programa en propietario. Para evitar esto, hemos dejado claro que cualquier patente debe ser pedida para el uso libre de cualquiera, o no ser pedida.

Los términos exactos y las condiciones para la copia, distribución y modificación se exponen a continuación.

Términos y condiciones para la copia, distribución y modificación

Esta Licencia se aplica a cualquier programa u otro tipo de trabajo que contenga una nota colocada por el tenedor del copyright diciendo que puede ser distribuido bajo los términos de esta Licencia Pública General. En adelante, Programa se referirá a cualquier programa o trabajo que cumpla esa condición y trabajo basado en el Programa se referirá bien al Programa o a cualquier trabajo derivado de él según la ley de copyright. Esto es, un trabajo que contenga el programa o una proción de él, bien en forma literal o con modificaciones y/o traducido en otro lenguaje. Por lo tanto, la traducción está incluida sin limitaciones en el término modificación. Cada concesionario (licenciatario) será denominado usted.

Cualquier otra actividad que no sea la copia, distribución o modificación no está cubierta por esta Licencia, está fuera de su ámbito. El acto de ejecutar el Programa no está restringido, y los resultados del Programa están cubiertos únicamente si sus contenidos constituyen un trabajo basado en el Programa, independientemente de haberlo producido mediante la ejecución del programa. El que esto se cumpla, depende de lo que haga el programa.

Usted puede copiar y distribuir copias literales del código fuente del Programa, según lo has recibido, en cualquier medio, supuesto que de forma adecuada y bien visible publique en cada copia un anuncio de copyright adecuado y un repudio de garantía, mantenga intactos todos los anuncios que se refieran a esta Licencia y a la ausencia de garantía, y proporcione a cualquier otro receptor del programa una copia de esta Licencia junto con el Programa.

Puede cobrar un precio por el acto físico de transferir una copia, y puede, según su libre albedrío, ofrecer garantía a cambio de unos honorarios.

Puede modificar su copia o copias del Programa o de cualquier porción de él, formando de esta manera un trabajo basado en el Programa, y copiar y distribuir esa modificación o trabajo bajo los términos del apartado 1, antedicho, supuesto que además cumpla las siguientes condiciones:

Debe hacer que los ficheros modificados lleven anuncios prominentes indicando que los ha cambiado y la fecha de cualquier cambio.
Debe hacer que cualquier trabajo que distribuya o publique y que en todo o en parte contenga o sea derivado del Programa o de cualquier parte de él sea licenciada como un todo, sin carga alguna, a todas las terceras partes y bajo los términos de esta Licencia.
Si el programa modificado lee normalmente órdenes interactivamente cuando es ejecutado, debe hacer que, cuando comience su ejecución para ese uso interactivo de la forma más habitual, muestre o escriba un mensaje que incluya un anuncio de copyright y un anuncio de que no se ofrece ninguna garantía (o por el contrario que sí se ofrece garantía) y que los usuarios pueden redistribuir el programa bajo estas condiciones, e indicando al usuario cómo ver una copia de esta licencia. (Excepción: si el propio programa es interactivo pero normalmente no muestra ese anuncio, no se requiere que su trabajo basado en el Programa muestre ningún anuncio).

Estos requisitos se aplican al trabajo modificado como un todo. Si partes identificables de ese trabajo no son derivadas del Programa, y pueden, razonablemente, ser consideradas trabajos independientes y separados por ellos mismos, entonces esta Licencia y sus términos no se aplican a esas partes cuando sean distribuidas como trabajos separados. Pero cuando distribuya esas mismas secciones como partes de un todo que es un trabajo basado en el Programa, la distribución del todo debe ser según los términos de esta licencia, cuyos permisos para otros licenciatarios se extienden al todo completo, y por lo tanto a todas y cada una de sus partes, con independencia de quién la escribió.

Por lo tanto, no es la intención de este apartado reclamar derechos o desafiar sus derechos sobre trabajos escritos totalmente por usted mismo. El intento es ejercer el derecho a controlar la distribución de trabajos derivados o colectivos basados en el Programa.

Además, el simple hecho de reunir un trabajo no basado en el Programa con el Programa (o con un trabajo basado en el Programa) en un volumen de almacenamiento o en un medio de distribución no hace que dicho trabajo entre dentro del ámbito cubierto por esta Licencia.

Puede copiar y distribuir el Programa (o un trabajo basado en él, según se especifica en el apartado 2, como código objeto o en formato ejecutable según los términos de los apartados 1 y 2, supuesto que además cumpla una de las siguientes condiciones:

Acompañarlo con el código fuente completo correspondiente, en formato electrónico, que debe ser distribuido según se especifica en los apartados 1 y 2 de esta Licencia en un medio habitualmente utilizado para el intercambio de programas, o
Acompañarlo con una oferta por escrito, válida durante al menos tres años, de proporcionar a cualquier tercera parte una copia completa en formato electrónico del código fuente correspondiente, a un coste no mayor que el de realizar físicamente la distribución del fuente, que será distribuido bajo las condiciones descritas en los apartados 1 y 2 anteriores, en un medio habitualmente utilizado para el intercambio de programas, o
Acompañarlo con la información que recibiste ofreciendo distribuir el código fuente correspondiente. (Esta opción se permite sólo para distribución no comercial y sólo si usted recibió el programa como código objeto o en formato ejecutable con tal oferta, de acuerdo con el apartado b anterior).

Por código fuente de un trabajo se entiende la forma preferida del trabajo cuando se le hacen modificaciones. Para un trabajo ejecutable, se entiende por código fuente completo todo el código fuente para todos los módulos que contiene, más cualquier fichero asociado de definición de interfaces, más los guiones utilizados para controlar la compilación e instalación del ejecutable.

Como excepción especial el código fuente distribuido no necesita incluir nada que sea distribuido normalmente (bien como fuente, bien en forma binaria) con los componentes principales (compilador, kernel y similares) del sistema operativo en el cual funciona el ejecutable, a no ser que el propio componente acompañe al ejecutable.

Si la distribución del ejecutable o del código objeto se hace mediante la oferta acceso para copiarlo de un cierto lugar, entonces se considera la oferta de acceso para copiar el código fuente del mismo lugar como distribución del código fuente, incluso aunque terceras partes no estén forzadas a copiar el fuente junto con el código objeto.

No puede copiar, modificar, sublicenciar o distribuir el Programa excepto como prevé expresamente esta Licencia. Cualquier intento de copiar, modificar sublicenciar o distribuir el Programa de otra forma es inválida, y hará que cesen automáticamente los derechos que te proporciona esta Licencia. En cualquier caso, las partes que hayan recibido copias o derechos de usted bajo esta Licencia no cesarán en sus derechos mientras esas partes continúen cumpliéndola.

No está obligado a aceptar esta licencia, ya que no la ha firmado. Sin embargo, no hay hada más que le proporcione permiso para modificar o distribuir el Programa o sus trabajos derivados. Estas acciones están prohibidas por la ley si no acepta esta Licencia. Por lo tanto, si modifica o distribuye el Programa (o cualquier trabajo basado en el Programa), está indicando que acepta esta Licencia para poder hacerlo, y todos sus términos y condiciones para copiar, distribuir o modificar el Programa o trabajos basados en él.

Cada vez que redistribuya el Programa (o cualquier trabajo basado en el Programa), el receptor recibe automáticamente una licencia del licenciatario original para copiar, distribuir o modificar el Programa, de forma sujeta a estos términos y condiciones. No puede imponer al receptor ninguna restricción más sobre el ejercicio de los derechos aquí garantizados. No es usted responsable de hacer cumplir esta licencia por terceras partes.

Si como consecuencia de una resolución judicial o de una alegación de infracción de patente o por cualquier otra razón (no limitada a asuntos relacionados con patentes) se le imponen condiciones (ya sea por mandato judicial, por acuerdo o por cualquier otra causa) que contradigan las condiciones de esta Licencia, ello no le exime de cumplir las condiciones de esta Licencia. Si no puede realizar distribuciones de forma que se satisfagan simultáneamente sus obligaciones bajo esta licencia y cualquier otra obligación pertinente entonces, como consecuencia, no puede distribuir el Programa de ninguna forma. Por ejemplo, si una patente no permite la redistribución libre de derechos de autor del Programa por parte de todos aquellos que reciban copias directa o indirectamente a través de usted, entonces la única forma en que podría satisfacer tanto esa condición como esta Licencia sería evitar completamente la distribución del Programa.

Si cualquier porción de este apartado se considera inválida o imposible de cumplir bajo cualquier circunstancia particular ha de cumplirse el resto y la sección por entero ha de cumplirse en cualquier otra circunstancia.

No es el propósito de este apartado inducirle a infringir ninguna reivindicación de patente ni de ningún otro derecho deç propiedad o impugnar la validez de ninguna de dichas reivindicaciones. Este apartado tiene el único propósito de proteger la integridad del sistema de distribución de software libre, que se realiza mediante prácticas de licencia pública. Mucha gente ha hecho contribuciones generosas a la gran variedad de software distribuido mediante ese sistema con la confianza de que el sistema se aplicará consistentemente. Será el autor/donante quien decida si quiere distribuir software mediante cualquier otro sistema y una licencia no puede imponer esa elección.

Este apartado pretende dejar completamente claro lo que se cree que es una consecuencia del resto de esta Licencia.

Si la distribución y/o uso de el Programa está restringida en ciertos países, bien por patentes o por interfaces bajo copyright, el tenedor del copyright que coloca este Programa bajo esta Licencia puede añadir una limitación explícita de distribución geográfica excluyendo esos países, de forma que la distribución se permita sólo en o entre los países no excluidos de esta manera. En ese caso, esta Licencia incorporará la limitación como si estuviese escrita en el cuerpo de esta Licencia.

La Free Software Foundation puede publicar versiones revisadas y/o nuevas de la Licencia Pública General de tiempo en tiempo. Dichas nuevas versiones serán similares en espíritu a la presente versión, pero pueden ser diferentes en detalles para considerar nuevos problemas o situaciones.

Cada versión recibe un número de versión que la distingue de otras. Si el Programa especifica un número de versión de esta Licencia que se refiere a ella y a cualquier versión posterior, tienes la opción de seguir los términos y condiciones, bien de esa versión, bien de cualquier versión posterior publicada por la Free Software Foundation. Si el Programa no especifica un número de versión de esta Licencia, puedes escoger cualquier versión publicada por la Free Software Foundation.

Si quiere incorporar partes del Programa en otros programas libres cuyas condiciones de distribución son diferentes, escribe al autor para pedirle permiso. Si el software tiene

copyright de la Free Software Foundation, escribe a la Free Software Foundation: algunas veces hacemos excepciones en estos casos. Nuestra decisión estará guiada por el doble objetivo de de preservar la libertad de todos los derivados de nuestro software libre y promover el que se comparta y reutilice el software en general.

AUSENCIA DE GARANTÍA

Como el programa se licencia libre de cargas, no se ofrece ninguna garantía sobre el programa, en todas la extensión permitida por la legislación aplicable. Excepto cuando se indique de otra forma por escrito, los tenedores del copyright y/u otras partes proporcionan el programa tal cual, sin garantía de ninguna clase, bien expresa o implícita, con inclusión, pero sin limitación a las garantías mercantiles implícitas o a la conveniencia para un propósito particular. Cualquier riesgo referente a la calidad y prestaciones del programa es asumido por usted. Si se probase que el Programa es defectuoso, asume el coste de cualquier servicio, reparación o corrección.

En ningún caso, salvo que lo requiera la legislaciónç aplicable o haya sido acordado por escrito, ningún tenedor del copyright ni ninguna otra parte que modifique y/o redistribuya el Programa según se permite en esta Licencia será responsable ante usted por daños, incluyendo cualquier daño general, especial, incidental o resultante producido por el uso o la imposibilidad de uso del Programa (con inclusión, pero sin limitación a la pérdida de datos o a la generación incorrecta de datos o a pérdidas sufridas por usted o por terceras partes o a un fallo del Programa al funcionar en combinación con cualquier otro programa), incluso si dicho tenedor u otra parte ha sido advertido de la posibilidad de dichos daños.

FIN DE TÉRMINOS Y CONDICIONES

Apéndice: Cómo aplicar estos términos a sus nuevos programas.

para el público en general, la mejor forma de conseguirlo es convirtiéndolo en software libre que cualquiera pueda redistribuir y cambiar bajo estos términos.

Para hacerlo, añada los siguientes anuncios al programa. Lo más seguro es añadirlos al principio de cada fichero fuente para transmitir lo más efectivamente posible la ausencia de garantía. Además cada fichero debería tener al menos la línea de copyright y un indicador a dónde puede encontrarse el anuncio completo.

<una línea para indicar el nombre del programa y una rápida idea de qué hace.>

Copyright (C) 19aa <nombre del autor
Este programa es software libre. Puede redistribuirlo y/o modificarlo bajo los términos de la Licencia Pública General de GNU según es publicada por la Free Software Foundation, bien de la versión 2 de dicha Licencia o bien (según su elección) de cualquier versión posterior.

Este programa se distribuye con la esperanza de que sea útil, pero SIN NINGUNA GARANTÍA, incluso sin la garantía MERCANTIL implícita o sin garantizar la CONVENIENCIA PARA UN PROPÓSITO PARTICULAR. Véase la Licencia Pública General de GNU para más detalles.

Debería haber recibido una copia de la Licencia Pública General junto con este programa. Si no ha sido así, escriba a la Free Software Foundation, Inc., en 675 Mass Ave, Cambridge, MA 02139, EEUU.

Añada también información sobre cómo contactar con usted mediante correo electrónico y postal.

Si el programa es interactivo, haga que muestre un pequeño anuncio como el siguiente, cuando comienza a funcionar en modo interactivo:

Gnomovision versión 69, Copyright (C) 19aa nombre del autor

Gnomovision no ofrece ABSOLUTAMENTE NINGUNA GARANTÍA. Para más detalles escriba show w.

Los comandos hipotéticos show w y show c deberían mostrar las partes adecuadas de la Licencia Pública General. Por supuesto, los comandos que use pueden llamarse de cualquier otra manera. Podrían incluso ser pulsaciones del ratón o elementos de un menú (lo que sea apropiado para su programa).

También deberías conseguir que su empleador (si trabaja como programador) o tu Universidad (si es el caso) firme un renuncia de copyright para el programa, si es necesario. A continuación se ofrece un ejemplo, altere los nombres según sea conveniente:

Yoyodyne, Inc. mediante este documento renuncia a cualquier interés de derechos de copyright con respecto al programa Gnomovision (que hace pasadas a compiladores) escrito por Pepe Programador.
<firma de Pepito Grillo>, 20 de diciembre de 1996
Pepito Grillo, Presidente de Asuntillos Varios.

Esta Licencia Pública General no permite que incluya sus programas en programas propietarios. Si su programa es una biblioteca de subrutinas, puede considerar más útil el permitir el enlazado de aplicaciones propietarias con la biblioteca. Si este es el caso, use la Licencia Pública General de GNU para Bibliotecas en lugar de esta Licencia.

Anexo IV-1

Encuestas Nº1

Nombre: _____ Teléfono: _____

Nombre de la Institución: _____

Ha utilizado Micromundo Si: __ No: __

Versión: 1.0: __ 2.0: __ Pro: __

Nivel de Conocimiento

Principiante: __ Intermedio: __ Avanzado: __

En su Institución cuenta con Micromundo Si: __ No: __

Versión: 1.0: __ 2.0: __ Pro: __

Estaría usted dispuesto a colaborar en un proyecto de creación de un software basado en Micromundo que se adapte a la realidad económica del país.

Si: __ No: __

¿Por qué?: _____

_____

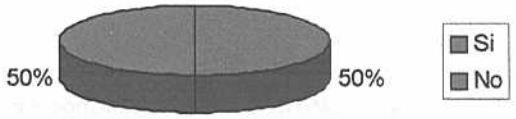
_____

Anexo IV-2

Resultados arrojados por las encuestas Nº 1

Una vez tabulados las respuestas obtenidas en las encuestas realizadas a los profesores, estos fueron los datos obtenidos:

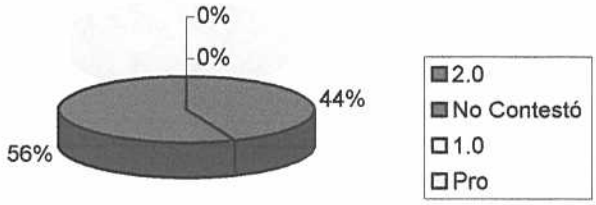
¿Ha utilizado Micromundo?



Gráfica 1: Uso de MicroMundos en las escuelas de Fe y Alegría

A la pregunta de si ha utilizado alguna vez MicroMundos en alguna de sus versiones, se observa que el 50% de la población lo ha usado.

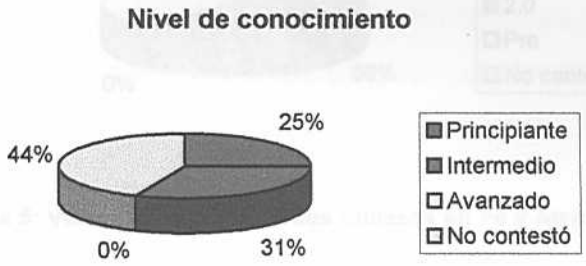
¿Qué versión?



Gráfica 2: Versión de MicroMundos utilizada por los profesores

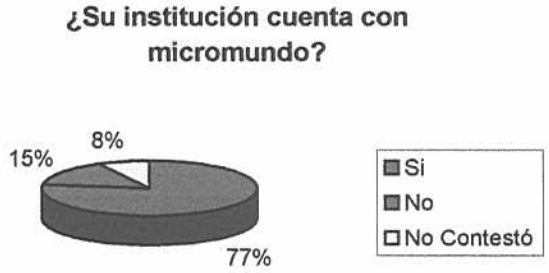
Con respecto a la versión, el 44% de la población ha utilizado MicroMundos 2.0 y el 56% no contestó, las versiones MicroMundos Pro (la

más actualizada del mercado)[1] y MicroMundos 1.0 (primera versión) no han sido utilizadas por la población muestra.



Gráfica 3: Nivel de conocimiento de MicroMundos por los profesores

El nivel de conocimiento por parte de los profesores de la herramienta se encuentra entre el intermedio (31%) y principiante (25%), el 44% de la población no contestó.



Gráfica 4: Utilización de MicroMundos dentro de las escuelas de Fe y Alegría

En total de 12 instituciones a las que pertenecían los profesores encuestados, el 77% (diez) de las instituciones poseen MicroMundos, un 15% (dos) no lo poseen y un 8% no contestó.

[1] Para la fecha de entrega de éste tomo se liberó MicroMundos EX.

Gráfica 5: Versión de MicroMundos utilizada en Fe y Alegría

Con respecto a las versiones de MicroMundos utilizadas en las instituciones de Fe y Alegría, un 56% de la población afirmó que se utilizaba la versión 2.0, el 38% no contestó y el 6% dice utilizar MicroMundos 1.0.

Anexo IV-3

Comentarios de profesores encuestados

A continuación se presentan algunas de las razones por las cuales los profesores encuestados estarían de acuerdo o no colaborar en el proyecto:

Profesora Celia Pita

Institución: Simón Bolívar – Caraballeda

Estuvo de acuerdo en colaborar con el proyecto.

"Deseo conocer MicroMundos, considero importante adaptarlo a la realidad económica ya que resulta costoso y es importante su contenido y por comentarios de algunos compañeros es un software que se adapta a muchos contenidos y a muchos niveles."

Profesora Nidia Teresa Mata

Institución: María Rosa Molas

Estuvo de acuerdo en colaborar con el proyecto.

"Le permite al niño crear y relacionarse con el computador. Es interesante que esto llegue al mayor número de alumnos del país."

Profesora Solanger Muñoz

Institución: Padre Joaquín López

Estuvo de acuerdo en colaborar con el proyecto.

"Porque es un software divertido, pedagógico y permite el desarrollo de habilidades cognitivas"

Profesor José A. Trejo

Institución: Sagrada Familia

Estuvo de acuerdo en colaborar con el proyecto.

"Permite desarrollar el desarrollo lógico matemático en el área de programación"

Profesor Demian Alcalá
Institución: Padre Joaquín López S.J
No estuvo de acuerdo en colaborar con el proyecto.
"No todos los equipos poseen MicroMundos, y desconozco el software del mismo, y por último soy suplente del área, por tiempo determinado y no poseo la disponibilidad del tiempo."

Profesor Ángel Chong
Institución: Jesús Maestro
No estuvo de acuerdo en colaborar con el proyecto.
"No tengo preparación necesaria para desarrollarlo. No lo manejo lo suficiente."

Anexo IV-4

Comparación de primitivas de MSWLogo y MicroMundos

Notación: las filas de cada tabla, que están resaltadas, representan aquellas primitivas de MSWLogo que tienen su equivalencia en MicroMundos. Para hacer más sencilla la ubicación de la primitiva equivalente en cada una de las tablas, éstas poseen una numeración igual en ambas tablas.

MSWLogo:

Nº	Comando	Abrevia	Parámetros	Característica
			GRÁFICOS	
Múltiples Tortugas				
	setturtle		tortuga	Selecciona una tortuga. Observe que el número de tortugas comienza con 0 (tortuga por defecto).
	turtle			Devuelve el número correspondiente a la tortuga actual.
	turtles			Devuelve el número máximo de tortugas que han sido seleccionadas desde la última vez que se ejecutó un *clearscreen*.
	bitmapturtle			La tortuga adquiere la forma del bitmap que se encuentra en el portapapeles.
	nobitmapturtle			Restaura la forma normal de la tortuga.
Movimiento de la Tortuga				
1	forward	fd	distancia	Mueve la tortuga hacia adelante
2	back	bk	distancia	Mueve la tortuga hacia atrás
11	left	lt	ángulo	La tortuga gira en sentido contrario a las agujas del reloj en el ángulo especificado.
10	right	rt	ángulo	La tortuga gira en el mismo sentido que las agujas del reloj en el ángulo especificado.
	leftroll	lr	ángulo	La tortuga gira a la izquierda en el ángulo especificado, medido en grados (1/360 de la circunferencia). Esta primitiva trabaja en modo PERSPECTIVA.
	rightroll	rr	ángulo	La tortuga gira a la derecha en el ángulo especificado, medido en

				grados (1/360 de la circunferencia). Esta primitiva trabaja en modo PERSPECTIVA.
	downpitch	down	ángulo	La tortuga inclina la nariz hacia abajo en el ángulo especificado, medido en grados (1/360 de la circunferencia). Esta primitiva trabaja en modo PERSPECTIVA.
	uppitch	up	ángulo	La tortuga inclina la nariz hacia arriba en el ángulo especificado, medido en grados (1/360 de la circunferencia). Esta primitiva trabaja en modo PERSPECTIVA.
15	setpos		[x y]	Mueve la tortuga a las coordenadas absolutas X, Y.
	setposxyz		[x y z]	Mueve la tortuga a las coordenadas absolutas X, Y, Z.
	setxy		xcor ycor	Mueve la tortuga a las coordenadas absolutas X, Y. Requiere dos argumentos numéricos, las coordenadas X , Y. (permite usar variables)
	setxyz		xcor ycor zcor	Mueve la tortuga a las coordenadas absolutas X, Y, Z. Requiere dos argumentos numéricos, las coordenadas X , Y, Z. (permite usar variables)
17	setx		xcor	Mueve la tortuga a lo largo del eje X desde la posición actual a una nueva coordenada absoluta X. El argumento es la nueva coordenada X.
18	sety		ycor	Mueve la tortuga a lo largo del eje Y desde la posición actual a una nueva coordenada absoluta Y. El argumento es la nueva coordenada Y.
	setz		zcor	Mueve la tortuga a lo largo del eje Z desde la posición actual a una nueva coordenada absoluta Z. El argumento es la nueva coordenada Z.
22	home			Sitúa la tortuga en el centro de la pantalla.
16	setheading	seth	ángulo	Cambia el rumbo de la tortuga por un nuevo rumbo absoluto. El argumento es un ángulo, el rumbo se mide en grados según el

				movimiento de las agujas del reloj desde el eje X.
	setroll		ángulo	La tortuga realiza un nuevo giro positivo, desde el eje positivo X hasta el eje negativo, rotando alrededor del eje Y. Esta primitiva se utiliza en modo PERSPECTIVA.
	setpitch		ángulo	Produce una nueva inclinación positiva desde el eje negativo Z hasta el eje Y, rotando alrededor del eje X. Esta primitiva se utiliza en modo PERSPECTIVA.
	setorientation		[x y z], lista (Proced. de la librería)	Cambia la orientación de la tortuga por una nueva orientación absoluta.
	ellipsearc *		ángulo menor mayor principioángulo (Proced. de la librería)	*Ellipsearc* no mueve la tortuga. Dibuja una parte o la totalidad de una elipse, según el rumbo, posición de la tortuga y los argumentos. El tamaño depende de los valores menor y mayor. La posición de la tortuga quedará en el centro de la elipse. Menor es la distancia más corta desde la elipse hasta el centro. Mayor es la distancia más larga desde la elipse hasta el centro. *Elipsa2* dibuja la elipse desde el borde donde queda finalmente posicionada.
	ellipse *		menor mayor	*Ellipse* no mueve la tortuga. Dibuja una elipse, según el rumbo, posición de la tortuga y argumentos. El tamaño viene determinado por los valores menor y mayor. Al finalizar, la tortuga queda posicionada en el centro de la elipse. Menor es la distancia más corta desde la elipse hasta el centro. Mayor es la distancia más larga desde la elipse hasta el centro.
	arc *		ángulo radio	Dibuja un arco (parte de la circunferencia), según el rumbo, posición de la tortuga y argumentos. El tamaño depende del radio. Al finalizar, la tortuga queda posicionada en el centro del arco. *arc* 360 dibuja una circunferencia.
	circle *		radio	*Circle* no mueve la tortuga. Dibuja

				un círculo, según el rumbo, posición de la tortuga y argumentos. El tamaño depende del radio.
Preguntas sobre el Movimiento de la Tortuga				
23	pos			Devuelve la posición actual de la tortuga, como una lista de dos números, las coordenadas X e Y.
	posxyz			Devuelve la posición actual de la tortuga, como una lista de tres números, las coordenadas X, Y y Z. Esta orden está diseñada para trabajar en modo PERSPECTIVA.
6	xcor			Devuelve un número, la coordenada X de la tortuga.
7	ycor			Devuelve un número, la coordenada Y de la tortuga.
	zcor			Devuelve un número, la coordenada Z de la tortuga.. Está orden está diseñada para trabajar en modo PERSPECTIVA.
24	heading			Devuelve un ángulo que indica el rumbo de la tortuga en grados.
	roll			Devuelve un ángulo, que indica el balanceo de la tortuga. En modo PERSPECTIVA.
	pitch			Devuelve un ángulo, que indica el cabeceo de la tortuga. En modo PERSPECTIVA.
	orientation			Devuelve una lista con el balanceo, cabeceo y rumbo de la tortuga en grados. En modo PERSPECTIVA.
	towards		pos [x y]	Devuelve un ángulo que indica el rumbo en grados que debe tomar la tortuga para moverse desde su posición actual hasta otra que se indica en el argumento **pos**.
	towardsxyz		pos [x y z]	Devuelve un ángulo que indica el rumbo en grados que debe tomar la tortuga para moverse desde su posición actual hasta otra que se indica en el argumento **pos**. Esta primitiva sólo trabaja en modo PERSPECTIVA.
	distance		pos [x y]	Devuelve un número que indica la distancia que la tortuga tiene que recorrer en línea recta hasta llegar a

				la posición que se indica en el argumento **pos**.
	distancexyz		pos [x y z]	Devuelve un número que indica la distancia que la tortuga tiene que recorrer en línea recta hasta llegar a la posición que se indica en el argumento pos, con las coordenadas X, Y y Z. Esta orden sólo se utiliza en modo PERSPECTIVA.
	píxel			Devuelve una lista con los números que representan la intensidad de los colores Rojo, Verde y Azul del píxel que se encuentra bajo la tortuga.
	scrunch			Devuelve una lista con dos números, X e Y que representan al radio.
La Tortuga y el Control de la Ventana				
21	showturtle	st		Hace visible la tortuga.
	hideturtle	ht		La tortuga es invisible. Es una buena idea emplear esta primitiva cuando la tortuga realiza un dibujo complicado, porque cuando está oculta aumenta la velocidad.
20	clean			Borra todas las líneas que la tortuga dibujó en la ventana de gráficos.
3	clearscreen	cs		Borra los gráficos de la ventana y envía la tortuga a su posición y rumbo inicial.
	wrap			Modo envolver, cuando la tortuga sobrepasa los límites de la pantalla por uno de sus bordes, aparece por el borde opuesto. Es el modo inicial de la tortuga.
	window			La tortuga se encuentra en el modo VENTANA: si se pide que la tortuga se mueva fuera de los límites de la pantalla, se alejará de ella. La ventana visible de gráficos se considera como una parte de un plano gráfico infinito; la tortuga puede estar en cualquier parte del plano.
	fence			La tortuga se encuentra en modo CERCA: si se pide que la tortuga se mueva fuera de los límites de la pantalla, se moverá hasta el límite

				de la pantalla, donde se detendrá, al tiempo que aparece un mensaje de error "fuera de límites".
	perspective			La tortuga se encuentra en modo PERSPECTIVA, con una tercera dimensión (eje Z).
	fill			Rellena de color una región de la ventana de gráficos, en la que se encuentra la tortuga y que está limitada por líneas previamente dibujadas.
	label		texto	Escribe en la pantalla de gráficos la entrada, que debe ser una palabra ó una lista; si es una lista, cualquier sub-lista estará delimitada por corchetes, pero la totalidad del objeto no estará entre corchetes.
	setpixel		colorvector	Pone el píxel que está debajo de la tortuga con el color **colorvector**.
	setlabelfont		font	La entrada es una lista con la estructura de la fuente.
	labelfont			Devuelve una lista que describe la fuente actual.
	labelsize		texto	Esta primitiva devuelve el tamaño de un texto.
	textscreen	ts		(No soportado por MSWLogo aún). Devuelve el tamaño y posición de la ventana al máximo del espacio disponible en la ventana de texto (la ventana es usada para la interacción con Logo).
	fullscreen	fs		(No soportado por MSWLogo aún). Devuelve el tamaño y posición de la ventana al máximo del espacio disponible en la ventana de gráficos (la ventana es usada para la interacción con Logo).
	splitscreen	ss		(No soportado por MSWLogo aún). Devuelve el tamaño y la posición de la ventana para dejar un espacio para interacción de texto mientras mantiene mas visible las ventanas de gráficos.
	setscrunch		xscale yscale	Ajusta el aspecto de la proporción y la escala de las imágenes gráficas. Después de usar esta primitiva, todos los movimientos de la tortuga se ajustarán multiplicando

				la extensión horizontal y vertical del movimiento por los dos números indicados en la entrada.
	refresh			(No soportado ó necesitado en MSWLogo aún). Le dice a Logo que recuerde los movimientos de la tortuga y que estos puedan ser reconstruidas en caso de que la ventana de gráficos sea cubierta.
	norefresh			(No soportado ó necesitado en MSWLogo aún). Le dice a Logo que no recuerde los movimientos de las tortugas.
	zoom		escala	Este primitiva deja que Logo controle la escala de la ventana "Pantalla". El argumento es el valor de la ampliación (**escala**).
	scrollx		deltax	Esta primitiva deja que Logo controle el desplazamiento horizontal de la ventana "Pantalla".
	scrolly		deltay	Esta primitiva deja que Logo controle el desplazamiento vertical de la ventana "Pantalla".
	setfocus		caption	Esta primitiva deja que Logo controle que ventana tiene el foco. La ventana deseada se especifica por su título.
	getfocus			Esta primitiva devuelve el título de la ventana que tiene el foco.
	icon		caption	Este procedimiento deja que Logo minimice la ventana especificada, en forma de un icono.
	unicon		caption	Esta primitiva deja que Logo restaure la ventana especificada.
	windowset		caption mode	Esta primitiva deja que Logo controle el modo de la ventana especificada (por ejemplo: oculta, visible etc.).
Preguntas sobre la Tortuga y la Ventana				
	shownp			Devuelve verdadero si la tortuga está en pantalla (visible), falso, si está oculta.
Control del Lápiz				
8	pendown	pd		Pone la posición de la pluma hacia abajo, sin cambiar el modo.
9	penup	pu		Pone la posición de la pluma hacia

				arriba, sin cambiar el modo.
8	penpaint	ppt		Pone la posición de la pluma hacia abajo (*pendown*), en el modo penpaint que es el modo habitual en que dibuja la tortuga.
	penerase	pe		Pone la posición de la pluma hacia abajo (*pendown*), en el modo *penerase*.
	penreverse	px		Pone la posición de la pluma hacia abajo (*pendown*), en el modo plumainversa.
	pennormal			Pone el estado de la pluma como si se usaran las primitivas *pendown* y *penup*.
12	setpencolor	setpc	[rojo verde azul]	Especifican la intensidad de los colores Rojo, Verde Azul de la pluma. Cada entrada representa qué cantidad de Rojo, Verde y Azul requiere el color. Cada entrada tiene un rango de 0-255. Al mezclar cantidades diferentes de colores puede crear 16,7 millones de colores diferentes.
	setfloodcolor	setfc	[rojo verde azul]	Especifican la intensidad de los colores Rojo, Verde Azul de relleno.
13	setscreencolor	setsc	[rojo verde azul]	Especifican la intensidad de los colores Rojo, Verde Azul del fondo.
14	setpensize		[ancho alto]	Pone las características de la pluma dependientes del "hardware". Es posible que esta primitiva no sea compatible con diferentes maquinas. El "tamaño" es una lista de dos miembros, anchura y altura. MSWLogo sólo usa la anchura. Por ello los dos miembros tienen el mismo valor.
	setpenpattern		pattern	(no soportado por MSWLogo aún). Toma las características de la pluma dependiendo del hardware.
	setpen		(Proced. de la librería)	Pone la posición de la pluma, el modo, y las características dependientes del hardware de acuerdo con la información de la entrada que debe ser una lista.
	clearpalette			Esta primitiva borra la paleta de colores.
Preguntas sobre el Lápiz				
	pendownp			Devuelve verdadero si la pluma está

				bajado y falso, si no lo está.
	penmode			Devuelve una de las palabras PAINT, ERASE, ó REVERSE, según el modo actual de la pluma.
4	pencolor			Devuelve información sobre el color de la pluma en una lista con las intensidades de los colores [Rojo Verde Azul] en un rango 0-255.
	floodcolor			Devuelve información sobre el color de relleno en una lista que contiene las intensidades de los colores [Rojo Verde Azul] en un rango 0-255.
5	screencolor			Devuelve información sobre el color de la pantalla en una lista que contiene las intensidades de los colores [Rojo Verde Azul] en un rango 0-255.
19	pensize			Devuelve información sobre el grosor del lápiz en una lista que contiene dos enteros [Ancho Alto]. Ancho no se usa en MSWLogo.
	penpattern			(no soportado por MSWLogo aún). Devuelve la información del lápiz según el hardware especifico.
	pen		lista	Devuelve una lista con información sobre el lápiz: posición, modo, y características específicas del hardware.
			ESTRUCTURAS DE CONTROL	
Ordenes de Control				
	run		listainstrucción	Ejecuta las instrucciones Logo que contiene la entrada.
	runresult		listainstrucción	Ejecuta las instrucciones de la entrada. Devuelve una lista vacía si las instrucciones no tienen salida, ó una lista con un solo miembro si hay una salida.
	repeat		nun listainstrucción	Ejecuta una lista de instrucciones tantas veces como indique la primera entrada (**num**).
	repcount		contador	Sólo puede usarse dentro de un *repeat*. Devuelve el número de repeticiones finalizadas, incluyendo la actual.

	if		objeto IF prueba listainstrucción	1º. Si la condición (**prueba**) tiene el valor verdadero , entonces *if* ejecuta una lista de instrucciones. Si prueba tiene como valor falso, entonces *if* no ejecuta nada. Se produce un error si la condición no tiene un valor verdadero o falso.
			objeto (IF prueba listainstrucción1 listainstrucción2)	2º. Si se cumple la condición se ejecuta la primera lista; si no se cumple, se ejecuta la segunda.
	ifelse		objeto IFELSE prueba listainstrucción1 listainstrucción2	Si la condición prueba tiene el valor verdadero, entonces *ifelse* ejecuta **listaainstrucción1**. Si la condición prueba tiene como valor falso, entonces *ifelse* ejecuta **listainstrucción2**. *ifelse* puede devolver un valor.
	test		prueba	Recuerda su entrada, que debe tener el valor verdadero o falso, que se usará después con las primitivas *iftrue* o *iffalse*. El efecto de *test* es local dentro del procedimiento en que se usa.
	iftrue	ift	listainstrucción	Ejecuta su entrada, si la última instrucción *test* tiene el valor verdadero.
	iffalse	iff	listainstrucción	Ejecuta su entrada, si la última instrucción *test* tiene el valor falso.
	true			Es una palabra especial que indica el resultado positivo de una condición.
	false			Es una palabra especial que indica el resultado negativo de una condición.
	stop			Termina la ejecución del procedimiento en que aparece. Devuelve el control al contexto en el que se llamó el procedimiento.
	output		valor	Termina la ejecución del procedimiento en que aparece y devuelve un valor.
	catch		tag listainstrucción	Ejecuta una lista de instrucciones y devuelve una salida, si la tiene. Si, mientras se ejecutan las instrucciones, se ejecuta un *throw* con una entrada igual a **tag**, entonces termina la ejecución. **Tag** debe ser una palabra.

	throw		etiqueta etiqueta valor	Debe usarse con una etiqueta con el mismo alcance que *catch*. Termina la ejecución de la lista de instrucciones de *catch*. Si *throw* se usa con solo una entrada, el correspondiente *catch* no devuelve ningún valor. Si se usa con dos entradas, la segunda proporciona la salida de *catch*.
	error		errorlista	Devuelve una lista que describe el error. Si no hay error después del último uso de *error*, devuelve una lista vacía.
	pause		valor	Introduce una pausa interactiva. El usuario recibe instrucciones en las que aparece el nombre del procedimiento en que se invocó *pause*. Las variables locales de este procedimiento están disponibles durante la pausa. Cuando termina *pause* devuelve *continue* con una entrada.
	continue	co	valor	Termina la pausa interactiva, volviendo al contexto desde el que se llamó a *pause*. Si *continue* se produce en una entrada, *pause* devuelve ese valor. Excepcionalmente, *continue* se puede usar sin su entrada y sin paréntesis con tal que esté en una línea de instrucciones.
	yield			Permite que otros programas usen la computadora mientras Logo trabaja.
	noyield			No permite que otros programas usen la computadora mientras Logo trabaja.
	eventcheck			Es similar a llamar a *yield* seguido de *noyield*. Chequea si hay eventos en espera de ser procesados. Las órdenes que producen "interrupciones" se ejecutan en modo *noyield*.
	setcursorwait			Pondrá el cursor en la forma de reloj de arena para indicar dos cosas. Una es que la operación tardará algún tiempo; la otra es que durante ese tiempo el usuario no

				tiene el control de Windows (*noyield*). Esto significa que debe usar *noyield* antes de cada *setcursorwait*.
	halt			Equivale a pulsar botón *stop* pero bajo el control de Logo.
	wait		tiempo	Retrasa la ejecución durante un tiempo 1/60 de segundo.
	settimer		id delay callback	Instala un cronómetro identificado por **id** (1-31) que produce una interrupción durante milisegundos hasta que se llame a *cleartimer*.
	cleartimer		id	Borra el tiempo puesto por *settimer* e identificado por **id** (1-31).
	bye			Sale de Logo y vuelve al sistema operativo.
	.maybeoutput		valor	Trabaja como la primitiva *output* excepto que la expresión que proporciona el valor de entrada podría no devolver un valor, en este caso el efecto sería el mismo de usar la primitiva *stop*. Esto está pensado para usar en la definición de estructuras de control, para los casos en los cuales no se sabe cual expresión produce un valor.
	ignore		valor	Ignora un efecto que no se considera importante.
	`		newlista ` lista	Devuelve una lista igual a su entrada pero con ciertas substituciones. Si un miembro de la lista es la palabra "," (coma) entonces el siguiente miembro debe ser una lista que produce una salida cuando se ejecuta. Esa salida reemplaza a la coma y a la lista.
	for		forcontrol listainstrucción	Tiene como primera entrada una lista con tres o cuatro miembros: (1) una palabra que es el nombre de una variable local; (2) una palabra o lista que contiene el valor inicial de la variable; (3) una palabra o lista que indica el valor límite de la variable; (4) y una palabra o lista que determina el incremento opcional.
	do.while		listainstrucción	Ejecuta una lista de instrucciones

			pruebalista	mientras se cumpla la condición.
	while		pruebalista listainstrucción	Ejecuta una lista de instrucciones mientras se cumpla la condición.
	do.until		listainstrucción pruebalista	Ejecuta una lista de instrucciones hasta que una condición tenga el valor falso.
	until		pruebalista listainstrucción	Ejecuta una lista de instrucciones hasta que una condición tenga el valor falso.
	goto		etiqueta	Esta primitiva puede ser usada solo dentro de un procedimiento. La entrada debe ser una palabra. Un *tag* de el mismo nombre debe aparecer dentro del mismo procedimiento. Durante el procesamiento continuará, siguiendo el rastro del *tag*. Es mejor evitar el uso de la primitiva *goto*, pero en algunos casos semejantes a "estado del computador" ésta puede ser muy poderosa.
	tag		etiqueta	Esta primitiva puede ser usada solo dentro de un procedimiento. Esta describe a donde se debe realizar un salto solicitado con la primitiva *goto*. La entrada debe ser una palabra.
template-based iteration				
	apply		template inputlist	Comando u operación que corre la plantilla, rellenando las ranuras con los miembros de la lista de entrada. El número de miembros de la lista debe ser un número aceptable con el número de ranuras de la plantilla.
	invoke		template input1 input2...	Comando u operación que es como un *apply* excepto que las entradas son dadas como expresiones separadas como en una lista.
	foreach		datalist1 datalist2... templatelist	Comando que evalúa la lista de plantillas repetidamente, pos cada elemento de la lista de datos.
	map		templatelist data1 data2...	Devuelve una palabra o lista, dependiendo del tipo de datos de entrada. Cada elemento que devuelve es el resultado de la evaluación de la lista de plantillas, llenando las ranuras con el correspondiente elemento de los

				datos de entrada.
	map.se		templatelist data1 data2...	Devuelve una lista formada por la evaluación de la lista de plantillas repetidamente y concatenando los resultados usando *sentence*. Esto es, los miembros de la salida son los miembros de los resultados de las evaluaciones.
	filter		tftemplatelist data	Devuelve una palabra o lista, dependiendo del tipo de la data de entrada, conteniendo a subset de los miembros (para una lista) ó caracteres (para una palabra) de la entrada. A plantilla es evaluada por cada miembro o carácter de la data, y este debe producir un valor verdadero o falso. Si el valor es verdadero, entonces la correspondiente entrada constituida es incluida en la salida.
	find		tftemplatelist data	Devuelve la primera constituida de la data de entrada (el primer miembro de una lista ó el primer carácter de una palabra) para el cual el valor producido por la evaluación de la plantilla con la constituida en las ranuras es verdadero.
	reduce		template data	Devuelve el resultado de aplicar la plantilla el acumulado de los elementos de la data de entrada. La plantilla debe tener dos funciones de ranura.
	crossmap		template data1 data2...	Devuelve una lista conteniendo los resultados de la plantilla de evaluación. Cada lista de datos corresponde a una ranura en la plantilla; el número de ranuras es igual al número de data de la lista de entrada.
	cascade		endtest template1 startvalue1 template2 startvalue2...	Devuelve el resultado de la aplicación de una plantilla (ó varias plantillas, ver Template-based iteration) repetidamente, tomando un valor manteniendo la ranura del primer tiempo, y el resultado de cada aplicación manteniendo la ranura para la siguiente aplicación.

	cascade.2		endtest template1 startvalue1 template2 startvalue2	Devuelve el resultado de la invocación de *cascade* con las mismas entradas. La única diferencia es que el número por defecto de las entradas es cinco en lugar de tres.
	transfer		endtest template inbasket	Devuelve el resultado de las repetidas evaluaciones de las plantillas de entrada. La plantilla es evaluada por cada miembro de la lista encestada.
MULTIMEDIA / MIDI / SONIDO				
Primitivas MIDI				
	midiopen		device midiopen device(midiopen id)	Abre el dispositivo MIDI mediante un controlador. **device**:(lista) El nombre del controlador, si la operación tuvo éxito. **id**:(entero) Es un índice que especifica qué controlador del dispositivo MIDI quiere abrir.
	midiclose			Cierra el dispositivo MIDI.
	midimessage		mensaje	La entrada debe ser una lista. Requiere que previamente haya llamado a *abremidi*.
	miditable			
	midicontrol			
	midiglossary			Canal: Un canal es un número, entre 0-15, que corresponde a los canales 1-16. Pitch: Un pitch es un número, entre 0-127, que corresponde a una nota en el instrumento. Velocidad: Es un número, entre 0-127, que indica la rapidez con que se pulsan y liberan las teclas o cuerdas. Presión: Es un número entre 0-127 y corresponde con las características de pulsación de las teclas. Programa: Es un número, entre 0-127, que se corresponde con el instrumento.
	midiinstrument			Piano, Percusión cromática, Órgano, Guitarra, Bajo, Cuerda, Conjunto, Cobre, Reed, Pipe, Synth Lead, Synth Pad, Efectos sintéticos,

				Étnicos, Percusión, Efectos de sonidos.
Primitiva SonidoWave				
	playwave		wavfile banderas	**wavfile** debe ser un fichero .WAV. banderas describe cómo quiere oír el sonido. Estas posibilidades también existen con *mci* pero con un control más fino.
Primitivas Sonido				
	soundon		frecuencia	La entrada es la frecuencia del sonido que se quiere oír. El sonido permanece hasta que se llame a *soundoff*.
	soundoff			Apaga el sonido emitido con *soundon*.
	sound		tonostream	La entrada debe ser una lista de pares. Cada par especifica la frecuencia (en hertzios) y la duración.
Primitivas Multimedia				
	mci		estado MCI mci-orden-lista estado (MCI mci-orden-lista callback)	La entrada mci-orden-lista debe ser una lista. La primitiva *mci* puede o no devolver una lista, dependiendo de la forma *mci* que emplee. En la segunda forma si mci-orden-lista "notifica" la opción entonces se ejecutará un programa cuando se complete la orden.
FUNCIONES DE BITMAP				
	bitcut		anchura altura	"Corta" una porción de la imagen y la guarda en la memoria de Logo. Logo cortará desde la posición de la tortuga el tamaño indicado por los argumentos anchura y altura.
	bitcopy		anchura altura	"Copia" una porción de la imagen y la guarda en la memoria Logo. Logo copiará desde la posición de la tortuga el tamaño indicado por los argumentos anchura y altura.
	bitpaste			"Pega" la imagen que se "cortó" "con *bitcut*. Logo siempre "pega" en la posición de la tortuga, la esquina inferior izquierda de la imagen.
	bitpastetoindex		indice x y	"Pega" a partir de las coordenadas x, y del bitmap identificado con el argumento índice, la imagen que se "cortó" con *bitcut* dentro del índice

				actual. Esta primitiva se utiliza en las animaciones.
	bitfit		anchura altura	"Ajusta" la actual imagen "cortada" (*bitcut*) ó "copiada" (*bitcopy*) a las dimensiones especificadas. Más tarde puede pegar (*bitpaste*) la imagen.
	setbitindex		indice	Pone el área cortada de un bitmap en el buffer que indica índice (rango entre 0 y 1023). Su propósito es guardar en memoria múltiples imágenes preparadas para ser pegadas con rapidez en animaciones.
	bitindex		indice	Devuelve el índice actual del bitmap fijado con *setbitindex*.
	setbitmode		modo	Pone el modo de un bitmap según su argumento. El modo tiene un rango entre 1 y 9. Su propósito es pegar imágenes usando diferentes métodos. modo 1: Copia de la memoria a la pantalla. modo 2: Operación "*or*" entre la copia en memoria y el fondo de la pantalla. modo 3: Operación "*and*" entre la copia en memoria y el fondo de la pantalla. modo 4: Operación "*xor*" entre la copia en memoria y el fondo de la pantalla. modo 5: Operación "*and*" entre la copia en memoria y el "*invert*" del fondo de la pantalla. modo 6: Copia el "*invert*" de la imagen en memoria en el fondo de la pantalla. modo 7: Operación "*or*" entre la copia en memoria y el "*invert*" del fondo de la pantalla. modo 8: Operación "*or*" entre el "*invert*" de la copia en memoria y el fondo de la pantalla. modo 9: Copia el "*invert*" del fondo de la pantalla.
	bitmode		modo	Devuelve el modo del bitmap

				puesto con *setbitmode*.
	bitblock		anchura altura	Dibuja un rectángulo relleno de las dimensiones indicadas en las entradas. El color se selecciona con *setfloodcolor*.
	bitload		nombrebitmap	Carga el bitmap que se indica en la entrada que debe ser una palabra. Esta primitiva es similar a la opción *Load* del Menú.
	bitsave		nombrebitmap	Guarda como un bitmap su trabajo en el disco de la computadora. El formato del fichero guardado es .BMP
	bitsize		tamaño nombrebitmap	Devuelve el tamaño del bitmap que se especifica en la entrada que debe ser una palabra.
	bitloadsize			
	gifload		nombrebitmap	Carga un fichero que contiene una imagen GIF. La entrada que debe ser una palabra especifica el nombre del fichero.
	gifsave		nombrebitmap delay append loop maxcolor transcolorvector	Guarda su trabajo como una imagen GIF. En la primera forma, la entrada es una palabra que especifica el nombre del fichero. La segunda forma tiene múltiples opciones: Puede guardar como GIF una animación.
	gifsize		tamaño nombrebitmap	Devuelve el tamaño de la imagen GIF.
	setactivearea		área	Es similar a *activearea* del Menú. Pone el área activa para imprimir y guardar Bitmaps.
	activearea		área	Devuelve el área activa que se puso con *setactivearea*.
			PRIMITIVAS DE DIRECTORIOS	
	dir			Escribe una lista de ficheros y directorios comprendidos en el directorio actual.
	files		listaficheros	Devuelve la lista de ficheros del directorio actual.
	directories		listadirectorios	Devuelve la lista de directorios comprendidos en el directorio actual.
	chdir	cd	directorio	Es exactamente igual que la orden *chdir* (CD) del DOS. El argumento debe ser una palabra.

	popdir			Le lleva al nivel superior de un directorio.
	mkdir		directorio	Crea un directorio y le lleva a él. El parámetro debe ser una palabra (el nombre del directorio que se quiere crear).
	rmdir		directorio	Este comando elimina un directorio. El nombre del directorio es una palabra que debe existir.
			ORDENES DE AYUDA	
	help		keypalabra	Tiene dos formas (con y sin argumento por teclado). Sin argumento, Logo entra en el nivel superior del fichero de ayuda de Logo. Con argumento, Logo buscará en el fichero de ayuda el tema o temas indicados por el argumento.
	winhelp		nombrefichero keypalabra	Tiene dos formas (con y sin segundo argumento por teclado). Sin un segundo argumento, Logo entra en el nivel superior del fichero de ayuda especificado. Con argumento, Logo buscará en el fichero de ayuda seleccionado el tema o temas indicados por el argumento.
			MANEJO DE LA MEMORIA	
Definición de Procedimientos				
	to		nombreproc :entrada1 :entrada2	Esta primitiva le dice a Logo que acepte una definición de procedimiento. Un procedimiento puede tener varias entradas o ninguna.
	end			Permite que termine un procedimiento.
	define		nombreproc texto	Define un procedimiento. Si existe un procedimiento con el mismo nombre, la nueva definición sustituye a la antigua. Posee una primera sublista que contiene las entradas y se parece a la línea *to*. Las demás sublistas de la entrada texto forman el cuerpo del procedimiento, con una sublista por cada línea de instrucciones. (No hay una línea *end* en la entrada texto.) No se puede redefinir un

				procedimiento, a no ser que la variable *redef?* tenga el valor verdadero.
	text		texto nombreproc	Devuelve el texto de un procedimiento en la forma que se requiere para un *define*: una lista con sublistas, la primera describe las entradas del procedimiento y las demás las líneas del cuerpo.
	fulltext		texto nombreproc	Devuelve una representación del procedimiento **nombreprocedimiento,** en el que el formato de información es preservado. Si el procedimiento fue definido con *for*, *edit* ó *load*, entonces la salida será una lista de palabras. Cada palabra representa una línea entera de definición, en el formato de salida de *readword*, incluyendo los espacios extras y las líneas de continuación. El último elemento de la salida representa la línea *end*. Si el procedimiento fue definido con *define*, entonces la salida será una lista de listas. Si estas listas son impresas, una por línea, el resultado lucirá como una definición usando *for*.
	copydef		nuevonom viejonom	Crea un procedimiento de nombre **nuevonom**, idéntico a otro llamado **viejonom**. El más reciente puede ser una primitiva. Si **nuevonom** es una primitiva, no se permite su redefinición a no ser que la variable *redef?* tenga el valor verdadero. Las definiciones creadas con *copydef* no se guardan con *save*; las primitivas no se guardan nunca, y los procedimientos creados con *copydef* se ocultan.
Definición de Variables				
	make		nombrevar valor	Asigna el valor indicado en valor a la variable **nombrevar**, que debe ser una palabra sin que importe si está en mayúsculas o minúsculas. Si la variable ya existe, cambia su valor. Si no, se crea una nueva variable global.

	name		valor nombrevar	Asigna el valor indicado en el argumento valor a la variable **nombrevar**, que debe ser una palabra, sin que importe si está en mayúsculas o minúsculas. Si la variable ya existe, cambia su valor. Si no, se crea una nueva variable global.
	local		nombrevar nombrevarlista	Acepta como entradas una o más palabras, o una lista de palabras. Se crea una variable por cada una de las entradas, con el nombre de esa palabra. Las variables son locales en el procedimiento en que se encuentran. Las variables creadas con *local* no tienen valor inicial; se les debe asignar un valor (por ejemplo con *make*) antes de que el procedimiento intente leer su valor.
	localmake		nombrevar valor	Es la abreviatura de: *local* **nombrevar** *make* **nombrevar** valor
	think	:	valor nombrevar	Devuelve valor de una variable cuyo nombre es la entrada. Tiene como abreviatura (:)
Propiedades de las Listas				
	pprop		lpropnombre propnombre valor	Añade una propiedad de nombre **propnombre** y con el valor **valor** a la lista **lpropnombre**.
	gprop		lpropnombre propnombre valor	Devuelve el valor de una propiedad (**propnombre**) de una lista de propiedades (**lpropnombre**), o una lista vacía si no existe tal propiedad.
	remprop		lpropnombre propnombre	Elimina la propiedad **propnombre** de la lista de propiedades **lpropnombre**.
	plist		lpropnombre	Devuelve una lista en la que los elementos impares son los nombres y los elementos pares los valores de las propiedades de la lista indicada en el argumento. Devuelve una copia de la actual lista de propiedades.
Predicados **Espacio de trabajo**				
	procedurep		verdad nombre	Devuelve verdadero si su entrada o argumento es el nombre de un

				procedimiento.
	primitivep		verdad nombre	Devuelve verdadero si la entrada es el nombre de una primitiva.
	definedp		verdad nombre	Devuelve verdadero si la entrada es el nombre de un procedimiento definido por el usuario, incluido en una librería de procedimiento.
	namep		verdad nombre	Devuelve verdadero si la entrada es el nombre de una variable.
	macrop		verdad nombre	Devuelve verdadero si la entrada es el nombre de una macro definida por el usuario, incluso de una macro de librería.
Preguntas				
	contents		contenidolista	Devuelve una lista, formada por tres listas que contiene los nombres de los procedimientos definidos, las variables y propiedades.
	buried		contenidolista	Devuelve todos los elementos que se encuentran el espacio de trabajo.
	procedures		proclista	Devuelve una lista con los nombres de todos los procedimientos definidos por el usuario y no tapados del espacio de trabajo.
	names		contenidolista	Devuelve una lista vacía (lo que indica que no son nombres de procedimiento), seguida de una lista con los nombres de las variable.
	plists		contenidolista	Devuelve una lista formada por dos listas vacías (para indicar que no son listas de procedimientos o variables) seguidas por una lista con todas las listas de propiedades no tapadas que se encuentran en memoria.
	namelist		contenidolista varnombrelista	Devuelve una lista formada por una lista vacía seguida de otra lista con el nombre o nombres de los argumentos.
	pllist		contenidolista plnombrelista	Devuelve una lista, formada por dos listas vacías a las que sigue una lista de nombres o los nombres de las entradas.
Inspección				
	po		contenidolista	Escribe las definiciones de todos los procedimientos, variables, y listas de propiedades, según la entrada **contenidolista**.

	pocontents	poall		Escribe todas las definiciones no tapadas del espacio de trabajo.
	poprocedures	pops		Escribe todas las definiciones no tapadas del espacio de trabajo.
	ponames	pons		Escribe todas las variables no tapadas del espacio de trabajo.
	poplist	popls		Escribe todas las propiedades de las listas del espacio de trabajo.
	ponamelist	pon	varnombrelista	Escribe los nombres de las variables y su valor.
	popllist	popl	pllistanombres	Escribe la definición de las propiedades de una lista.
	pot		contenidolista	Esta primitiva imprime la línea de titulo de los procedimientos y las definiciones de las variables y las propiedades de las listas. Para las listas de propiedades, las listas enteras se muestran en una línea, en lugar de, en una serie de instrucciones *pprop* como en *po*.
	potprocedures	pots		Imprime las líneas de titulo de todo lo destapado en el espacio de procedimientos.
Control de Memoria				
	erase	er	contenidolista	Borra de la memoria los procedimientos, variables y listas de propiedades de la entrada. Los procedimientos no se borrarán a no ser que la variable *redef?* tenga el valor verdadero.
	erall			Borra todos los procedimientos, variables, y listas de propiedades de memoria.
	erps			Borra todos los procedimientos no tapados de la memoria.
	erns			Borra todas las variables no tapadas de la memoria.
	erpls			Borra todas las listas no tapadas de la memoria.
	ern		nombrevariable	Borra de la memoria la variable(s).
	erpl		plnombre	Borra de la memoria la lista(s) de propiedades.
	bury		contenidolista	Oculta los procedimientos, variables y listas de propiedades contenidos que aparecen en su entrada.
	buryall			Es la abreviatura de *bury contents*.
	buryname		nombrevariable	Es la abreviatura de *bury namelist* **nombrevariable** (lista).

	unbury		contenidolista	Destapa los procedimientos, variables y listas de propiedades que aparecen en la entrada **contenidolista**.
	unburyall			Es la abreviatura de *unburyburied*.
	unburyname		nombrevariable	Es la abreviatura de *unbury namelist* **nombrevariable** (lista).
	trace		contenidolista	Marca un elemento para ser ejecutado en modo trazar. En este modo y cuando se ejecuta un procedimiento, Logo envía un mensaje con los valores actuales de las entradas, siempre que el procedimiento se ejecute en modo trazado y con las primitivas *stop* ó *output*.
	untrace		contenidolista	Desactiva el modo *trace*.
	traced			Devuelve una lista de contenidos consistente de una lista de procedimientos rastreados seguidos por dos listas vacías.
	step		contenidolista	Marca un elemento para su ejecución en modo *step*. Siempre que se invoca un procedimiento en este modo, Logo escribe cada línea de instrucción del cuerpo del procedimiento antes de ejecutarla y espera una pulsación del usuario para continuar con una nueva línea.
	unstep		contenidolista	Suprime la ejecución en modo *step*.
	stepped			Devuelve una lista de contenidos consistente de una lista de procedimientos *stepped* seguidos por dos listas vacías.
	edit	ed	contenidolista	Edita las definiciones de elemento(s), usando el Editor de Logo.
	edall			Es la abreviatura de *editcontents*.
	edps			Es la abreviatura de *editprocedures*
	edns			Es la abreviatura de *editnames*
	edpls			Es la abreviatura de *editplist*
	edn		nombrevariable	Es la abreviatura de *editnamelist*
	edpl		plnombre	Es la abreviatura de *editpllist*
	save		nombrefichero	Guarda las definiciones de todos los procedimientos, variables y listas de propiedades, si no están tapados.
	savel		contenidolista	Guarda las definiciones de los

			nombrefichero	procedimientos, variables y listas de propiedades que se especifican en **contenidolista** en el fichero.
	load		nombrefichero	Carga un fichero. Si Logo ve un *to* definirá ese procedimiento. Si Logo ve un make creará la variable. Si Logo ve código fuera de la definición de un procedimiento, lo ejecutará en el orden en que aparezca.
	nostatus			Esto es, mata el estado popup de la ventana.
	status			Esto es, hace *pops* sobre el estado de la ventana.
			FUNCIONES DE WINDOWS	
Ventana				
	windowcreate		madre, nombre, título, xpos, ypos, anchura, altura, setup	Crea una ventana. A una ventana se le puede añadir otros objetos ó controles de Windows, como botones, barras de desplazamiento, cajas de lista, etc.
	windowdelete		nombre	Borra (cierra) la ventana, identificada por nombre. Cierra también las ventanas y controles dependientes de esa ventana.
	windowenable		nombre state	Este comando habilita ó deshabilita la ventana tomando el nombre y el status.
Diálogo				
	dialogcreate		madre, nombre, título, xpos, ypos, anchura, altura, setup	Crea una ventana de diálogo. Se le puede añadir otros objetos ó controles, como botones, barras de desplazamiento, cajas de lista, etc.
	dialogdelete		nombre	Borra (cierra) la ventana de diálogo de nombre **nombre**. Observe que las ventanas hijas y controles se cerrarán también.
	dialogenable		nombre state	Este comando habilita ó deshabilita la ventana de dialogo.
Listbox				
	listboxcreate		madre, nombre, xpos, ypos, anchura, altura	Crea un control listbox.
	listboxdelete		nombre	Borra (cierra) el control listbox identificado por nombre.
	listboxenable		nombre	Este comando habilita o deshabilita

			state	el control listbox.
	listboxgetselect		texto nombre	Devuelve una copia del elemento seleccionado en el control listbox.
	listboxaddstring		nombre elemento	Añade un elemento al control listbox identificado por nombre.
	listboxdeletestring		nombre índice	Borra el elemento número **índice** de un control listbox.
Combobox				
	comboboxcreate		madre nombre xpos ypos anchura altura	Crea un control combobox. El control combobox es la unión de dos controles en uno sólo (el control listbox y el control edit). Si quiere crear un control edit (combobox sin listbox) sólo tiene que fijar la altura a un tamaño en el que no encaje el tamaño del control listbox.
	comboboxdelete		nombre	Borra (cierra) el control combobox identificado por nombre.
	comboboxenable		nombre state	Este comando habilita ó deshabilita el control combobox.
	comboboxgettext		texto nombre	Lee el contenido del control edit que forma parte del control combobox (puede ser o no ser un elemento seleccionado).
	comboboxsettext		nombre texto	Pone texto en el control edit que es un componente de combobox.
	comboboxaddstri ng		nombre elemento	Añadirá un "elemento" en el control combobox identificado con "nombre". Observe que los elementos que se insertan se ordenan automáticamente en el control combobox.
	comboboxdeletes tring		nombre indice	Borra el elemento número "índice" del combobox identificado como "nombre".
Barra Desplazamiento				
	scrollbarcreate		madre nombre xpos ypos anchura altura callback	Crea un control scrollbar. El control scrollbar se usa para pedir al usuario el valor de una variable (aunque puede diseñar una función con otro motivo). Es frecuente su utilización junto con el control static para informar al usuario de una situación.
	scrollbardelete		nombre	Borra (cierra) el control scrollbar identificado por nombre.
	scrollbarenable		nombre	Este comando habilita ó deshabilita

			state	el control scrollbar.
	scrollbarset		nombre, mínimo máximo, posición	Fija el rango (entre mínimo y máximo) y la posición actual del control scrollbar.
	scrollbarget		pos nombre	Devuelve la posición del control scrollbar identificado por nombre.
Botón				
	buttoncreate		madre, nombre etiqueta, xpos, ypos, anchura, altura, callback	Crea un control button. El control button se usa para que el usuario provoque eventos.
	buttondelete		nombre	Borra (cierra) el control button identificado por nombre
	buttonenable		nombre state	Este control habilita ó deshabilita el control button.
	buttonupdate		nombre texto	Cambia el contenido del control button por un nuevo "texto".
Estático				
	staticcreate		madre, nombre, texto, xpos, ypos, anchura, altura	Crea un control static. El control static se usa sólo para mostrar un texto. Su nombre puede inducir a error, ya que el control static puede ser dinámico si se utiliza la primitiva staticupdate.
	staticdelete		nombre	Borra (cierra) el control static identificado por nombre.
	staticupdate		nombre texto	Cambiará el contenido del control static por **texto**.
Groupbox				
	groupboxcreate		madre nombre xpos ypos anchura altur	Crea un control groupbox. Un control groupbox es diferente de los otros controles de Windows porque agrupa de forma gráfica y lógica los grupos radiobuttom (radiobuttoncreate) y checkbox (checkboxcreate). Radiobuttom y checkBox deben pertenecer a un groupbox. Para complicar las cosas, los radiobuttom y checkBox colocados en un groupbox tienen una "madre" distinta a la de groupbox.
	groupboxdelete		nombre	Borra (cierra) el control groupbox identificado por nombre.
Checkbox				
	checkboxcreate		madre, grupo, nombre etiqueta, xpos, ypos anchura, altura	Crea un control checkbox.

	checkboxdelete		nombre	Borra (cierra) el control checkbox identificado por nombre.
	checkboxenable		nombre state	Este comando habilita ó deshabilita el control checkbox.
	checkboxget		verdad nombre	Pregunta al control checkbox por su estado (Verdadero ó Falso).
	checkboxset		nombre estado	Fija el valor del estado del checkbox como Verdadero ó Falso.
Radiobuttom				
	radiobuttoncreate		madre, grupo, nombre etiqueta, xpos, ypos anchura, altura	Crea un control radiobutton.
	radiobuttondelete		nombre	Borra (cierra) el control radiobutton identificado por nombre.
	radiobuttonenable		nombre state	Este comando habilita ó deshabilita el control radiobutton.
	radiobuttonget		verdad nombre	Pregunta a radiobutton por su estado (verdadero ó falso).
	radiobuttonset		nombre estado	Fija el estado de radiobutton como verdadero ó falso.
Depuración				
	debugwindows		nombre	Escribe el árbol (jerarquía de ventanas) comenzando con la ventana especificada con nombre y, por otra parte, listará todas las ventanas y controles.
Ventanas Predefinidas				
	messagebox		bandera cuerpo	Detiene el proceso y despliega una ventana con un mensaje (cuerpo). En la barra de título aparece una etiqueta (bandera). El proceso anterior continua cuando el usuario pulsa el botón Aceptar ó Cancelar.
	yesnobox		siono bandera cuerpo	Detiene el proceso y despliega una ventana con un mensaje (cuerpo). En la barra de título aparece bandera. El proceso anterior continua cuando el usuario pulsa uno de los botones Si, No ó Cancelar.
	selectbox		eleccion bandera listaopciones	Detiene el proceso y despliega una ventana de mensaje en la que en la barra de título aparece bandera y con las opciones **listaopciones**. El proceso anterior continua cuando el usuario pulsa botón Ok ó Cancelar.
	questionbox		datos bandera	Detiene el proceso y despliega una ventana con un mensaje (cuerpo).

			cuerpo	En la barra de título aparece bandera. El proceso anterior continua cuando el usuario pulsa uno de los botones Si ó Cancelar.
	dialogfileopen		ruta nombrefichero	Despliega la ventana de diálogo "Abrir". No abre un archivo sino que deja que el usuario seleccione el fichero deseado usando la Interfaz Gráfica de Usuario (GUI).
	dialogfilesave		ruta nombrefichero	Despliega la ventana de diálogo "**save as**". No guarda un fichero sino que deja que el usuario seleccione el fichero deseado usando la GUI.
	windowfileedit		nombrefichero callback	Edita el fichero especificado en el Editor por defecto. Cuando el usuario sale del Editor se ejecuta el contenido de **callback**.
			ARITMÉTICA	
Operaciones Numéricas				
28	sum	+	número1 número2.....	Devuelve la suma del valor numérico de las entradas.
25	difference	-	número1 número2.....	Devuelve la diferencia del valor numérico de las entradas.
29	minus	-	numero	Devuelve el opuesto del valor numérico de la entrada.
26	product	*	número1 número2	Devuelve el producto de los valores numéricos de las entradas.
27	quotient	/	número1 número2	Devuelve el cociente de la división de los valores numéricos de las entradas.
41	remainder		número1 número2	Devuelve el resto de la división entre número1 y número2.
	modulo		número1 número2	Devuelve el resto de dividir número1 y número2; ambos deben ser enteros y el resultado es un entero con el mismo signo de número2.
39	int		número	Devuelve la entrada, sin la parte decimal, es decir, un entero.
40	round		número	Devuelve el entero más cercano al valor numérico de la entrada.
30	abs		número1 (Proced. de la librería)	Devuelve el valor absoluto de la entrada.
	sign		número (Proced. de la librería)	Devuelve el signo de la entrada (-1 si es negativo, 0 si es cero, 1 si es positivo).
31	sqrt		número1	Devuelve la raíz cuadrada del valor

				numérico de la entrada.
32	power		número1 número2	Devuelve número1 elevado a la potencia número2.
33	exp		número1	Devuelve el número e (2.718281828+) elevado a la potencia número1.
34	log10		número1	Devuelve el logaritmo común de número1.
35	ln		número1	Devuelve el logaritmo natural de número1.
36	pi		(Proced. de la librería)	Devuelve el valor del número pi.
43	tan		ángulo	Devuelve la tangente de ángulo en grados.
	radtan		ángulo	Devuelve la tangente de ángulo, en radianes.
42	sin		ángulo	Devuelve el seno de ángulo, en grados.
	radsin		ángulo	Devuelve el seno de ángulo en radianes.
37	cos		ángulo	Devuelve el coseno de ángulo en grados.
	radcos		ángulo	Devuelve el coseno de ángulo en radianes.
38	arctan		número x, y	Con una entrada, devuelve el arco tangente de la entrada, en grados. Con dos entradas, devuelve el arco tangente en grados de y/x.
	radarctan		número x, y	Con una entrada, devuelve el arco tangente en radianes de la entrada. Con dos entradas, devuelve el arco tangente en radianes de y/x.
	arcsin		número	Devuelve el arco seno en grados de la entrada.
	radarcsin		ángulo	Devuelve el arco seno en radianes de la entrada.
	arccos		número	Devuelve el arco coseno en grados de la entrada.
	radarccos		número	Devuelve el arco coseno en radianes de la entrada.
Predicados (aritmética)				
44	less?	<	número1 número2	Devuelve verdadero si la primera entrada es menor que la segunda.
45	greater?	>	número1 número2	Devuelve verdadero si el primer número es mayor que el segundo
Números aleatorios				
46	random		número	Devuelve un número aleatorio, entero positivo menor que la entrada que debe ser un número

				entero.
47	rerandom		número	Esta primitiva reproduce los resultados de *random*. Generalmente la secuencia de números aleatorios es distinta cada vez que se usa. Si necesita repetidamente la misma secuencia de números pseudo-aleatorios, debe llamar a *rerandom* antes de llamar a *random*.
Escribir con Formato				
	form		palabra número ancho precisión	Devuelve una palabra que representa un número, posiblemente precedido por espacios (y por ello no es un número con el que se pueda efectuar operaciones aritméticas), con tantos caracteres como indique ancho, y con los dígitos después del punto decimal que indique precisión. (Si "precisión" es 0, la salida no tendrá punto decimal.)
Operaciones con Bit				
	bitand		número1 número2.....	Devuelve el resultado de la operación *and* tomando como datos los valores enteros de las entradas.
	bitor		número1 número2.....	Devuelve el resultado de la operación *or* con los valores enteros de las entradas.
	bitxor		número1 número2.....	Devuelve el resultado de la operación *xor* con los valores enteros de las entradas.
	bitnot		número	Devuelve el resultado de la operación *not* tomando como dato el valor entero de la entrada.
	ashift		número1 número2	Devuelve número1 con un desplazamiento aritmético a la izquierda de tantos bits como indique número2. Si número2 es negativo, el desplazamiento es a la derecha.
	lshift		número1 número2	Devuelve número1 con un desplazamiento lógico a la izquierda de tantos bits como indique número2. Si número2 es negativo, el desplazamiento es a la derecha.

COMUNICACIONES			
Transmisores			
print	pr	objeto1 objeto2...	Esta primitiva escribe la entrada ó entradas actuales. Si la entrada es una lista, no escribe los corchetes, sí se escriben corchetes en las sublistas. Las matrices se escriben entre llaves.
type		objeto1 objeto2...	Esta orden escribe la entrada ó entradas igual que *print*, pero no produce un retorno de carro y las entradas no se escribirán separadas por espacios.
show		objeto1 objeto2...	Esta primitiva escribe la entrada ó entradas, al igual que *print*, pero si la entrada es una lista, la escribe entre corchetes.
Receptores			
readlist	rl	lista	Lee una línea desde el lector de cadenas (inicialmente el terminal) y devuelve la línea como una lista.
readword	rw	palabra	Lee una línea desde el lector de cadenas y devuelve esa línea como una palabra. Devuelve una palabra aunque la línea tenga espacios, corchetes, etc.
readchar	rc	carácter	Lee un sólo carácter de una cadena y lo devuelve como una palabra (carácter). Si la cadena se lee desde un fichero, y se lee el final del fichero, *readchar* devuelve una lista vacía. Si la cadena se lee desde un terminal, y se llama a *readchar* no devuelve la palabra carácter hasta que se llame a *readlist* ó *readword*.
readchars	rcs	palabra número	Lee un número de caracteres desde el lector de cadenas y devuelve esos caracteres como una palabra. Si se lee de un archivo y se llega al final, *readchars* devuelve una lista vacía (no una palabra vacía). Las barras a la izquierda y verticales, y los caracteres tipo tilde no tienen un tratamiento especial. (No soportado por MSWLogo cuando la lectura es desde el teclado).
shell		orden	Devuelve el resultado de la orden

				que se ejecuta tras la orden shell. Si la orden está en una lista en la línea de instrucciones, y si se quiere enviar un carácter backslash, se debe usar \\. Devuelve verdadero ó falso, según el éxito de la operación.
Comunicaciones				
	portopen		puerto	Esta orden se usa para acceder a los puertos series y paralelo.
	portclose			Cierra el puerto.
	portflush		queue	Esta orden se usa para limpiar la cola del puerto de lectura o escritura
	portmode		modo	Se usa para fijar el modo (velocidad, paridad, bits de datos, y bits de parada) del puerto.
	portreadarray		contador buffer	Lee el puerto abierto y escribe los datos en el buffer. Intentará leer todos los caracteres que indique contador y que estén disponibles. Devuelve el número de bytes leídos.
	portreadchar			Esta orden leerá un byte del puerto abierto y devolverá un entero. Un resultado "-1" indica que el carácter no está disponible.
	portwritearray		contador buffer	Esa primitiva escribirá en el puerto, previamente abierto, los datos que se encuentran en el buffer de cadenas. Intentará escribir el mayor número posible de caracteres (contador). Devuelve el número actual de bytes escritos.
	portwritechar		data	Esta orden escribirá un byte de datos en el puerto previamente abierto y devolverá el actual número de bytes escritos (0 o 1).
	inport		idpuerto	Esta orden leerá 2 bytes (16 bits) del especificado **idpuerto** y devolverá los datos como un entero.
	outport		idpuerto	Esta orden escribirá 2 bytes de datos en el especificado **idpuerto**. El **idpuerto** se refiere a un Puerto "Hardware". No se debe confundir esta orden con *portopen* y *portwritechar* que sirven para las comunicaciones a través de los puertos Serie y Paralelo.

	inportb		idpuerto	Esta primitiva leerá 1 byte (carácter) desde el especificado **idpuerto** y devolverá los datos como un entero.
	outportb		idpuerto	Esta primitiva escribirá los datos como 1 byte en el **idpuerto** especificado.
	ingameport		idpuerto, máscara	Esta primitiva leerá el valor del canal del puerto de juego especificado por la máscara. Se puede leer directamente el estado del botón usando *inport* 513. Si no hay nada conectado al canal de puerto de juego la primitiva eventualmente terminará su tiempo activo devolviendo como salida un -1.
	dllload		dllnombre	Esta orden carga la DLL (Dynamic Linked Library) de nombre dllnombre en MSWLogo. Una vez haya concluido el trabajo con la DLL o necesite utilizar otra DLL, debe cerrar la DLL.
	dllcall		función	Esta primitiva llama a una determinada función y devuelve el tipo. Pasará los argumentos arg1 como type1 y arg2 como type2 y así sucesivamente. Debe contener 4 apartados (tipo función tipo1 arg1) aunque no se use arg1.
	dllfree			Esta orden cierra la DLL previamente cargada con *dllload*.
Acceso al Teclado y al Ratón				
	keyp			(No soportada en MSWLogo). Devuelve verdadero si se ha pulsado una tecla.
	keyboardon		tecla_pulsada tecla_bajada tecla_liberada	Esta orden captura los eventos del teclado. Para obtener el carácter pulsado se llama a *keyboardvalue*. Observe que la ventana "Pantalla de MSWLogo" debe tener el foco, lo que se logra con *setfocus* [Pantalla de MSWLogo]. La segunda forma detecta independientemente los sucesos de tecla bajada y liberada.
	keyboardoff			Esta orden desactiva la captura de los eventos del teclado.
	keyboardvalue		tecla	Esta primitiva devuelve el valor de la tecla presionada o liberada.

	mouseon		boton_izq_abajo boton_izq_arriba boton_der_abajo boton_der_arriba mover	Esta orden captura los eventos del ratón. Para obtener la posición del ratón usar *mousepos*. Observe que la ventana "Pantalla de MSWLogo" debe tener el foco (NO la ventana de trabajo) para poder capturar los eventos del ratón.
	mouseoff			Desactiva la captura de los eventos del ratón.
	mousepos		posición	Devuelve la posición del puntero del ratón, tras el último evento. Previamente se utilizó *mouseon*.
	cleartext	ct		Borra el texto. En MSWLogo borra las salidas/órdenes de la caja de listas de la ventana Trabajo.
	setcursor		vector	(No soportado en MSWLogo) Comando donde la entrada es una lista de dos números, las coordenadas x e y de una de pantalla. El cursor de la pantalla es movido a la posición pedida.
	cursor			(No soportado en MSWLogo) Devuelve una lista que contiene las actuales coordenadas x e y del cursor de la pantalla.
	setmargins		vector	(No soportado en MSWLogo) Comando donde la entrada debe ser una lista de dos números. El efecto de esto es limpiar la pantalla y entonces adapta para toda a nueva impresión un cambio abajo y a la derecha acorde con los márgenes indicados.
Acceso a Archivos				
	openread		filename binarymode	Comando que abre el archivo para lectura. La posición de lectura es inicialmente la de comienzo del archivo.
	openwrite		filename binarymode	Comando que abre el archivo para escritura. Si el archivo ya existe, la vieja versión es eliminada y un nuevo archivo vacío es creado.
	openappend		filename binarymode	Comando que abre los archivos para escritura. Si este archivo ya existe la posición de escritura es inicialmente tomada del final del viejo archivo, y es escrita nuevamente la data como un anexo.

	openupdate		filename binarymode	Comando que abre archivos para lectura y escritura. La posición de lectura y escritura es inicialmente tomada del final del viejo archivo.
	close		filename	Comando para cerrar los archivos.
	allopen		lista	Devuelve una lista donde los miembros son los nombres de todos los archivos abiertos frecuentemente.
	closeall			Comando que cierra todos los archivos abiertos.
	erasefile		filename	Comando que borra (elimina, remueve) un archivo, el cual no a sido abierto frecuentemente.
	dribble		filename	Comando que crea un nuevo archivo con el nombre como entrada, y comienza registrando en que archivo se leyó por teclado o se escribió desde un terminal.
	nodribble			Comando que detiene el copiado de información dentro de un archivo abierto, y cierra el archivo.
	setread		filename	Comando que hace al archivo un flujo de lectura, usado para *readlist*, etc. El archivo debe estar abierto con *openread* ó *openupdate*.
	setwrite		filename	Comando que hace al archivo un flujo de escritura, para ser usado por *print*, etc. El archivo debe estar abierto con *openwrite*, *openappend* ó *openupdate*.
	reader		filename	Devuelve el nombre del mas frecuente flujo de lectura de un archivo, o una lista vacía si el flujo de lectura es el terminal.
	writer		filename	Devuelve el nombre del mas frecuente flujo de escritura de un archivo, o una lista vacía si el flujo de escritura es el terminal.
	setreadpos		charpos	Comando que toma el puntero al archivo del flujo de lectura para el próximo *readlist*, etc; para comenzar leyendo la posición del carácter.
	setwritepos		charpos	Comando que toma el puntero al archivo del flujo de escritura para el próximo *print*, etc; para comenzar a escribir desde la posición del carácter.

	readpos		charpos	Devuelve la posición del carácter del archivo, como un número, del mas frecuente flujo de lectura de un archivo.
	writepos		posición	Devuelve la posición del archivo, como un número, del mas frecuente flujo de escritura de un archivo.
	eofp		truth	Devuelve un salida verdadera si no hay mas caracteres para ser leídos del flujo de lectura de archivo, sino devuelve falso.
			ESTRUCTURAS DE DATOS	
Constructores				
	word		palabra1 palabra2....	Devuelve una palabra formada por la unión de las entradas.
	list		cosa1 cosa2...	Devuelve una lista cuyos miembros son las entradas, que pueden ser cualquier objeto de Logo (palabra, lista, o arreglo).
	sentence	se	objeto1 objeto2	Devuelve una lista cuyos miembros son sus entradas. Si las entradas son listas o más de dos objetos, se debe utilizar paréntesis.
	fput		objeto lista	Toma como argumento un objeto y una lista y devuelve otra lista igual, pero con el primer argumento, que será el primer miembro de la nueva lista.
	lput		objeto lista	Toma como argumento un objeto y una lista y devuelve otra lista, pero con el primer argumento, que será el último miembro de la nueva lista.
	array		tamaño índice	Devuelve un arreglo con tantos elementos enteros positivos como indique tamaño. Los elementos del arreglo pueden seleccionarse con *item* y cambiarse con *setitem*. Los arreglos se escriben con *print*.
	mdarray		tamañolista (Proced. de la librería)	Devuelve un arreglo multi-dimensional. La primera entrada debe ser una lista con uno o más enteros. La segunda entrada, si la hay, debe ser un único entero que se aplica a cada dimensión del arreglo.
	listtoarray		lista (Proced. de la librería)	Convierte una lista en arreglo, ya que devuelve un arreglo del mismo tamaño que la entrada lista, cuyos

				elementos son los miembros de la entrada lista.
	arraytolist		matriz (Proced. de la librería)	Convierte un arreglo en lista, ya que devuelve una lista cuyos miembros son los elementos del arreglo de entrada. El primer miembro de la lista es el primer elemento del arreglo.
	combine		objeto1 objeto2 (Proced. de la librería)	Si objeto2 es una palabra, devuelve *word* objeto1 objeto2. Si objeto2 es una lista, devuelve *fput* objeto1 objeto2.
	reverse		lista (Proced. de la librería)	Toma como entrada una lista y devuelve otra con los mismos miembros, pero en orden inverso.
	gensym			Devuelve una única palabra. Las palabras tienen la forma G1, G2, etc.
Selectores				
49	first		objeto	Si la entrada es una palabra, devuelve el primer carácter de la palabra. Si la entrada es una lista, devuelve el primer miembro de la lista. Si la entrada es un arreglo devuelve el índice del arreglo.
	firsts		lista	Devuelve una lista que contiene los primeros elementos de cada una de las listas de entrada.
50	last		objeto	Si la entrada es una palabra, devuelve el último carácter de la palabra. Si la entrada es una lista, devuelve el último miembro de la lista.
51	butfirst	bf	objeto	Si la entrada es una palabra, devuelve una palabra que contiene todos los caracteres de la palabra, menos el primero. Si la entrada es una lista, devuelve una lista que contiene todos los miembros de la lista de entrada, menos el primero.
	butfirsts	bfs	lista	Toma como entrada una lista formada por varias sublistas y la devuelve con todos los elementos menos los primeros elementos de cada sublista.
52	butlast	bl	objeto	Si la entrada es una palabra,

				devuelve una palabra que contiene todos los caracteres de la entrada, menos el último. Si la entrada es una lista, devuelve una lista que contiene todos los miembros de la entrada, menos el último.
	item		índice objeto	Si el segundo argumento (objeto) es una palabra, devuelve el carácter número índice de la palabra; si es una lista, devuelve el miembro número índice de la lista y si es un arreglo, devuelve el elemento número índice del arreglo.
	mditem		índicelista mdmatriz (Proced. de la librería)	Toma un arreglo multidimensional como entrada y devuelve el elemento indicado por su primer argumento.
	pick		objeto (Proced. de la librería)	Devuelve un elemento escogido al azar de un objeto de entrada.
	remove		objeto1 objeto2 (Proced. de la librería)	Toma como argumentos dos objetos y devuelve un nuevo objeto igual al que se corresponde con el objeto2, pero sin los miembros iguales al objeto1.
	remdup		objetos (Proced. de la librería)	Devuelve una copia de objetos, pero sin los miembros repetidos. Si dos ó más miembros de la entrada son iguales, sólo permanece uno.
	quoted		objeto (Proced. de la librería)	Devuelve la entrada, si es una lista. Devuelve la entrada con comillas, si es una palabra.
Mutadores				
	setitem		índice matriz valor	Cambia el valor del elemento número índice de un arreglo por un nuevo valor.
	mdsetitem		índicelista mdmatriz valor (Proced. de la librería)	Cambia el valor de un elemento de un arreglo multidimensional (**mdmatriz**), por un nuevo valor. El elemento se selecciona con el argumento **índicelista**.
	.setfirst		lista valor	Esta primitiva cambia el valor del primer elemento de una lista por un nuevo valor.
	.setbf		lista valor	Esta primitiva cambia el valor de los elementos de una lista –menos el del primero- por un nuevo valor.
	.setitem		índice, matriz, valor	Cambia el elemento número índice

				de un arreglo por un nuevo valor.
	push		nombrestack objeto (Proced. de la librería)	Este procedimiento añade un objeto a la pila (**nombrestack**). El primer argumento es una lista, inicialmente vacía. Los nuevos miembros se añaden al principio de la lista. Después, con *pop* se pueden sacar los objetos de la pila.
	pop		nombrestack (Proced. de la librería)	Devuelve el último objeto que entró en la pila, usando *push*, y que sale fuera de la pila.
	queue		nombrecola objeto (Proced. de la librería)	Añade un objeto a una cola. Una cola es una lista inicialmente vacía, en la que los nuevos miembros se añaden al final de la lista. Después, con *dequeue* el primer objeto sale de la cola.
	dequeue		nombrecola (Proced. de la librería)	Devuelve el objeto que entró primero en la cola, usando *queue*, y que saldrá de la cola.
Predicados (Datos)				
	word?		objeto	Devuelve verdadero si la entrada objeto es una palabra, falso si no lo es.
	list?		objeto	Devuelve verdadero si la entrada es una lista; falso, si no lo es.
	array?		objeto	Devuelve verdadero si la entrada es una matriz, falso si no lo es.
	empty?		objeto	Devuelve verdadero si la entrada es una palabra o lista vacía, falso si no lo es.
48	equal?		objeto1 objeto2	Devuelve verdadero si las entradas objeto1 y objeto2 son iguales, falso si no lo son.
	before?		palabra1 palabra2	Devuelve verdadero si palabra1 es anterior en orden alfabético a palabra2, según la secuencia ASCII.
	.eq		objeto1 objeto2	Devuelve verdadero si las dos entradas objeto1 y objeto2 son el mismo objeto. En otro caso, devuelve falso aunque las entradas tengan el mismo valor.
	member?		objeto1 objeto2	Permite conocer si un objeto es miembro de otro. Si objeto2 es una lista o un arreglo, devuelve verdadero si objeto1 y un miembro o elemento de objeto2 son

				equal?, falso si no lo son. Si objeto2 es una palabra, devuelve verdadero si objeto1 y una parte de objeto2 son *equal?*, falso si no lo son.
	substring?		objeto1 objeto2	Permite conocer si un objeto forma parte de una lista o arreglo. Si objeto2 es una lista o una matriz, devuelve verdadero si objeto1 y un miembro o elemento de objeto2 son *equal?*, falso si no lo son. Si objeto2 es una palabra, devuelve verdadero si objeto1 y una parte de objeto2 son *equal?*, falso si no lo son.
	number?		objeto	Devuelve verdadero si la entrada es un número, falso si no lo es.
	backslashed?		carácter	Devuelve verdadero si la entrada tiene una barra a la izquierda (\), en otro caso, falso.
Consultas				
	count		objeto	Si la entrada es una palabra, devuelve el número de caracteres. Si la entrada es una lista o un arreglo, devuelve el número de miembros o elementos.
	ascii		carácter	Devuelve un que representa la entrada de un carácter en código ASCII.
	rawascii		caracter	Devuelve un número (en EEUU, entre 0 y 127) que representa la entrada de un carácter en código ASCII.
	char		número	Devuelve el carácter que corresponde a un código ASCII entre 0 y 127.
	member		objeto1 objeto2	Devuelve parte de una palabra o lista (objeto2) desde la primera aparición de objeto1 hasta el fin. Devuelve la palabra o lista vacía, si objeto1 no es una parte de objeto2.
	lowercase		palabra	Devuelve una palabra, copia de la palabra de entrada, pero en la que se han cambiado las mayúsculas por las correspondientes minúsculas.
	uppercase		palabra	Devuelve una palabra, copia de la palabra de entrada, pero en la que

				se han cambiado las minúsculas por las correspondientes mayúsculas.
	standout		objeto	Devuelve una nueva palabra que, cuando es impresa, parece la entrada pero es mostrada en modo *standout* (boldface, reverse video, ó cualquier terminal para *standout*) (No soportada en MSWLogo aun)
	parse		palabra	Toma una palabra como entrada y la devuelve como lista, como si hubiese sido leída mediante *readlist*.
	runparse		objeto	Devuelve la lista que resultaría si la entrada (palabra ó lista) hubiera sido una línea de instrucciones.
	time			Devuelve en una lista la hora actual del sistema.
	timemilli			Devuelve el tiempo transcurrido desde que se inició Windows, con una precisión de milisegundos.
	OPCIONES DE LAS LÍNEAS DE COMANDOS			
	-width	-w	ancho	Pone la anchura del espacio de trabajo como anchura virtual de dibujo.
	-height	-h	altura	Fija el tamaño del espacio de dibujo virtual para la altura.
	-f			Le dice a MSWLogo que no use srollbars. Si se usa esta opción sin usar -h o -w este automáticamente calculará el especio de trabajo hasta ajustarlo a la pantalla completa exactamente y la tortuga siempre estará a la vista.
	-p			Habilita el modo *perspectiva* como modo estándar.
	-e			Habilita el modo "Experto".
	-l		archivo1 archivo2	MSWLogo puede cargar uno o más archivos desde la línea de comandos cuando se carga MSWLogo.
	SALIR DE MSWLOGO			
	bye			Para salir de MSWLogo.
	OPERACIONES LOGICAS			
	and		entrada1 entrada2.....	Devuelve verdadero si las entradas tienen el valor verdadero, en otro caso, falso.
	or		entrada1 entrada2.....	Devuelve verdadero si una de las entradas es verdadero, en otro

				caso, falso.
	not		entrada	Devuelve verdadero si la entrada tiene el valor falso, y viceversa.
			GESTIÓN DE REDES	
	netstartup		synchronous_dns	Permite la gestión de redes en MSWLogo. Puede llamar a esta primitiva con la bandera **synchronous_dns** a 0 (asincrónico por defecto) o 1 (síncrono).
	netshutdown			Finaliza la gestión de redes.
	netaccepton			Acepta una conexión.
	netacceptoff			Desactiva la aceptación a conexiones entrantes.
	netacceptreceive value		datos sent	Envía los datos. Cuando envíe los datos, esta primitiva puede devolver falso (significa que no están listo para enviar).
	netacceptsendvalue		datos	Devuelve el valor de los datos recibidos.
	netconnecton		machinename, socket, sendready, receiveready, status	Comienza una conexión.
	netconnectoff			Desactiva el envío de datos.
	netconnectsendvalue		sent datos	Envía los datos. Cuando envía los datos, esta primitiva puede devolver falso (significa que no está listo para enviar).
	netconnectreceivevalue		datos	Devuelve el valor de los últimos datos recibidos.
			COMANDOS ESPECIALES	
	caseignoredp			Si es verdadero, indica que las mayúsculas y minúsculas deben ser consideradas igual por *equalp*, *beforep*, *memberp*, etc.
	erract			Esta instrucción será ejecutada si hay un error. Típicamente tiene el valor [PAUSE] para un debugging interactivo.
	printdepthlimit			Si es un entero no negativo, indica el máximo de profundidad de la estructura de sublista que será impresa por *print*, etc.
	printwidthlimit			Si es un entero no negativo, indica el máximo numero de elementos en cualquier lista que será impresa por *print*, etc.
	redefp			Si es verdadero, deja las primitivas

				que serán borradas (*erase*) ó redefinidas (*copydef*).
	startup			Si se asignó el valor de una lista al cargar un archivo con *load*, el valor es ejecutado como una instrucción lista después de cargarse.
	machine		lista	Este comando devuelve las características de la maquina en una lista. Donde la lista está seguida por el formato [Windows WindowsVersion BitMapWidth BitMapHeight Palette]. Donde palette es 1 si es verdadero y 0 si es falso.
			MACROS	
	.macro		procname :input1 :input2 ...	Una macro es una clase especial de procedimiento cuya devolución es evaluada como instrucciones Logo en el contexto de las llamadas de macros. *.macro* es como *to* excepto que el nuevo procedimiento resulta ser una macro.
	.defmacro		procname text	*.defmacro* es como *define* con las mismas excepciones de *.macro*.
	macroexpand		macrocall	Comando que toma como entradas expresiones Logo (*macrocall*) que invoca una macro (esto es, que comienzan con el nombre de una macro) y devuelve las expresiones Logo (*macroexpand*).

Tabla 1: Listado de primitivas de MSWLogo y sus funciones

* Estás primitivas pueden ser escritas con la terminación de un 2 lo que generará que la tortuga quede posicionada en el borde de la figura y no en el centro de la misma.

MicroMundos:

Nº	Comando	Abrevia.	Parámetros	Característica
			GRÁFICOS	
	calca			Copia el fondo que está detrás de la tortuga en la figura actual de la tortuga.
	calcaárea		figura-número [x y] [tamañox tamañoy]	Copia los gráficos en el área definida y los pega en una figura de tortuga.

	colordebajo			Devuelve el número correspondiente al color que se encuentra debajo de la tortuga actual.
	congelaf			Significa congela fondo. Congela los gráficos del fondo en su estado actual.
	conserva			Conserva una copia del fondo de la pantalla. La siguiente vez que se use un mando restaura, el fondo volverá a ser como era en el momento en que conserva guardó su copia.
	descongelaf			Descongela el fondo que fue congelado por *congelaf*.
	desliza		distancia velocidad	Hace que la tortuga se deslice la distancia indicada. La segunda entrada indica la velocidad con la que se deslizará.
	distancia		nombre-de-tortuga	Devuelve la distancia entre la tortuga actual y la tortuga especificada.
	estampa			Estampa una copia de la tortuga en el fondo.
	et			Significa esconde tortuga. Oculta la tortuga actual.
	ffigura	ffig	nombre-o-número, lista-de-nombres-o-números	Fija la figura de una tortuga. Si la entrada es un nombre, éste debe estar precedido por comillas.
	figura	fig		Devuelve el nombre o número de la figura, o una lista de nombres o números de figuras de la tortuga en uso.
	ftamaño	ft	número	Fija el tamaño de la tortuga. El tamaño original es 40 y el tamaño máximo es 160.
	hacia		nombre-tortuga	Fija el rumbo de la tortuga actual para que apunte hacia la tortuga cuyo nombre se ha dado como entrada.
	pinta			Pinta una figura cerrada o toda la pantalla con el color de la tortuga.
	quien			Devuelve el nombre de la tortuga actual.
	restaura			Restaura el fondo utilizando como original el fondo que el mando conserva conservó como copia.
	tamaño			Devuelve el tamaño de la tortuga en

				uso.
	tortuganueva		nombre	Crea una nueva tortuga con el nombre indicado.
1	adelante	ad	número	Mueve la tortuga hacia delante.
2	atrás	at	Número	Mueve la tortuga hacia atrás.
3	bg			Borra Gráficos.
4	color			Devuelve el número del color de la tortuga.
5	colorf			Significa color de fondo.
6	coorx			Devuelve la coordenada x de la tortuga actual.
7	coory			Devuelve la coordenada y de la tortuga actual.
8	cp			Significa con pluma. Pone la pluma a la tortuga actual.
10	derecha	de	número	Gira la tortuga hacia la derecha.
12	fcolor		nombre-o-número	Significa fija color. Fija el color del lápiz de la tortuga. Si la tortuga conserva su figura original, será también de ese color.
13	fcolorf		nombre-o-número	Significa fija color de fondo. Fija el color de fondo de la página.
14	fgrosor		número	Fija el grosor de la pluma de la tortuga que determina el espesor de las líneas que dibuja.
15	fpos		[x y]	Significa fija posición. Lleva a la tortuga a la posición indicada por las coordenadas x y.
16	frumbo		número	Significa fija rumbo. Fija el rumbo de la tortuga en la dirección especificada (en grados).
17	fx		número	Fija la coordenada x de la tortuga en uso.
18	fy		número	Fija la coordenada y de la tortuga en uso. La coordenada x no cambia.
19	grosor			Devuelve un número que representa el grosor de la pluma de la tortuga en uso.
11	izquierda	iz	número	Gira la tortuga a la izquierda.
20	limpia			Limpia los gráficos sin cambiar la posición de la tortuga.
21	mt			Significa muestra tortuga. Muestra la tortuga en uso.
22	origen			Mueve la tortuga al centro de la pantalla (coordenadas 0,0), apuntando hacia arriba.
23	pos			Devuelve la posición de la tortuga.
24	rumbo			Devuelve el rumbo de la tortuga en

				grados.
9	sp			Significa sin pluma. Retira la pluma de la tortuga en uso.
			OBJETOS	
	botónnuevo		nombre, [x y] lista-instrucción	Crea un botón nuevo con el nombre y la instrucción especificada, en la posición [x y]. La posición [x y] es la esquina superior izquierda del botón.
	congela		palabra-o-lista nombre-de-página	Congela objetos para que no se puedan mover, cambiar de tamaño ó eliminar usando el ratón. También se puede usar como entrada un nombre de página para congelar todos los elementos que contiene la página.
	controlnuevo		nombre, [x y] [mín máx actual]	Crea un control nuevo bajo el nombre especificado en la posición indicada. La última entrada es una lista de tres números que representan el valor mínimo, máximo y actual del control.
	descongela		palabra-o-lista nombre-de-página	Descongela el botón, caja de texto, tortuga, control, ó cualquier objeto de una página, para que pueda ser cambiado usando el ratón.
	dile		Tortuga -o- lista-de-tortugas caja-texto	Activa la tortuga ó la caja de texto. Es la única forma de conseguir que muchas tortugas hagan lo mismo al mismo tiempo.
	estampatexto		nombre-caja-texto	Deja una copia de la caja de texto transparente designado en el fondo.
	etexto			Oculta la caja de texto actual. No es posible escribir en una caja de texto oculto, pero las primitivas *escribe*, *inserta*, y *bt* siguen funcionando.
	fija		objeto propiedad valor	Fija una propiedad de un objeto al valor especificado. La primera entrada es el nombre de un objeto. La segunda entrada es el nombre de una propiedad y la última es su valor.
	mtexto			Significa muestra texto. Hace visible la caja de texto actual.
	obtiene		objeto propiedad	Devuelve la propiedad de un objeto del proyecto en uso. La primera entrada es el nombre de un objeto, un color o una página. La segunda

				entrada es un nombre de propiedad.
	opaca		nombre-caja-texto	Hace opaca la caja de texto designada.
	pide		quien lista-de-instrucción	Pide temporalmente a cada elemento quien de la primera entrada que active la lista de instrucciones.
	reiniciavídeo		palabra	Reinicia el vídeo al comienzo.
	remueve		nombre	Elimina un objeto, página, variable de proyecto, creado con *creavarproyecto*, o una variable de tortuga creada con *tortugastienen*.
	textoactual			Devuelve el nombre de la caja de texto actual.
	textonuevo		nombre [x y] [tamañox tamañoy]	Crea una caja de texto nueva bajo el nombre y tamaño especificados en la posición [x y] indicada. El tamaño máximo [tamañox tamañoy] es el tamaño de la página del proyecto.
	tortugastienen		palabra	Asigna una variable a todas las tortugas del proyecto en uso. Este mando también crea una nueva primitiva formada por la palabra fija seguida del nombre de la variable.
	transparente		nombre-caja-texto	Hace transparente la caja de texto especificado.
			EDICIÓN DE TEXTOS	
	anotador			Devuelve el contenido del Portapapeles.
	borra			Suprime el carácter a la derecha del punto de inserción, en la caja de texto actual.
	borratexto	bt		Borra el texto de la caja de texto actual.
	busca		palabra	Indica a MicroMundos que busque y seleccione (resalte) la palabra en la caja de texto actual.
	copia			Inserta una copia en el texto seleccionado del Portapapeles.
	corta			Corta el texto seleccionado en la caja de texto actual y coloca una copia en el Portapapeles.
	ct			Significa color texto. Devuelve el número del color de texto usado en la caja de texto actual, en la posición del cursor.
	cuab			Significa cursor abajo. Mueve el

				cursor, en la caja de texto actual, una línea hacia abajo.
	cuad			Significa cursor adelante. Mueve el cursor, en la caja de texto, un carácter hacia adelante.
	cuar			Significa cursor arriba. Mueve el cursor, en la caja de texto actual, una línea hacia arriba.
	cuat			Significa cursor atrás. Mueve el cursor, en la caja de texto, un carácter hacia atrás.
	cuentalíneas		nombre-caja-texto	Devuelve el número de líneas de la caja de texto en uso.
	desmarca			Deshace el efecto de resaltado de marca o busca.
	eligelínea		nombre-caja-texto	Devuelve el texto en una línea escogida al azar de la caja de texto especificado.
	encontrada?			Devuelve cierto si la búsqueda más reciente tuvo éxito.
	escribe	es	palabra-o-lista	Escribe una palabra o lista en la caja de texto actual. El texto va seguido de un retorno de línea y de una secuencia de avance de línea.
	fcolortexto	fct	nombre-o-numero	Significa fija color texto. Fija el color del texto de la caja de texto en uso. La entrada puede ser el nombre de un color o un número.
	fdl			Significa fin de línea. Mueve el cursor, en la caja de texto actual, al final de la línea en la que se encuentra.
	fdt?			Significa fin de texto. Devuelve cierto si el cursor, en la caja de texto actual, está al final del texto.
	festilo		palabra-o-lista	Fija el estilo de la fuente de la caja de texto actual.
	final			Lleva el cursor (punto de inserción) de la caja de texto actual al final del texto.
	fletra		palabra	Fija el tipo de letra para el texto seleccionado de la caja de texto en uso.
	ftamañoletra	ftl	número	Fija el tamaño de letra para la caja de texto en uso.
	inserta		palabra-o-lista	Escribe la entrada en la posición del cursor, en la caja de texto actual.
	ítemtexto		número-línea	Devuelve la línea especificada de la

			nombre-caja-texto	caja de texto indicado.
	marca			Indica a MicroMundos que comience a seleccionar texto en la caja de texto actual.
	marcado			Devuelve la copia de los caracteres seleccionados en la caja de texto actual como una palabra.
	muestra		palabra-o-lista	Escribe una palabra o una lista en el Centro de Mando.
	pdl			Mueve el cursor, en la caja de texto actual, al principio de la línea.
	pega			Inserta una copia de texto que está en el Portapapeles en la caja de texto actual.
	principio			Desplaza el cursor al comienzo del texto en la caja de texto actual.
	tamañoletra			Devuelve el tamaño de la letra en el punto de inserción de la caja de texto actual.
			PALABRAS Y LISTAS	
	ascii		carácter	Devuelve el número ascii que representa el carácter.
	car		número	Devuelve el carácter cuyo número ascii es el número dado como entrada.
	cuenta		palabra-o-lista	Devuelve el número de elementos de la palabra o lista.
	elige		palabra-o-lista	Devuelve un elemento elegido al azar de la palabra o de la lista.
	frase	fr	palabra-o-lista1 palabra-o-lista2...	Devuelve una lista formada por sus entradas (palabras o listas).
	hazlista		palabra	Transforma una serie de caracteres en una lista.
	idéntico?		palabra-o-lista1 palabra-o-lista2	Devuelve cierto si las dos entradas son idénticas.
	igual?		palabra-o-lista1 palabra-o-lista2	Devuelve cierto si las dos entradas son iguales.
	item		número palabra-o-lista	Devuelve el elemento especificado de una palabra o lista.
	lista		palabra-o-lista1 palabra-o-lista2	Devuelve una lista formada por dos entradas (palabras o listas).
	lista?		palabra-o-lista	Devuelve cierto si la entrada es una lista.
51	menosprimero	mpr	palabra-o-lista	Devuelve la palabra o lista sin el primer elemento.
52	menosúltimo	múl	palabra-o-lista	Devuelve la palabra o lista sin el último componente.
	miembro?		palabra-o-lista1	Devuelve cierto si la primera

			palabra-o-lista2	entrada es un elemento del segundo.
	número?		palabra-o-lista	Devuelve cierto si la entrada es un número.
	palabra		palabra-o-lista1 palabra-o-lista2	Combina sus entradas en una palabra y devuelve una sola palabra.
	palabra?		palabra-o-lista	Devuelve cierto si la entrada es una palabra.
	ponprimero	ppr	palabra-o-lista lista	Devuelve una lista creada al añadir la primera entrada al comienzo de la segunda entrada.
	ponúltimo	púl	palabra-o-lista lista	Devuelve la lista creada al añadir la primera entrada al final de la segunda.
49	primero	pr	palabra-o-lista	Devuelve el primer elemento de la palabra o lista.
50	último	úl	palabra-o-lista	Devuelve el último elemento de la palabra o la lista.
	vacía?		palabra-o-lista	Devuelve cierto si la entrada es una palabra o lista vacía.
			GESTIÓN DE LA PANTALLA	
	bm			Borra el texto en el Centro de Mando.
	fpiedepág		palabra-o-lista	Fija los contenidos del pie de página en las copias impresas.
	ftamañoproyecto		lista-de-números	Fija el tamaño de la página en la pantalla de la computadora para nuevos proyectos. La entrada es una lista de dos números: el ancho y el alto de la página en pasos de tortuga.
	imptexto			Imprime, en la impresora, el contenido de la caja de texto actual o de la página de Procedimientos, dependiendo de lo que se está mostrando.
	mezcla		nombre-proyecto palabra-o-lista-de-tipos	Copia páginas, procedimientos o figuras de otro proyecto en el proyecto en uso.
	modopresentación			Oculta el Centro de Mando, la paleta de herramientas, la barra de título del proyecto y los menúes de MicroMundos.
	nombrapág	np	nombre-página	Nombra la página que se muestra con la entrada. Nombrapág se corresponde con el elemento Nombrar página del menú Páginas.

	páginanueva			Abre una nueva página llamada Páginax
	páginas			Devuelve una lista que contiene los nombres de todas las páginas del proyecto en uso.
	procedimientos			Abre la página de Procedimientos.
	tamañoproyecto			Devuelve el tamaño del proyecto en uso, en pasos de tortuga.
	traepág	tp	nombre-de-página	Muestra la página.
	traeproyecto		nombre-de-proyecto	Trae el proyecto indicado.
			ACCESO AL DISCO	
	archivos		tipo-de-archivo	Devuelve la lista de archivos del tipo de archivo especificado.
	archtextos			Devuelve una lista que contiene los nombres de los archivos de texto del directorio en uso.
	cambiadir		nombre de ruta	Cambia la unidad en uso y/o el nombre del subdirectorio al nombre de ruta
	cargadib		nombre de ruta	Carga un dibujo en la página en uso
	cargafigura		archivo número-figura	Carga el fichero de figuras especificado en esa figura.
	cargatexto		nombre-de-ruta	Carga el archivo de texto en la caja de texto actual ó en la página de Procedimientos.
	colocadib		nombre-archivo-dib posición tamaño	Importa un archivo de dibujo, lo inserta en la posición designada y adapta su tamaño para que se adapte al tamaño indicado.
	dibujos			Devuelve una lista que contiene los nombres de los archivos de dibujo del directorio actual.
	diractual			Devuelve el nombre del directorio en uso.
	directorios			Devuelve una lista de nombres de subdirectorios.
	elarchivo		nombre-de-ruta	Borra cualquier tipo de archivo que no esté protegido.
	guardadib		nombre de ruta	Guarda el fondo de la página en uso como un archivo de dibujo.
	guardafig		nombre-archivo número	Guarda la figura especificada como un archivo de imágenes.
	guardahtml		directorio	Guarda el proyecto en uso como una serie de páginas HTML en un nuevo directorio creado en la carpeta actual.
	guardaproyecto			Guarda el proyecto en uso sin cerrarlo.

	guardatexto	nombre-de-ruta	Guarda el texto de la caja de texto actual o de la página de Procedimientos en un formato de archivo de texto.
	proyectos		Devuelve una lista que contiene los nombres de todos los proyectos de MicroMundos en el directorio.
	CONTROL DE FLUJO Y LÓGICA		
	acabado?	lista-de-instrucción	Devuelve cierto si el proceso indicado ha finalizado.
	activa	palabra-o-lista-a-activar	Activa una palabra o una lista de instrucción.
	alto		Detiene el procedimiento que está activo.
	apaga		Simula un clic de ratón sobre la tortuga actual, apagándola si estaba encendida.
	concuidado	palabra-o-lista-a-activar1 palabra-o-lista-a-activar2	Activa la primera lista de instrucciones. Si la primera lista contiene un error, *concuidado* activará la segunda lista de instrucciones y registra *mensajedeerror* para el error que acaba de ocurrir.
	cuando	lista-instrucción-cierto-o-falso lista-instrucción	Inicia un proceso paralelo que comprueba repetidamente si la primera lista de instrucción devuelve cierto o falso.
	cumplelista	serie lista-de-instrucción	Activa la lista de instrucción para cada elemento de la lista.
	cumpleveces	serie lista-de-instrucción	Activa la lista de instrucción por cada uno de los valores dentro de su serie.
	detén	instrucción-lista	Detiene el proceso dado como entrada.
	deténme		Detiene el proceso en el cual está activo el mando.
	deténtodo		Detiene todos los procedimientos y procesos activos incluyendo tortugas y botones.
	enciende		Simula un clic de ratón sobre la tortuga actual, encendiéndola si estaba apagada.
	escucha		Establece qué tortuga será "quien global" (la tortuga que obedece instrucciones del Centro de Mando).
	esperahasta	cierto-o-falso-lista-a-activar	Le indica a MicroMundos que espere hasta que cierto-o-falso-

				lista-a-activar sea cierto antes de activar otra instrucción.
	finstrucción		lista-instrucción	Fija la instrucción de la tortuga en uso.
	lanza		palabra-o-lista-a-activar	Activa la entrada como un proceso paralelo independiente.
	mensajedeerror			Devuelve el último mensaje de error que ha registrado *concuidado*.
	no		cierto-o-falso	Devuelve el valor lógico inverso de su entrada.
	o		cierto-o-falso1 cierto-o-falso2	Devuelve cierto si alguna de sus entradas devuelve cierto.
	porsiempre		palabra-o-lista-a-activar	Activa la entrada repetidamente como un proceso paralelo independiente.
	repite		número lista-a-activar	Activa la lista de instrucciones el número de veces que se especifica.
	reporta	re	palabra-o-lista	Detiene el procedimiento y devuelve una palabra o lista.
	si		cierto-o-falso lista-a-activar	Activa la lista de instrucción sólo si la condición (primera entrada) devuelve cierto.
	siotro		cierto-o-falso instrucción-lista1 instrucción-lista2	Activa la primera lista de instrucción si la condición es verdadera. Activa la segunda instrucción si la condición es falsa.
	tocando?		nombre-tortuga nombre-tortuga	Devuelve cierto si las dos tortugas se están tocando. Devuelve falso si no se están tocando; siempre devuelve falso si una de ellas es invisible.
	todas		lista-de-instrucción	Hace que todas las tortugas de la página activen la lista de instrucción, una después de otra.
	y		cierto-o-falso1 cierto-o-falso2	Devuelve cierto si todas sus entradas devuelven cierto.
			ASIGNACIÓN	
	asigna		lista-de-nombres valores	Crea una o varias variables temporales. Las variables sólo existirán mientras se active el procedimiento que contiene la instrucción asigna y los procedimientos llamados por este procedimiento.
	bnombre		palabra	Borra de la memoria una variable global.
	bnombres			Borra de la memoria todas las variables globales.

	cosa		nombre	Devuelve el valor de la variable indicada. Se corresponde con el uso de dos puntos (:) antes de una palabra.
	creavarproyecto		palabra	Crea una variable de proyecto representada por un mando y un reportero.
	da		palabra palabra-o-lista	Crea una variable y le asigna el valor palabra-o-lista.
	local		palabra-o-lista	Crea una variable local dentro del procedimiento donde se usó local. Local sólo se puede usar dentro de un procedimiento.
	nombra		palabra-o-lista palabra	Crea una variable y le asigna el valor palabra-o-lista
	nombre?		palabra	Devuelve cierto si la entrada es el nombre de una variable.
	nombres			Devuelve los nombres de todas las variables con sus valores.
	varproyecto			Devuelve la lista de las variables de proyecto actualmente definidas.
			MATEMÁTICAS	
25	diferencia	-	número1 número2	Devuelve la diferencia entre el número1 y el número2.
26	producto	*	número1 número2	Devuelve el producto del número1 multiplicado por el número2.
27	cociente	/	número1 número2	Devuelve el resultado del cociente del número1 dividido por número2.
28	suma	+	número1 número2	Devuelve la suma del número1 y el número2.
44	menor?	<	número1 número2	Devuelve cierto si el número1 es menor que el número2.
48	=		número1 número2	Devuelve cierto si palabra-o-lista1 es igual a palabra-o-lista2.
45	mayor?	>	número1 número2	Devuelve cierto si el número1 es mayor que el número2.
30	abs		número	Devuelve el valor absoluto del número.
38	arctan		número	Devuelve el arco tangente de su entrada (la función inversa de la tangente).
46	azar		número	Devuelve un entero positivo menor que número.
37	cos		número	Devuelve el coseno de su entrada.
39	entero		número	Devuelve la parte entera de su entrada.
33	exp		número	Devuelve e a la potencia del número.

35	ln		número	Devuelve el logaritmo natural (el logaritmo en base e) del número.
34	log		número	Devuelve el logaritmo del número.
29	menos	-	número	Devuelve el número con el signo invertido.
36	pi			Devuelve la constante pi.
32	potencia		número1 número2	Devuelve número1 elevado a la potencia de número2.
31	rc		número	Devuelve la raíz cuadrada de su entrada.
47	reazar			Reproduce la misma secuencia de números generados al azar.
40	redondeo		número	Devuelve el número dado como entrada redondeado al entero más próximo.
41	residuo		número1 número2	Devuelve el resto que queda después de dividir número1 por número2.
42	sen		número	Devuelve el seno del número de grados.
43	tan		número	Devuelve la tangente de su entrada.
			ENTRADA DE DATOS	
	anuncia		palabra-o-lista	Muestra el mensaje en una caja de alerta. Haga clic en Aceptar para cerrar la caja.
	leecar			Detiene la ejecución y espera que se introduzca un carácter por el teclado.
	posratón			Devuelve las coordenadas de la página que representan la posición del ratón actual en la pantalla.
	pregunta		palabra-o-lista	Abre una caja de diálogo que muestra la pregunta y una zona para escribir la respuesta.
	respuesta			Devuelve el contenido de la última respuesta escrita en el cuadro de diálogo de pregunta.
	tecla?			Devuelve cierto si se ha pulsado una tecla del teclado.
			TIEMPO	
	cronómetro			Devuelve un número que representa el tiempo pasado desde que comenzó el programa, o desde que se activó el último mando *iniciacron*.
	espera		número	Origina una pausa en la ejecución de un programa o instrucción. El tiempo se mide en décimas de

				segundo.
	iniciacron			Significa inicia cronómetro. Inicializa el cronómetro a 0. El cronómetro empieza a correr cuando se inicia MicroMundos.
			SONIDO	
	finstrumento		nombre-o-número	Fija el instrumento musical que ejecutará la siguiente nota usando el mando nota. Hay 7 nombres de instrumentos: piano, clavicordio, vibráfono, guitarra, violín, clarinete y calimba.
	nota		número duración	Ejecuta una nota con el instrumento en uso. La primera entrada debe ser un número MIDI de nota y la segunda es la duración en décimas de segundo.
	silencio		duración	Inserta un silencio en una secuencia de notas. La duración viene dada en décimas de segundo, y tiene un máximo de 255.
			ESPACIO DE TRABAJO	
	espacio			Devuelve la cantidad de espacio Logo libre en bytes.
	recicla			Libera el espacio de memoria que no está siendo utilizado.

Tabla 2:Listado de primitivas de MicroMundos y sus funciones

Anexo IV-5

Diagrama de clases de MSWLogo

A continuación se presénta el diagrama de clases de MSWLogo. Para facilitar su entendimiento se decidió utilizar la siguiente notación:

Clase	Clases que pertenecen a MSWLogo
Clase	Clases que pertenecen a las librerías de Borland C++
Clase	Clases que se repiten en más de un diagrama

Por motivos de espacio se dividió el diagrama de clases de MSWLogo en seis partes; cada una de éstas se encuentra numerada. Además, por el mismo motivo, los atributos y métodos de cada una de las clases de MSWLogo se listan en la tabla 3, del anexo IV-6.

Diagrama de Clases de MSWLogo

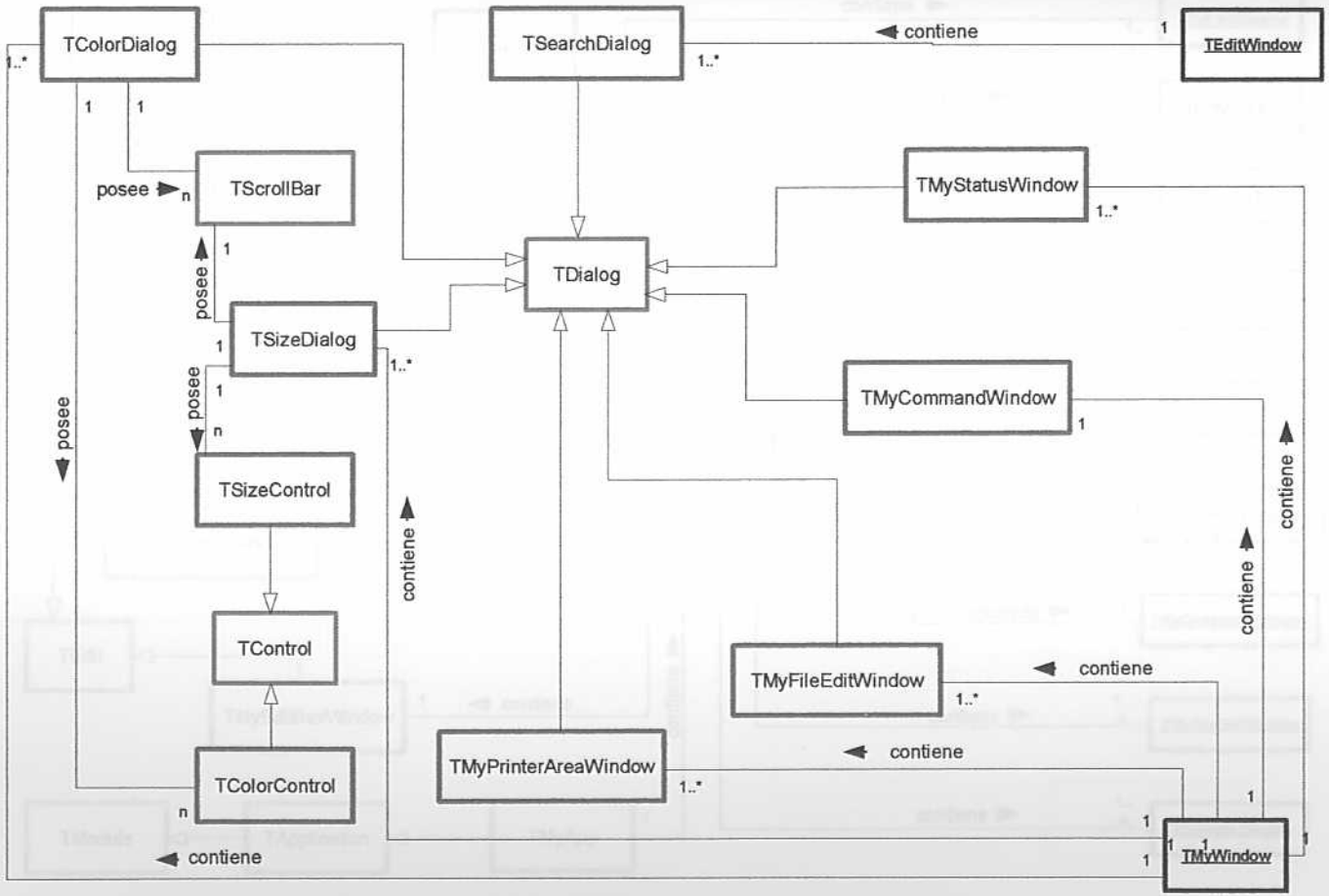


Diagrama 1: Diagrama de Clase de MSWLogo. Parte 1

Diagrama 2: Diagrama de Clases de MSWLogo. Parte 2

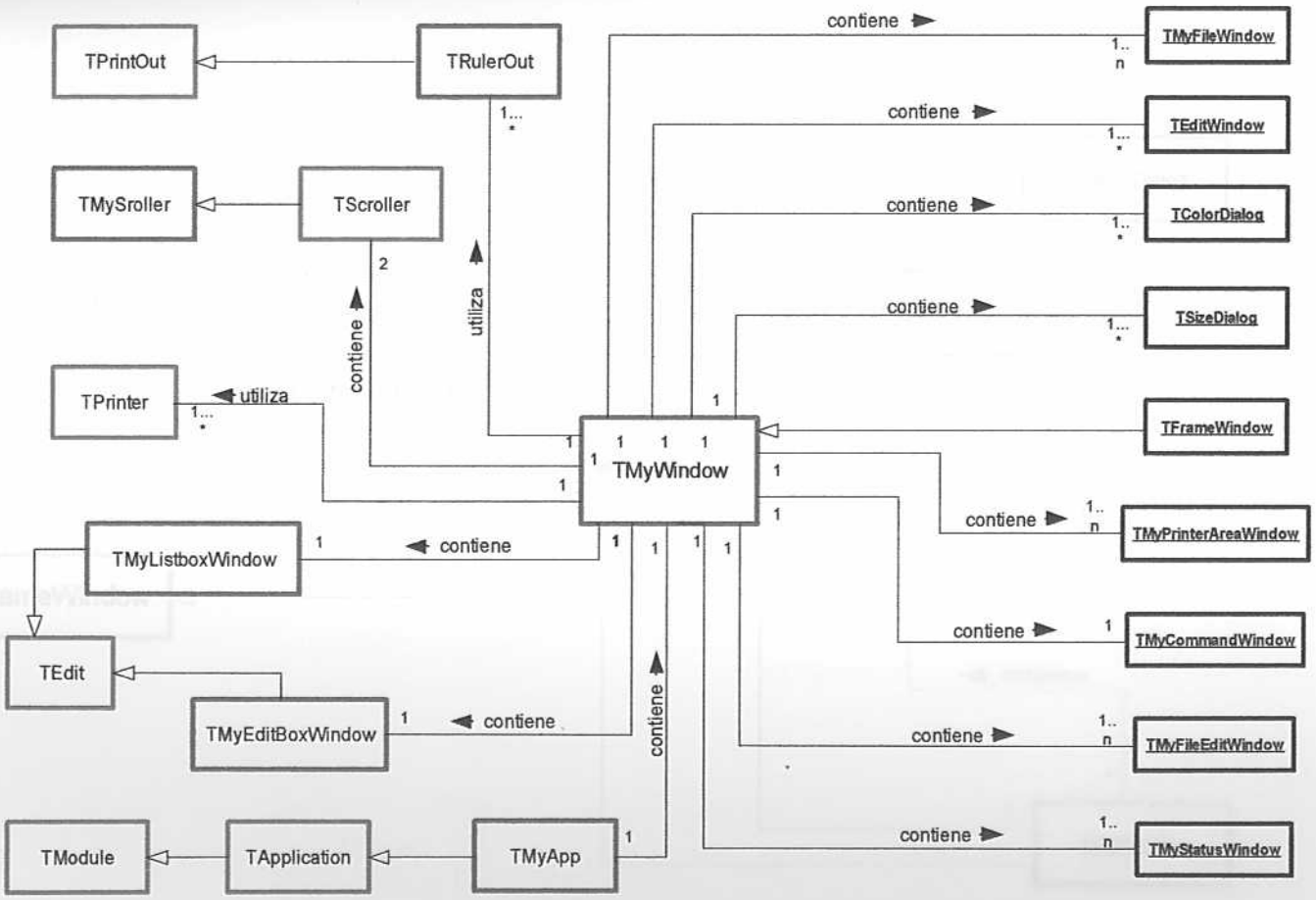


Diagrama 2: Diagrama de Clase de MSWLogo. Parte 2

Diagrama de Clases de MSWLogo

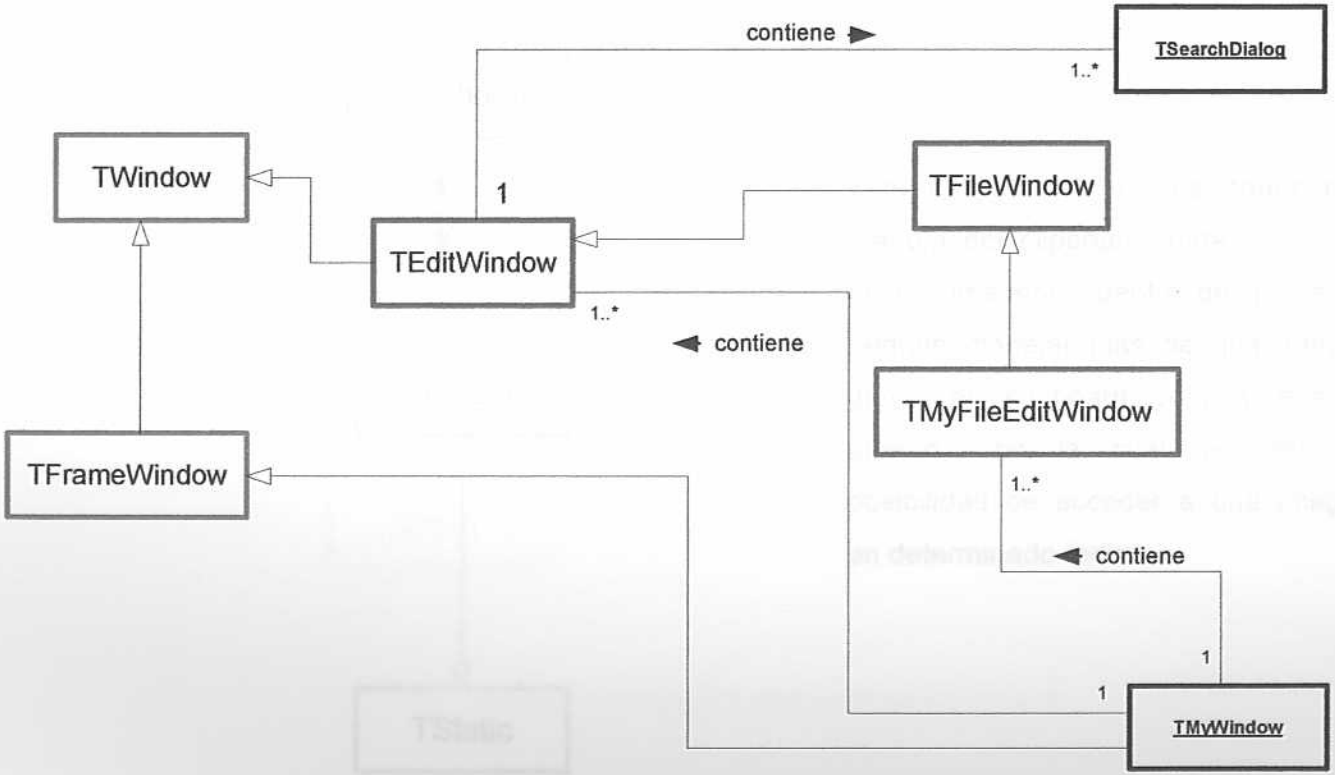
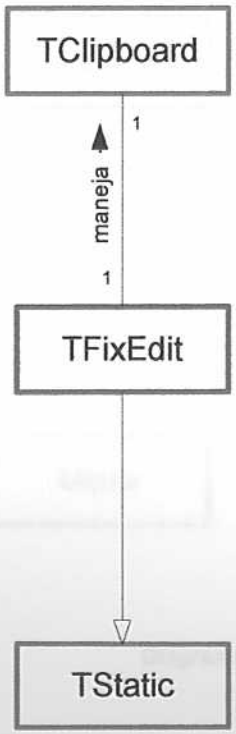


Diagrama 3: Diagrama de Clase de MSWLogo. Parte 3

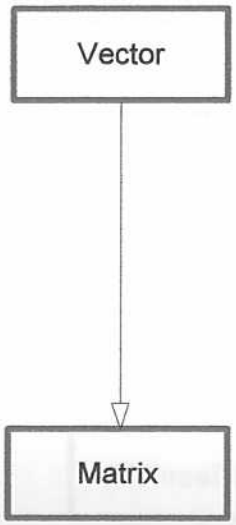
Diagrama de Clases de MSWLogo



Este conjunto de clases controla el manejo del clipboard, para el trabajo de las imágenes dentro del proyecto, permite manejar más de una imagen dentro del clipboard, semejante a un arreglo, con la cual se tiene la posibilidad de acceder a una imagen en determinado índice.

Diagrama 4: Diagrama de Clase de MSWLogo. Parte 4

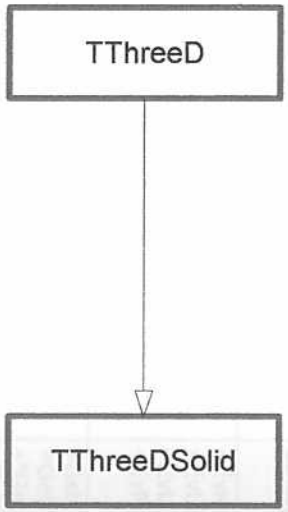
Diagrama de Clases de MSWLogo



Este conjunto de clases controla el manejo de la posición y tamaño de la tortuga, permite trabajar en dos o tres dimensiones.

Diagrama 5: Diagrama de Clase de MSWLogo. Parte 5

Diagrama de Clases de MSWLogo



Este conjunto de clases controla el manejo de los objetos en tercera dimensión, tales como polígonos y sólidos, dibujados en el área de trabajo.

Diagrama 6: Diagrama de Clase de MSWLogo. Parte 6

Anexo IV-6

Cuadro de Atributos y Métodos de las Clases de MSWLogo

El siguiente cuadro muestra las clases pertenecientes a MSWLogo con sus respectivos atributos y métodos.

Nombre de la Clase	Atributos	Métodos
	CLASS.H	
TMyFileWindow	+*args_list: NODE +hEditHeap +*pm +hEdtFont	+CanClose(): bool +EndEdit(): int +EvDestroy(): void +DefWindowProc: <sin especificar +CMExit(): void +CMTest(): void +CMSaveAndExit(): void +CMSaveToWorkspace(): void +CMEditSetFont(): void +CMHelp(): void +CMHelpEdit():void +CMHelpEditTopic():void +SetupWindow():void
TMyEditboxWindow		+EvKeyDown(): void +EvRButtonUp():void
TMyListboxWindow		+EvKeyDown(): void +EvLButtonDown(): void +EvLButtonDblClk(): void +EvRButtonUp():void
TMyCommandWindow		+SetupWindow(): void +GetWindowClass(): void +*GetClassName(): char +EvDestroy(): void +EvSize(): void +CmCancel(): void +CmOk(): void +EvClose(): void +DoListBox: void +DoEditBox: void +DoButtonExecute(): void +DoButtonEdall(): void +DoButtonHalt(): void +DoButtonStatus(): void +DoButtonYield(): void +DoButtonPause: void +DoButtonTrace(): void +DoButtonReset(): void

TMyStatusWindow		
TMyPrinterAreaWindow		+SetupWindow(): void +CloseWindow(): void +ResetWindow(): void +DoReset(): void
TMyFileEditWindow	FileEditAll: int	+CanClose(): bool +SetupWindow(): void +DoAll(): void +DoCombo(): void
TMyApp	-RR	+InitMainWindow(): void +InitInstance(): void +IdleAction(): bool +ProcessAppMsg(): bool +EvSysColorChange(): void
TMyWindow	-size -hListboxHeap; -HIcon; -ScreenDC; -MemDC; -PrintMemDC; -OldBitmap; -OldBitmap2; -OldPalette; -OldPalette2; -hCmdFont; -R; -RR; -ii: long -PenSize: int -FileName: char -BitmapName: char -IsNewFile: bool -IsNewBitmap: bool -*pm; -*Printer; -*EditWindow: TMyFileWindow -*CommandWindow: TMyCommandWindow -*EditboxWindow: TMyEditboxWindow -*ListboxWindow: TMyListboxWindow -*StatusWindow: TMyStatusWindow -*PrinterAreaWindow: TMyPrinterAreaWindow -*FileEditWindow: TMyFileEditWindow	+MyPopupEdit(): void +MyPopupCommander(): void +MyPopupStatus(): void +MyPopupStatusKill(): void +MyPopupInput(): int +LoadBitmapFile(): bool +OpenDIB(): bool +GetBitmapData(): void +DumpBitmapFile(): bool +WriteDIB(): bool +PutBitmapData(): void +Printit(): void +SaveFile(): void +SaveFileAs(): void +SaveBitmap(): void +SaveBitmapAs(): void +SetupWindow(): void +CanClose(): bool +GetWindowClass(): void +Paint(): void +EvDestroy(): void +EvKeyDown(): void +EvKeyUp(): void +EvChar(): void +EvLButtonDown(): void +EvLButtonUp(): void +EvRButtonDown(): void +EvRButtonUp(): void +EvMouseMove(): void +EvTimer(): void +EvSize(): void +EvVScroll(): void +EvHScroll(): void +CMFileNew(): void +CMFileOpen(): void +CMFileSave(): void +CMFileSaveAs(): void +CMFileEdit(): void +CMFileErase(): void

		+CMExit(): void +CMBitmapNew(): void +CMBitmapOpen(): void +CMBitmapSave(): void +CMBitmapSaveAs(): void +CMBitmapPrint(): void +CMBitmapPrinterSetup(): void +CMBitmapPrinterArea(): void +CMHelp(): void +CMHelpMCI(): void +CMHelpHelp(): void +CMHelpDemo(): void +CMHelpExamples(): void +CMHelpReleaseNotes(): void +CMHelpTutorial(): void +CMHelpAbout(): void +CMHelpAboutMS(): void +CMControlExecute(): void +CMSetFont(): void +CMSetCommanderFont(): void +CMSetPenSize(): void +CMSetPenColor(): void +CMSetFloodColor(): void +CMSetScreenColor(): void +CMZoomIn(): void +CMZoomOut(): void +CMZoomNormal(): void +MMMCINotify(): <sin especificar> +WMCheckQueue(): <sin especificar> +OnNetworkConnectSendAck(): <sin especificar> +OnNetworkConnectSendFinish(): <sin especificar> +OnNetworkListenReceiveAck(): <sin especificar> +OnNetworkListenReceiveFinish(): <sin especificar> protected: TMyWindow(StreamableInit) TFrameWindow(streamableInit)
TRulerOut		+PrintPage(): void +GetDialogInfo(): void +SetBanding(): void +HasPage(): bool
TMyScroller	-Xpos: long -Ypos: long	
	COLORDLG.H	
TColorControl	+Color	+SetColor(): void +GetColor(): void -*GetClassName(): char -Transfer(): <sin especificar> -EvPaint(): void -EvLButtonDown(): void -EvLButtonDblClk(): void

TColorDialog	+*ColorBar1 +*ColorBar2 +*ColorBar3 +*SelColor +*colorcaption	+SetupWindow(): void +TransferData(): void +UpdateBars(): void +ClickFmControl1(): void +ClickFmControl2(): void +ClickFmControl3(): void +ClickFmControl4(): void +ClickFmControl5(): void +ClickFmControl6(): void +ClickFmControl7(): void +ClickFmControl8(): void +SetColorFmControl(): void +SetColorFmSlider(): void
	VECTOR.H	
VECTOR	-x: double -y: double -z: double	
MATRIX	-e11: double -e12: double -e13: double -e21: double -e22: double -e23: double -e31: double -e32: double -e33: double	
TThreeD	-A, B, C, D, Dval: double -From, At, Up, Light: vector -Angle: double -A1, A2, A3: vector -Vl, Vt, Vr, Vb: long -Offset: vector -Zmin: double -Zmax: double -Dist: vector	+WORLDtoPC(): void +Code(): void +SetAt(): void +SetUp(): void +SetClip(): void +SetFrom(): void +SetEye(): void +SetVolume(): void +SetLight(): void +TransformSegment(): bool +Clip3DSegment(): bool +TransformPoint(): bool +Clip3DPoint(): bool
	3DSOLID.H	
TThreeDSolid	+MemDC +ScreenDC +PolyCount: long +iSplitPolyCount: long +TraverseTree_s: vector +Tree +Ambient: double +Diffuse: double +PolyClass: int	+View(): void +DisposeTree(): void +DisposePolygon(): void +AddPoint(): void +RemovePoint(): void +InsertPoint(): void +DisposeVertices(): void +DisposeFalseShares(): void +CalcPlaneEq(). void

	+ThisPointClass: int +LastInOutPointClass: int +iPolyDepth: long +bDepthExceeded: bool +polyA +polyB +sideA: double +sideB: double +pt +ptA +ptB +*VerticesStart +*Vertices +*VerticesIn +*shareA +*shareB +*DeferVerticesOut +*DeferVerticesIn +bDefer: bool	+CalcPolyNormal(): void +PrecomputeCentroid(): void +DisposeBSP(): void +AddPolygon(): void +MakeBSPNode(): <sin especificar> +AddList(): void +CalcSign(): double +Intersect(): void +AddToBSPTree(): void +WorldToDisplay(): bool +ComputeColor(): <sin especificar> +DisplayPolygon(): void +TraverseTree(): void
	EDITWND.H	
TEditWindow	-*Editor -*Editor -SearchStruct -IsReplaceOp: bool	+DoSearch(): void +EvSize(): void +EvRButtonUp(): void +EvSetFocus(): void +CmEditFind(): void +CmEditFindNext(): void +CmEditReplace(): void +CmFilePrint(): void
TSearchDialog		#TSearchDialog(): void
	FILEWND.H	
TFileWindow	+*FileName +IsNewFile: bool	+CanClear(): void +CanClose(): bool +AssignMenu(): bool +NewFile(): void +Open(): void +Read(): bool +ReplaceWith(): void +Save(): bool +SaveAs(): bool +SetFileName(): void +Write(): bool +StripClipboard(): void #CmFileNew(): void #CmFileOpen(): void #CmFileSave(): void #CmFileSaveAs(): void #CmEditUndo(): void #CmEditCopy(): void #CmEditCut(): void #CmEditPaste(): void #CmEditClear(): void #CmEditDelete(): void

		#SetupWindow(): void
	SIZEDLG.H	
TSizeControl	-Size	+SetSize(): void +GetSize(): void +GetClassName(): char +Transfer(): <sin especificar> +EvPaint(): void +EvLButtonDown(): void +EvLButtonDblClk(): void
TSizeDialog	#SizeBar #SelSize #sizecaption: char	+SetupWindow(): void +TransferData(): void +UpdateBars(): void +ClickFmControl1(): void +ClickFmControl2(): void +ClickFmControl3(): void +ClickFmControl4(): void +ClickFmControl5(): void +ClickFmControl6(): void +ClickFmControl7(): void +ClickFmControl8(): void +SetSizeFmControl(): void +SetSizeFmSlider(): void
	FIXEDIT.H	
TFixEdit	+id: int +text: char +x, y, w, h: int +textLen: unit +multiline: bool +module	+GetNumLines(): int +GetLineLength(): int +GetLine():bool +GetSubText(): void +GetSelection(): void +IsModified(): bool +ClearModify()void +GetLineFromPos(): int +GetLineIndex(): unit +Transfer(): unit +LockBuffer(): char +UnlockBuffer(): void +DeleteSubText(): bool +DeleteLine(): bool +DeleteSelection(): bool +SetSelection(): bool +Scroll(): void +Insert(): void +Search(): int +GetRect(): void +SetRect(): void +SetRectNP(): void +FormatLines(): void +SetTabStops(): void +GetHandle(): <sin especificar> +SetHandle(): void +SetPasswordChar(): void +GetFirstVisibleLine(): int +SetReadOnly(): void

		+GetPasswordChar(): uint +GetWordBreakProc(): <sin especificar> +SetWordBreakProc(): void +CanUndo(): bool +EmptyUndoBuffer(): void +Undo(): void +Paste(): void +Copy(): void +Cut(): void +IsValid(): bool +SetValidator(): void +ValidatorError(): void +Clear(): void #CmEditCut(): void #CmEditCopy(): void #CmEditPaste(): void #CmEditDelete(): void #CmEditClear(): void #CmEditUndo(): void #CmSelectEnable(): void #CmPasteEnable(): void #CmCharsEnable(): void #CmModEnable(): void #ENErrSpace(): void #GetClassName(): char #SetupWindow(): void #Validator(): <sin especificar> #EvChar(): void #EvKeyDown(): void #EvGetDlgCode(): uint #EvSetFocus(): void #EvKillFocus(): void #CanClose(): bool #EvChildInvalid(): void -TFixEdit(): <sin especificar> -TFixEdit&(): <sin especificar>

Tabla 3: Atributos y métodos de MSWLogo

Fuente: Elaboración propia

Anexo IV-7

Código fuente de la Caja de Herramientas "MOVER"

```
PARA MOVER
Local "TECLA
Haz "TECLA LC
SI :TECLA = 1 [BP CUADRO]
SI :TECLA = 2 [BP RECTANGU]
SI :TECLA = 3 [BP TRIANGU]
Si :TECLA = "a [AV 10]
Si :TECLA = "r [RE 10]
SI :TECLA = "h [GD 30]
Si :TECLA = "d [GD 10]
Si :TECLA = "i [GI 10]
SI :TECLA = "z [GI 30]
Si :TECLA = "p [BL]
Si :TECLA = "n [SL]
Si :TECLA = "g [GOMA]
Si :TECLA = "b [BP BL]
SI :TECLA = "c [SL CENTRO BL]
SI :TECLA = "ñ [PONCL 10]
SI :TECLA = "l [PONCL 13]
SI :TECLA = "k [PONCL 1]
SI :TECLA = "m [MT]
SI :TECLA = "o [OT]
SI :TECLA = "e [RECICLA]
SI :TECLA = "s [ADIÓS]
MOVER BP BL
Fin

Para a :AVA
Av :AVA
Fin

Para azul
PonCL 2
Fin

Para amari
PonCL 15
Fin

Para b
BP
Fin
Para c
SL Centro BL
Fin
```

```
Para d :gra
GD :gra
Fin

Para f :rad
Repite 360 [gd 1 av 6.28 * :rad / 360]
Fin

Para fonazul
PonF 12
Fin

Para fonmora
PonF 14
Fin

Para fonrojo
PonF 13
Fin

Para fongris
PonF 8
Fin

Para fonblan
PonF 16
Fin

Para g
Goma
Fin

Para g1
PonGrosor 1
Fin

Para g2
PonGrosor 3
Fin

Para g3
PonGrosor 5
Fin

Para i :gra
GI :gra
Fin

Para l
SL Av 10
```

```
BL Rellena
SL Re 10 BL
Fin

Para m
MT
Fin

Para mora
PonCL 6
Fin

Para n
SL
Fin

Para negro
PonCL 1
Fin

Para p
BL
Fin

Para r :RET
Re :RET
Fin

Para rojo
PonCL 5
Fin

Para v :rad :gra
Repite :gra [gd 1 av 6.28 * :rad / 360 ]
Fin

Para verde
PonCL 3
Fin

Para rec :LARGO :ANCHO
Repite 2 [AV :LARGO GD 90 AV :ANCHO GD 90]
Fin

Para poli :NUM :LONG
Repite :NUM [AV :LONG GD 360 / :NUM]
Fin

Para t
OT
fin
```

```
Para blanco
ponCL 16
fin

Para naran
ponCL 7
fin

Para fonamari
PonF 15
fin

Para fonnegro
PonF 1
fin

Para fonverde
ponF 11
fin
```

Anexo IV-8

Diagrama de Clases de MundoLogo

A continuación se presénta el diagrama de clases de MundoLogo. Para facilitar su entendimiento se decidió utilizar la siguiente notación:

Clase	Nuevas clases incorporadas al software
Clase	Clases que pertenecen a MSWLogo
Clase	Clases que pertenecen a las librerías de Borland C++
Clase	Clases que se repiten en más de un diagrama

Por motivos de espacio se dividió el diagrama de clases de MundoLogo en seis partes; cada una de éstas se encuentra numerada (La parte 4 del diagrama de clases fue presentado en la página 60 de tomo). Además, por el mismo motivo, los atributos y métodos de cada una de las clases de MSWLogo se listan en la tabla 4, del anexo IV-9.

Diagrama de Clases de MundoLogo

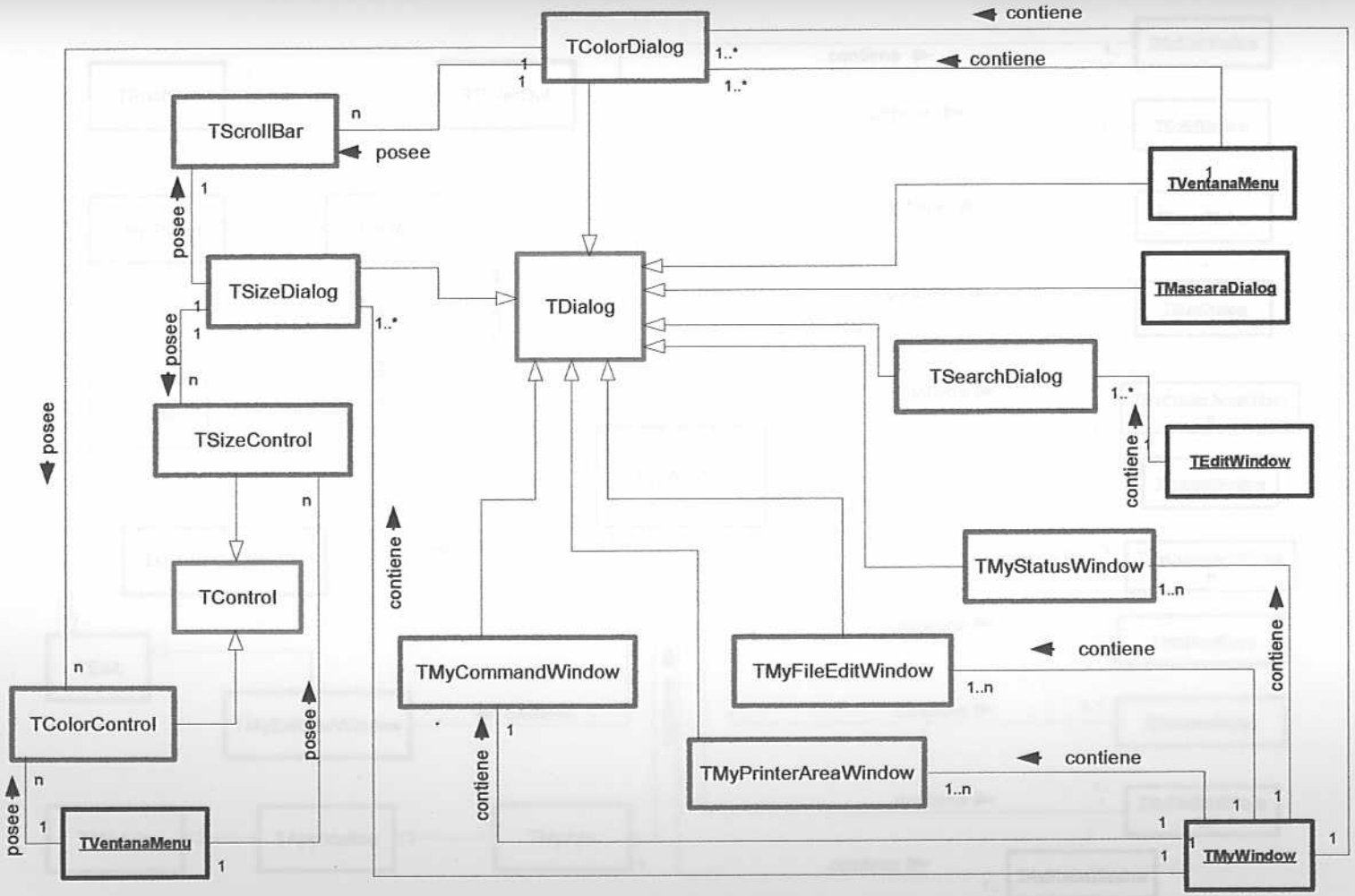


Diagrama 7: Diagrama de Clase de MundoLogo. Parte 1

Diagrama de Clases de MundoLogo

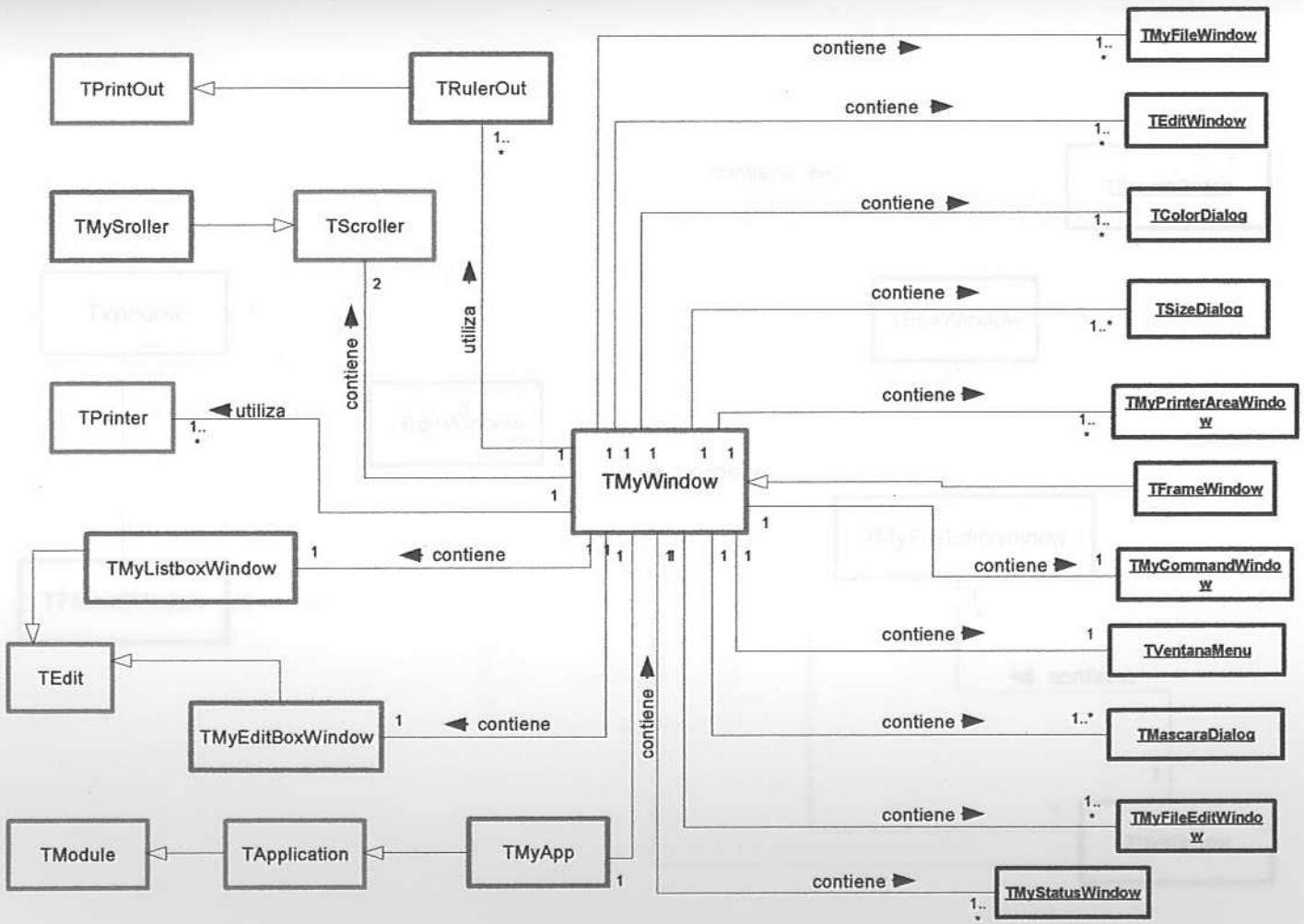


Diagrama 8: Diagrama de Clase de MundoLogo. Parte 2

Diagrama de Clases de MundoLogo

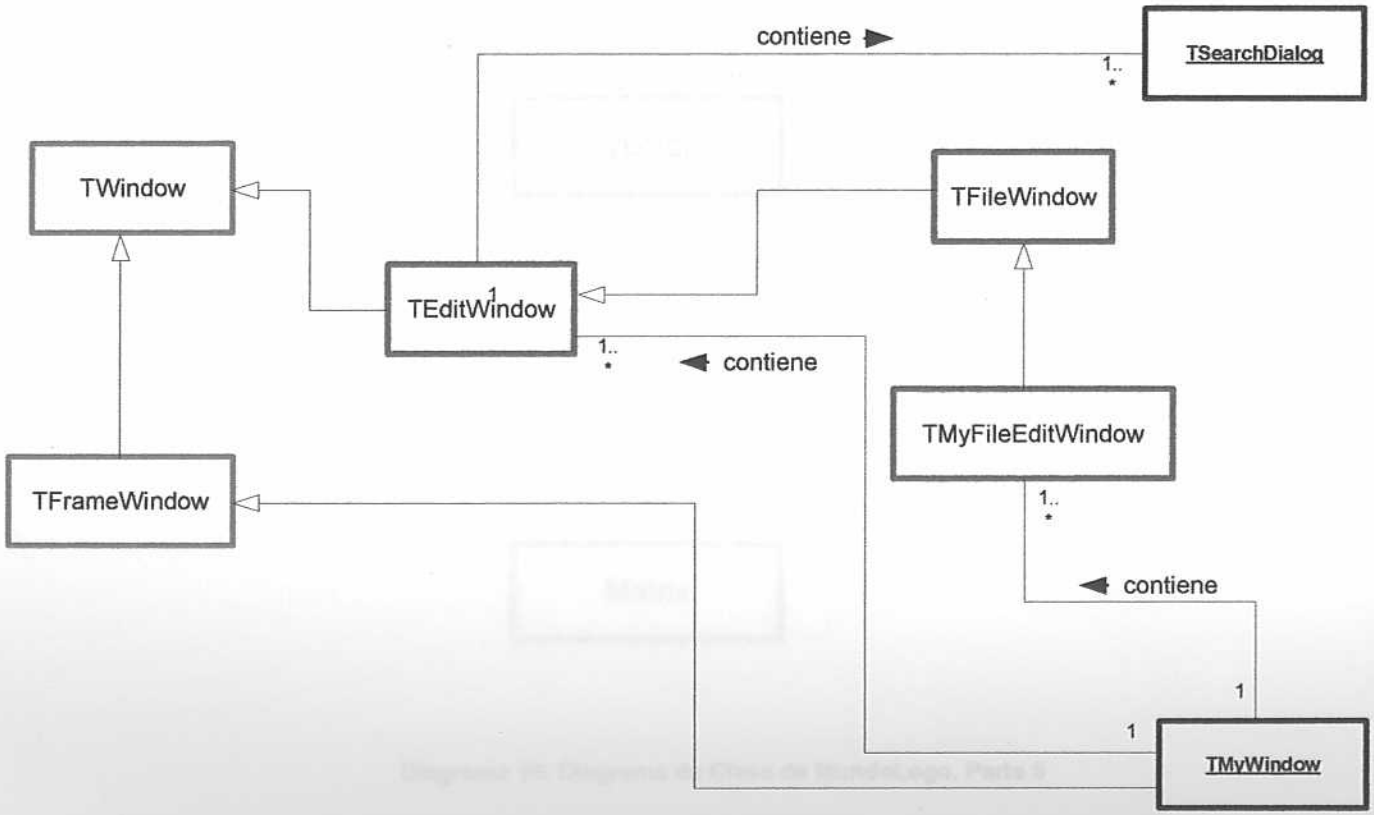


Diagrama 9: Diagrama de Clase de MundoLogo. Parte 3

Diagrama de Clases de MundoLogo

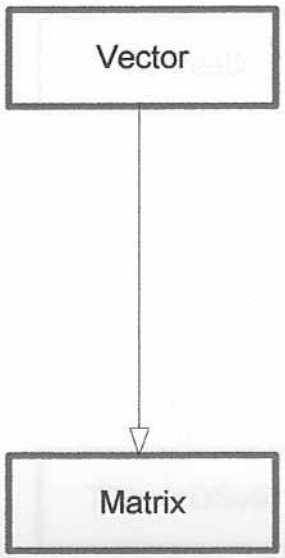


Diagrama 10: Diagrama de Clase de MundoLogo. Parte 5

Diagrama de Clases de MundoLogo

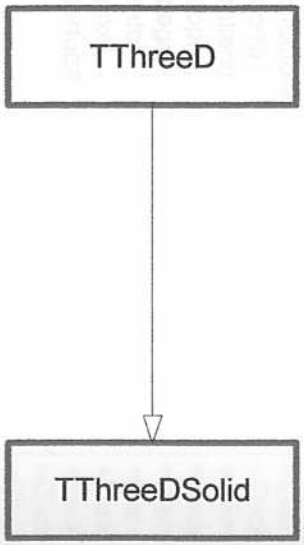


Diagrama 11: Diagrama de Clase de MundoLogo. Parte 6

Anexo IV-9

Cuadro de Atributos y Métodos de las Clases de MundoLogo

La siguiente tabla muestra las clases pertenecientes a MundoLogo con sus respectivos atributos y métodos.

Nombre de la Clase	Atributos	Métodos
	CLASS.H	
TVentanaMenu	+*SelColorV +*SelSizeV +*blapiz +*blinea +*bcuadrado +*bcuadradore +*bcirculo +*bcirculore +*bborra +*brelleno +*bcopiar +*bpegar +*braton +*bpixel +*btortuga	+SetupWindow(): void +GetWindowClass(): void +AsignaColor(): void +*GetClassName(): char +EvDestroy(): void +EvSize(): void +DoButtonTortuga(): void +ClickFmControl1(): void +ClickFmControl2(): void +ClickFmControl3(): void +ClickFmControl4(): void +ClickFmControl5(): void +ClickFmControl6(): void +ClickFmControl7(): void +ClickFmControl8(): void +ClickFmControl9(): void +ClickFmControl10(): void +ClickFmControl11(): void +ClickFmControl12(): void +ClickFmControl13(): void +ClickFmControl14(): void +ClickFmControl15(): void +ClickFmControl16(): void +ClickFmControl17(): void +ClickFmControl18(): void +BotonMas() : void +ClickSizeControl1(): void +ClickSizeControl2(): void +ClickSizeControl3(): void +ClickSizeControl4(): void +DibLapiz(): void +DibLinea(): void +DibCuadrado(): void +DibCuadradoRe(): void +DibCirculo(): void +DibCirculoRe(): void +DibBorra(): void +DibRelleno(): void +DibCortar(): void

		+DibRaton(): void +DibPegar(): void +DibPixel(): void +CmCancel(): void +CmOk(): void +EvClose(): void
MASCARA.H		
TMascara	+*mascara1 +*mascara2 +*mascara3 +*mascara4 +*mascara5 +*mascara6 +*mascara7 +*mascara8	+SetupWindow(): void +ClickMascara1(): void +ClickMascara2(): void +ClickMascara3(): void +ClickMascara4(): void +ClickMascara5(): void +ClickMascara6(): void +ClickMascara7(): void +ClickMascara8(): void +PonFigura(): void
GLYPHBUTTONEX.H		
TGlyphButtonEx	+Id: int +text, +X, Y, W, H: int +isDefault: bool +module -LayoutStyle -GlyphResId -BorderDistanceX: int -BorderDistanceY: int -UserDefinedStyle: int -Xglyph: int -Yglyph: int -Xtext: int -Ytext: int	+SetGlyphEx(): void +SetBorderDistance(): void +GetBorderDistanceX(): int +GetBorderDistanceY(): int +SetLayoutStyleEx(): void +SetUserDefinedLayoutStyle(): void +SetTextOriginEx(): void +SetGlyphOriginEx(): void #Paint(): void #LayoutTextGlyph(): void #PaintButtonEx(): void #PaintFrameEx(): void #PaintDefaultRectEx(): void #PaintFaceEx(): void #PaintFocusRectEx(): void

Tabla 4: Atributos y métodos de MundoLogo

Fuente: Elaboración propia

Anexo IV-10

Casos de Uso de MundoLogo

A continuación se presentan los casos de Uso de MundoLogo (el caso de Uso de Nivel 1 fue presentado en la página 72 del tomo).

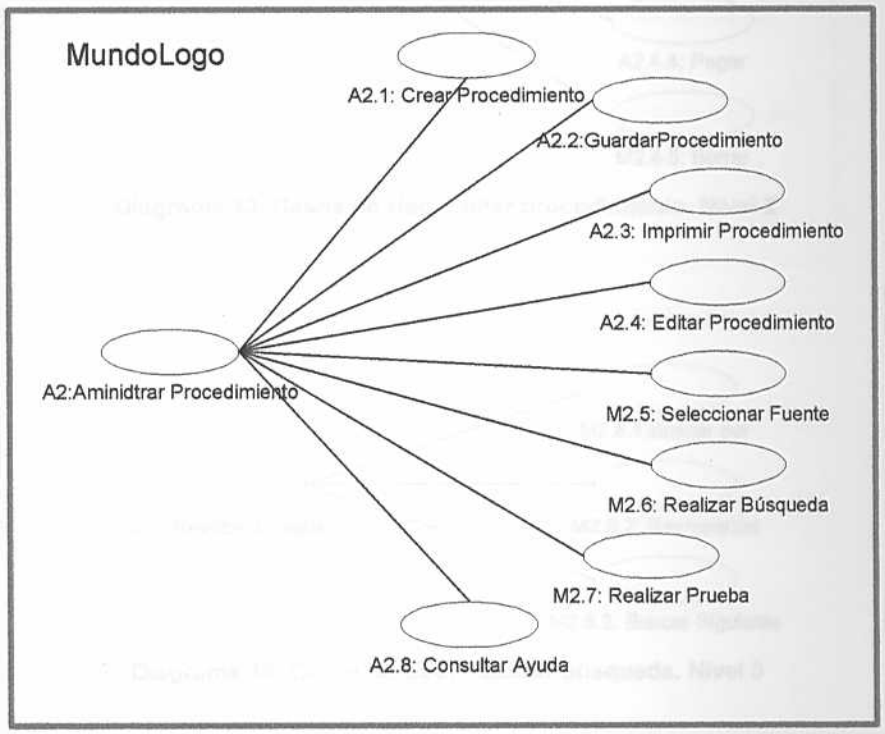


Diagrama 12: Casos de Uso, Administrar procedimiento. Nivel 2

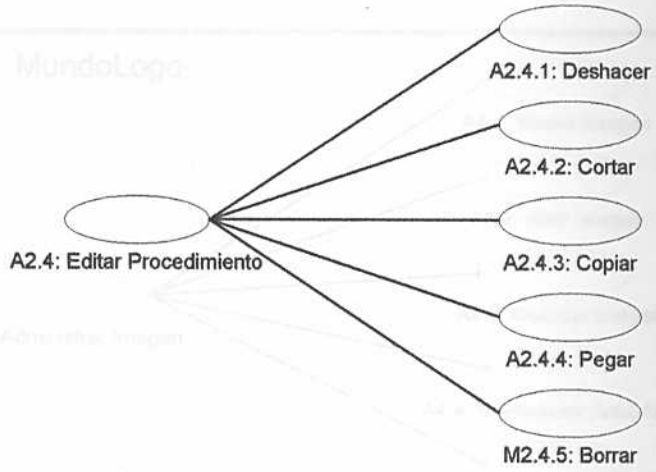


Diagrama 13: Casos de Uso, Editar procedimiento. Nivel 2

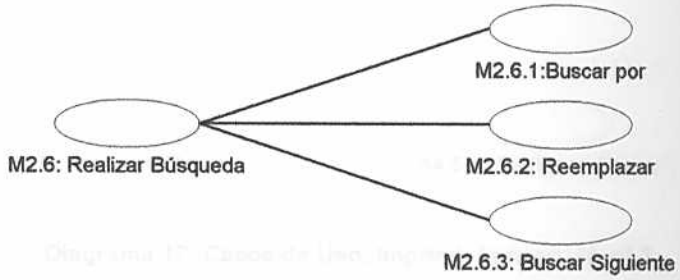


Diagrama 14: Casos de Uso, Realizar búsqueda. Nivel 3

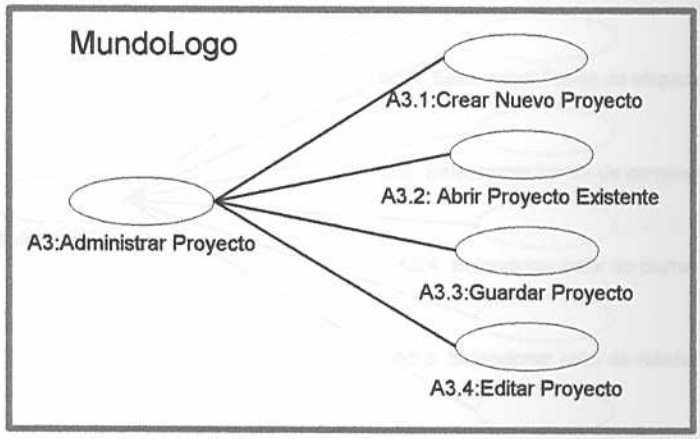


Diagrama 15: Casos de Uso, Administrar proyecto. Nivel 2

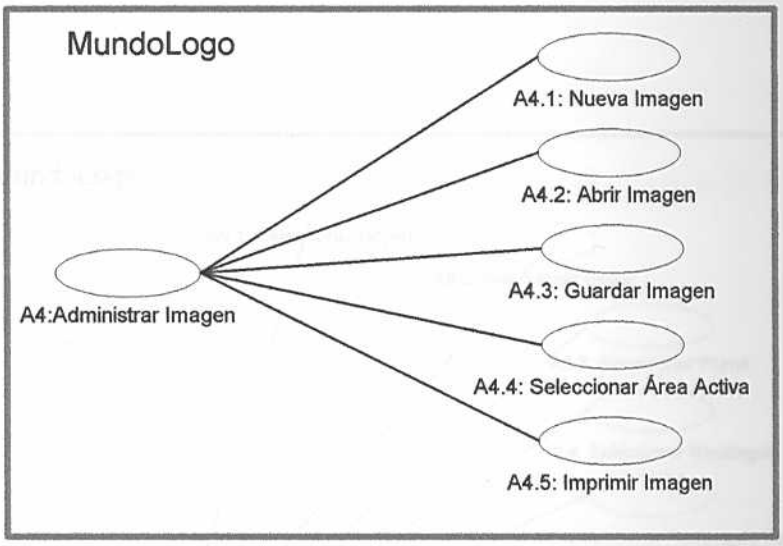


Diagrama 16: Casos de Uso, Administrar imagen. Nivel 2

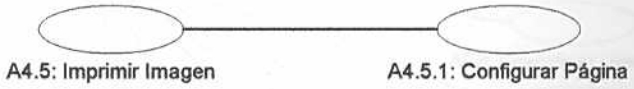


Diagrama 17: Casos de Uso, Imprimir imagen. Nivel 3

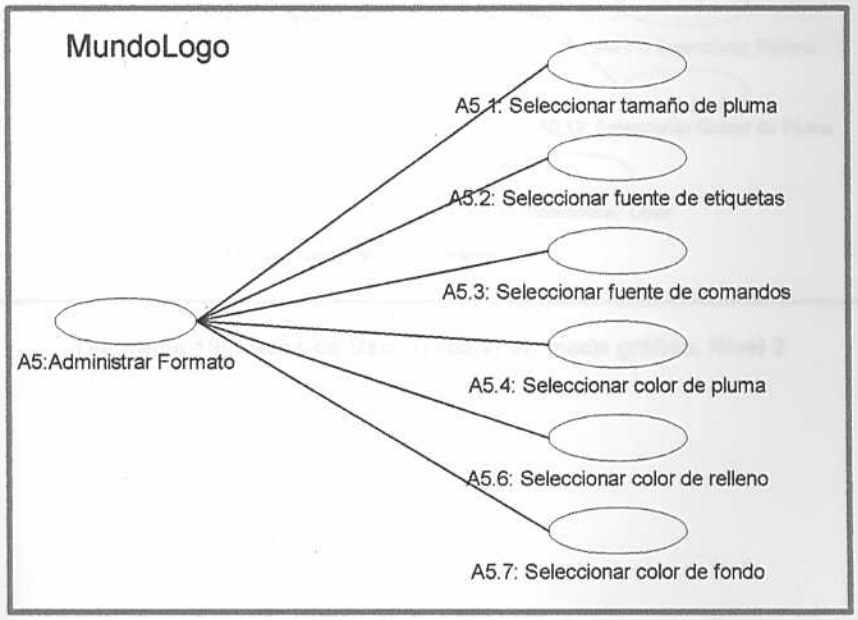


Diagrama 18: Casos de Uso, Administrar formato. Nivel 2

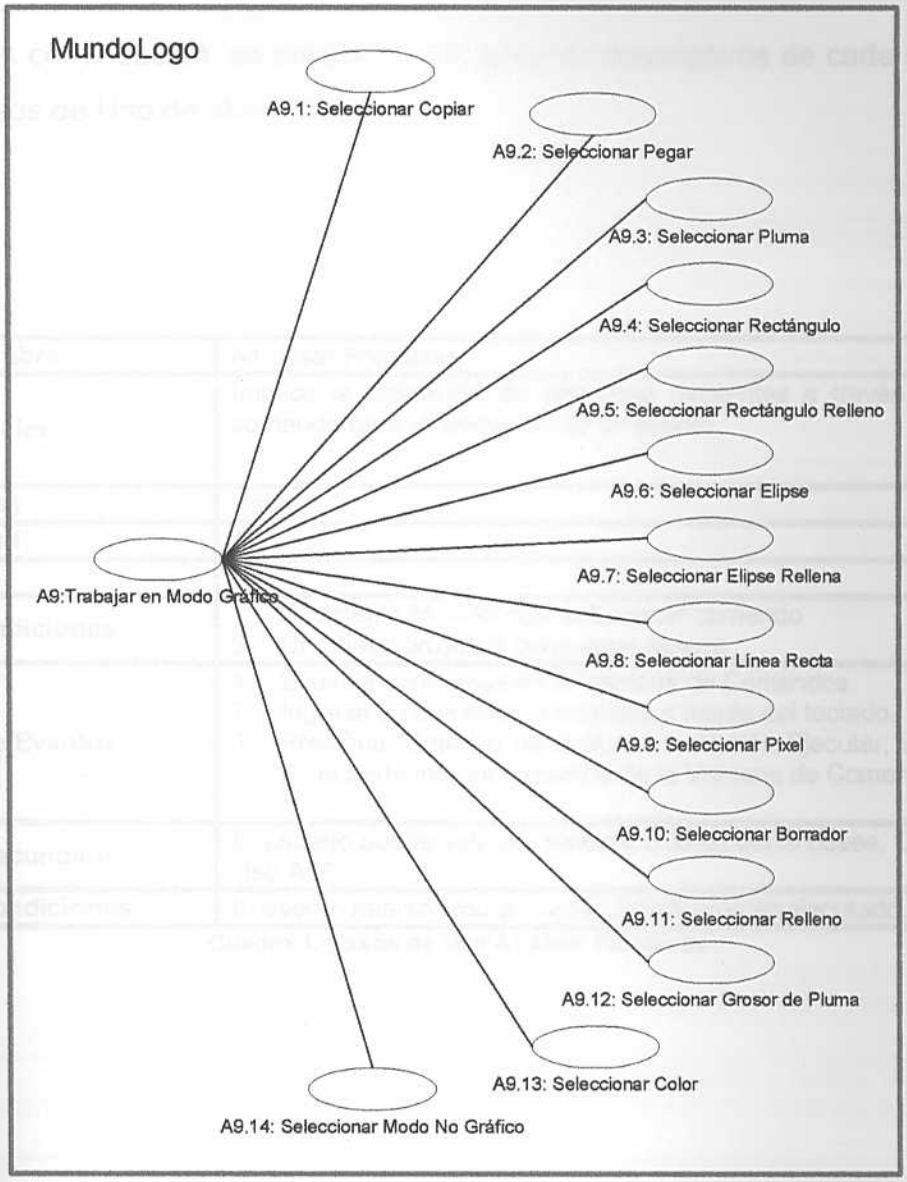


Diagrama 19: Casos de Uso, Trabajar en modo gráfico. Nivel 2

Anexo IV-11

Cuadros descriptivos de los Casos de Uso de MundoLogo

A continuación, se presentan los cuadros descriptivos de cada uno de los Casos de Uso de MundoLogo.

ID y Nombre	A1: Usar Primitivas
Descripción	Implica la utilización de primitivas existentes a través de un comando para la ejecución de un evento.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo 2. Un nuevo proyecto debe estar abierto
Flujo de Eventos	1. El usuario se ubica en la Ventana de Comandos. 2. Ingresa un comando ó primitiva a través del teclado. 3. Presiona "Enter" o hace click en el botón Ejecutar, ubicado en la parte inferior izquierda de la Ventana de Comandos.
Flujo Secundario	El usuario puede salir del sistema cuando así lo desee, Caso de Uso A16.
Post-Condiciones	El evento relacionado al mando introducido es ejecutado.

Cuadro 1: Casos de Uso A1:Usar Primitivas

ID y Nombre	A2: Administrar procedimientos
Descripción	Permite al usuario administrar todo lo relacionado con la creación y el desarrollo de procedimientos.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario presiona con el botón izquierdo del *mouse*, el botón Editar, ubicado en la parte inferior derecha de la Ventana de Comandos, y se despliega la ventana "Editor" de procedimientos (ó flujo alterno 1). 2. Si el usuario desea crear un nuevo procedimiento: Caso de Uso A2.1. 3. Si el usuario desea guardar un procedimiento: Caso de Uso A2.2. 4. Si el usuario desea imprimir el código de los procedimientos creados: Caso de Uso A2.3. 5. Si el usuario desea realizar alguna edición en un procedimiento: Caso de Uso A2.4. 6. Si el usuario desea modificar la fuente del código: Caso de Uso M2.5. 7. Si el usuario desea realizar una búsqueda en el código de los procedimientos: Caso de Uso M2.6. 8. Si el usuario desea realizar una prueba: Caso de Uso M2.7. 9. Si el usuario desea consultar la ayuda: Caso de Uso M2.8.
Flujo Alterno	1. El usuario selecciona con el *mouse* la opción Editar del Menú Archivo de la Ventana Principal y continúa el Caso de Uso A3.4.
Flujo Secundario	El usuario puede salir del sistema cuando así lo desee, Caso de Uso A16.
Post-Condiciones	La ventana "Editor" de procedimientos es desplegada.

Cuadro 2: Casos de Uso A2:Administrar Procedimientos

ID y Nombre	A2.1: Crear procedimientos
Descripción	Implica la utilización de primitivas existentes para crear un nuevo procedimiento, que podrá ser llamado por su nombre dentro del Ventana de Comandos.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario inicia la creación del nuevo procedimiento, colocándole un nombre al mismo. 2. Luego, el usuario introduce a través del teclado, las primitivas necesarias para crear el nuevo procedimiento que desea ejecutar. 3. A continuación, el procedimiento es guardado en el proyecto actual, Caso de Uso A2.2.
Flujo Alterno	
Flujo Secundario	El usuario puede cerrar la ventana de procedimientos en cualquier momento.
Post-Condiciones	Son creados nuevos procedimientos en el proyecto actual, los cuales ahora funcionarán como nuevas primitivas.

Cuadro 3: Casos de Uso A2.1:Crear Procedimientos

ID y Nombre	A2.2: Guardar procedimientos
Descripción	Permite al usuario guardar en disco, como parte del proyecto actual, los procedimientos que ha creado en la ventana Editor.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario despliega presionando con el botón izquierdo del *mouse* el Menú Archivo de la ventana Editor: 1.1. Si selecciona "Guardar proyecto" (ó flujo alterno 1): 1.1.1. El procedimiento será guardado y el usuario podrá continuar trabajando en el editor. 1.2. Si no, si selecciona "Guardar y Salir" (ó flujo alterno 2): 1.2.1. El procedimiento será guardado. 1.2.2. Se cerrará la ventana Editor desplegada.
Flujo Alterno	1. El usuario puede "Guardar proyecto", presionando las teclas Alt+A seguida de la tecla P. 2. El usuario puede "Guardar y Salir", presionando las teclas Alt+A seguida de la tecla G.
Flujo Secundario	El usuario puede cerrar la ventana de procedimientos sin guardar ningún cambio, en el momento que así lo desee.
Post-Condiciones	Son creados y guardados nuevos procedimientos que pueden ser ahora utilizados como nuevas primitivas del proyecto.

Cuadro 4: Casos de Uso A2.2:Guardar Procedimientos

ID y Nombre	A2.3: Imprimir Procedimientos
Descripción	Permite al usuario realizar una impresión en papel, del código de todos los procedimientos que se encuentran creados en la ventana "Editor".
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario selecciona la opción "Imprimir" del Menú Archivo de la Ventana Editor (ó flujo alterno 1). 2. Luego, configura las características de la impresión que realizará. 3. El usuario, presiona con el *mouse* el botón Aceptar, ubicado en la parte inferior izquierda de la ventana desplegada. Si no, flujo secundario 1.
Flujo Alterno	1. El usuario presiona las teclas Alt+A seguido por la tecla I.
Flujo Secundario	1. El usuario presiona con el *mouse* el botón Cancelar, ubicado en la parte inferior derecha de la ventana desplegada. Ésta se cierra y no se realiza la impresión.
Post-Condiciones	Los procedimientos del proyecto son impresos por el dispositivo de impresión seleccionado.

Cuadro 5: Casos de Uso A2.3:Imprimir Procedimientos

ID y Nombre	A2.4: Editar procedimientos
Descripción	Permite al usuario realizar cualquier modificación ó edición en los procedimientos que están en la ventana "Editor".
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario despliega el Menú Editar de la ventana "Editor" (ó flujo alterno 1). 2. Si el usuario desea deshacer la última modificación realizada en el código de algún procedimiento: Caso de Uso A2.4.1. 3. Si el usuario desea "cortar" parte del código de algún procedimiento: Caso de Uso A2.4.2. 4. Si el usuario desea "copiar" parte del código de algún procedimiento: Caso de Uso A2.4.3. 5. Si el usuario desea "pegar" parte de algún procedimiento que contenga en el portapapeles: Caso de Uso A2.4.4. 6. Si el usuario desea borrar parte del código de algún procedimiento: Caso de Uso A2.4.5.
Flujo Alterno	1. El usuario puede presionar directamente las teclas Alt+E, para desplegar el Menú de Edición.
Flujo Secundario	El usuario puede cerrar la ventana "Editor" en cualquier momento.
Post-Condiciones	El Menú Editar de la Ventana "Editor" es desplegado.

Cuadro 6: Casos de Uso A2.4:Editar Procedimientos

ID y Nombre	A2.4.1: Deshacer
Descripción	Permite al usuario deshacer el último cambio que realizó en el código de algún procedimiento.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada. 4. Debe haberse realizado una modificación en el código.
Flujo de Eventos	1. El usuario selecciona la opción "Deshacer" del Menú Editar de la Ventana Editor (ó flujo alterno 1 ó flujo alterno 2).
Flujo Alterno	1. El usuario presiona las teclas Alt+BkSp. 2. Una vez desplegado el menú Editar, el usuario presiona la tecla D.
Flujo Secundario	El usuario puede cerrar la ventana "Editor" en cualquier momento.
Post-Condiciones	Se deshace el último cambio realizado en el código.

Cuadro 7: Casos de Uso A2.4.1:Deshacer

ID y Nombre	A2.4.2: Cortar
Descripción	Permite al usuario "cortar" parte del código de algún procedimiento para luego ser pegado en otro lugar de la ventana Editor.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario selecciona con el *mouse* ó con ayuda del teclado la parte del texto de la ventana Editor que desea "cortar". 2. Luego, selecciona la opción "Cortar" del Menú Editar de la Ventana Editor (ó flujo alterno 1 ó flujo alterno 2 ó flujo alterno 3). 3. El texto cortado pasa al portapapeles y desaparece de la ventana Editor.
Flujo Alterno	1. El usuario presiona las teclas Shift+Sup. 2. El usuario presiona las teclas Ctrl+X. 3. Una vez desplegado el menú Editar, el usuario presiona la tecla C.
Flujo Secundario	El usuario puede cerrar la ventana "Editor" en cualquier momento.
Post-Condiciones	El texto cortado pasa al portapapeles de la aplicación.

Cuadro 8: Casos de Uso A2.4.2: Cortar

ID y Nombre	A2.4.3: Copiar
Descripción	Permite al usuario "copiar" parte del código de algún procedimiento para luego ser pegado en otro lugar de la ventana Editor.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario selecciona la parte del texto de la ventana Editor que desea "copiar". 2. Luego, selecciona la opción "Copiar" del Menú Editar de la Ventana Editor (ó flujo alterno 1 ó flujo alterno 2 ó flujo alterno 3). 3. El texto copiado pasa al portapapeles, pero no es borrado de la ventana Editor.
Flujo Alterno	1. El usuario presiona las teclas Ctrl+Ins. 2. El usuario presiona las teclas Ctrl+C. 3. Una vez desplegado el menú Editar, el usuario presiona la tecla O.
Flujo Secundario	El usuario puede cerrar la ventana "Editor" en cualquier momento.
Post-Condiciones	El texto copiado pasa al portapapeles de la aplicación.

Cuadro 9: Casos de Uso A2.4.3: Copiar

ID y Nombre	A2.4.4: Pegar
Descripción	Permite al usuario "pegar" en cualquier lugar de la ventana Editor, parte del código de algún procedimiento que haya sido previamente introducido (cortado ó copiado) en el portapapeles.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada. 4. El portapapeles debe contener datos, es decir, previamente debe haberse cortado ó copiado algún trozo de texto.
Flujo de Eventos	1. El usuario coloca el puntero en el lugar de la ventana Editor donde desea pegar la información. 2. Luego, selecciona la opción "Pegar" del Menú Editar de la Ventana Editor (ó flujo alterno 1 ó flujo alterno 2). 3. El texto es pegado en la ventana Editor.
Flujo Alterno	1. El usuario presiona las teclas Shift+Ins. 2. El usuario presiona las teclas Ctrl+V. 3. Una vez desplegado el menú Editar, el usuario presiona la tecla P.
Flujo Secundario	El usuario puede cerrar la ventana "Editor" en cualquier momento.
Post-Condiciones	El texto contenido en el portapapeles es pegado en la ventana Editor.

Cuadro 10: Casos de Uso A2.4.4: Pegar

ID y Nombre	A2.4.5: Borrar
Descripción	Permite al usuario borrar parte del código de algún procedimiento en la Ventana Editor.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. Si el usuario desea borrar sólo un trozo del código de algún procedimiento: 1.1. Selecciona el segmento de código que desea borrar. 1.2. A continuación, selecciona la opción "Borrar" del Menú Editar de la ventana Editor (o flujo alterno 1 ó flujo alterno 2). 2. Si el usuario desea limpiar por completo la pantalla de la ventana Editor: 2.1. Selecciona la opción "Borrar todo" del Menú Editar de la Ventana Editor (ó flujo alterno 3 ó flujo alterno 4).
Flujo Alterno	1. El usuario presiona la tecla Sup. 2. Una vez desplegado el menú Editar, el usuario presiona la tecla B. 3. El usuario presiona las teclas Ctrl+Sup. 4. Una vez desplegado el menú Editar, el usuario presiona la tecla O, seguida nuevamente de la tecla O.
Flujo Secundario	El usuario puede cerrar la ventana "Editor" en cualquier momento.
Post-Condiciones	En el caso 1, el texto seleccionado es borrado del editor. En el caso 2, toda la pantalla del editor es limpiada.

Cuadro 11: Casos de Uso A2.4.5: Borrar

ID y Nombre	M2.5: Seleccionar fuente
Descripción	Permite al usuario modificar el estilo de la fuente con la que está trabajando en el editor de procedimientos.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario selecciona la opción "Fuente", en el Menú Ambiente de la Ventana Editor (ó flujo alterno 1). 2. Se despliega la ventana de Fuente. 3. El usuario puede seleccionar, presionado con el botón izquierdo del *mouse*, la combinación que prefiera, entre: El tipo de fuente. El estilo de la fuente. El tamaño de la fuente. 4. El usuario presiona con el *mouse* el botón Aceptar, ubicado en la parte superior derecha de la ventana desplegada para aceptar los cambios, si no flujo secundario 1.
Flujo Alterno	1. El usuario presiona las teclas Alt+M seguida de la tecla F.
Flujo Secundario	1. El usuario presiona con el *mouse* el botón Cancelar, ubicado en la parte derecha de la ventana desplegada. Esta se cerrará sin realizar ninguna modificación de la fuente.
Post-Condiciones	Se modifica el estilo de la fuente utilizada en el editor.

Cuadro 12: Casos de Uso M2.5: Seleccionar Fuente

ID y Nombre	M2.6: Realizar búsqueda
Descripción	Permite al usuario realizar búsquedas y reemplazar palabras ó frases dentro del código de los procedimientos en la ventana Editor.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario despliega presionando con el *mouse* el menú Búsqueda de la ventana Editor (ó flujo alterno 1). 2. Si el usuario desea buscar una palabra ó frase dentro del código: Caso de Uso M2.6.1. 3. Si el usuario desea remplazar una palabra ó frase en el código por otra palabra ó frase: Caso de Uso M2.6.2. 4. Si el usuario desea continuar una búsqueda ya iniciada: Caso de Uso M2.6.3.
Flujo Alterno	1. El usuario puede presionar directamente las teclas Alt+B, para desplegar el Menú de Búsqueda.
Flujo Secundario	El usuario puede cerrar la ventana "Editor" en cualquier momento.
Post-Condiciones	El Menú Búsqueda de la Ventana "Editor" es desplegada.

Cuadro 13: Casos de Uso M2.6: Realizar Búsqueda

ID y Nombre	M2.6.1: Buscar por
Descripción	Permite al usuario buscar una palabra ó frase, dentro del código de los procedimientos.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario selecciona la opción "Buscar", en el Menú Búsqueda de la Ventana Editor (ó flujo alterno 1). 2. Se despliega la ventana Buscar. 3. El usuario introduce la palabra ó frase a buscar en el texto. 4. El usuario puede seleccionar la opción "coincidir mayúsculas y minúsculas", si así lo desea. 5. Para finalizar el usuario presiona con el *mouse* el botón Aceptar, ubicado al lado izquierdo de la ventana Buscar, para realizar la búsqueda, en caso contrario, flujo secundario 1.
Flujo Alterno	1. Una vez desplegado el Menú Búsqueda, el usuario presiona la tecla B.
Flujo Secundario	1. El usuario puede cerrar en cualquier momento la ventana Buscar sin realizar ninguna búsqueda, presionando con el *mouse* el botón Cancelar, ubicado a la derecha de dicha ventana desplegada.
Post-Condiciones	La palabra ó frase introducida por el usuario es buscada en el código.

Cuadro 14: Casos de Uso M2.6.1: Búscar por

ID y Nombre	M2.6.2: Reemplazar
Descripción	Permite al usuario reemplazar una palabra ó frase del código por otra palabra ó frase introducida por el usuario.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario selecciona la opción "Reemplazar", en el Menú Búsqueda de la Ventana Editor (ó flujo alterno 1). 2. Se despliega la ventana Reemplazar. 3. El usuario introduce la palabra ó frase que se encuentra en el código y que desea reemplazar. 4. Luego introduce la nueva palabra o frase con la que va a realizar el reemplazo en el código. 5. El usuario puede seleccionar las opciones "coincidir mayúsculas y minúsculas", "todas las ocurrencias" y "Prompt on replace", si así lo desea. 6. Para finalizar el usuario presiona con el *mouse* el botón Aceptar, ubicado al lado izquierdo de la ventana Reemplazar, y se realiza el reemplazo. En caso contrario, flujo secundario 1.
Flujo Alterno	1. Una vez desplegado el Menú Búsqueda, el usuario presiona la tecla R.
Flujo Secundario	1. El usuario puede cerrar en cualquier momento la ventana Reemplazar, presionando con el *mouse* el botón Cancelar, ubicado a la derecha de dicha ventana.
Post-Condiciones	La palabra ó frase buscada, será reemplazada por la nueva palabra ó frase introducida por el usuario.

Cuadro 15: Casos de Uso M2.6.2: Reemplazar

ID y Nombre	M2.6.3: Buscar siguiente
Descripción	Permite al usuario continuar una búsqueda anterior de una palabra ó frase, en el código de procedimientos.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario selecciona la opción "Siguiente", en el Menú Búsqueda de la Ventana Editor (ó flujo alterno 1 ó flujo alterno 2).
Flujo Alterno	1. El usuario puede presionar directamente la tecla F3, para continuar la búsqueda. 2. Una vez desplegado el Menú Búsqueda, el usuario presiona la tecla S.
Flujo Secundario	El usuario puede cerrar la ventana Editor en cualquier momento.
Post-Condiciones	Se continúa con la búsqueda anterior.

Cuadro 16: Casos de Uso M2.6.3: Buscar Siguiente

ID y Nombre	M2.7: Realizar prueba
Descripción	Permite al usuario realizar la prueba de un segmento de código, en el editor, previamente seleccionada, sin necesidad de cerrar la ventana Editor, ni de guardar lo procedimientos.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada. 4. Debe existir código en el editor de texto.
Flujo de Eventos	1. El usuario selecciona un segmento de código, en el editor de procedimientos. 2. A continuación, el usuario selecciona con el *mouse* la opción Prueba! de la ventana Editor (ó flujo alterno 1).
Flujo Alterno	1. El usuario presiona las teclas Alt+P.
Flujo Secundario	El usuario puede cerrar la ventana Editor en cualquier momento.
Post-Condiciones	El sistema ejecuta cada línea, del segmento de código seleccionado en el editor.

Cuadro 17: Casos de Uso M2.7: Realizar prueba

ID y Nombre	M2.8: Consultar ayuda
Descripción	Permite al usuario consultar la ayuda, que el sistema ofrece específicamente para el trabajo en el editor de procedimientos.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. La ventana "Editor" de procedimientos debe estar desplegada.
Flujo de Eventos	1. El usuario despliega el Menú Ayuda de la ventana Editor (ó flujo alterno 1). 2. Si el usuario desea consultar la ayuda por el índice, selecciona con el *mouse* la opción Índice de éste menú (ó flujo alterno 2). 3. Si el usuario desea consultar la ayuda para el manejo del Editor, selecciona con el *mouse* la opción Editor, de éste menú (ó flujo alterno 3). 4. Si el usuario desea consultar por un tópico especifico, selecciona con el *mouse* la opción Búsqueda por Tópicos, de éste menú (ó flujo alterno 4 ó flujo alterno 5).
Flujo Alterno	1. El usuario presiona las teclas Alt+?. 2. Una vez desplegado el Menú de Ayuda, el presiona la tecla I. 3. Una vez desplegado el Menú de Ayuda, el presiona la tecla E. 4. Una vez desplegado el Menú de Ayuda, el presiona la tecla B. 5. El usuario también puede consultar directamente la ayuda por tópicos, presionando la tecla F1.
Flujo Secundario	El usuario puede cerrar la ventana Editor en cualquier momento.
Post-Condiciones	Se despliega la ayuda para ser consultada.

Cuadro 18: Casos de Uso M2.8: Consultar Ayuda

ID y Nombre	A3: Administrar Proyecto
Descripción	Permite al usuario administrar todo lo relacionado con la creación, el desarrollo y el trabajo con proyectos, en MundoLogo.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo.
Flujo de Eventos	1. El usuario despliega con el *mouse*, el Menú Archivo de la Ventana Principal de MundoLogo (ó flujo alterno 1). 2. Si el usuario desea crear un nuevo proyecto: Caso de Uso A3.1. 3. Si el usuario desea abrir un proyecto existente: Caso de Uso A3.2. 4. Si el usuario desea guardar en disco un nuevo proyecto: Caso de Uso A3.3. 5. Si el usuario desea editar los procedimientos creados y contenidos en el proyecto actual: Caso de Uso A3.4.
Flujo Alterno	1. El usuario presiona las teclas Alt+A.
Flujo Secundario	En cualquier momento el usuario puede salir del sistema, Caso de Uso A16.
Post-Condiciones	El Menú Archivo es desplegado, para administrar los proyectos.

Cuadro 19: Casos de Uso A3: Administar Proyecto

ID y Nombre	A3.1: Crear nuevo Proyecto
Descripción	Permite al usuario, crear un nuevo proyecto de MundoLogo, donde comenzará a trabajar.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo.
Flujo de Eventos	1. El usuario selecciona la opción Nuevo del Menú Archivo (ó flujo alterno). 2. El sistema despliega una ventana preguntando al usuario, si efectivamente desea iniciar un nuevo proyecto de MundoLogo, sin guardar el proyecto que venía trabajando. 3. Si el usuario no desea guardar el ultimo proyecto que estaba trabajando, presiona con el botón izquierdo del *mouse* la opción Aceptar, ubicado en la parte inferior izquierda de la ventana desplegada. El nuevo proyecto es creado. En caso contrario, flujo secundario 1.
Flujo Alterno	1. Una vez desplegado el Menú Archivo, el usuario presiona la tecla N.
Flujo Secundario	1. Si el usuario desea guardar el proyecto con que estaba trabajando, antes de crear el nuevo proyecto presiona con el botón izquierdo del *mouse* la opción Cancelar, ubicado en la parte inferior derecha de la ventana desplegada, y continua con el Caso de Uso A3.3. 2. El usuario puede salir del sistema cuando lo desee, Caso de Uso A16.
Post-Condiciones	Se crea un nuevo proyecto desde el inicio, es decir sin procedimientos almacenados y con la pantalla limpia.

Cuadro 20: Casos de Uso A3: Crear Nuevo Proyecto

ID y Nombre	A3.2: Abrir un Proyecto
Descripción	Permite al usuario, abrir un proyecto de MundoLogo que se encuentra guardado en el disco.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. El proyecto que se desea abrir, debe existir.
Flujo de Eventos	1. El usuario selecciona la opción Abrir del Menú Archivo (ó flujo alterno 1). 2. Si el usuario estaba trabajando con algún proyecto, se desplegará un ventana informándole que la acción que está realizando va a borrar el proyecto que está trabajando, si no ha guardado previamente los cambios. 3. Si el usuario no desea guardar los cambios, presiona con el botón izquierdo del *mouse*, el botón Aceptar ubicado en la parte inferior izquierda de la ventana desplegada y ésta se cerrará. En caso contrario, flujo secundario 1. 4. A continuación se desplegará una nueva ventana "Abrir", donde el usuario podrá buscar en el disco y seleccionar, el proyecto que desea abrir. 5. El usuario presiona con el *mouse* el botón Abrir, ubicado en la parte inferior derecha de la ventana desplegada. En caso contrario, flujo secundario 2.
Flujo Alterno	1. Una vez desplegado el Menú Archivo, el usuario presiona la tecla A.
Flujo Secundario	1. Si el usuario desea guardar los cambios del proyecto actual, presiona con el *mouse* el botón Cancelar, ubicado en la parte inferior derecha de la ventana y continua el Caso de Uso A3.3. 2. Una vez desplegada la ventana "Abrir", el usuario puede salir de ésta sin abrir ningún proyecto, presionando con el *mouse* el botón Cancelar ubicado en la parte inferior derecha de la ventana, la cual se cerrará. 3. El usuario puede salir del sistema cuando lo desee, Caso de Uso A16.
Post-Condiciones	Es cargado el proyecto seleccionado.

Cuadro 21: Casos de Uso A3.2: Abrir un Proyecto

ID y Nombre	A3.3: Guardar Proyecto
Descripción	Permite al usuario guardar en disco, el proyecto actual, incluyendo los procedimientos creados dentro de este proyecto.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. Si el usuario desea guardar el proyecto actual en el disco: 1.1. Selecciona con el *mouse* la opción Guardar en el Menú Archivo (ó flujo alterno 1). 1.2. El proyecto actual guarda automáticamente todos los cambios en el disco. Si el proyecto no ha sido guardado anteriormente, flujo secundario 1. 2. Si el usuario desea guardar el proyecto actual en el disco pero con un nuevo nombre: 2.1. Selecciona con el *mouse*, la opción "Guardar como" del Menú Archivo (ó flujo alterno 2). 2.2. Se desplegará una ventana donde el usuario puede seleccionar el lugar en disco donde desea guardar su proyecto. 2.3. El usuario introduce el nombre del proyecto a guardar y luego presiona con el *mouse* el botón Aceptar, ubicado en la parte inferior derecha de la ventana desplegada. Si no, flujo secundario 2. 2.4. El proyecto es guardado y la ventana desplegada se cierra. El usuario podrá continuar trabajando en el proyecto.
Flujo Alterno	1. Una vez desplegado el Menú Archivo, el usuario presiona la tecla G. 2. Una vez desplegado el Menú Archivo, el usuario presiona la tecla C.
Flujo Secundario	1. Se despliega la ventana de "Guardar como", donde el usuario seleccionará el lugar en el disco y el nombre, con el cual guardará el proyecto. 2. Una vez desplegada la ventana "Guardar como", el usuario puede salir de ésta sin guardar el proyecto, presionando con el *mouse* el botón Cancelar ubicado en la parte inferior derecha de la ventana, la cual se cerrará. 3. El usuario puede salir del sistema en cualquier momento.
Post-Condiciones	El nuevo proyecto es guardado en el disco.

Cuadro 22: Casos de Uso A3.3: Guardar Proyecto

ID y Nombre	A3.4: Editar Proyecto
Descripción	Permite al usuario editar uno ó mas procedimientos pertenecientes al proyecto actual.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. Debe existir algún procedimiento creado en el proyecto actual.
Flujo de Eventos	1. El usuario despliega con el *mouse* el menú Archivo y selecciona la opción Editar (ó flujo alterno 1). 2. Se despliega la ventana Editar, donde el usuario selecciona el nombre del procedimiento que desea editar. A continuación, el usuario presiona con el *mouse* el botón Aceptar, ubicado en la parte inferior izquierda de la ventana. Si no, flujo secundario1. 3. Si el usuario desea editar varios ó todos los procedimientos del proyecto, entonces presiona con el *mouse* el botón Todo, ubicado en la parte inferior central de la ventana.
Flujo Alterno	1. Una vez desplegado el menú Archivo, el usuario presiona la tecla E.
Flujo Secundario	1. Si el usuario desea cerrar la ventana sin editar ningún procedimiento, presiona con el *mouse* el botón Cancelar, ubicado en la parte inferior derecha de la ventana. 2. En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	La ventana Editor se abre, con el ó los procedimientos seleccionados.

Cuadro 23: Casos de Uso A3.4: Editar Proyecto

ID y Nombre	A4: Administrar imagen
Descripción	Corresponde a todos los procesos asociados al control y manejo de imágenes.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar ejecutándose. 2. Un proyecto debe estar abierto.
Flujo de Eventos	1. El usuario se ubica en el menú superior de la pantalla. 2. Hace click con el botón derecho del mouse en la opción "Imagen".
Flujo Alterno	1. El usuario presiona la tecla "Alt". 2. Presiona la tecla "I" del teclado.
Post-Condiciones	Se despliega el sub-Menú de la opción Imagen.

Cuadro 24: Casos de Uso A4.1: Nueva imagen

ID y Nombre	A4.1: Nueva imagen
Descripción	Limpia la pantalla para colocar una nueva imagen como fondo del área de trabajo.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A4
Flujo de Eventos	1. El usuario se ubica en la opción "Nuevo" del sub-Menú 2. Hace click con el botón derecho del mouse.
Flujo Alterno	1. El usuario presiona la tecla "Alt". 2. Presiona la tecla "N" del teclado.
Post-Condiciones	El fondo el fondo del área de trabajo se borra.

Cuadro 25: Casos de Uso A4.1: Nueva Imagen

ID y Nombre	A4.2: Abrir imagen
Descripción	Abre una imagen existente y la coloca como fondo del proyecto, en la posición de la tortuga.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A4
Flujo de Eventos	1. El usuario hace click en la opción "Abrir" del sub-Menú "Imagen". 2. Se despliega la ventana "Abrir" de Windows. 2.1. El usuario busca y selecciona el archivo el archivo BMP o GIF que desea incluir en el proyecto. 2.2. Hace click el en botón "Aceptar" de la sub-ventana. 3. Se cierra la ventana "Abrir" de Windows.
Escenarios secundarios	1. El usuario hace click en el botón "cancelar" de la ventana "Abrir" de Windows. 2. Se cierra la ventana "Abrir", termina el caso de uso
Post-Condiciones	Se agrega la imagen seleccionada en el área de trabajo.

Cuadro 26: Casos de Uso A4.2: Abrir Imagen

ID y Nombre	A4.3: Guardar imagen
Descripción	Guarda el fondo del área de trabajo como imagen BMP o GIF.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A4.
Flujo de Eventos	1. El usuario hace click en la opción "Guardar" del sub-Menú "Imagen". 2. Si el usuario no ha guardado anteriormente la imagen: 2.1. Se despliega la ventana "Guardar" de Windows. 2.2. El usuario busca y se ubica en el directorio donde desea guardar la imagen. 2.3. Escribe el nombre de la imagen. 2.4. Selecciona el tipo de archivo a guardar (BMP o GIF). 2.5. Hace click el en botón "Aceptar" de la sub-ventana. 2.6. Se cierra la ventana "Guardar" de Windows. 3. Si no, el usuario ya había guardado la imagen con anterioridad: 3.1. El sistema guarda los cambios realizados a la imagen.
Escenarios secundarios	1. El usuario hace click en la opción "Guardar Como" del sub-Menú "Imagen". 1.1. Se despliega la ventana "Guardar" de Windows. 1.2. El usuario busca y se ubica en el directorio donde desea guardar la imagen. 1.3. Escribe el nombre de la imagen. 1.4. Selecciona el tipo de archivo a guardar (BMP o GIF). 1.5. Hace click el en botón "Aceptar" de la sub-ventana. 1.6. Se cierra la ventana "Guardar" de Windows. 2. El usuario hace click en el botón "Cancelar" de la ventana "Guardar" de Windows. 2.1. Se cierra la ventana, el caso de uso termina.
Post-Condiciones	La imagen de fondo del área de trabajo queda guardada en el directorio correspondiente.

Cuadro 27: Casos de Uso A4.3: Guardar Imagen

ID y Nombre	A4.4: Seleccionar área activa
Descripción	Controla las dimensiones del área de trabajo.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A4
Flujo de Eventos	1. El usuario hace click en la opción "Área Activa" del sub-Menú "Imagen". 2. Se despliega la ventana "Seleccionar área activa". 2.1. El usuario modifica los valores correspondientes al área activa. 2.2. Hace click el en botón "Aceptar" de la sub-ventana. 3. Se cierra la ventana "Seleccionar área activa".
Escenarios secundarios	1. Si el usuario hace click en el botón "Reiniciar" de la ventana "Seleccionar área activa" 1.1. Los valores del área de trabajo: X, Y y pasos de la tortuga por pulgada, toman sus valores por defecto. 2. Si el usuario hace click en el botón "cancelar" de la ventana "Seleccionar área activa" 2.1. Se cierra la Ventana "Seleccionar área activa", se termina el caso de uso.
Post-Condiciones	El área de trabajo adquiere las dimensiones determinadas.

Cuadro 28: Casos de Uso A4.4: Seleccionar Área Activa

ID y Nombre	A4.5: Imprimir Imagen
Descripción	Imprime el fondo del área de trabajo.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A4
Flujo de Eventos	1. El usuario hace click en la opción "Imprimir" del sub-Menú "Imagen". 2. Se despliega la ventana "Imprimir" de Windows. 2.1. El usuario selecciona la impresora a utilizar. 2.2. Determina el intérvalo de impresión. 2.3. Selecciona el número de copias a imprimir 2.4. Hace click el en botón "Aceptar" de la sub-ventana. 3. Se despliega una ventana que muestra la condición de la impresora.
Escenarios secundarios	1. Si el usuario hace clik en el botón "cancelar" de la ventana "Imprimir" 1.1. Se cierra la Ventana "Imprimir", se termina el caso de uso.
Post-Condiciones	El fondo del área de trabajo es impreso

Cuadro 29: Casos de Uso A4.5: Imprimir Imagen

ID y Nombre	A4.5.1: Configurar Página
Descripción	Especifica los valores necesarios para configurar la página a imprimir.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A4.5
Flujo de Eventos	1. El usuario hace click en la opción "Configurar página" del sub-Menú "Imagen". 2. Se despliega la ventana "Configurar página" de Windows. 2.1. El usuario selecciona tamaño y origen del papel. 2.2. Determina la orientación de la página. 2.3. Valores de los márgenes derecho, superior, izquierdo e inferior. 2.4. Hace click el en botón "Aceptar" de la sub-ventana. 3. Se cierra la ventana, termina el caso de uso.
Escenarios secundarios	1. Si el usuario hace clik en el botón "Impresora" de la ventana "Configurar Página" 1.1. Se despliega una nueva sub-ventana de "Configurar Página" 1.2. El usuario selecciona la impresora a utilizar. 1.3. Si el usuario presiona el botón "Red" de la ventana. 1.3.1 Se despliega una ventana donde puede seleccionar entre las distintas computadoras conectadas a la red. 1.4. El usuario hace click en el botón "Aceptar" de la ventana. 2. Si el usuario hace clik en el botón "cancelar" de la ventana "Configurar Página" 1.2. Se cierra la Ventana "Configurar Página", se termina el caso de uso.
Post-Condiciones	La página a imprimir queda configurada de acuerdo a las condiciones señaladas.

Cuadro 30: Casos de Uso A4.5.1: Configurar Página

ID y Nombre	A5: Administrar Formato
Descripción	Corresponde a todos los procesos asociados al control y manejo del formato de etiquetas y dibujo.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar ejecutándose. 2. Un proyecto debe estar abierto.
Flujo de Eventos	1. El usuario se ubica en el menú superior de la pantalla. 2. Hace click con el botón derecho del mouse en la opción "Formato".
Flujo Alterno	1. El usuario presiona la tecla "Alt". 2. Presiona la tecla "F" del teclado.
Post-Condiciones	Se despliega el sub-Menú de la opción Formato.

Cuadro 31: Casos de Uso A5: Administrar Formato

ID y Nombre	A5.1: Seleccionar tamaño de pluma
Descripción	Determina el tamaño de la pluma de la tortuga.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A5
Flujo de Eventos	1. El usuario hace click en la opción "Tamaño de la Pluma" del sub-Menú "Formato". 2. Se despliega la ventana "Tamaño de la Pluma". 3. Si el usuario hace click con el botón derecho del mouse, sobre alguno de los recuadros de tamaño de la pluma. 3.1. El recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra la imagen seleccionada. 4. Si el usuario hace click y arrastra el mouse sobre la barra de scroll "Tamaño" de la ventana. 4.1. En el recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra el tamaño seleccionado, que puede variar en un rango de 32 tamaños. 5. El usuario hace click sobre el botón "OK" de la ventana. 6. Se cierra la ventana "Tamaño de la Pluma".
Escenarios secundarios	1. El usuario hace click en el botón "Cancelar" de la ventana "Tamaño de la Pluma". 2. Se cierra la ventana "Tamaño de la Pluma", termina el caso de uso.
Post-Condiciones	Se modifica el tamaño de la pluma de la tortuga.

Cuadro 32: Casos de Uso A5.1: Seleccionar tamaño de pluma

ID y Nombre	A5.2: Seleccionar Fuente de Etiquetas
Descripción	Determina la fuente a utilizar por las etiquetas, que serán creadas.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A5
Flujo de Eventos	1. El usuario hace click en la opción "Fuente de Etiquetas" del sub-Menú "Formato". 2. Se despliega la ventana "Fuente de Etiquetas". 2.1. El usuario selecciona el Nombre de la fuente de la lista, el estilo de la fuente, tamaño y el alfabeto a utilizar. 2.2. Hace click en el botón "Aceptar" de la ventana. 3. Se cierra la ventana "Fuente de Etiquetas".
Escenarios secundarios	1. El usuario hace click en el botón "Cancelar" de la ventana "Fuente de Etiquetas". 2. Se cierra la ventana "Fuente de Etiquetas", el caso de uso termina.
Post-Condiciones	Se modifica el tamaño de la pluma de la tortuga.

Cuadro 33: Casos de Uso A5.2: Seleccionar Fuentes de Etiquetas

ID y Nombre	A5.3: Seleccionar Fuente de Comandos
Descripción	Determina la fuente que será utilizada en la ventana de comandos.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A5
Flujo de Eventos	1. El usuario hace click en la opción "Fuente de Comandos" del sub-Menú "Formato". 2. Se despliega la ventana "Fuente de Comandos". 2.1. El usuario selecciona el Nombre de la fuente de la lista, el estilo de la fuente, tamaño y el alfabeto a utilizar. 2.2. Hace click en el botón "Aceptar" de la ventana. 3. Se cierra la ventana "Fuente de Comandos".
Escenarios secundarios	1. El usuario hace click en el botón "Cancelar" de la ventana "Fuente de Comandos". 2. Se cierra la ventana "Fuente de Comandos", el caso de uso termina.
Post-Condiciones	Se modifica el tamaño de la pluma de la tortuga.

Cuadro 34: Casos de Uso A5.3: Seleccionar Fuentes de Comandos

ID y Nombre	A5.4: Seleccionar color de la pluma
Descripción	Determina el color de la pluma de la tortuga.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A5
Flujo de Eventos	1. El usuario hace click en la opción "Color de la Pluma" del sub-Menú "Formato". 2. Se despliega la ventana "Color de la Pluma". 3. Si el usuario hace click con el botón derecho del mouse, sobre alguno de los recuadros de color. 3.1. El recuadro indicador del color de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado. 4. Si el usuario hace click y arrastra el mouse sobre la barra de scroll "Rojo" de la ventana. 4.1. En el recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado, que puede variar en un rango de 256 tonos de rojo. 5. Si el usuario hace click y arrastra el mouse sobre la barra de scroll "Verde" de la ventana. 5.1. En el recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado, que puede variar en un rango de 256 tonos de verde. 6. Si el usuario hace click y arrastra el mouse sobre la barra de scroll "Azul" de la ventana. 6.1. En el recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado, que puede variar en un rango de 256 tonos de azul. 7. El usuario hace click sobre el botón "OK" de la ventana. 8. Se cierra la ventana "Color de la Pluma".
Escenarios secundarios	1. El usuario hace click en el botón "Cancelar" de la ventana "Color de la Pluma". 2. Se cierra la ventana "Color de la Pluma", el caso de uso termina.
Post-Condiciones	Se modifica el color de la pluma de la tortuga.

Cuadro 35: Casos de Uso A5.4: Seleccionar color de la pluma

ID y Nombre	A5.5: Seleccionar color de relleno
Descripción	Determina el color de relleno de la tortuga.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A5
Flujo de Eventos	1. El usuario hace click en la opción "Color de Relleno" del sub-Menú "Formato". 2. Se despliega la ventana "Color de Relleno". 3. Si el usuario hace click con el botón derecho del mouse, sobre alguno de los recuadros de color. 3.1. El recuadro indicador del color de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado. 4. Si el usuario hace click y arrastra el mouse sobre la barra de scroll "Rojo" de la ventana. 4.1. En el recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado, que puede variar en un rango de 256 tonos de rojo. 5. Si el usuario hace click y arrastra el mouse sobre la barra de scroll "Verde" de la ventana. 5.1. En el recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado, que puede variar en un rango de 256 tonos de verde. 6. Si el usuario hace click y arrastra el mouse sobre la barra de scroll "Azul" de la ventana. 6.1. En el recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado, que puede variar en un rango de 256 tonos de azul. 7. El usuario hace click sobre el botón "OK" de la ventana. 8. Se cierra la ventana "Color de Relleno".
Escenarios secundarios	1. El usuario hace click en el botón "Cancelar" de la ventana "Color de Relleno". 2. Se cierra la ventana "Color de Relleno", el caso de uso termina.
Post-Condiciones	Se modifica el color de relleno de la tortuga.

Cuadro 36: Casos de Uso A5.5: Seleccionar color de relleno

ID y Nombre	A5.6: Seleccionar color de fondo
Descripción	Determina el color de fondo de la tortuga.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. Caso de Uso A5
Flujo de Eventos	1. El usuario hace click en la opción "Color de Fondo" del sub-Menú "Formato". 2. Se despliega la ventana "Color de Fondo". 3. Si el usuario hace click con el botón derecho del mouse, sobre alguno de los recuadros de color. 3.1. El recuadro indicador del color de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado. 4. Si el usuario hace click y arrastra el mouse sobre la barra de scroll "Rojo" de la ventana. 4.1. En el recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado, que puede variar en un rango de 256 tonos de rojo. 5. Si el usuario hace click y arrastra el mouse sobre la barra de scroll "Verde" de la ventana. 5.1. En el recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado, que puede variar en un rango de 256 tonos de verde. 6. Si el usuario hace click y arrastra el mouse sobre la barra de scroll "Azul" de la ventana. 6.1. En el recuadro indicador del tamaño de la pluma, ubicado en la parte superior derecha de la ventana, muestra el color seleccionado, que puede variar en un rango de 256 tonos de azul. 7. El usuario hace click sobre el botón "OK" de la ventana. 8. Se cierra la ventana "Color de Fondo".
Escenarios secundarios	1. El usuario hace click en el botón "Cancelar" de la ventana "Color de Fondo". 2. Se cierra la ventana "Color de Fondo", el caso de uso termina.
Post-Condiciones	Se modifica el color de fondo de la tortuga.

Cuadro 37: Casos de Uso A5.6: Seleccionar color de fondo

ID y Nombre	M6: Insertar máscara a la tortuga
Descripción	Permite al usuario colocar una máscara sobre la tortuga.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo 2. Un nuevo proyecto debe estar abierto
Flujo de Eventos	1. El usuario selecciona con el *mouse* la opción Insertar Máscara del Menú Formato (ó flujo alterno 1). 2. Se despliega la ventana Máscara, mostrando todas las posibles máscaras que se pueden asignar a las tortugas. 3. Con el *mouse*, el usuario selecciona el botón de opción ubicado en la parte inferior de la máscara que desea insertar. 4. Presiona con el botón izquierdo del *mouse*, el botón Aceptar ubicado en la parte inferior izquierda de la ventana. Si no, flujo secundario 1. 5. Para finalizar, presiona con el *mouse*, el botón Salir ubicado en la parte inferior derecha de la ventana.
Flujo Alterno	1. El usuario presiona las teclas Alt+F, seguido de la tecla M.
Flujo Secundario	1. Si el usuario desea cerrar la ventana Máscara sin insertar ninguna máscara al proyecto, presiona con el *mouse* el botón Salir, ubicado en la parte inferior derecha de la ventana. 2. En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	La tortuga ahora posee la máscara seleccionada.

Cuadro 38: Casos de Uso M6: Insertar Máscara a la tortuga

ID y Nombre	B7: Seleccionar vista
Descripción	Permite al usuario seleccionar entre tres modos distintos de vistas del proyecto.
Actor(es)	Usuario
Prioridad	Baja
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario despliega con el *mouse* el Menú Zoom de la Ventana Principal de MundoLogo (ó flujo alterno 1). 2. Si el usuario desea ver su proyecto en el modo Zoom a doble tamaño, selecciona la opción Aumentar del menú (ó flujo alterno 2). 3. Si el usuario desea ver su proyecto en el modo Zoom a mitad de tamaño, selecciona la opción Disminuir del menú (ó flujo alterno 3). 4. Si el usuario desea ver su proyecto en el modo Zoom al tamaño 1.0, selecciona la opción Normal del menú (ó flujo alterno 4).
Flujo Alterno	1. El usuario presiona las teclas Alt+Z. 2. Una vez desplegado el Menú Zoom, el usuario presiona la tecla A. 3. Una vez desplegado el Menú Zoom, el usuario presiona la tecla D. 4. Una vez desplegado el Menú Zoom, el usuario presiona la tecla N.
Flujo Secundario	En cualquier momento el usuario puede salir del sistema.
Post-Condiciones	El modo de la vista del proyecto se adapta al seleccionado por el usuario.

Cuadro 39: Casos de Uso B7: Seleccionar vista

ID y Nombre	A8: Consultar Ayuda
Descripción	Permite al usuario consultar toda la ayuda del sistema y algunas notas de versión.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo 2. Un nuevo proyecto debe estar abierto
Flujo de Eventos	1. El usuario despliega con el *mouse* el Menú Ayuda de la ventana principal de MundoLogo (ó flujo alterno 1). 2. Si el usuario desea consultar la ayuda del sistema por el índice, selecciona la opción Índice del menú (ó flujo alterno 2). 3. Si el usuario desea consultar como utilizar la ayuda, selecciona la opción Usar Ayuda del menú (ó flujo alterno 3). 4. Si el usuario desea consultar el Tutorial del sistema, selecciona la opción Tutorial del menú (ó flujo alterno 4). 5. Si el usuario desea consultar el Demo del sistema, selecciona la opción Demo del menú (ó flujo alterno 5). 6. Si el usuario desea consultar algunos ejemplos de proyectos programados en lenguaje Logo, selecciona la opción Ejemplos del menú (ó flujo alterno 6). 7. Si el usuario desea ver la notas sobre estos ejemplos, selecciona la opción Notas del menú (ó flujo alterno 7). 8. Si el usuario desea consultar información acerca de Logo, selecciona la opción Acerca de Logo del menú (ó flujo alterno 8). 9. Si el usuario desea conocer información acerca de la versión de MundoLogo, selecciona la opción Acerca de MundoLogo del menú (ó flujo alterno 9).
Flujo Alterno	1. El usuario presiona las teclas Alt+?. 2. Una vez desplegado el Menú Zoom, el usuario presiona la tecla I. 3. Una vez desplegado el Menú Zoom, el usuario presiona la tecla M. 4. Una vez desplegado el Menú Zoom, el usuario presiona la tecla U. 5. Una vez desplegado el Menú Zoom, el usuario presiona la tecla T. 6. Una vez desplegado el Menú Zoom, el usuario presiona la tecla E. 7. Una vez desplegado el Menú Zoom, el usuario presiona la tecla N. 8. Una vez desplegado el Menú Zoom, el usuario presiona la tecla A. 9. Una vez desplegado el Menú Zoom, el usuario presiona la tecla A, seguida nuevamente de la tecla C.
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	Se despliega la información seleccionada por el usuario.

Cuadro 40: Casos de Uso A8: Consultar Ayuda

ID y Nombre	A9: Trabajar en Modo Gráfico
Descripción	Permite al usuario crear figuras a través del uso de la ventana de herramientas de dibujo.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo 2. Un nuevo proyecto debe estar abierto
Flujo de Eventos	1. El usuario se ubica en la barra de herramientas de dibujo. 2. Luego, selecciona con el *mouse* la herramienta de dibujo que desea utilizar: 2.1. Si el usuario desea copiar un área, Caso de Uso A9.1. 2.2. Si el usuario desea pegar un área previamente copiada, en algún lugar del área de trabajo, Caso de Uso A9.2. 2.3. Si desea realizar un grafico, haciendo uso de la herramienta lápiz, Caso de Uso A9.3. 2.4. Si desea crear un rectángulo, Caso de Uso A9.4. 2.5. Si desea crear un rectángulo relleno, Caso de Uso A9.5. 2.6. Si desea crear una elipse, Caso de Uso A9.6. 2.7. Si desea crear una elipse rellena, Caso de Uso A9.7. 2.8. Si desea dibujar una línea recta, Caso de Uso A9.8. 2.9. Si desea utilizar la herramienta píxel, Caso de Uso A9.9. 2.10. Si desea borrar parte de los gráficos dibujados en el área de trabajo, Caso de Uso A9.10. 2.11. Si desea rellenar un área del grafico, Caso de Uso A9.11. 2.12. Si desea seleccionar un grosor de pluma, Caso de Uso A9.12. 2.13. Si desea seleccionar un color, Caso de Uso A9.13. 2.14. Si desea volver a trabajar en el modo no grafico, Caso de Uso A9.14.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	El sistema cambia a modo gráfico y se crea un nuevo dibujo en el área de trabajo.

Cuadro 41: Casos de Uso A9: Trabajar en Modo Gráfico

ID y Nombre	A9.1: Seleccionar Copiar
Descripción	Permite al usuario copiar en el portapapeles de la aplicación, el área que él seleccione de la ventana de trabajo.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo 2. Un nuevo proyecto debe estar abierto
Flujo de Eventos	1. El usuario selecciona la herramienta "copiar", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. A continuación, el usuario selecciona el área que desea copiar, manteniendo presionando el botón izquierdo del *mouse* y arrastrando el puntero para así definir dicha área. 3. Luego, el usuario suelta el botón del *mouse* y el área definida es automáticamente copiada en el portapapeles.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	El área seleccionada queda almacenada en el portapapeles de la aplicación.

Cuadro 42: Casos de Uso A9.1: Seleccionar Copiar

ID y Nombre	A9.2: Seleccionar Pegar
Descripción	Permite al usuario pegar en cualquier lugar del área de trabajo, una imagen o área que haya sido previamente almacenada en el portapapeles.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. Debe existir un área o imagen almacenada en el portapapeles.
Flujo de Eventos	1. El usuario selecciona la herramienta "pegar", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. A continuación, presiona con el botón izquierdo del *mouse* el lugar donde desea colocar la imagen en el área de trabajo, y ésta es pegada desde el portapapeles.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	El área previamente copiada, es pegada en el área de trabajo.

Cuadro 43: Casos de Uso A9.2: Seleccionar Pegar

ID y Nombre	A9.3: Seleccionar Pluma
Descripción	Permite al usuario dibujar sobre el área de trabajo usando la herramienta pluma.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona la herramienta "pluma", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. Luego, sobre el área de trabajo, con el botón izquierdo del *mouse* presionado, el usuario dibuja con la pluma arrastrando el puntero por dicha área.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	Se crea un nuevo dibujo en el área de trabajo, usando la pluma.

Cuadro 44: Casos de Uso A9.3: Seleccionar Pluma

ID y Nombre	A9.4: Seleccionar Rectángulo
Descripción	Permite al usuario dibujar sobre el área de trabajo un rectángulo.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona la herramienta "rectángulo", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. Luego, sobre el área de trabajo, manteniendo el botón izquierdo del *mouse* presionado, el usuario dibuja el rectángulo, arrastrando el puntero diagonalmente en cualquier dirección. 3. Por último, el usuario suelta el botón del *mouse* y el rectángulo queda dibujado sobre el área de trabajo.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	Se dibuja un rectángulo en el área de trabajo.

Cuadro 45: Casos de Uso A9.4: Seleccionar Rectángulo

ID y Nombre	A9.5: Seleccionar Rectángulo Relleno
Descripción	Permite al usuario dibujar sobre el área de trabajo un rectángulo relleno.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona la herramienta "rectángulo relleno", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. Luego, sobre el área de trabajo, manteniendo el botón izquierdo del *mouse* presionado, el usuario dibuja el rectángulo, arrastrando el puntero diagonalmente en cualquier dirección. 3. Por último, el usuario suelta el botón del *mouse* y el rectángulo relleno queda dibujado sobre el área de trabajo.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	Se dibuja un rectángulo relleno en el área de trabajo.

Cuadro 46: Casos de Uso A9.5: Seleccionar Rectángulo Relleno

ID y Nombre	A9.6: Seleccionar Elipse
Descripción	Permite al usuario dibujar sobre el área de trabajo una elipse.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona la herramienta "elipse", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. Luego, sobre el área de trabajo, manteniendo el botón izquierdo del *mouse* presionado, el usuario dibuja la elipse, arrastrando el puntero diagonalmente en cualquier dirección. 3. Por último, el usuario suelta el botón del *mouse* y la elipse queda dibujada sobre el área de trabajo.
Flujo Alternativo	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	Se dibuja una elipse en el área de trabajo.

Cuadro 47: Casos de Uso A9.6: Seleccionar Elipse

ID y Nombre	A9.7: Seleccionar Elipse rellena
Descripción	Permite al usuario dibujar sobre el área de trabajo una elipse rellena.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona la herramienta "elipse rellena", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. Luego, sobre el área de trabajo, manteniendo el botón izquierdo del *mouse* presionado, el usuario dibuja la elipse, arrastrando el puntero diagonalmente en cualquier dirección. 3. Por último, el usuario suelta el botón del *mouse* y la elipse rellena queda dibujada sobre el área de trabajo.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	Se dibuja una elipse rellena en el área de trabajo.

Cuadro 48: Casos de Uso A9.7: Seleccionar Elipse Rellena

ID y Nombre	A9.8: Seleccionar Línea recta
Descripción	Permite al usuario dibujar sobre el área de trabajo una línea recta.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona la herramienta "línea recta", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. Luego, sobre el área de trabajo, manteniendo el botón izquierdo del *mouse* presionado, el usuario dibuja la línea, arrastrando el puntero en cualquier dirección. 3. Por último, el usuario suelta el botón del *mouse* y la línea recta queda dibujada sobre el área de trabajo.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	Se dibuja una línea recta en el área de trabajo.

Cuadro 49: Casos de Uso A9.8: Seleccionar Línea Recta

ID y Nombre	A9.9: Seleccionar Pixel
Descripción	Permite al usuario copiar el color de un área de la ventana de trabajo para utilizarlo luego con otra herramienta.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona la herramienta "pixel", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. Luego, el usuario hace clic con el botón izquierdo del *mouse* sobre el área cuyo color desea copiar. 3. El color queda copiado.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	El color queda copiado para poder ser utilizado luego con otra herramienta de dibujo.

Cuadro 50: Casos de Uso A9.9: Seleccionar Píxel

ID y Nombre	A9.10: Seleccionar Borrador
Descripción	Permite al usuario borrar un área ó dibujo del área de trabajo.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona la herramienta "borrador", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. Luego, con el botón izquierdo del *mouse* presionado, el usuario arrastra el puntero por el área que desea borrar.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	Es borrado del área de trabajo los gráficos ó dibujos que decidió borrar el usuario.

Cuadro 51: Casos de Uso A9.10: Seleccionar Borrador

ID y Nombre	A9.11: Seleccionar Relleno
Descripción	Permite al usuario rellenar de color un área cerrada u objeto del área de trabajo.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona la herramienta "relleno", presionando con el *mouse* el botón correspondiente a dicha herramienta en la ventana de herramientas de dibujo. 2. Luego, el usuario hace clic con el botón izquierdo del *mouse*, en el área ó en el objeto que desea rellenar.
Flujo Alternativo	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	El área u objeto es rellenado.

Cuadro 52: Casos de Uso A9.11: Seleccionar Relleno

ID y Nombre	A9.12: Seleccionar Grosor de la Pluma
Descripción	Permite al usuario seleccionar el grosor de la herramienta con que la va a trabajar.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona el grosor de la pluma, presionando con el botón izquierdo del *mouse* el control correspondiente al grosor deseado (ó flujo alterno 1). 2. Luego el usuario, selecciona la herramienta de dibujo con la cual desea combinar el grosor escogido: 2.1. Lápiz, Caso de Uso A9.3. 2.2. Rectángulo, Caso de Uso A9.4. 2.3. Elipse, Caso de Uso A9.6. 2.4. Línea recta, Caso de Uso A9.8. 2.5. Borrador, Caso de Uso A9.10.
Flujo Alterno	1. El usuario despliega el Menú Formato de la Ventana Principal, y selecciona la opción "Tamaño de la Pluma", y continúa el Caso de Uso A5.1.
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	El grosor de la herramienta queda seleccionado.

Cuadro 53: Casos de Uso A9.12: Seleccionar Grosor de Pluma

ID y Nombre	A9.13: Seleccionar Color
Descripción	Permite al usuario seleccionar el color de la herramienta con que la va a trabajar.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto.
Flujo de Eventos	1. El usuario selecciona el color de la pluma, presionando con el botón izquierdo del *mouse* el control correspondiente al color deseado (ó flujo alterno 1). 2. Luego el usuario, selecciona la herramienta de dibujo con la cual desea combinar el color escogido: 2.1. Lápiz, Caso de Uso A9.3. 2.2. Rectángulo, Caso de Uso A9.4. 2.3. Rectángulo relleno, Caso de Uso A9.5. 2.4. Elipse, Caso de Uso A9.6 2.5. Elipse rellena, Caso de Uso A9.7. 2.6. Línea recta, Caso de Uso A9.8. 2.7. Relleno, Caso de Uso A9.11. 3. Si el usuario desea escoger entre una mayor variedad de colores, presiona con el *mouse* el botón Más de la ventana de Herramientas de Dibujo y se despliega la ventana "Color de la Pluma". Continua el Caso de Uso A5.4.
Flujo Alterno	1. El usuario despliega el Menú Formato de la Ventana Principal, y selecciona la opción "Color de la Pluma", y continúa el Caso de Uso A5.4.
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	El color de la herramienta queda seleccionado.

Cuadro 54: Casos de Uso A9.13: Seleccionar Color

ID y Nombre	A9.14: Seleccionar Modo no Grafico
Descripción	Permite al usuario salir del modo grafico (trabajo con el menú de herramientas de dibujo) y volver al modo normal de trabajo con primitivas y procedimientos.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Alto
Pre-Condiciones	1. El sistema MundoLogo debe estar corriendo. 2. Un nuevo proyecto debe estar abierto. 3. El sistema debe estar en modo grafico.
Flujo de Eventos	1. El usuario selecciona presionando con el *mouse*, el botón correspondiente a la herramienta "puntero". 2. El sistema muestra nuevamente en pantalla las tortugas del proyecto y la ventana de comandos se activa para seguir trabajando en ella.
Flujo Alterno	
Flujo Secundario	En cualquier instante el usuario puede abandonar el sistema.
Post-Condiciones	El sistema vuelve al modo no grafico.

Cuadro 55: Casos de Uso A9.14: Seleccionar Modo no Gráfico

ID y Nombre	A10: Agregar tortuga
Descripción	Crea y coloca en el área de trabajo una nueva tortuga.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar ejecutándose. 2. Un proyecto debe estar abierto.
Flujo de Eventos	1. El usuario se ubica en la ventana de "herramientas de dibujo" 2. Hace click en el botón de agregar nueva tortuga, en la parte superior de la ventana, representado por la ilustración de una tortuga.
Post-Condiciones	La nueva tortuga es posicionada en el área de trabajo.

Cuadro 56: Casos de Uso A10: Agregar Tortuga

ID y Nombre	A11: Borrar Pantalla
Descripción	Borra el fondo del área de trabajo y coloca la tortuga en la posición [0,0]
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar ejecutándose. 2. Un proyecto debe estar abierto.
Flujo de Eventos	1. El usuario se ubica en la ventana de "Comandos" 2. Hace click en el botón "Borrar", en la parte superior derecha de la ventana. 3. Si existe más de una tortuga en el área de trabajo. 4. La tortuga cero es posicionada en el centro del área de trabajo.
Post-Condiciones	El fondo del área de trabajo es borrado y la tortuga cero se encuentran en el centro de la pantalla.

Cuadro 57: Casos de Uso A11: Borrar Pantalla

ID y Nombre	M12: Detener Proceso
Descripción	Detiene procesos en ejecución.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar ejecutándose. 2. Un proyecto debe estar abierto.
Flujo de Eventos	1. El usuario se ubica en la ventana de "Comandos" 2. Hace click en el botón "Detener", en la parte superior derecha de la ventana. 3. En la lista de la ventana de comandos aparece el mensaje "Deteniendo..."
Post-Condiciones	El procedimiento o proceso que se esté ejecutando en ese momento es detenido.

Cuadro 58: Casos de Uso M12: Detener Proceso

ID y Nombre	M13: Pausar Proceso
Descripción	Pausa los procesos en ejecución.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar ejecutándose. 2. Un proyecto debe estar abierto.
Flujo de Eventos	1. El usuario se ubica en la ventana de "Comandos" 2. Hace click en el botón "Pausar", en la parte superior derecha de la ventana.
Post-Condiciones	El proceso que se esté ejecutando en ese momento es pausado o despausado.

Cuadro 59: Casos de Uso M13: Pausar Proceso

ID y Nombre	M14: Realizar traza
Descripción	Coloca la ejecución en un modo traza, que facilita el proceso de depuración.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar ejecutándose. 2. Un proyecto debe estar abierto.
Flujo de Eventos	1. El usuario se ubica en la ventana de "Comandos" 2. Hace click en el botón "Traza", en la parte superior derecha de la ventana.
Post-Condiciones	El modo traza es activado o desactivado.

Cuadro 60: Casos de Uso M14: Realizar traza

ID y Nombre	M15: Correr por pasos
Descripción	Coloca la ejecución en un modo por pasos, que facilita el proceso de depuración.
Actor(es)	Usuario
Prioridad	Media
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar ejecutándose. 2. Un proyecto debe estar abierto.
Flujo de Eventos	1. El usuario se ubica en la ventana de "Comandos". 2. Hace click en el botón "Paso", en la parte superior derecha de la ventana. 2.1. Si los el modo por pasos no está activo, se activa el modo. Los procedimientos se ejecutan línea por línea. 2.2. Si no, el modo por pasos se desactiva.
Post-Condiciones	El modo por pasos es activado o desactivado.

Cuadro 61: Casos de Uso M15: Correr por Pasos

ID y Nombre	A16: Salir del sistema
Descripción	Detiene los procesos en ejecución y se cierra el sistema.
Actor(es)	Usuario
Prioridad	Alta
Riesgo	Bajo
Pre-Condiciones	1. El sistema MundoLogo debe estar ejecutándose.
Flujo de Eventos	1. El usuario se ubica en el menú superior de la pantalla. 2. Hace click en la opción "Archivo". 3. Se despliega el sub-Menú Archivo. 4. El usuario hace click en la opción salir del sub-Menú. 5. Si el usuario no ha guardado anteriormente el proyecto 5.1. Se despliega la ventana "No ha guardado el proyecto". 5.2. Aparece la "¿Está seguro de que desea salir del sistema?". 5.3.Si el usuario presiona el botón "Aceptar" de la ventana, el sistema se cierra. 5.4.Si no, el usuario presiona el botón "Cancelar", la ventana "No ha guardo el proyecto se cierra" el caso de uso termina. 6. Si no, el proyecto está guardado. 7. Se cierra el sistema.
Flujo Alterno	1. El usuario hace click sobre la "X" en la parte superior derecha de la ventana principal. 2. Continua el caso de uso en el flujo de eventos 5
Post-Condiciones	El sistema se cierra.

Cuadro 62: Casos de Uso A16: Salir del Sistema

Anexo IV-12
Diagramas de Actividades de MundoLogo

A continuación se presentan los diagramas de actividades de MundoLogo.

Actividad: Insertar Máscara a la Tortuga

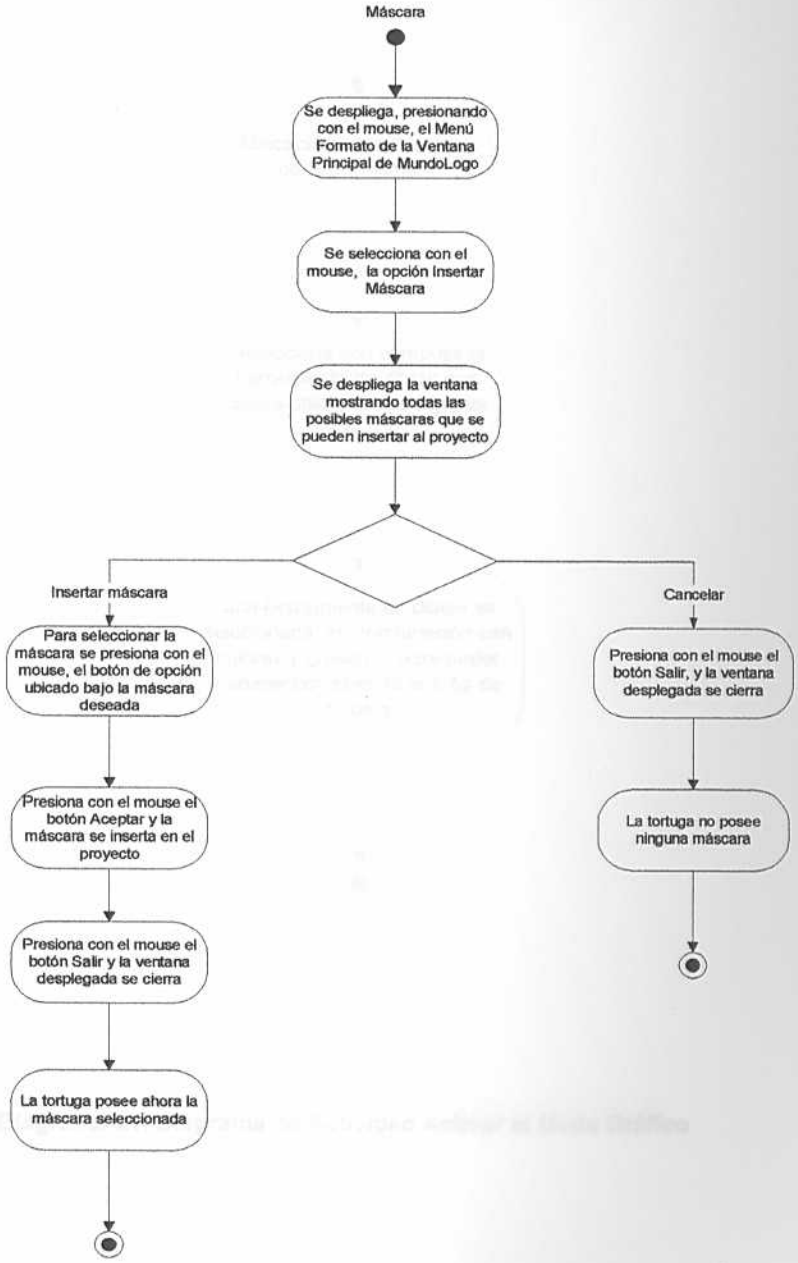


Diagrama 20: Diagrama de Actividad Insertar Máscara a la tortuga

Actividad: Activar el Modo Gráfico

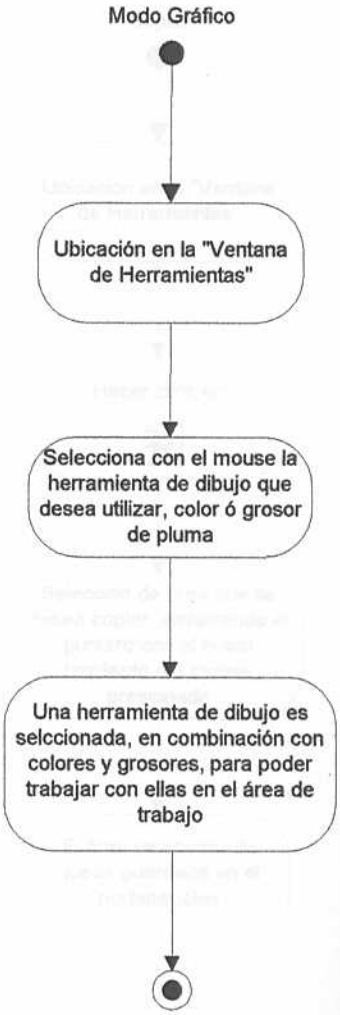


Diagrama 21: Diagrama de Actividad Activar el Modo Gráfico

Actividad: Seleccionar Copiar

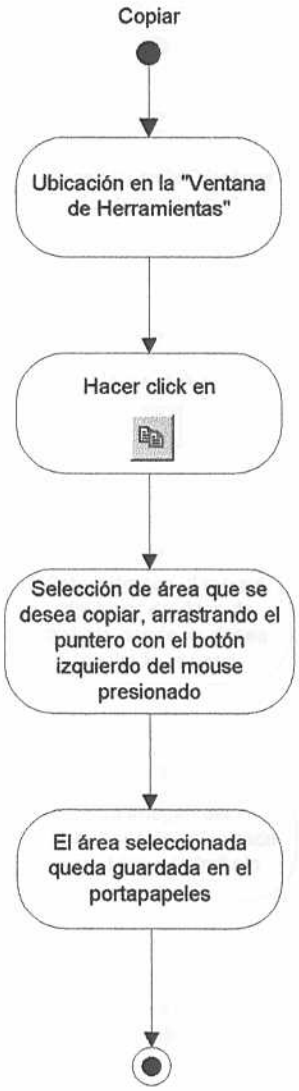


Diagrama 22: Diagrama de Actividad Seleccionar Copiar

Actividad: Seleccionar Pegar

Diagrama 23: Diagrama de Actividad Seleccionar Pegar

Actividad: Seleccionar Pluma

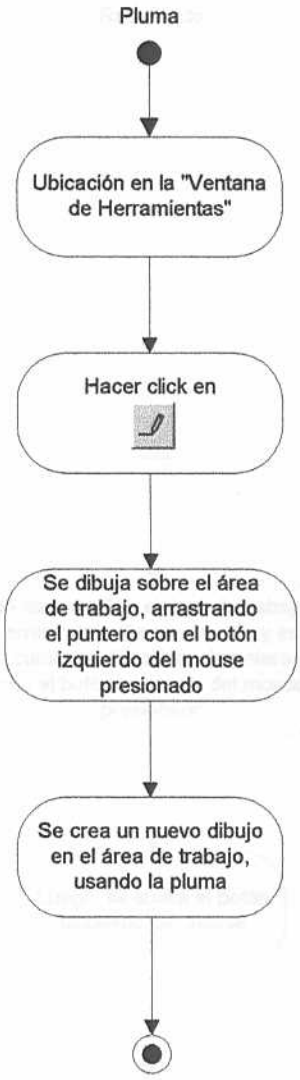


Diagrama 24: Diagrama de Actividad Seleccionar Pluma

Actividad: Seleccionar Rectángulo

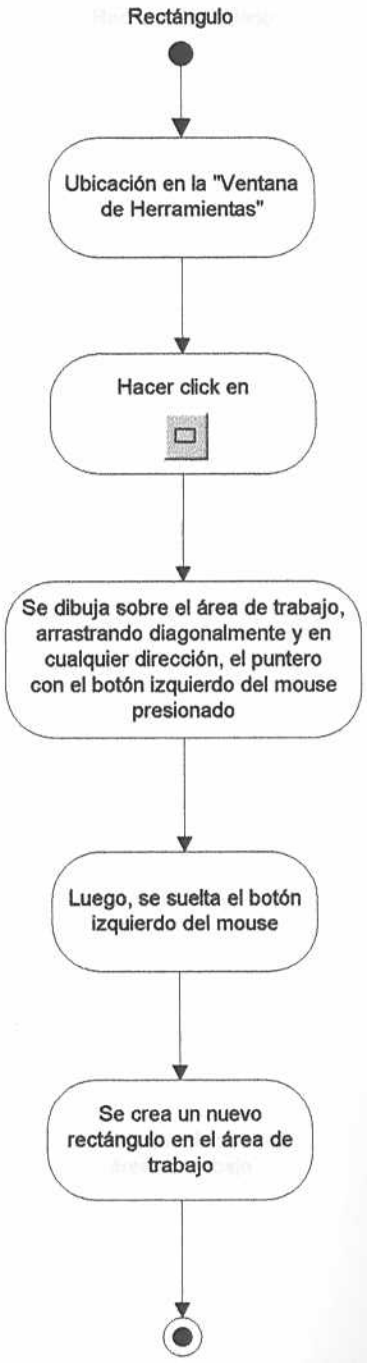


Diagrama 25: Diagrama de Actividad Seleccionar Rectángulo

Actividad: Seleccionar Rectángulo Relleno

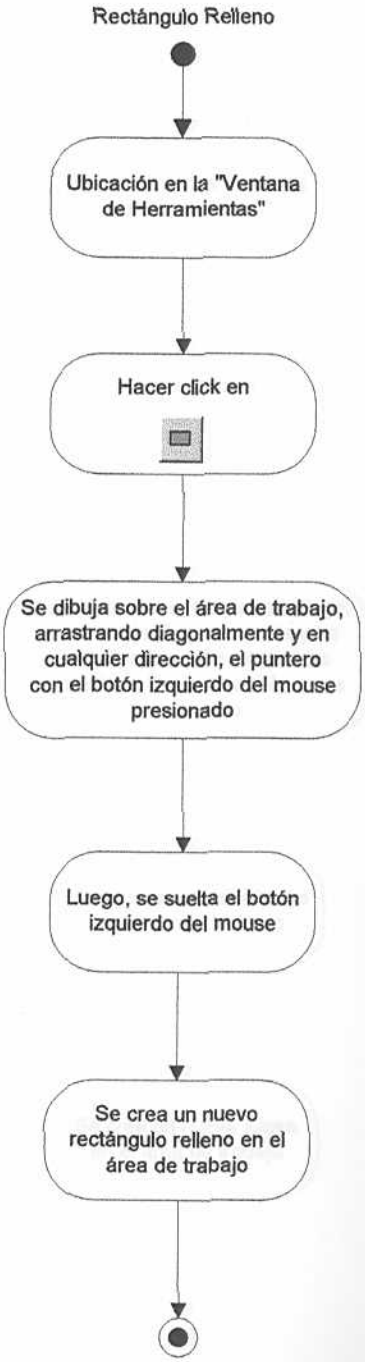


Diagrama 26: Diagrama de Actividad Seleccionar Rectángulo Relleno

Actividad: Seleccionar Elipse

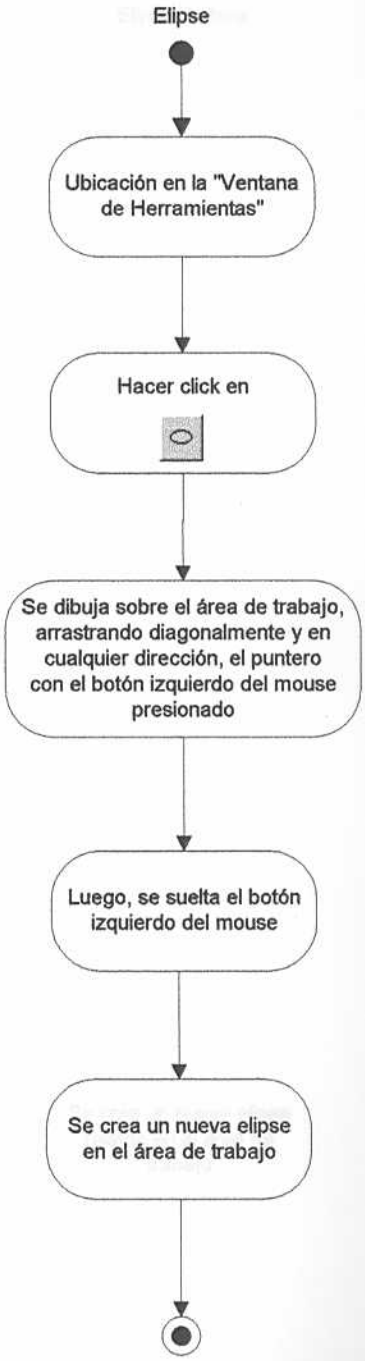


Diagrama 27: Diagrama de Actividad Seleccionar Elipse

Actividad: Seleccionar Elipse Rellena

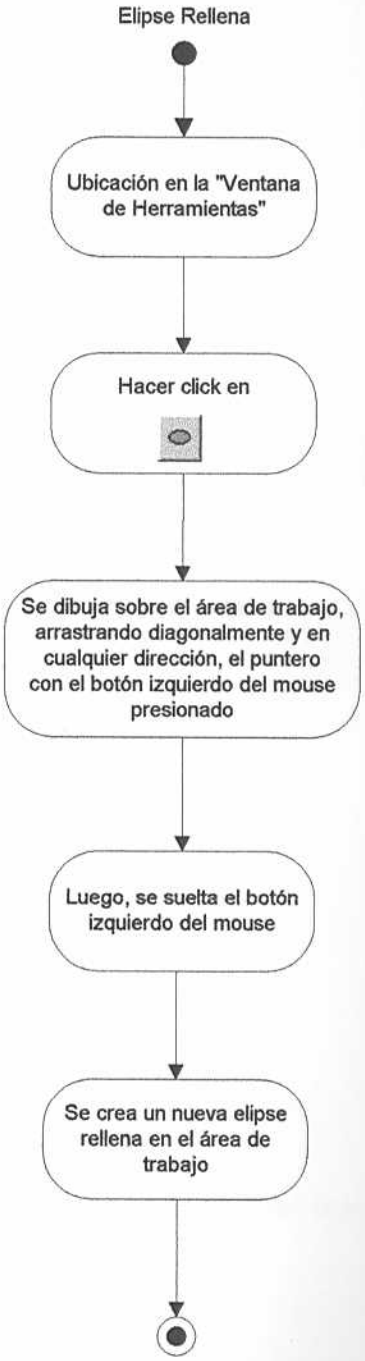


Diagrama 28: Diagrama de Actividad Seleccionar Elipse Rellena

Actividad: Seleccionar Línea Recta

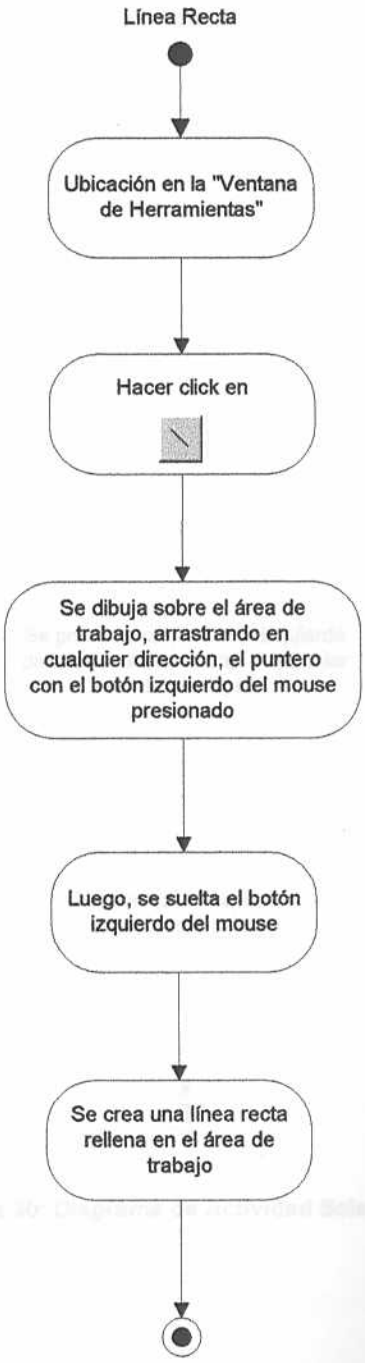


Diagrama 29: Diagrama de Actividad Seleccionar Línea Recta

Actividad: Seleccionar Pixel

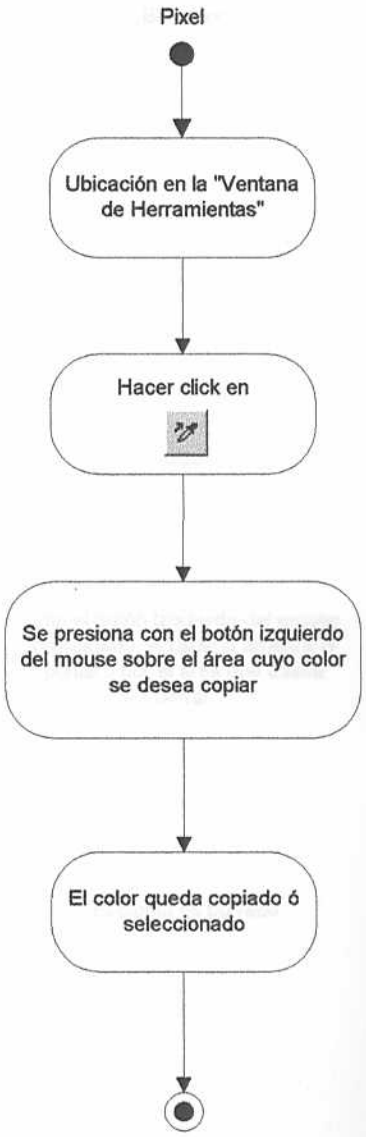


Diagrama 30: Diagrama de Actividad Seleccionar Pixel

Actividad: Seleccionar Borrador

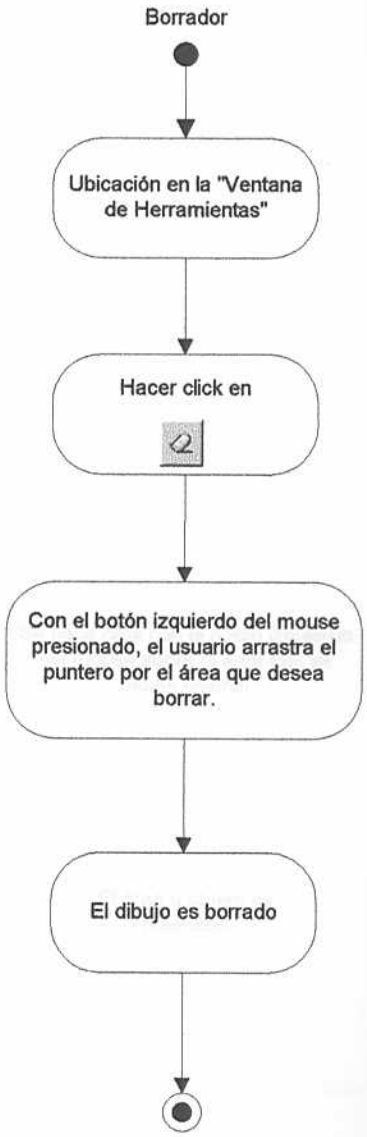


Diagrama 31: Diagrama de Actividad Seleccionar Borrador

Actividad: Seleccionar Relleno

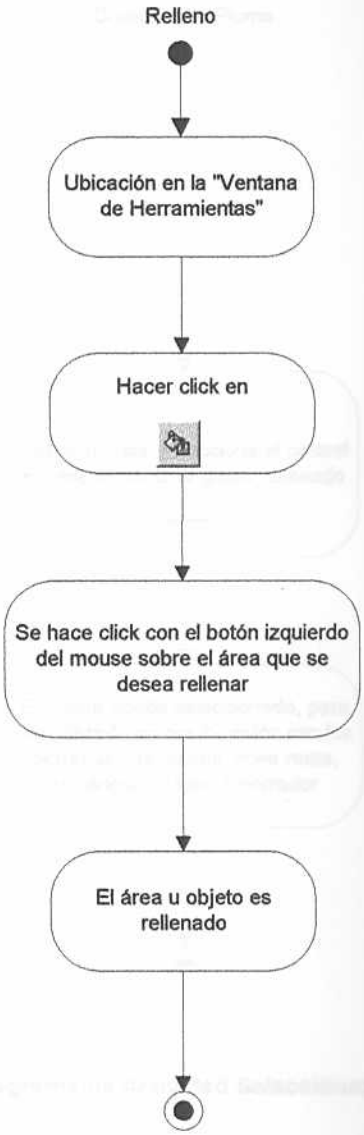


Diagrama 32: Diagrama de Actividad Seleccionar Relleno

Actividad: Seleccionar Grosor de la Pluma

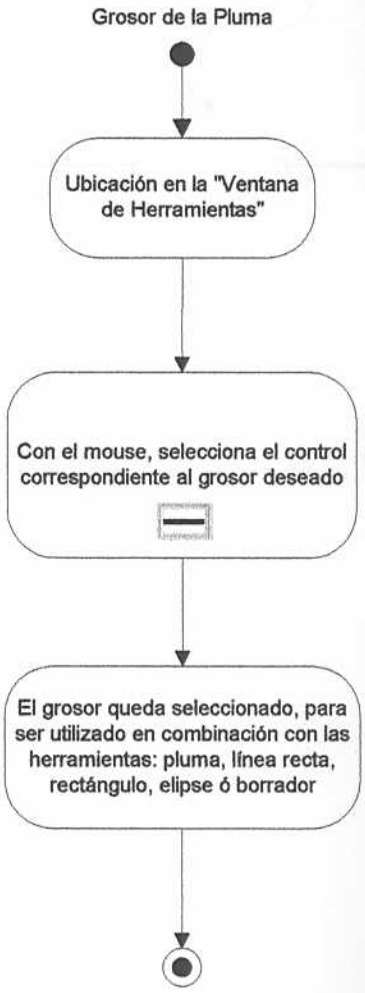


Diagrama 33: Diagrama de Actividad Seleccionar Grosor de la Pluma

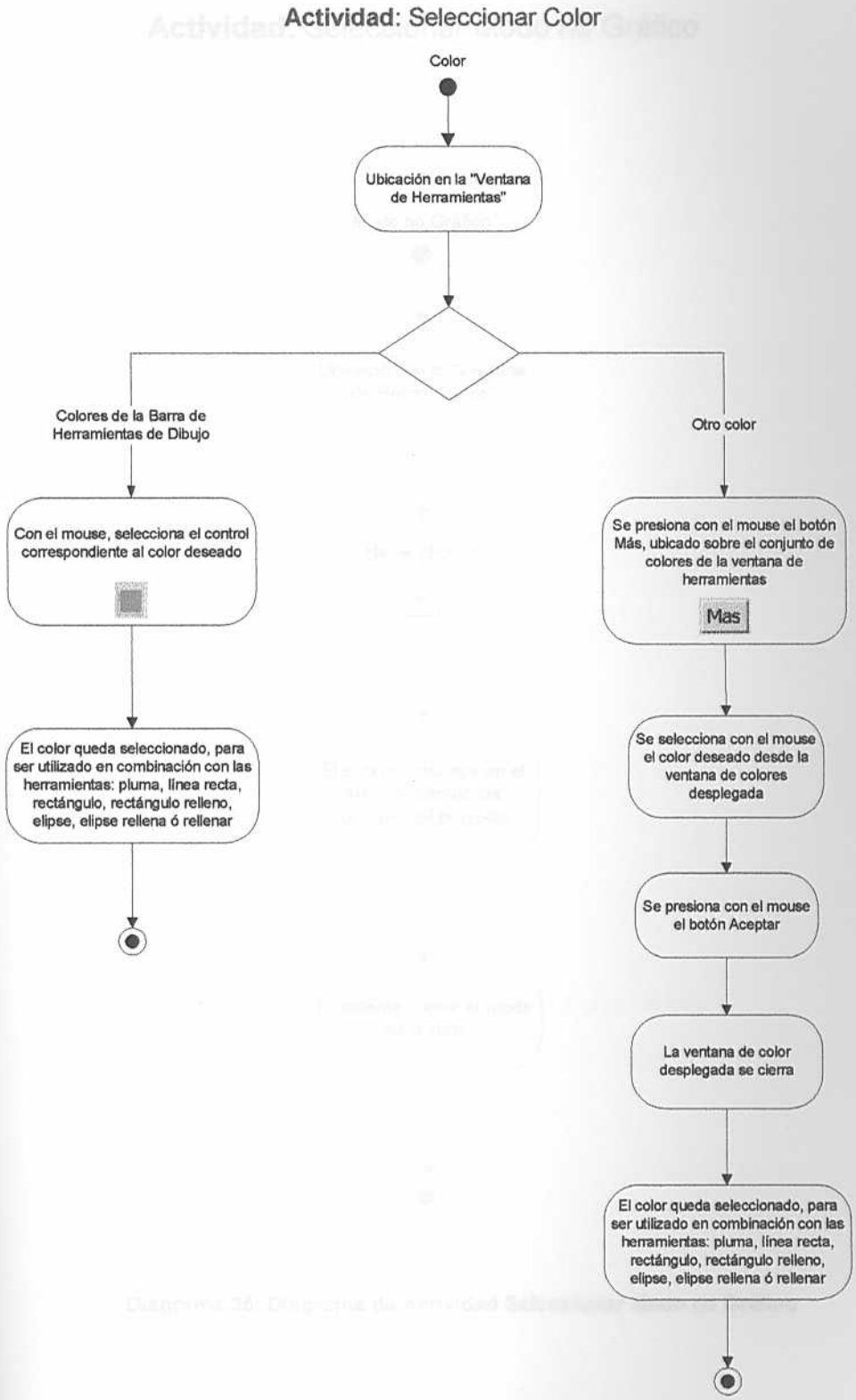


Diagrama 34: Diagrama de Actividad Seleccionar Color

Actividad: Seleccionar Modo no Gráfico

Diagrama 35: Diagrama de Actividad Seleccionar Modo no Gráfico

Actividad: Insertar Nueva Tortuga

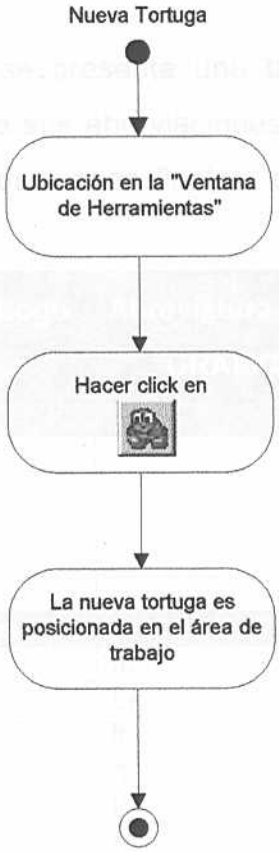


Diagrama 36: Diagrama de Actividad Insertar Nueva Tortuga

Anexo IV-13

Traducción al español de las Primitivas de MSWLogo

A continuación, se presenta una tabla con todas las primitivas de MSWLogo, así como de sus abreviaciones, agrupadas por categorías, junto con las respectivas traducciones finales, que son las que se utilizan en la versión de MundoLogo.

	Primitiva en MSWLogo	Abreviatura	Primitiva en MundoLogo	Abreviatura
	GRÁFICOS			
Múltiples Tortugas				
	setturtle		pontortuga	
	turtle		tortuga	
	turtles		tortugas	
	bitmapturtle		ffigura	
	nobitmapturtle		noffigura	
Movimiento de la Tortuga				
	forward	fd	adelante	ad
	back	bk	atrás	at
	left	lt	giraizquierda	gi
	right	rt	giraderecha	gd
	leftroll	lr	ruedaizquierda	ri
	rightroll	rr	ruedaderecha	rd
	downpitch	down	bajacabeza	bajac
	uppitch	up	subecabeza	subec
	setpos		ponpos	
	setposxyz		ponposxyz	
	setxy		ponxy	
	setxyz		ponxyz	
	setx		ponx	
	sety		pony	
	setz		ponz	
	home		inicio	
	setheading	seth	ponrumbo	ponr
	setroll		ponrueda	
	setpitch		poninclinación	
	setorientation		ponorientación	
	ellipsearc		arcoelipse	
	ellipsearc2		arcoelipse2	
	ellipse		elipse	
	ellipse2		elipse2	

	arc		arco	
	arc2		arco2	
	circle		círculo	
	circle2		círculo2	
Preguntas sobre el Movimiento de la Tortuga				
	pos		pos	
	posxyz		posxyz	
	xcor		coorx	
	ycor		coory	
	zcor		coorz	
	heading		rumbo	
	roll		rueda	
	pitch		inclinación	
	orientation		orientación	
	towards		haciaxy	
	towardsxyz		haciaxyz	
	distance		distancia	
	distancexyz		distanciaxyz	
	píxel		píxel	
	scrunch		escala	
La Tortuga y el Control de la Ventana				
	showturtle	st	muestratortuga	mt
	hideturtle	ht	ocultatortuga	ot
	clean		limpia	
	clearscreen	cs	limpiaventana	lv
	wrap		envolver	
	window		ventana	
	fence		cerca	
	perspective		perspectiva	
	fill		pinta	
	label		etiqueta	
	setpixel		ponpixel	
	setlabelfont		ponletraetiqueta	
	labelfont		letraetiqueta	
	labelsize		tamañoetiqueta	
	textscreen	ts	tamposventana	tpv
	fullscreen	fs	modopresentación	mp
	splitscreen	ss	dividirpantalla	dp
	setscrunch		ajustaescala	
	refresh		recordar	
	norefresh		norecordar	
	zoom		zoom	
	scrollx		scrollx	
	scrolly		scrolly	
	setfocus		ponfoco	
	getfocus		obténfoco	
	icon		icono	

	unicon		quitaicono	
	windowset		ponventana	
Preguntas sobre la Tortuga y la Ventana				
	shownp		esvisible	
	shown?		visible?	
Control del Lápiz				
	pendown	pd	conpluma	cp
	penup	pu	sinpluma	sp
	penpaint	ppt	ponpluma	ppl
	penerase	pe	borrapluma	bp
	penreverse	px	plumainversa	pinv
	pennormal		plumanormal	
	setpencolor	setpc	poncolorpluma	poncp
	setfloodcolor	setfc	poncolorrelleno	poncr
	setscreencolor	setsc	poncolorfondo	poncf
	setpensize		pongrosor	
	setpenpattern		ponpatrónpluma	
	setpen		pontipopluma	
	clearpalette		limpiapaleta	
Preguntas sobre el Lápiz				
	pendownp		esconpluma	
	pendown?		conpluma?	
	penmode		modopluma	
	pencolor	pc	colorpluma	cpl
	floodcolor		colorrelleno	
	screencolor		colorfondo	
	pensize		grosor	
	penpattern		patrónpluma	
	pen		pluma	
Dibujando Sólidos en 3D				
	polystart		ipolígono	
	polyend		fpolígono	
	polyview		mpolígono	
	setlight		ponluz	
	light		luz	
ESTRUCTURAS DE CONTROL				
Ordenes de Control				
	run		ejecuta	
	runresult		ejecutaresultado	
	repeat		repite	
	repcount		repeticiones	
	if		si	
	ifelse		siotro	
	test		prueba	
	iftrue	ift	siverdadero	siv
	iffalse	iff	sifalso	sif

	true		verdadero	
	false		falso	
	stop		parar	
	output	op	devuelve	dev
	catch		captura	
	throw		salto	
	error		error	
	pause		pausa	
	continue	co	continúa	co
	yield		exclusivo	
	noyield		noexclusivo	
	eventcheck		hayevento	
	setcursorwait		poncursorespera	
	halt		detén	
	wait		espera	
	settimer		ponreloj	
	cleartimer		relojacero	
	bye		chao	
	.maybeoutput		.quizádevuelve	
	ignore		ignora	
	`		'	
	for		desde	
	forever		porsiempre	
	do.while		haz.mientras	
	while		mientras	
	do.until		haz.hasta	
	until		hasta	
	goto		ir	
	tag		tag	
Iteración basada en plantillas				
	apply		aplicar	
	invoke		invocar	
	foreach		paracadauno	
	map		mapa	
	map.se		mapa.or	
	filter		filtro	
	find		buscar	
	reduce		reducir	
	crossmap		mezcla	
	cascade		cascada	
	cascade.2		cascada.2	
	transfer		transferir	
	iseq		esec	
	rseq		rsec	

MULTIMEDIA / MIDI / SONIDO				
Primitivas MIDI				
	midiopen		abremidi	
	midiclose		cierramidi	
	midimessage		mensajemidi	
	miditable		tablamidi	
	midicontrol		controlmidi	
	midiglossary		glosariomidi	
	midiinstrument		instrumentosmidi	
Primitiva SonidoWave				
	playwave		reproducewav	
Primitivas Sonido				
	soundon		enciendesonido	
	soundoff		apagasonido	
	sound		sonido	
Primitivas Multimedia				
	mci		mci	
FUNCIONES DE BITMAP				
Primitivas de Bitmap				
	bitcut		cortaárea	
	bitcopy		copiaárea	
	bitpaste		pegaárea	
	bitpastetoindex		pegaáreaíndice	
	bitfit		ajustaárea	
	setbitindex		poníndiceárea	
	bitindex		índiceárea	
	setbitmode		ponmodoárea	
	bitmode		modoárea	
	bitblock		rellenaárea	
	bitload		cargaimg	
	bitsave		guardaimg	
	bitsize		tamañoárea	
	bitloadsize		tamañoimg	
	gifload		cargagif	
	gifsave		guardagif	
	gifsize		tamañogif	
	setactivearea		ponáreaactiva	
	activearea		áreaactiva	
PRIMITIVAS DE DIRECTORIOS				
	dir		dir	
	files		archivos	
	directories		directorios	
	chdir	cd	cambiardir	cd
	popdir		directoriopadre	
	mkdir		creadir	

	rmdir		eliminardir	
		ORDENES DE AYUDA		
	help		ayuda	
	winhelp		winayuda	
		MANEJO DE LA MEMORIA		
Definición de Procedimientos				
	to		para	
	end		fin	
	define		define	
	text		texto	
	fulltext		textocompleto	
	copydef		copiaproc	
Definición de Variables				
	make		haz	
	name		variable	
	local		local	
	localmake		hazlocal	
	thing	:	valor	:
Propiedades Listas				
	pprop		ponprop	
	gprop		propiedad	
	remprop		eliminaprop	
	plist		plista	
Predicados - Espacio de trabajo				
	procedurep		esprocedimiento	
	procedure?		procedimiento?	
	primitivep		esprimitiva	
	primitive?		primitiva?	
	definedp		estádefinido	
	defined?		definido?	
	namep		esvariable	
	name?		variable?	
	macrop		esmacro	
	macro?		macro?	
Preguntas				
	contents		contenidos	
	buried		tapado	
	procedures		procedimientos	
	names		variables	
	plists		plistas	
	namelist		listavariables	
	pllist		pllista	
Inspección				
	po		im	
	pocontents	poall	impcontenidos	imptodo
	poprocedures	pops	impprocedimientos	impproc

	ponames	pons	impvariables	impvar
	poplist	popls	impplistas	imppls
	ponamelist	pon	implistavariables	implv
	popllist	popl	impplista	imppl
	pot		imptítuloproc	
	potprocedures	pots	imptítuloprocs	imptps
Control de Memoria				
	erase	er	borrar	br
	erall		borrartodo	
	erps		borrarproc	
	erns		borrarvar	
	erpls		borrarplistas	
	ern		borrarlistavar	
	erpl		borrarpllista	
	bury		tapa	
	buryall		tapatodo	
	buryname		tapavariable	
	unbury		destapa	
	unburyall		destapatodo	
	unburyname		destapavariable	
	trace		traza	
	untrace		quitatraza	
	traced		trazado	
	step		paso	
	unstep		quitapaso	
	stepped		pasado	
	edit	ed	editar	ed
	edall		editartodo	
	edps		editarproc	
	edns		editarvariables	
	edpls		editarplistas	
	edn		editarlistavar	
	edpl		editarpllista	
	save		guarda	
	savel		guardal	
	load		carga	
	nostatus		quitaestatus	
	status		estatus	
FUNCIONES DE WINDOWS				
Ventana				
	windowcreate		creaventana	
	windowdelete		borraventana	
	windowenable		habilitaventana	
Diálogo				
	dialogcreate		creadiálogo	
	dialogdelete		borradiálogo	

	dialogenable		habilitadiálogo	
Listbox				
	listboxcreate		crealistbox	
	listboxdelete		borralistbox	
	listboxenable		habilitalistbox	
	listboxgetselect		leeseleccióniistbox	
	listboxaddstring		agregalínealistbox	
	listboxdeletestring		borralínealistbox	
Combobox				
	comboboxcreate		creacombobox	
	comboboxdelete		borracombobox	
	comboboxenable		habilitacombobox	
	comboboxgettext		leetextocombobox	
	comboboxsettext		pontextocombobox	
	comboboxaddstring		agregalíneacombobox	
	comboboxdeletestring		borralíneacombobox	
Barra Desplazamiento				
	scrollbarcreate		creascrollbar	
	scrollbardelete		borrascrollbar	
	scrollbarenable		habilitascrollbar	
	scrollbarset		ponscrollbar	
	scrollbarget		leescrollbar	
Botón				
	buttoncreate		creabotón	
	buttondelete		borrabotón	
	buttonenable		habilitabotón	
	buttonupdate		actualizabotón	
Estático				
	staticcreate		creaestático	
	staticdelete		borraestático	
	staticupdate		actualizaestático	
Groupbox				
	groupboxcreate		creagroupbox	
	groupboxdelete		borragroupbox	
Checkbox				
	checkboxcreate		creacheckbox	
	checkboxdelete		borracheckbox	
	checkboxenable		habilitacheckbox	
	checkboxget		obténcheckbox	
	checkboxset		poncheckbox	
Radiobuttom				
	radiobuttoncreate		crearadiobutton	
	radiobuttondelete		borraradiobutton	
	radiobuttonenable		habilitaradiobutton	
	radiobuttonget		obténradiobutton	
	radiobuttonset		ponradiobutton	

Depuración				
	debugwindows		ventanadepurador	
Ventanas Predefinidas				
	messagebox		mensaje	
	yesnobox		ventanasino	
	selectbox		ventanaselección	
	questionbox		ventanapregunta	
	dialogfileopen		abrirarchivo	
	dialogfilesave		guardarcomo	
	windowfileedit		editaarchivo	
ARITMÉTICA				
Operaciones Numéricas				
	sum	+	suma	+
	difference	-	diferencia	-
	minus	-	menos	-
	product	*	producto	*
	quotient	/	divide	/
	remainder		resto	
	modulo		módulo	
	int		entero	
	round		redondea	
	abs		abs	
	sign		signo	
	sqrt		raízcuadrada	
	power		potencia	
	exp		exp	
	log10		log10	
	ln		ln	
	pi		pi	
	tan		tan	
	radtan		radtan	
	sin		sen	
	radsin		radsen	
	cos		cos	
	radcos		radcos	
	arctan		arctan	
	radarctan		radarctan	
	arcsin		arcsen	
	radarcsin		radarcsen	
	arccos		arccos	
	radarccos		radarccos	
Predicados (aritmética)				
	less?	<	menor?	
	lessp	<	esmenor	
	greater?	>	mayor?	
	greaterp	>	esmayor	

Números aleatorios				
	random		azar	
	rerandom		reazar	
Escribir c/ Formato				
	form		formatonúmero	
Operaciones con Bit				
	bitand		bity	
	bitor		bito	
	bitxor		bitxor	
	bitnot		nobit	
	ashift		adesplaza	
	lshift		ldesplaza	
COMUNICACIONES				
Transmisores				
	print	pr	imprimir	imp
	type		tipea	
	show		muestra	
Receptores				
	readlist	rl	leelista	ll
	readword	rw	leepalabra	lp
	readchar	rc	leecaracter	lc
	readchars	rcs	leecaracteres	lcs
	shell		shell	
Comunicaciones				
	portopen		abrepuerto	
	portclose		cierrapuerto	
	portflush		limpiapuerto	
	portmode		modopuerto	
	portreadarray		leecadenapuerto	
	portreadchar		leecaracterpuerto	
	portwritearray		escribecadenapuerto	
	portwritechar		escribecaracterpuerto	
	inport		puertoentrada	
	outport		puertosalida	
	inportb		puertoentradab	
	outportb		puertosalidab	
	ingameport		puertodejuego	
	dllload		cargardll	
	dllcall		llamardll	
	dllfree		liberardll	
Acceso al Teclado y al Ratón				
	keyp		estecla	
	key?		tecla?	
	keyboardon		prendeteclado	
	keyboardoff		apagateclado	
	keyboardvalue		valortecla	

	mouseon		ratónactivo	
	mouseoff		ratónianctivo	
	mousepos		posratón	
	cleartext	ct	limpiatexto	lt
	setcursor		poncursor	
	cursor		cursor	
	setmargins		ponmárgenes	
Acceso a Archivos				
	openread		archivolectura	
	openwrite		archivoescritura	
	openappend		archivoagregar	
	openupdate		archivomodificar	
	close		cierraarchivo	
	allopen		archivosabiertos	
	closeall		cerrartodos	
	erasefile	erf	borraarchivo	bar
	dribble		creaarchivo	
	nodribble		finalarchivo	
	setread		ponlectura	
	setwrite		ponescritura	
	reader		últimoleído	
	writer		últimoescrito	
	setreadpos		ponposlectura	
	setwritepos		ponposescritura	
	readpos		leepos	
	writepos		escribepos	
	eofp		esfinal	
	eof?		final?	
ESTRUCTURAS DE DATOS				
Constructores				
	word		palabra	
	list		lista	
	sentence	se	oración	or
	fput		ponprimero	
	lput		ponúltimo	
	array		arreglo	
	mdarray		mdarreglo	
	listtoarray		listaaarreglo	
	arraytolist		arregloalista	
	combine		combina	
	reverse		inverso	
	gensym		gensym	
Selectores				
	first		primero	
	firsts		primeros	
	last		último	

	butfirst	bf	menosprimero	mpr
	butfirsts	bfs	menosprimeros	mprs
	butlast	bl	menosúltimo	mu
	item		elemento	
	mditem		mdelemento	
	pick		sacaalazar	
	remove		quita	
	remdup		quitarepetidos	
	quoted		entrecomillas	
Mutadores				
	setitem		ponelemento	
	mdsetitem		mdponelemento	
	.setfirst		.ponprimero	
	.setbf		.ponmp	
	.setitem		.ponelemento	
	push		añade	
	pop		pop	
	queue		cola	
	dequeue		decola	
Predicados (Datos)				
	word?		palabra?	
	wordp		espalabra	
	list?		lista?	
	listp		eslista	
	array?		arreglo?	
	arrayp		esarreglo	
	empty?		vacío?	
	emptyp		esvacío	
	equal?		igual?	
	equalp		esigual	=
	before?		anterior?	
	beforep		esanterior	
	.eq		.ig	
	member?		miembro?	
	memberp		esmiembro	
	substring?		subcadena?	
	substringp		essubcadena	
	number?		número?	
	numberp		esnúmero	
	backslashed?		backslash?	
	backslashedp		esbackslash	
Consultas				
	count		cuenta	
	ascii		ascii	
	rawascii		rascii	
	char		caracter	
	member		miembro	

	lowercase		minúscula	
	uppercase		mayúscula	
	standout		mantenerfuera	
	parse		analiza	
	runparse		analizacorrida	
	time		tiempo	
	timemilli		tiempomili	
	OPCIONES DE LAS LÍNEAS DE COMANDOS			
	-w		-an	
	-h		-al	
	-f		-f	
	-p		-p	
	-e		-e	
	-l		-c	
	SALIR DEL SISTEMA			
	bye		chao	
	OPERACIONES LÓGICAS			
	and		y	
	or		o	
	not		no	
	GESTIÓN DE REDES			
	netstartup		iniciared	
	netshutdown		finred	
	netaccepton		sired	
	netacceptoff		nored	
	netacceptreceivevalue		leered	
	netacceptsendvalue		escribered	
	netconnecton		activared	
	netconnectoff		desactivared	
	netconnectsendvalue		envíavalorred	
	netconnectreceivevalue		leevalorred	
	COMANDOS ESPECIALES			
Comandos Especiales				
	caseignoredp		seignorómaymin	
	erract		errado	
	printdepthlimit		immaxprofundidad	
	printwidthlimit		immaxelementos	
	redefp		estáredefinido	
	startup		principio	
	machine		computador	
	MACROS			
	.macro		.macro	
	.defmacro		.definemacro	

	macroexpand		expandemacro	

Tabla 5: Traducción al español de Primitivas de MSWLogo
Fuente: Elaboración propia

Anexo IV-14

Traducción al español de las Librerías de Logo

A continuación, se presenta el listado de las primitivas contenidas dentro de la carpeta Logolib.

- #
- '
- abs
- arc
- arc2
- arraytolist
- ask
- buryall
- buryname
- cascade
- circle
- circle2
- closeall
- combine
- crossmap
- demo
- dequeue
- dir
- distance
- distancexyz
- do.until
- do.while
- edall
- edallbtn
- edn
- edns
- edpl
- edpls
- edps
- ellipsa2
- ellipse
- ellipse2
- erall
- ern
- erns
- erpl
- erpls
- erps
- filep
- filter
- find
- for
- foreach
- gensym
- goldenratio
- icon
- ignore
- invoke
- iseq
- listtoarray
- localmake
- macroexpand
- maketurtle
- map
- map.se
- mdarray
- mditem
- mdsetitem
- mousecopy
- name
- namelist
- orientation
- pen
- pennormal
- pi
- pick

- pllist
- poall
- pon
- pons
- pop
- popl
- popls
- pops
- pots
- push
- queue
- quoted
- reduce
- remdup
- remove
- rest
- reverse
- rseq
- savel
- setorientation
- setpen
- settextfont
- sign
- slowdraw
- textfont
- tone
- transfer
- tutor
- unburyall
- unburyname
- unicon
- until
- while
- xcor
- ycor

La traducción al español del nombre de estas primitivas, se encuentran en la tabla 5, mostrada el el anexo IV-13.

Anexo IV-15

Traducción al español de los mensajes de error de MSWLogo

A continuación se presenta una tabla con todos los mensajes de error en inglés que maneja MSWLogo, y su respectiva traducción al español para MundoLogo.

Error	Traducción
"I Lost an Error Message"	"Perdí un Mensaje de Error"
"Double Error"	"Error Doble"
"Logo: Fatal Internal Error.\n"	"Logo: Error Interno.\n"
"Logo: Out of Memory.\n"	"Logo: Fuera de Memoria"
"out of space"	"fuera del espacio"
"stack overflow"	"pila desbordada"
"turtle out of bounds"	"tortuga fuera del área de trabajo"
"couldn't initialize graphics"	"no se pude inicializar las gráficas"
"doesn\'t like"	"no me gusta"
"as input"	"como entrada"
"didn\'t output to"	"no era una salida de"
"not enough inputs to"	"no hay suficientes entradas para"
"APPLY doesn\'t like"	"No me gusta APLICA"
"too much inside ()\'s"	"demasiados parámetros dentro de ()\'"
"You don\'t say what to do with"	"No sé qué hacer con"
"too many (\'s"	"demasiados (\'s"
"has no value"	"no tiene valor"
"unexpected \')\'"	"no se esperaba \')\'"
"unexpected \']\'"	"no se esperaba \']\'"
"unexpected \'}\'"	"no se esperaba \'}\'"
"I don't know how to"	"Yo no sé cómo hacer"
"Can't find catch tag for"	"No encontré una etiqueta para \'capturar\'"
"is already defined"	"ya ha sido definido"
"Stopping..."	"Parando..."
"Already dribbling"	"Already dribbling"
"File system error:"	"Error del archivo del sistema:"
"Assuming you mean IFELSE, not IF"	"Asumo que te referías a SIOTRO, y no a SI"
"shadowed by local in procedure call"	"fue oculto en una llamada local a un procedimiento"
"Throw \"Error"	"Error del \'Throw\'"
"is a primitive"	"es una primitiva"
"Can't use TO inside a procedure"	"No se puede usar PARA dentro de un procedimiento"

"Can only use"	"Solo se puede usar"
"inside a procedure"	"dentro de un procedimiento"
"without TEST"	"sin PRUEBA"
"Macro returned"	"Devuelve un Macro"
"instead of a list"	"en lugar de una lista"
"END inside multi-line instruction.\n"	"FIN dentro de una instrucción multi-línea.\n"
"END inside multi-line instruction in"	"FIN dentro de una instrucción multi-línea en"
"Unknown error condition - internal error.\n"	"Error desconocido– error interno"
"You don't say what to do with %s\n"	"No sé que hacer con %s\n"
"Erract loop\n"	"Ciclo fallido\n"
"Pausing..."	"Pausando..."
"alloc=%d, freed=%d, used=%d\n"	"alloc=%d, liberado=%d, utilizado=%d\n"
"Print driver does not support this function"	"Los driver de la impresora no soportan esta función"
"Cannot draw image"	"No puedo dibujar la imagen"
"Cannot extract image"	"No puedo extraer la imagen"
"No Memory to Print"	"No hay memoria para imprimir"
"Changes in this edit session may be lost.\n\nDo you really want to exit MSWLogo?"	"Los cambios en esta sesión quizás se puedan perder.\n\n¿Realmente quieres salir de MundoLogo?"
"An Edit session is running"	"Una sesión está corriendo"
"MSWLogo does not like exiting while not halted.\n\nDo you really want to exit MSWLogo?"	"A MundoLogo no le gusta salir, mientras no esté detenido.\n\n¿Realmente tú quieres salir de MundoLogo?"
"Logo is not Halted"	"Logo no se ha detenido"
"MSWLogo does not like exiting while not halted.\n\nDo you really want to Halt MSWLogo?"	"A MundoLogo no le gusta salir mientras no haya sido detenido.\n\n¿Realmente quieres detener MundoLogo?"
"You will lose any changes if you don't save your workspace to disk.\n\nDo you really want to exit MSWLogo?"	"Perderás cualquier cambio si no salvas tu proyecto en el disco.\n\n¿Realmente quieres salir de MundoLogo?"
"You have not saved to disk"	"No has salvado la información en el disco"
"Write failed, Possibly no Memory"	"La escritura ha fallado, posiblemente no hay suficiente memoria"
"Could not Write .BMP"	"No puedo escribir un .BMP"
"Could not Open .BMP"	"No puedo abrir un .BMP"
"Unable to create Windows 3.0 bitmap"	"No habilitado para crear un bitmap de Windows 3.0"
"Not a Windows 3.0 bitmap"	"No es un bitmap de Windows 3.0"
"Cannot open bitmap file"	"No puedo abrir un archivo de bitmap"
"Executing a new will erase all definitions.\n\nContinue with New?"	"Ejecutando \'Nuevo\' se borrarán todos los procedimientos.\n\n¿Deseas ejecutar \'Nuevo\'?"
"You have not saved to disk"	"No has salvado en el disco"
"The file being loaded may over write your	"El archivo está siendo cargado, quizás

changes.\n\nContinue with Load?"	sobre escriba tus cambios.\n\n¿Deseas continuar cargando?"
"You must be in a procedure to use OUTPUT or STOP.\n"	"Debes estar en un procedimiento para usar SALIDA o PARA.\n"
"Bad arguments"	"Los argumentos son errados"
"Active Area"	"Área activa"
"Unexpected Error, Network may be shutdown"	"Ha ocurrido un error, puede que la red deje de funcionar"
"Network Error"	"Error en la red"
"The Environment variable TEMP is not defined\nUsing C:\\ as TEMP"	"El ambiente variable "
"Warning"	"Advertencia"
"Invalid Command Line"	"Línea de comando inválida"
"MSWLogo Screen"	"MundoLogo"
"AddList: Out of Memory"	"AddList: Fuera de memoria"

Tabla 6: Traducción al español de mensajes de error de MSWLogo
Fuente: Elaboración propia

Anexo IV-16

Traducción al español de los Menús de MSWLogo

A continuación se presentan las tablas que contienen la traducción al español de los menús y otros botones de MSWLogo, para MundoLogo.

Menú Principal

MSWLogo	MundoLogo
File Menu	Menú Archivo
• New	• Nuevo
• Load...	• Abrir...
• Save	• Guardar
• Save as...	• Guardar como...
• Edit	• Editar
• Erase	• Borrar
• Exit	• Salir
Bitmap Menu	Menú Imagen
• New	• Nuevo
• Load	• Abrir
• Save	• Guardar
• Save as...	• Guardar como...
• Print	• Imprimir
• Print setup	• Configurar página
• Active Area	• Área Activa
Set Menu	Menú Formato
• Pen size	• Tamaño de la Pluma
• Label font	• Fuente de Etiquetas
• Commander font	• Fuente de Comandos
• Pen color	• Color de la Pluma
• Flood color	• Color de Relleno

• Screen color	• Color de Fondo
Zoom Menu	Menú Zoom
• In	• Aumentar
• Out	• Disminuir
• Normal	• Normal
Help Menu	Ayuda
• Index	• Índice
• MCI	• MCI
• Using Help	• Usar Ayuda
• Tutorial	• Tutorial
• Demo	• Demo
• Examples	• Ejemplos
• Release Notes	• Notas
• About Logo	• Acerca de Logo
• About MSWLogo	• Acerca de MundoLogo

Tabla 7: Traducción al español de menús de MSWLogo
Fuente: Elaboración propia

Menú Editor de Procedimientos:

MSWLogo	MundoLogo
File Menu	Menú Archivo
• Save and Exit	• Guardar y Salir
• Save to Workspace	• Gardar Proyecto
• Print	• Imprimir
• Exit	• Salir
Edit Menu	Menú Editar
• Undo	• Deshacer
• Cut	• Cortar
• Copy	• Copiar
• Paste	• Pegar

• Delete • Clear All	• Borrar • Borrar Todo
Search Menu • Find • Replace • Next	Menú Búsqueda • Buscar • Reemplazar • Siguiente
Set Menu • Font	Menú Ambiente • Fuente
Test! Menu	Menú Prueba!
Help Menu • Index • Editor • Topic search	Menú Ayuda • Índice • Editor • Búsqueda por Tópicos

Tabla 8: Traducción al español de menús de MSWLogo
Fuente: Elaboración propia

Botones de la Ventana de Comandos

MSWLogo	MundoLogo
Halt	Detener
Trace	Pausa
Pause	Traza
Status	Estatus
Step	Pasos
Reset	Borrar
Execute	Ejecutar
Edall	Editar

Tabla 9: Traducción al español de menús de MSWLogo
Fuente: Elaboración propia

Anexo IV-17

Código fuente de la nuevas primitivas incorporadas a MundoLogo

LÍNEA

```
para DrawLine :box
haz "x primero :box
haz "y primero menosprimero :box
haz "w primero menosprimero menosprimero :box
haz "h último :box
sp ponxy :x :y cp
ponxy :x+:w :y-:h
fin

para LineResize :pt
plumainversa
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
DrawLine (lista primero :StartPoint último :StartPoint -:w :h)
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
DrawLine (lista primero :StartPoint último :StartPoint -:w :h)
fin

para OnLeftDown
haz "pt posratón
plumainversa
haz "StartPoint :pt
haz "EndPoint :pt
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
DrawLine (lista primero :StartPoint último :StartPoint -:w :h)
haz "Down "Verdad
fin

para OnLeftUp
haz "pt posratón
plumainversa
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
DrawLine (lista primero :StartPoint último :StartPoint -:w :h)
ponpluma
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
DrawLine (lista primero :StartPoint último :StartPoint -:w :h)
haz "Down "Falso
fin

para OnMove
haz "pt posratón
si no :Down [parar]
si :Down [LineResize :pt]
```

```
fin
para línea
ratóninactivo
ponpluma
ocultatortugas
haz "Down "Falso
ratónactivo [OnLeftDown] [OnLeftUp] [] [] [OnMove]
fin

oculta "línea
oculta "DrawLine
oculta "LineResize
oculta "OnLeftDown
oculta "OnLeftUp
oculta "OnMove
```

CUADRADO

```
para DrawBox :box
haz "x primero :box
haz "y primero menosprimero :box
haz "w primero menosprimero menosprimero :box
haz "h último :box
sp ponxy :x :y cp
ponxy :x+:w :y
ponxy :x+:w :y-:h
ponxy :x  :y-:h
ponxy :x  :y
fin

para DrawToolMove :pt
plumainversa
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
DrawBox (lista primero :StartPoint último :StartPoint -:w :h)
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
DrawBox (lista primero :StartPoint último :StartPoint -:w :h)
fin

para CuBotonAb
haz "pt posratón
plumainversa
haz "StartPoint :pt
haz "EndPoint :pt
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
DrawBox (lista primero :StartPoint último :StartPoint -:w :h)
haz "Down "Verdad
fin

para CuBotonAr
haz "pt posratón
plumainversa
haz "w (primero :StartPoint) - (primero :EndPoint)
```

```
haz "h (último :StartPoint) - (último :EndPoint)
DrawBox (lista primero :StartPoint último :StartPoint -:w :h)
ponpluma
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
DrawBox (lista primero :StartPoint último :StartPoint -:w :h)
haz "Down "Falso
fin
para CuMove
haz "pt posratón
si no :Down [parar]
si :Down [DrawToolMove :pt]
fin
para cuadrado
ratóninactivo
ponpluma
ocultatortugas
haz "Down "Falso
ratónactivo [CuBotonAb] [CuBotonAr] [] [] [CuMove]
fin
oculta "cuadrado
oculta "DrawBox
oculta "DrawToolMove
oculta "CuBotonAb
oculta "CuBotonAr
oculta "CuMove
```

CUADRADORELLENO

```
para CenterBox :box
haz "x primero :box
haz "y primero menosprimero :box
haz "w primero menosprimero menosprimero :box
haz "h último :box
haz "relleno colorrelleno
poncolorrelleno colorpluma
si y :h > 0 :w > 0 [sp ponxy :x - :w :y - :h cp rellenaárea :w :h];bueno
si y :h < 0 :w > 0 [sp ponxy :x - :w :y cp rellenaárea :w -:h];bueno
si y :h < 0 :w < 0 [sp ponxy :x :y cp rellenaárea -:w -:h];bueno
si y :h > 0 :w < 0 [sp ponxy :x :y-:h cp rellenaárea -:w :h]
poncolorrelleno :relleno
fin
para RDrawBox :box
haz "x primero :box
haz "y primero menosprimero :box
haz "w primero menosprimero menosprimero :box
haz "h último :box
sp ponxy :x :y cp
ponxy :x+:w :y
ponxy :x+:w :y-:h
ponxy :x    :y-:h
ponxy :x    :y
fin
```

```
para RDrawToolMove :pt
plumainversa
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
RDrawBox (lista primero :StartPoint último :StartPoint -:w :h)
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
RDrawBox (lista primero :StartPoint último :StartPoint -:w :h)
fin

para CuRBotonAb
haz "pt posratón
plumainversa
haz "StartPoint :pt
haz "EndPoint :pt
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
RDrawBox (lista primero :StartPoint último :StartPoint -:w :h)
haz "Down "Verdad
fin

para CuRBotonAr
haz "pt posratón
plumainversa
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
RDrawBox (lista primero :StartPoint último :StartPoint -:w :h)
ponpluma
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
RDrawBox (lista primero :StartPoint último :StartPoint -:w :h)
centerbox (lista primero :StartPoint último :StartPoint -:w :h)
haz "Down "Falso
fin

para CuRMove
haz "pt posratón
si no :Down [parar]
si :Down [RDrawToolMove :pt]
fin

para cuadradorelleno
ratóninactivo
ponpluma
ocultatortugas
haz "Down "Falso
ratónactivo [CuRBotonAb] [CuRBotonAr] [] [] [CuRMove]
fin

oculta "cuadradorelleno
oculta "RDrawBox
oculta "CenterBox
oculta "RDrawToolMove
oculta "CuRBotonAb
oculta "CuRBotonAr
```

```
oculta "CuRMove
```

RCÍRCULO

```
para DibujaCírculo :pt
plumainversa
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
elipse :w :h
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
Elipse :w :h
fin
para CirBotonAb
haz "pt posratón
plumainversa
haz "StartPoint :pt
haz "EndPoint :pt
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
Elipse :w :h
haz "Down "Verdad
fin
para CirBotonAr
haz "pt posratón
plumainversa
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
elipse :w :h
ponpluma
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
elipse :w :h
haz "Down "Falso
fin
para CirMove
haz "pt posratón
si no :Down [parar]
si :Down [DibujaCírculo :pt]
fin
para rcírculo
ratóninactivo
ponpluma
sp
haz "Down "Falso
ocultatortugas
ratónactivo [CirBotonAb] [CirBotonAr] [] [] [CirMove]
fin
oculta "rcírculo
```

```
oculta "DibujaCírculo
oculta "CirBotonAb
oculta "CirBotonAr
oculta "CirMove
```

CÍRCULORELLENO

```
para RDibujaCírculo :pt
plumainversa
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
elipse :w :h
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
Elipse :w :h
fin

para RCirBotonAb
haz "pt posratón
plumainversa
haz "StartPoint :pt
haz "EndPoint :pt
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
Elipse :w :h
haz "Down "Verdad
fin

para RCirBotonAr
haz "pt posratón
plumainversa
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
elipse :w :h
ponpluma
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
elipse :w :h
rellenar :w :h
haz "Down "Falso
fin

para rellenar :w :h
haz "tamaño grosor
pontamañopluma [3 3]
si (o (y (:w > 0) (:h > 0)) (y (:w < 0) (:h > 0))) [haz.mientras [elipse :w :h haz "w :w-1] ~
  [:w > 0] ~
  haz.mientras [elipse :w :h haz "h :h-1] ~
  [:h > 0]]
si (y (:w < 0) (:h < 0)) [haz.mientras [elipse :w :h haz "h :h+1] ~
  [:h < 0] ~
  haz.mientras [elipse :w :h haz "w :w+1] ~
  [:w < 0]]
```

```
si (y (:w > 0) (:h < 0)) [haz.mientras [elipse :w :h haz "h :h+1] ~
  [:h < 0] ~
  haz.mientras [elipse :w :h haz "w :w-1] ~
  [:w > 0]]
pontamañopluma :tamaño
fin

para RCirMove
haz "pt posratón
si no :Down [parar]
si :Down [RDibujaCírculo :pt]
fin

para círculorelleno
ratóninactivo
ponpluma
sp
haz "Down "Falso
ocultatortugas
ratónactivo [RCirBotonAb] [RCirBotonAr] [] [] [RCirMove]
fin

oculta "círculorelleno
oculta "RDibujaCírculo
oculta "RCirBotonAb
oculta "RCirBotonAr
oculta "RCirMove
```

COPIA

```
para CDrawBox :box
haz "x primero :box
haz "y primero menosprimero :box
haz "w primero menosprimero menosprimero :box
haz "h último :box
sp ponxy :x :y cp
ponxy :x+:w :y
ponxy :x+:w :y-:h
ponxy :x    :y-:h
ponxy :x    :y
fin

para CDrawToolMove :pt
plumainversa
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
CDrawBox (lista primero :StartPoint último :StartPoint -:w :h)
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
CDrawBox (lista primero :StartPoint último :StartPoint -:w :h)
fin

para CoBotonAb
haz "pt posratón
```

```
plumainversa
haz "StartPoint :pt
haz "EndPoint :pt
sp ponpos :StartPoint cp
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
CDrawBox (lista primero :StartPoint último :StartPoint -:w :h)
haz "Down "Verdad
fin

para CoBotonAr
haz "pt posratón
plumainversa
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
CDrawBox (lista primero :StartPoint último :StartPoint -:w :h)
ponpluma
haz "EndPoint :pt
haz "w (primero :StartPoint) - (primero :EndPoint)
haz "h (último :StartPoint) - (último :EndPoint)
;copiaárea -:w :h
si y :h > 0 :w > 0 [sp ponxy :x - :w :y - :h copiaárea :w :h];bueno
si y :h < 0 :w > 0 [sp ponxy :x - :w :y copiaárea :w -:h];bueno
si y :h < 0 :w < 0 [sp ponxy :x :y copiaárea -:w -:h];bueno
si y :h > 0 :w < 0 [sp ponxy :x :y-:h copiaárea -:w :h]
haz "Down "Falso
fin

para CoMove
haz "pt posratón
si no :Down [parar]
si :Down [CDrawToolMove :pt]
fin

para copia
ratóninactivo
ponpluma
ocultatortuga
haz "Down "Falso
ratónactivo [CoBotonAb] [CoBotonAr] [] [] [CoMove]
fin

oculta "copia
oculta "CDrawBox
oculta "CDrawToolMove
oculta "CoBotonAb
oculta "CoBotonAr
oculta "CoMove
```

RATÓN

```
para posición
haz "ts tortugas
haz "t tortuga
```

```
siotro (= :ts 0) [hayuna] [haymuchas :ts]
fin

para haymuchas :ts
haz "i 0
mientras [:i < :ts+1] ~
        [pontortuga :i haz "x primero pos ~
        haz "y último pos ~
        haz "xr primero posratón ~
        haz "yr último posratón ~
        haz "ru rumbo ~
        haz "i :i+1 ~
        si (o (:ru > 329) (:ru < 31)) ~
        [si (y (:x+6 > :xr) (:x-6 < :xr)) ~
                [si (y (:y+9 > :yr) (:y-1 < :yr)) ~
                        [haz "i :i+:ts mueveTortuga]]] ~
        si (y (:ru > 30) (:ru < 61)) ~
        [si (y (:x+8 > :xr) (:x-2 < :xr)) ~
                [si (y (:y+9 > :yr) (:y-6 < :yr)) ~
                        [haz "i :i+:ts mueveTortuga]]] ~
        si (y (:ru > 60) (:ru < 91)) ~
        [si (y (:x+10 > :xr) (:x-2 < :xr)) ~
                [si (y (:y+9 > :yr) (:y-9 < :yr)) ~
                        [haz "i :i+:ts mueveTortuga]]] ~
        si (y (:ru > 90) (:ru < 121)) ~
        [si (y (:x+11 > :xr) (:x-3 < :xr)) ~
                [si (y (:y+8 > :yr) (:y-10 < :yr)) ~
                        [haz "i :i+:ts mueveTortuga]]] ~
        si (y (:ru > 120) (:ru < 151)) ~
        [si (y (:x+10 > :xr) (:x-5 < :xr)) ~
                [si (y (:y+5 > :yr) (:y-10 < :yr)) ~
                        [haz "i :i+:ts mueveTortuga]]] ~
        si (y (:ru > 150) (:ru < 211)) ~
        [si (y (:x+7 > :xr) (:x-7 < :xr)) ~
                [si (y (:y+1 > :yr) (:y-11 < :yr)) ~
                        [haz "i :i+:ts mueveTortuga]]] ~
        si (y (:ru > 210) (:ru < 241)) ~
        [si (y (:x+5 > :xr) (:x-7 < :xr)) ~
                [si (y (:y+5 > :yr) (:y-10 < :yr)) ~
                        [haz "i :i+:ts mueveTortuga]]] ~
        si (y (:ru > 240) (:ru < 271)) ~
        [si (y (:x+1 > :xr) (:x-10 < :xr)) ~
                [si (y (:y+8 > :yr) (:y-8 < :yr)) ~
                        [haz "i :i+:ts mueveTortuga]]] ~
        si (y (:ru > 270) (:ru < 301)) ~
        [si (y (:x+4 > :xr) (:x-10 < :xr)) ~
                [si (y (:y+8 > :yr) (:y-7 < :yr)) ~
                        [haz "i :i+:ts mueveTortuga]]] ~
        si (y (:ru > 300) (:ru < 330)) ~
        [si (y (:x+6 > :xr) (:x-6 < :xr)) ~
                [si (y (:y+9 > :yr) (:y-5 < :yr)) ~
                        [haz "i :i+:ts mueveTortuga]]]]
fin

para mueveTortuga
ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]
fin

para hayuna
haz "x primero pos
```

```
haz "y último pos
haz "xr primero posratón
haz "yr último posratón
haz "ru rumbo
si (o (:ru > 329) (:ru < 31)) ~
[si (y (:x+6 > :xr) (:x-6 < :xr)) ~
        [si (y (:y+9 > :yr) (:y-1 < :yr)) ~
                [ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]]]]

si (y (:ru > 30) (:ru < 61)) ~
[si (y (:x+8 > :xr) (:x-2 < :xr)) ~
        [si (y (:y+9 > :yr) (:y-6 < :yr)) ~
                [ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]]]]

si (y (:ru > 60) (:ru < 91)) ~
[si (y (:x+10 > :xr) (:x-2 < :xr)) ~
        [si (y (:y+9 > :yr) (:y-9 < :yr)) ~
                [ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]]]]

si (y (:ru > 90) (:ru < 121)) ~
[si (y (:x+11 > :xr) (:x-3 < :xr)) ~
        [si (y (:y+8 > :yr) (:y-10 < :yr)) ~
                [ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]]]]

si (y (:ru > 120) (:ru < 151)) ~
[si (y (:x+10 > :xr) (:x-5 < :xr)) ~
        [si (y (:y+5 > :yr) (:y-10 < :yr)) ~
                [ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]]]]

si (y (:ru > 150) (:ru < 211)) ~
[si (y (:x+7 > :xr) (:x-7 < :xr)) ~
        [si (y (:y+1 > :yr) (:y-11 < :yr)) ~
                [ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]]]]

si (y (:ru > 210) (:ru < 241)) ~
[si (y (:x+5 > :xr) (:x-7 < :xr)) ~
        [si (y (:y+5 > :yr) (:y-10 < :yr)) ~
                [ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]]]]

si (y (:ru > 240) (:ru < 271)) ~
[si (y (:x+1 > :xr) (:x-10 < :xr)) ~
        [si (y (:y+8 > :yr) (:y-8 < :yr)) ~
                [ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]]]]

si (y (:ru > 270) (:ru < 301)) ~
[si (y (:x+4 > :xr) (:x-10 < :xr)) ~
        [si (y (:y+8 > :yr) (:y-7 < :yr)) ~
                [ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]]]]

si (y (:ru > 300) (:ru < 330)) ~
[si (y (:x+6 > :xr) (:x-6 < :xr)) ~
        [si (y (:y+9 > :yr) (:y-5 < :yr)) ~
                [ratónactivo [sp] [sp ratóninactivo] [] [] [sp ponpos posratón]]]]

fin

para ratón
ratónactivo [posición] [] [] [] []
fin
```

```
oculta "ratón
oculta "posición
oculta "hayuna
oculta "haymuchas
oculta "mueveTortuga
```

OCULTATORTUGAS

```
para ocultatortugas
haz "ts tortugas
pontortuga 0
ocultatortuga
si (:ts > 0) [desde [i 0 :ts 1] [pontortuga :i ocultatortuga]]
fin

oculta "ocultatortugas
```

MUESTRATORTUGAS

```
para muestratortugas
haz "ts tortugas
pontortuga 0
ocultatortuga
si (:ts > 0) [desde [i 0 :ts 1] [pontortuga :i muestratortuga]]
fin

oculta "muestratortugas
```

Anexo IV-18

Encuesta de Evaluación del software para profesores

FICHA DE EVALUACIÓN DEL SOFTWARE MUNDOLOGO

PARTE I: Identificación del evaluador	
Nombre del organismo, institución o empresa evaluadora:	
Persona evaluadora:	Cargo:
Profesión:	Email:
Lugar y fecha de la evaluación:	

PARTE II: Descripción del software	
¿Con que tipo de usuarios trabajaría el software?	Preescolar:____ 1era Etapa de Básica:____ 2da Etapa de Básica:____ 3era Etapa de Básica:____ Ciclo Diversificado:____ Educación Superior:____ Adultos:____ Público en General:____
¿Para que áreas del Conocimiento utilizaría el software?	Lenguaje:___ Matemática:___ Ciencias Sociales:___ Ciencias Naturales:___ Cultura General:___ Valores:___ Otra, especifique: _____ Tema(s) Específico(s):

PARTE III: Evaluación			
Diseño Conceptual			
		Si	**No**
Las reglas básicas para trabajar con el software se captan fácilmente			
Es posible trabajar con el software sin leer las instrucciones			
Las "reglas del juego" se mantienen a lo largo del software			
El software puede trabajarse bajo un enfoque lúdico			
Comentarios adicionales:			
Diseño Conceptual		**Aprobado con reservas**	**Reprobado**
Aspectos Pedagógicos			
		Si	**No**
Puede trabajarse contenido del currículum con el software			

Se adapta al nivel educativo del usuario		
Puede ser utilizado en actividades de enseñanza y aprendizaje con distintos niveles de dificultad		
Presenta caminos alternos adecuados a los distintos ritmos de aprendizaje		
Incentiva la creatividad de los usuarios		
Presenta retroalimentación efectiva y oportuna		
Es interactivo		
El léxico utilizado se adapta al contexto nacional		
Cumple con las normas básicas de redacción y ortografía		
Comentarios adicionales:		

Aspectos Pedagógicos		**Aprobado con reservas**	**Reprobado**

Aspectos Técnicos y de Interfaz		
	Si	**No**
Corre apropiadamente		
Existe confiabilidad en la ejecución		
Provee ayudas adecuadas y suficientes		
El tiempo de respuesta es adecuado		
Se explotan adecuadamente los elementos multimedia		
El sonido funciona correctamente		
Es fácil salir de la aplicación		
Comentarios adicionales:		

Aspectos Técnicos y de Interfaz		**Aprobado con reservas**	**Reprobado**

PARTE III: Evaluación (Continuación)		
Aspectos de Diseño Gráfico		
	Si	**No**
El tamaño de los elementos presentados favorece una buena visualización		
Se mantiene una identidad gráfica constante		
El tipo de letra y el estilo de diseño son coherentes		
El diseño gráfico es armónico con el tratamiento del tema		
Comentarios adicionales:		

Aspectos de Diseño Gráfico		Aprobado con reservas	Reprobado

Juicio Global sobre el Software

Nota: Versión original por el Lic. Assaf Yamin.
Adaptado y modificado por Dojelin Quijada y Sandra Quezada.

Anexo IV-19

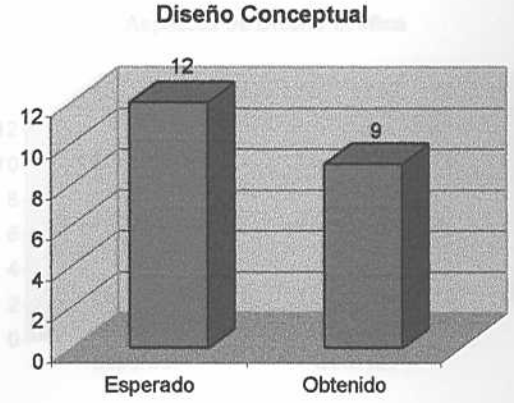
Resultados arrojados por las encuestas aplicadas a los profesores

Para el análisis de los resustados obtenidos en las encuestas aplicadas a los niños, el primer paso del proceso consistió en clasificar las preguntas según la variable sobre la cual se quería recabar información: Diseño Conceptual, Aspectos Pedagógicos, Aspectos Técnicos y de Interfaz y Aspecto de diseño Gráfico.

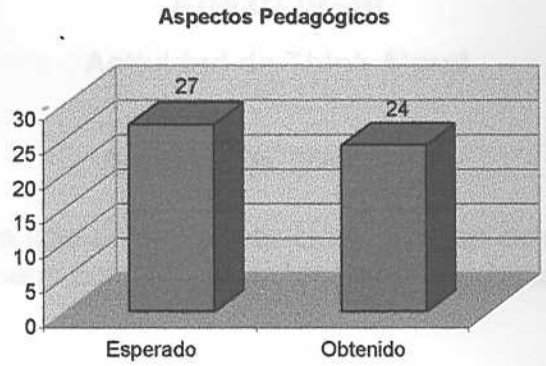
Para cada una de las variables se dispuso que se sumaría un punto (1) cada vez que la respuesta obtenida fuera igual a la esperada (Respuestas afirmativa "SI"). A partir de aquí se procedió a tabular los valores esperado y obtenidos para cada una de las variables.

Para la primera variable "Diseño Conceptual" el valor esperado se calculó, multiplicando el número de preguntas correspondientes a esta variable (4) por el número de encuestas aplicadas (3);

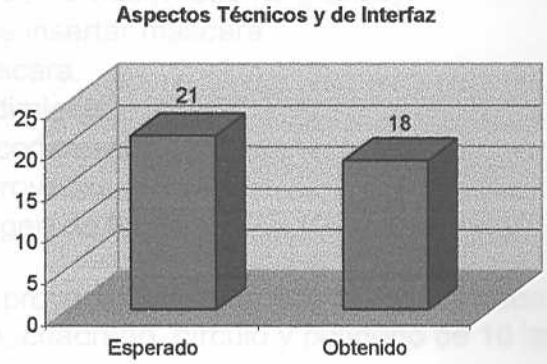
4 x 3 = 12 Valor esperado para la variable Diseño Conceptual;
y así sucesivamente para cada una de las variables.



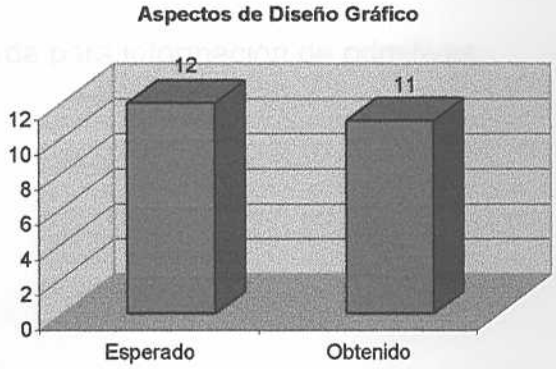
Gráfica 6: Valor esperado y obtenido para la variable Diseño Conceptual



Gráfica 7: Valor esperado y obtenido para la variable Aspectos Pedagógicos



Gráfica 8: Valor esperado y obtenido para la variable Aspectos Técnicos y de Interfaz



Gráfica 9: Valor esperado y obtenido para la variable Aspectos de Diseño Gráfico

Anexo IV-20

Actividad de Think Aloud

Actividad de Think Aloud

1.- Pintar un paisaje que contenga por lo menos una casa, dos árboles y un espacio "corral" y luego colocar una máscara del cochinito y del hombre a la tortuga.

- Seleccionar línea o lápiz.
- Seleccionar color.
- Realizar el dibujo en el área de gráfico.
- Seleccionar otras herramientas de dibujo.
- Volver al modo normal (modo no gráfico).
- Seleccionar e insertar máscara.
- Cambiar máscara.
- Crear procedimiento.
- Ejecutar procedimiento.
- Guardar el proyecto.
- Guardar imagen de fondo.

2.- Crear un nuevo proyecto y realizar cuatro figuras geométricas con primitivas (triángulo, cuadrado, círculo y polígono de 10 lados).

- Crear un nuevo proyecto.
- Borrar procedimientos.
- Realizar procedimientos.
- Ejecutar procedimientos.

3.- Consultar la ayuda para información de primitivas.

Anexo IV-21

Encuesta de Evaluación del software para niños

Instrucciones

En las páginas siguientes aparece una serie de enunciados relacionados con el programa que haz utilizado. Nos interesa saber que opinas sobre cada afirmación.
La escala utilizada es la siguiente:

3.- Siempre
2.- Algunas Veces
1.- Nunca

Marque con una "X" la opción que más te parezca, por ejemplo, si marca 3 en cualquiera de las afirmaciones, eso indica que estás totalmente de acuerdo con ella.

	Siempre	Algunas Veces	Nunca
Disfruté el uso de este programa.			
En ocasiones sentí que perdía el gusto por utilizar este programa.			
Las respuestas dadas por el programa, fueron buenas para ayudarme a utilizarlo.			
Utilizar el programa es emocionante.			
Sentí que cuando fallaba en mis respuestas, el programa NO me daba pistas para hallar el error.			
Si yo quiero, el programa me permite ir despacio o rápido en mi aprendizaje.			
Los contenidos me parecieron fáciles.			
La ayuda me permitía aprender elementos que anteriormente NO había entendido.			
Los contenidos presentados por el programa son de poco uso práctico.			
El programa me dio la oportunidad de ejercitarme suficientemente.			
Algunas veces sentí desmotivación por el tipo de respuestas dadas en el computador.			
El programa me permitió hacer prácticas verdaderamente importantes.			

El uso de la ayuda me desmotiva en mi aprendizaje.			
Me agradó la forma en como este programa me impulsaba a seguir en mi proceso de aprendizaje.			
El programa NO me permite ir a mi propio ritmo.			
Pienso que las personas que utilizan el computador para aprender tienen ventajas sobre los que no lo usan.			
Después de haber utilizado el programa me siento en capacidad de aplicar lo aprendido.			
Mientras utilizaba el programa, me mantuve animado a realizar las actividades propuestas.			
Los colores utilizados por el programa son agradables.			
La letra utilizada permite leer con facilidad.			
Utiliza el tipo de letra NO adecuado.			

	Si	No
Me gustaría volver a participar en otra prueba de materiales educativos computarizados		

¿Por qué?

_____

_____

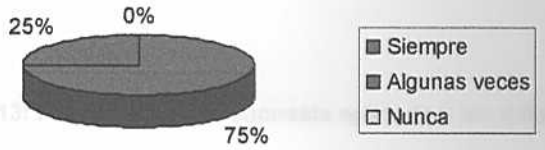
_____

_____

Anexo IV-22

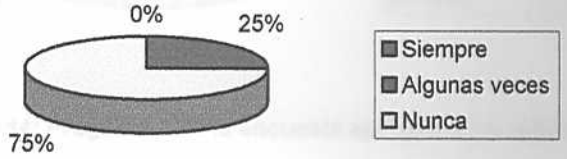
Resultados arrojados por las encuestas aplicadas a los niños

Disfrute el uso de éste programa



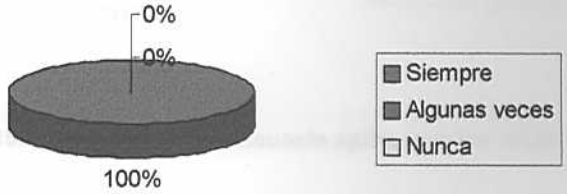
Gráfica 10: Pregunta 1 de la encuesta aplicada a los niños

En ocasiones sentí que perdía el gusto por utilizar este programa

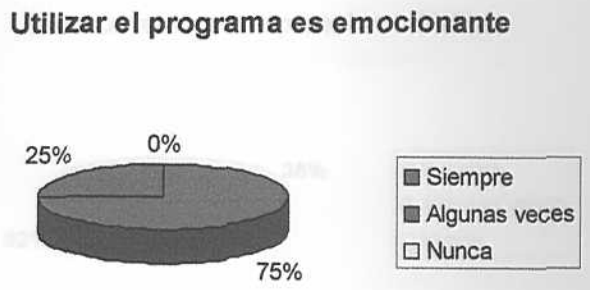


Gráfica 11: Pregunta 2 de la encuesta aplicada a los niños

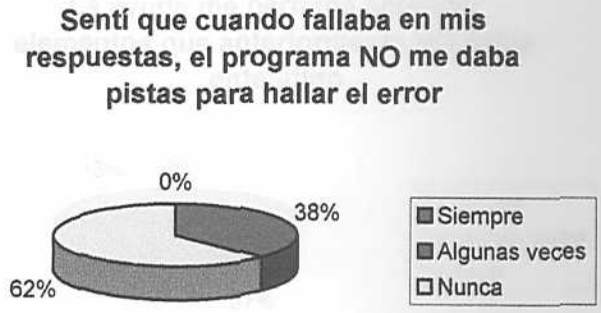
Las respuestas dadas por el programa, fueron buenas para ayudarme a utilizarlo



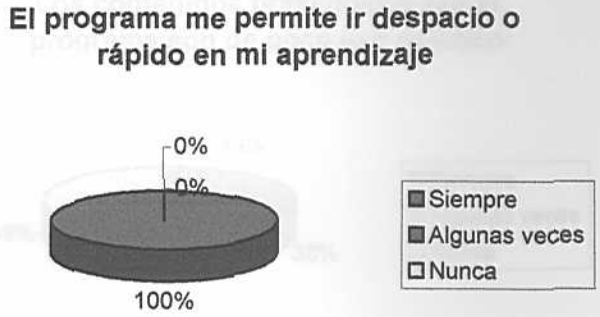
Gráfica 12: Pregunta 3 de la encuesta aplicada a los niños



Gráfica 13: Pregunta 4 de la encuesta aplicada a los niños

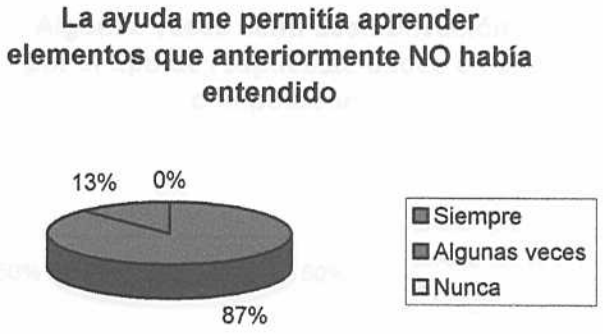


Gráfica 14: Pregunta 5 de la encuesta aplicada a los niños

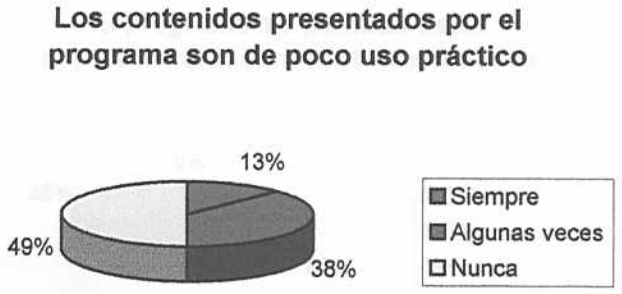


Gráfica 15: Pregunta 6 de la encuesta aplicada a los niños

Gráfica 16: Pregunta 7 de la encuesta aplicada a los niños

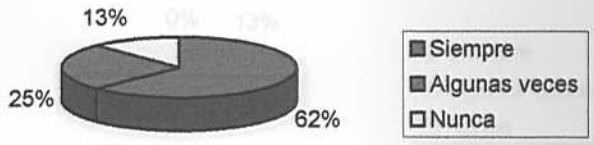


Gráfica 17: Pregunta 8 de la encuesta aplicada a los niños



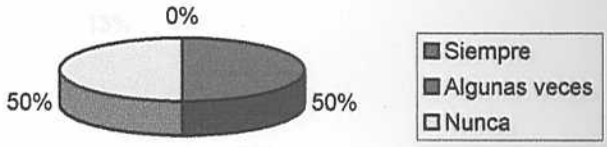
Gráfica 18 Pregunta 9 de la encuesta aplicada a los niños

El programa me dio la oportunidad de ejercitarme suficientamente



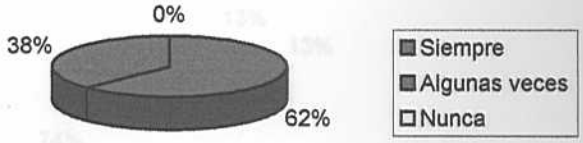
Gráfica 19: Pregunta 10 de la encuesta aplicada a los niños

Algunas veces sentí desmotivación por el tipo de respuestas dadas en el computador

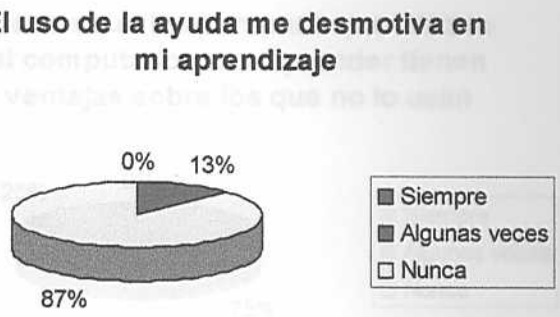


Gráfica 20: Pregunta 11 de la encuesta aplicada a los niños

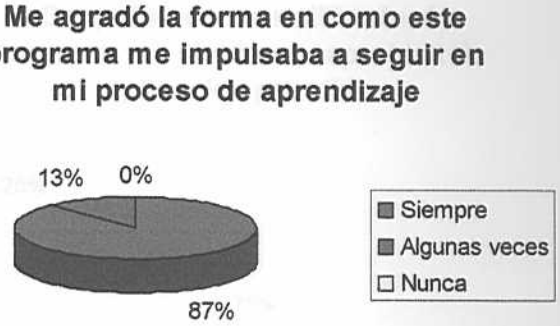
El programa me permitió hacer prácticas verdaderamente importantes



Gráfica 21: Pregunta 12 de la encuesta aplicada a los niños

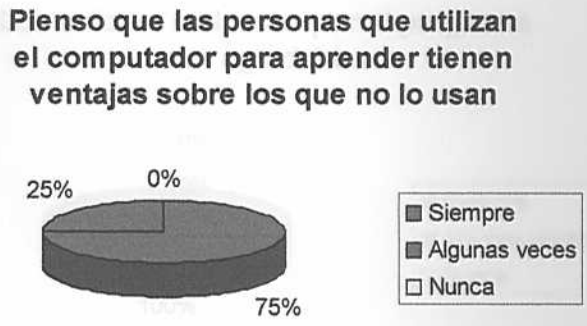


Gráfica 22: Pregunta 13 de la encuesta aplicada a los niños

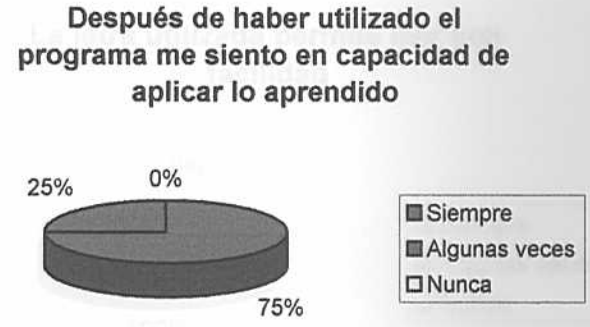


Gráfica 23: Pregunta 14 de la encuesta aplicada a los niños

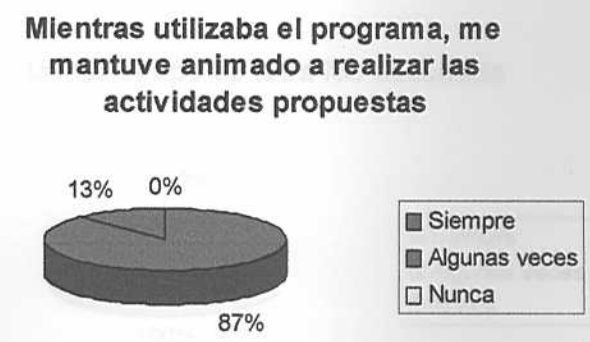
Gráfica 24: Pregunta 15 de la encuesta aplicada a los niños



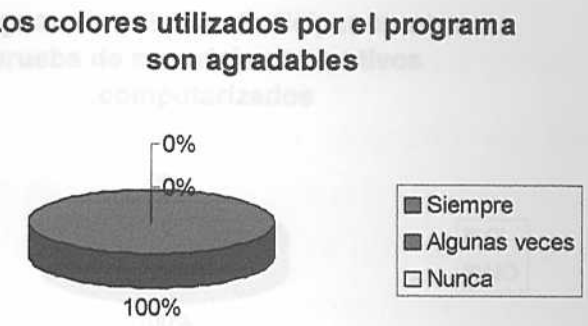
Gráfica 25: Pregunta 16 de la encuesta aplicada a los niños



Gráfica 26: Pregunta 17 de la encuesta aplicada a los niños

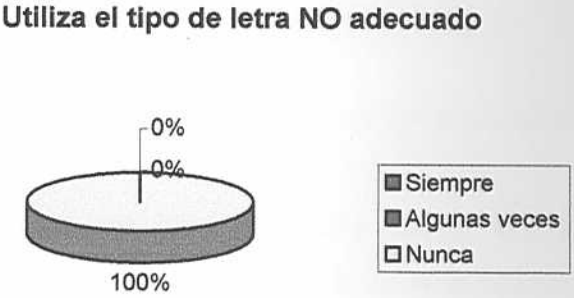


Gráfica 27: Pregunta 18 de la encuesta aplicada a los niños



Gráfica 28: Pregunta 19 de la encuesta aplicada a los niños

Gráfica 29: Pregunta 20 de la encuesta aplicada a los niños



Gráfica 30: Pregunta 21 de la encuesta aplicada a los niños

Me gustaría volver a participar en otra prueba de materiales educativos computarizados

Gráfica 31: Pregunta 22 de la encuesta aplicada a los niños

Anexo IV-23

Análisis de las encuestas aplicadas a los niños

Para el análisis de los resustados obtenidos en las encuestas aplicadas a los niños, el primer paso del proceso consistió en clasificar las preguntas según la variable sobre la cual se quería recabar información: Motivación, Contenidos, Ejercitación y práctica, Evaluación, Aprendizaje, Ritmo, Interfaz y Actitud global hacia el uso de materiales educativos computarizados (MEC). La estructura de la encuesta se sintetiza en el cuadro siguiente, en el que para cada variable se detalla que preguntas corresponde y su tendencia (directa: + e inversa: -) .

Motivación		Contenidos		Ejercitación y Práctica		Evaluación	
+	-	+	-	+	-	+	-
1	2	7		10	9	3	5
4	11			12			
14	13			17			
18							

Aprendizaje		Ritmo		Interfaz		Actitud Global MEC	
+	-	+	-	+	-	+	-
8		6	15	19	21	16	
				20		22	

Cuadro : Estructura de la encuesta aplicada a los niños

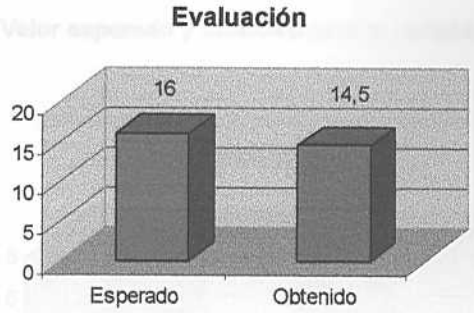
Para cada una de las variables se dispuso que se sumaría un punto (1) cada vez que la respuesta obtenida fuera igual a la esperada y medio punto (0.5) cuando la respuesta obtenida fuera "Algunas Veces". A partir de aquí se procedió a tabular los valores esperado y obtenidos para cada una de las variables.

Para la primera variable "Motivación" el valor esperado se calculó, multiplicando el número de preguntas correspondientes a esta variable (7) por el número de encuestas aplicadas (8);

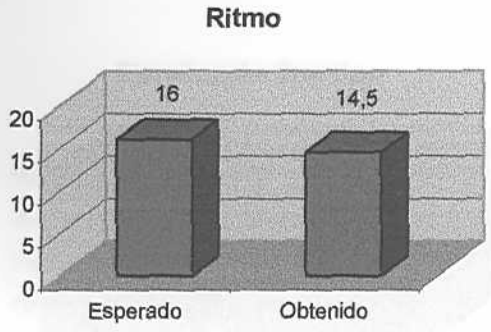
7 x 8 = 56 Valor esperado para la variable motivación;

y así sucesivamente para cada una d elas variables.

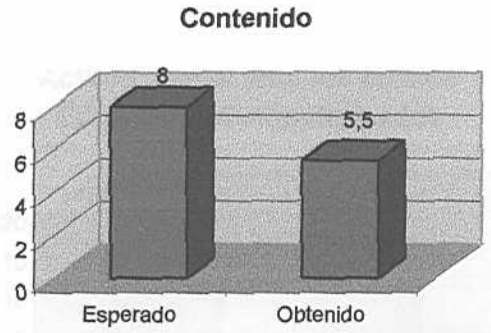
Grafica 32: Valor esperado y obtenido para la variable Motivación



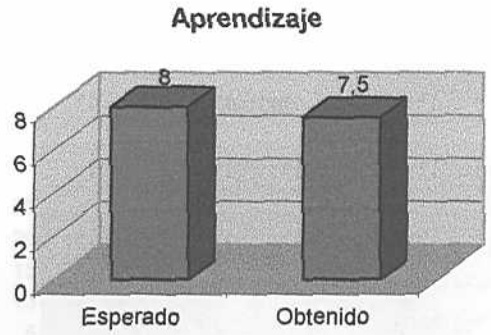
Grafica 33: Valor esperado y obtenido para la variable Evaluación



Grafica 34: Valor esperado y obtenido para la variable Ritmo

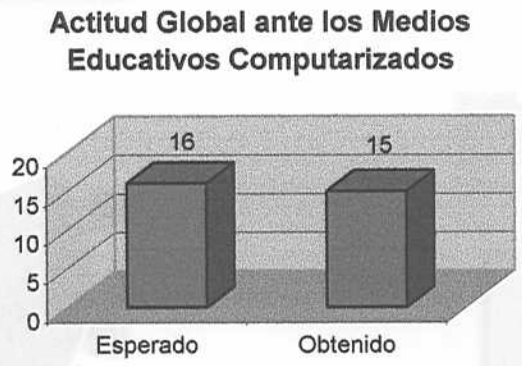


Grafica 35: Valor esperado y obtenido para la variable Contenido

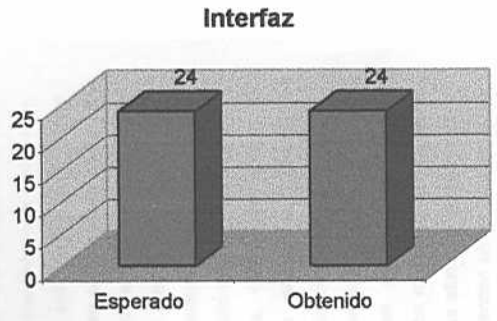


Grafica 36: Valor esperado y obtenido para la variable Apredizaje

Grafica 37: Valor esperado y obtenido para la variable Ejercitación Práctica



Grafica 38: Valor esperado y obtenido para la variable Actitud ante los MEC



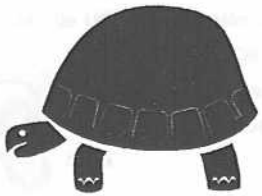
Grafica 39: Valor esperado y obtenido para la variable Interfaz

Anexo IV-24

Tríptico presentado durante la IV Feria de Tecnología Educativa 2004

La Computadora como Aprendiz

En el enfoque de la computadora como aprendiz, el estudiante que aprende ve la necesidad de enseñar lo que sabe a la computadora, no solo lo que el tiene que aprender sino también como están hechas estas cosas, las relaciones que tienen, etc. Todas estas actividades además de ser de mucho interés y altamente formativas, contribuyen a que el alumno refuerce la memorización. Como resultados de múltiples experiencias, los maestros han señalado que sus alumnos verdaderamente entienden un tema cuando se ven en la necesidad de explicárselo a otro, ya que esto no solo los obliga a repetir el tema, sino también a interiorizarlo.



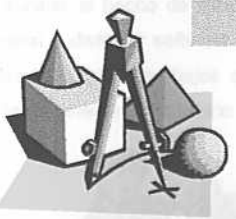
UCAB

Sandra Quezada (sandra_quezada_1@ucab.edu.ve
Dojelin Quijada (dojelin_quijada_1@ucab.edu.ve)
Alejandro del Mar (adelmar@ucab.edu.ve)

UCAB

MundoLogo Versión 1.0

La Experiencia de Aprender



Desarrolladoras:
Dojelin Quijada y Sandra Quezada

Ilustración 1: Parte 1 del tríptico de MundoLogo

Historia

A raíz del auge de las computadoras personales a finales de los años 70, se comenzó ha extender el uso del lenguaje Logo, creado por Seymour Papert, matemático nacido en África del Sur y uno de los pioneros de la inteligencia artificial. Durante muchos años Papert se dedicó al estudio de las posibilidades de la informática en la educación, así fue como nació Logo, un lenguaje de programación completo, cuyo objetivo es el de incentivas a los estudiantes a aprender y pensar. En el año de 1980, fue creada la compañía Logo Computer Systems, Inc. (LCSI), desarrolladora del conocido software Micromundos. Pero esta notable herramienta de desarrollo en lenguaje logo, tiene costos que van desde 99$ la vesión individual a 999$ la licencia por escuela.

De MSWLogo a MundoLogo

En vista del alto costo que tienen muchas de los softwares propietarios, aparece la idea de software Libre creada por Richard Stallman, la cual fue asumida por muchas persona en el mundo. Dentro de este campo se han desarrollado también, importantes ambientes Logo, como es el caso de UCBLogo, versión compatible para Macintosh, MSDOS, Windows y Unix, creada por Brian Hervey, autor del clásico Computer Science Logo Style.

Mas adelante, George Mills, usó como base de su desarrollo el núcleo de UCBLogo para crear su MSWLogo, el cual corre bajo Windows, incluyendo mejoras que son posibles en ese sistema operativo. Actualmente existen numerables variaciones y adapciones de este sofware, Entre estas adaptaciones podemos mencionar las traducciones al francés, chino, portugués y al español (México y España), además algunas de estas adaptaciones incluyen mejoras de la estabilidad del sistema (España), cambió de la imagen del cursor de un triángulo a una tortuga (Portugal), entre otras.

En octubre del 2003, a través del Profesor Alejandro del Mar, nace la idea de realizar una serie de mejoras y adaptaciones al software MSWLogo, las cuales se enfocan principalmente en realizar una traducción al español, lo más adaptado a las necesidades de los colegios de Fe y Alegría y Venezuela. Además la agregación de un modulo de dibujo, que permita facilitar la interacción del usuario con el sistema, al momento de realizar gráficos y dibujos. A través de esta herramienta podemos dibujar figuras geométricas en nuestra tarea de trabajo, modificar colores, copiar fragmentos, pegar, etc.

Una los aspectos más interesantes de este proyecto es que MundoLogo por su carácter de código abierto, es solo una parte de una idea que puede crecer y que con el tiempo se convierta en una herramienta cada vez más potente y superior. Todo esto aunado al hecho de que los costos de generar y distribuir software GNU son considerablemente más bajos que su contraparte, el software propietario.

Ilustración 2: Parte 2 del tríptico de MundoLogo