

Proyecto Fin de Carrera

Ingeniería Informática

Junio de 2011

AraWord:
Un procesador de textos
para comunicación aumentativa y adaptativa

Joaquín Pérez Marco

Director: Dr. Joaquín Ezpeleta Mateo



Departamento de
Informática e
Ingeniería de Sistemas
Universidad Zaragoza

Índice general

A. Documentación del desarrollo del <i>software</i>	1
A.1. Metodología de trabajo	1
A.2. Requisitos del software	2
A.2.1. <u>Requisitos funcionales</u>	2
A.2.2. <u>Requisitos no funcionales</u>	2
A.3. AraWord: casos de uso	3
A.4. Diseño de la aplicación	3
A.4.1. <u>Diseño de la base de datos</u>	4
A.4.2. <u>Subsistemas internos</u>	6
A.4.3. <u>Bibliotecas externas</u>	7
A.5. Pruebas	7
A.5.1. Pruebas unitarias	7
A.5.2. Pruebas de integración	8
A.5.3. Pruebas de sistema	8
A.5.4. Pruebas de usuario	8
A.6. Herramientas utilizadas	9
B. Distribución temporal del proceso	11
C. Manual de usuario	13
C.1. Introducción	13
C.2. Instalación de la aplicación	13
C.3. Primeros pasos	13
C.4. Manual de uso de AraWord	15
D. The GNU General Public License	19

Apéndice A

Documentación del desarrollo del *software*

A.1. Metodología de trabajo

Para el desarrollo de las tareas se ha seguido la metodología de trabajo llamada Scrum, que se enmarca dentro de las llamadas metodologías ágiles de programación. Scrum es una metodología para la gestión y desarrollo de *software* basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software. Popularizada a principios de este siglo, Scrum es un modelo de referencia que define un conjunto de prácticas y roles, y que puede tomarse como punto de partida para definir el proceso de desarrollo que se ejecutará durante un proyecto. Los roles se han simplificado dadas las características del proyecto (un único programador, un director de proyecto y algunos clientes externos) y se ha utilizado más el sistema de pequeños avances, llamados *sprints*.

Durante cada *sprint*, un periodo entre 15 y 30 días (la magnitud es definida por el equipo), el equipo crea un incremento de software potencialmente entregable (utilizable) mediante un pequeño ciclo del modelo clásico en cascada (requisitos-análisis-diseño-implementación-pruebas). El conjunto de características que forma parte de cada *sprint* viene derivado del *Product Backlog*, que es un conjunto de requisitos de alto nivel priorizados que definen el trabajo a realizar.

Un principio clave de Scrum es el reconocimiento de que durante un proyecto los clientes pueden cambiar de idea sobre lo que quieren y necesitan (a menudo llamado *requirements churn*), y que los desafíos impredecibles no pueden ser fácilmente enfrentados de una forma predictiva y planificada. Por lo tanto, Scrum adopta una aproximación pragmática, aceptando que el problema no puede ser completamente entendido o definido al principio, y centrándose en maximizar la capacidad del equipo de entregar rápidamente y responder a requisitos emergentes.

Existen varias implementaciones de sistemas para gestionar el proceso de Scrum, que van desde notas amarillas post-it y pizarras hasta paquetes de *software*. Una de las mayores ventajas de Scrum es que es muy fácil de aprender, y requiere muy poco esfuerzo para comenzarse a utilizar.

A.2. Requisitos del software

La primera etapa del desarrollo de cada parte consistió en establecer los requisitos que debía cumplir y así conseguir una mayor comprensión del problema a resolver. Para ello, antes de comenzar el desarrollo, se realizaron reuniones con el director del proyecto y los profesores del CPEE Alborada. Se expusieron las nuevas necesidades o mejoras que había que introducir en la aplicación y de esta forma se llegó a un enfoque común de la misma.

Los requisitos recogidos para cada parte sufrieron varias modificaciones a lo largo del desarrollo del *software* correspondiente. A continuación se listan de forma conjunta, tal y como quedaron en su versión final, agrupados en funcionales y no funcionales. Los requisitos funcionales definen el comportamiento del *software*, así como las características de sus funcionalidades. Los requisitos no funcionales describen las restricciones tanto *software* como *hardware* de la aplicación y las facilidades que debe proporcionar.

A.2.1. Requisitos funcionales

RF-1: La aplicación permitirá la edición sencilla de texto con pictogramas asociados.

RF-2: La aplicación proporcionará mecanismos para cortar, copiar y pegar texto en el propio programa, y también desde o hacia otras aplicaciones (navegadores web, otros procesadores de textos, etc.)

RF-3: La aplicación proporcionará mecanismos para buscar texto en el documento, así como crear nuevos documentos, cargarlos y guardarlos. También opciones para deshacer y rehacer las últimas acciones realizadas.

RF-4: Se podrá configurar el idioma del documento, así como el de la aplicación, y cuantos parámetros se estimen convenientes (longitud máxima de búsqueda automática de palabras compuestas, nivel máximo de deshacer, tamaño de los pictogramas, etc.)

RF-5: La interfaz gráfica de usuario será lo más sencilla posible, amén de plenamente operativa.

RF-6: La aplicación deberá incluir una forma sencilla para importar los pictogramas de la base de datos de ARASAAC.

RF-7: La aplicación dispondrá de las opciones necesarias para trabajar con estos pictogramas en el documento, así como las solicitadas por el equipo de profesores del CPEE Alborada, como ocultar o mostrar el borde asociado a un pictograma, ocultar el pictograma asociado a una imagen, seleccionar entre los distintos pictogramas asociados a una palabra, etc. Todo ello de la forma más clara y fácil posible.

RF-8: La aplicación permitirá la búsqueda automática de palabras compuestas a las que corresponde un único pictograma (por ejemplo, *cepillo de dientes*).

A.2.2. Requisitos no funcionales

RNF-1: La aplicación deberá ser multilinguaje. Soportará los idiomas español e inglés pudiendo ser ampliados con facilidad.

RNF-2: El lenguaje de programación será Java.

RNF-3: La aplicación deberá poder ejecutarse sobre una máquina virtual Java 1.6 o superior.

RNF-4: La ejecución de la aplicación deberá ser independiente de la plataforma.

RNF-5: La aplicación deberá disponer de un paquete de imágenes que se pueda incluir de forma directa en la Galería de pictogramas.

RNF-6: Los archivos Araword deben ser independientes de la plataforma sobre la que se han creado.

RNF-7: El código fuente de la aplicación y los comentarios que en él aparecen deberán ser escritos en inglés.

RNF-8: La aplicación estará liberada bajo una licencia GNU General Public License, versión 3.

RNF-9: El manual de usuario tendrá una licencia *Creative Commons - Non commercial - Share Alike*, versión 3.0.

A.3. AraWord: casos de uso

Durante las primeras fases de definición de requisitos de AraWord y para comprender mejor el funcionamiento que debía tener se realizaron varios diagramas de casos de uso. Éstos sirven para mostrar qué debe hacer el sistema desde el punto de vista de la persona que lo va a utilizar y describen, de forma gráfica, la interacción de los posibles usuarios con el programa. En nuestro caso los diferentes roles que pueden desempeñar los usuarios de la aplicación son:

- **Profesor/Monitor:** Es el usuario habitual de este programa. Se encarga de crear los documentos que estime convenientes, y en general del manejo habitual de la aplicación.
- **Alumno:** Podría utilizar este programa, sin ser él el usuario habitual de éste, según su grado de discapacidad, con o sin ayuda de un monitor, para distintas actividades educativas.

A continuación se presenta el diagrama de casos de uso final de la aplicación.

A.4. Diseño de la aplicación

A continuación se expone el diseño final de la aplicación, explicando de forma breve cada uno de los subsistemas desarrollados en este proyecto o utilizados por la aplicación para cumplir los objetivos fijados, para obtener una visión global del sistema.

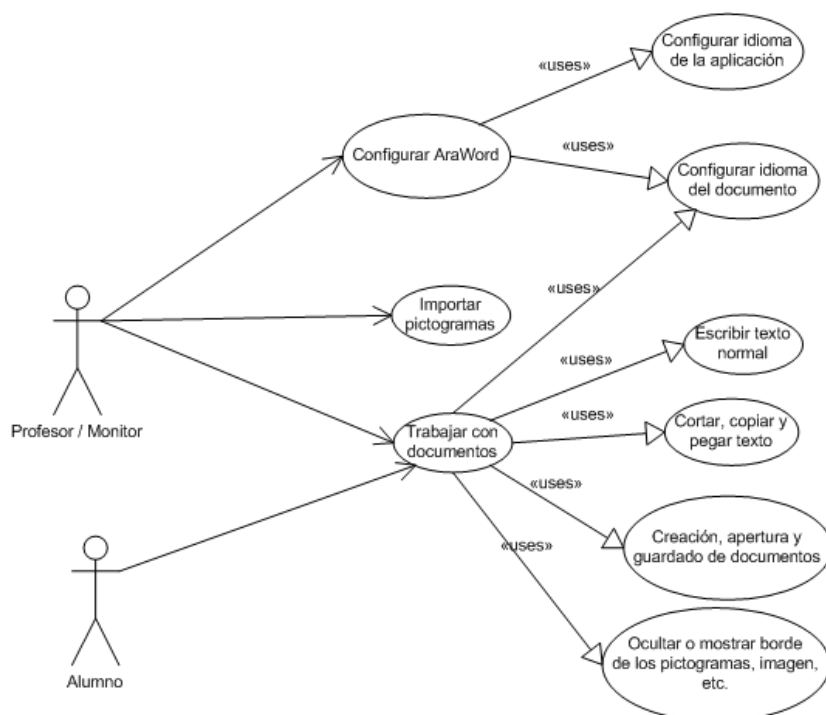


Figura A.1: Diagrama de casos de uso de AraWord

A.4.1. Diseño de la base de datos

La versión de base de datos final, que se espera sea común con TICO, AraPicto y otras aplicaciones similares que puedan surgir, en el marco de colaboración con el CPEE Alborada, es una base de datos bastante sencilla. Consta de tres tablas: Languages, WordTypes y Images. De ellas, sólo la última tiene alguna complejidad, ya que las otras dos son combinaciones sencillas ID - nombre a guardar de lenguajes y del tipo de palabra. Se crean al iniciar la aplicación por primera vez, quedando guardadas para futuras ejecuciones, y es poco probable su modificación (salvo que desde ARASAAC se decidiese incorporar soporte para un lenguaje o tipo de palabras adicional). Así pues, la tabla Languages tiene esta forma:

ID=1 Castellano.

ID=2 English.

ID=3 Français.

ID=4 Catalan.

ID=5 Deutsch. (etc)

Del mismo modo, la tabla WordTypes es parecida a esta:

ID=1 Nombre común.

ID=2 Verbo.

ID=3 Descriptivo (adjetivo o adverbio).

ID=4 Contenido social (fechas señaladas, eventos). (etc)

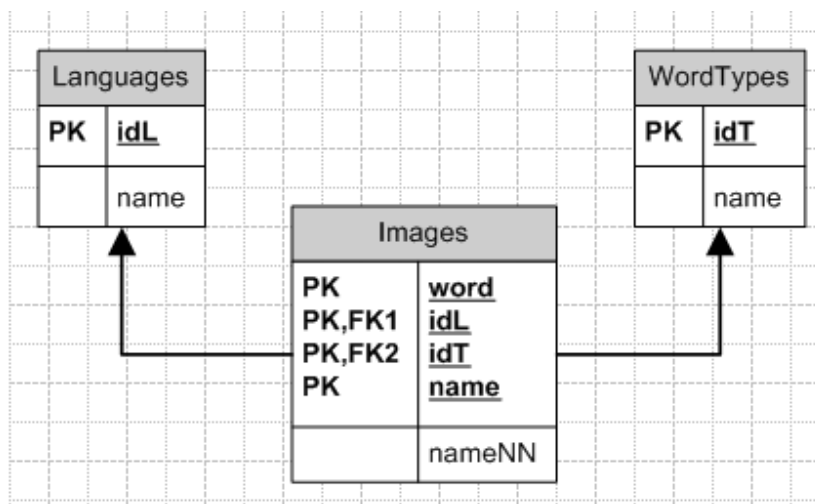


Figura A.2: Base de datos de AraWord

Apenas hay unas 10 tuplas por cada tabla. Por el contrario, la tabla Images es bastante grande (unas 73000 filas con la base de datos estándar de ARASAAC), y es donde recae casi todo el peso de cara a guardar la información necesaria. Cada entrada de la tabla es una 5-tupla de la siguiente forma:

word Palabra o concepto (por ejemplo, *abeja* o *cepillo de dientes*)

idL Clave ajena del lenguaje de la palabra.

idT Clave ajena del tipo de la palabra (utilizado fundamentalmente para escoger el color del borde del pictograma).

name El nombre normalizado de la imagen en disco (por ejemplo, *2283.png* o *laCasaNueva.jpg*).

nameNN El nombre no normalizado de la imagen (por ejemplo, *!‘QUÉ BUEN DÍA HACER!.png*). No utilizado en AraWord, se mantiene por retrocompatibilidad con TICO, aunque podría desaparecer en un futuro próximo sin plantear mayor problema.

Adicionalmente, para implementar la funcionalidad de encontrar formas verbales y su pictograma asociado (el del infinitivo correspondiente, por ejemplo, *he comido* tiene la misma imagen que *comer*) se emplea otra tabla, no relacionada de ninguna forma con las anteriores, con parejas (forma verbal, verbo en infinitivo asociado). Por ejemplo, parejas del tipo (comeré, comer), (he saltado, saltar), (reiría, reír).

form Forma verbal.

verb Verbo en infinitivo.

Esta tabla, debido a que incorpora muchos verbos (unos 9.400) con todas sus formas verbales, tiene más de 870.000 filas. Sin embargo, y pese a la frecuencia de la operación de búsqueda en esta tabla, el tiempo de respuesta se mantiene en unos niveles aceptables para el usuario.

A.4.2. Subsistemas internos

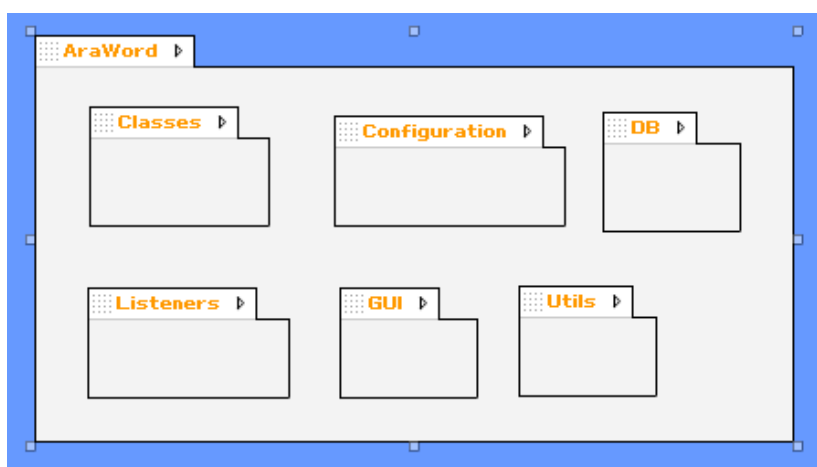


Figura A.3: Subsistemas internos

Los subsistemas internos o paquetes *software* de la aplicación son los siguientes:

- **araword**: Paquete principal de la aplicación que contiene todos los subsistemas. Además, posee dos clases: una conveniente para lanzar la aplicación y otra para almacenar variables accesibles desde diferentes subsistemas de la aplicación (globales).
- **araword.classes**: Contiene las clases que implementan los componentes que pueden aparecer en un documento (AWElement, AWTextField).
- **araword.configuration**: Contiene las clases que permiten la configuración del programa.
- **araword.db**: Contiene las clases para el acceso a la base de datos.
- **araword.gui**: Contiene las clases que implementan la interfaz gráfica de la aplicación.
- **araword.listeners**: Contiene las clases que implementan el comportamiento de los elementos que componen un documento, concretamente su reacción a eventos de teclado, ratón y ganancia de foco de ratón.
- **araword.utils**: Contiene las clases que proporcionan utilidades empleadas en otros sitios (manejo de rutas de fichero, modificación de tamaño de imágenes, rutinas para tratamiento de textos, manejo de archivos, etc.).

A.4.3. Bibliotecas externas

Las bibliotecas externas, requeridas para el funcionamiento de la aplicación (y que por supuesto se distribuyen junto a ella, siendo su manejo totalmente transparente al usuario) son:

- **Swing:** Es una biblioteca gráfica para Java que se distribuye junto con la máquina virtual Java (JVM) a partir de la versión 1.2. Incluye todo tipo de componentes para generar interfaces gráficas de usuario.
- **JDOM:** Es una biblioteca de código abierto para la manipulación de datos XML optimizados para Java. Se barajaron otras metodologías como DOM o SAX pero, a diferencia de éstas, JDOM está creada específicamente para Java, por lo que es más sencillo de usar.
- **SQLiteJDBC:** Es un *driver* JDBC de Java para SQLite. SQLite es un sistema de gestión de bases de datos relacional compatible con *ACID*¹, y que está contenido en una pequeña biblioteca en C.
- **AbsoluteLayout:** Pequeña biblioteca para posicionamiento sencillo de elementos en interfaces gráficas. Incluido con NetBeans, es necesario incluirlo aparte al trabajar desde Eclipse.
- **iText:** Es una biblioteca disponible para Java y otros lenguajes que sirve para creación de documentos PDF, entre sus múltiples posibilidades.
- **JFontChooser:** Pequeña librería que implementa un diálogo para seleccionar una fuente del sistema.
- **JavaCSV:** Pequeña librería que implementa un sistema de lectura y escritura de ficheros CSV (comma-separated values). Útil para importar la lista de formas verbales conjugadas a la base de datos que utilizamos.

A.5. Pruebas

Las pruebas de un producto *software* son los procesos que permiten verificar su calidad se han utilizado para identificar posibles fallos de implementación o usabilidad. Se han realizado durante todo el proceso de desarrollo del proyecto y no solamente al final, para poder solucionar el mayor número de errores en cada etapa, y no ir arrastrándolos con el avance del proyecto.

En esta sección se explican los diferentes tipos de *test* empleados y la finalidad de cada uno de ellos.

A.5.1. Pruebas unitarias

Este tipo de pruebas se utilizan para probar cada módulo de código desarrollado y permiten asegurar que cada uno de ellos funciona correctamente por separado. Una de las principales

¹*Atomicity, Consistency, Isolation and Durability*: Atomicidad, Consistencia, Aislamiento y Durabilidad; propiedades que puede tener una base de datos.

ventajas de este tipo de pruebas es que simplifican la integración de los módulos, ya que proporcionan un alto nivel de garantía de que el código funciona de la manera esperada.

A.5.2. Pruebas de integración

Después de realizar las pruebas unitarias de cada uno de los componentes, estas pruebas sirven para verificar que las distintas partes del *software* funcionan correctamente cuando se integran entre sí. De esta forma se prueba que cada parte desarrollada y verificada de forma individual se comporta de igual manera al integrarse con las demás funcionalidades.

A.5.3. Pruebas de sistema

Este tipo de pruebas permite comprobar el funcionamiento y el rendimiento global de la aplicación. El *software* ya validado con las pruebas de integración se probó con el resto del sistema para verificar los siguientes aspectos:

- Rendimiento: Se han realizado diferentes pruebas para determinar el tiempo de respuesta de determinadas operaciones que ofrece la aplicación, fundamentalmente, la operación de importar y exportar la base de datos con los pictogramas. Asimismo, en todo momento el tiempo de respuesta de la aplicación al trabajo normal (cortar, copiar, pegar texto; escritura normal, borrado de elementos...) se tenía que mantener en unos límites aceptables y no molestos para el usuario final de la aplicación (lo que ha sido determinante a la hora de afrontar ciertas optimizaciones).
- Robustez: Las pruebas realizadas tenían por objetivo determinar la capacidad del programa para soportar entradas incorrectas y asegurar que la aplicación era capaz de responder sin fallos.

A.5.4. Pruebas de usuario

Con este tipo de pruebas se mide el nivel de adaptación del sistema a las necesidades del usuario. Es importante tener en cuenta la facilidad de uso durante todo el desarrollo del proyecto para conseguir unos resultados satisfactorios para la mayoría de los futuros usuarios. Son esenciales para el tipo de programa desarrollado debido a los usuarios a los que va dirigida la aplicación. Algunas de los aspectos evaluados con este tipo de pruebas son los siguientes:

- Facilidad de aprendizaje: Se ha buscado en todo momento que la aplicación sea sencilla y manejable por los usuarios, realizando para ello interfaces gráficas intuitivas y similares entre sí. Esto facilita al usuario el aprendizaje de la aplicación de manera que pueda manejar la aplicación sin necesidad de tener unos conocimientos de informática elevados. Con estas pruebas se comprobó el tiempo empleado en la configuración y utilización de la aplicación por personas que no la conocían.
- Memoria: Un aspecto importante en el uso de la aplicación es que su manejo sea fácil de recordar aunque haya pasado bastante tiempo sin utilizarla. Para lograrlo se aseguró que la interfaz fuera intuitiva y sencilla.

- Satisfacción del usuario: Se determina la calidad de la experiencia de un usuario en su interacción con el programa. Para evaluar este aspecto se han realizado distintas pruebas en las aulas del colegio CPEE Alborada, donde actualmente la aplicación ya está en uso. La satisfacción en el uso de la aplicación es consecuencia de la facilidad de uso del programa y el cumplimiento de los objetivos previamente establecidos.

A.6. Herramientas utilizadas

A lo largo del desarrollo del proyecto se han utilizado distintas herramientas y tecnologías que han servido de apoyo en el desarrollo de las distintas tareas realizadas. Se recogen en esta sección explicando brevemente en algunos casos el uso que se ha hecho de ellas.

- **Desarrollo y pruebas**
 - Eclipse 3.6: Entorno de desarrollo (general).
 - NetBeans 6.9.1: Entorno de desarrollo (para la interfaz gráfica de usuario).
 - Subversion: Control de versiones. Subclipse 1.6: *plugin* para Eclipse que permite trabajar con el sistema de control de versiones desde el entorno de desarrollo.
 - Java SDK 1.6: Máquina virtual Java.
 - PangoScrum: Servicio web gratuito para gestionar los elementos de la metodología seguida (Scrum).
- **Sistemas operativos**
 - Windows XP, Windows Vista y Windows 7.
 - KUbuntu GNU/Linux 10.01
- **Generación de ejecutables e instaladores**
 - Nullsoft Scriptable Install System 2.46
 - HM NIS Edit 2.0.3
 - Any to Icon 2.0
- **Documentación**
 - MiKTeX 2.9: Compilador de $\text{\LaTeX}$
 - T_EXnicCenter 1.0RC1: Editor de $\text{\LaTeX}$, para elaboración de la memoria y apéndices.
 - Microsoft Project 2007: Editor de diagramas de Gantt.
 - Microsoft Visio 2007: Generador de diagramas (de casos de uso, de base de datos, de subsistemas, etc).
- **Propósito general**
 - Mozilla Firefox 3.6.13: Navegador web.
 - Notepad++ 5.3.1: Editor de texto y código fuente que soporta varios lenguajes de programación, así como el cambio de codificación de ficheros (Windows, Unix, Mac; UTF8, ANSI, ISO-8859-1).

Apéndice B

Distribución temporal del proceso

Este apéndice tiene por objetivo explicar cómo se ha distribuido el tiempo para llevar a cabo cada una de las distintas partes de este PFC. Comienza en septiembre de 2010 y finaliza en junio de 2011 realizándose a tiempo parcial. A continuación se muestra el Diagrama de Gantt donde se reflejan las diferentes tareas realizadas y el tiempo real dedicado al desarrollo de cada una de ellas.

La primera fase de este PFC se dedicó a la lectura sobre los sistemas aumentativos y alternativos de comunicación para obtener conocimientos sobre ese campo y así comprender mejor el objetivo de AraWord. Una vez adquiridos y para tener una visión de conjunto del programa, se invirtió tiempo en el manejo del software anteriormente utilizado (Escribir con Símbolos).

Para el desarrollo de AraWord se recogieron los requisitos que debía cumplir, así como los distintos cambios y ampliaciones que debían introducirse en la nueva versión. Junto con el director del proyecto se elaboró un diseño y se establecieron varios hitos cronológicos donde se debía tener distintas versiones de prueba de la aplicación, en las que se iban añadiendo funcionalidades. Cuando el proyecto estaba ya bastante avanzado, estas versiones de prueba se proporcionaban a los profesores del CPEE Alborada, para que con el uso detectaran posibles errores y pudiesen ser corregidos durante la implementación de la aplicación.

Además de los errores detectados en las distintas funcionalidades y corregidos durante el desarrollo, se realizaron pruebas en distintos sistemas operativos para comprobar que la aplicación funcionaba de la misma forma en todos ellos.

En la última etapa del tiempo que ha durado este proyecto y a la vez que se iniciaba la elaboración de la documentación del mismo, se han realizado una serie de mejoras en la aplicación. La fase de documentación de este PFC se ha extendido hasta la finalización del mismo.

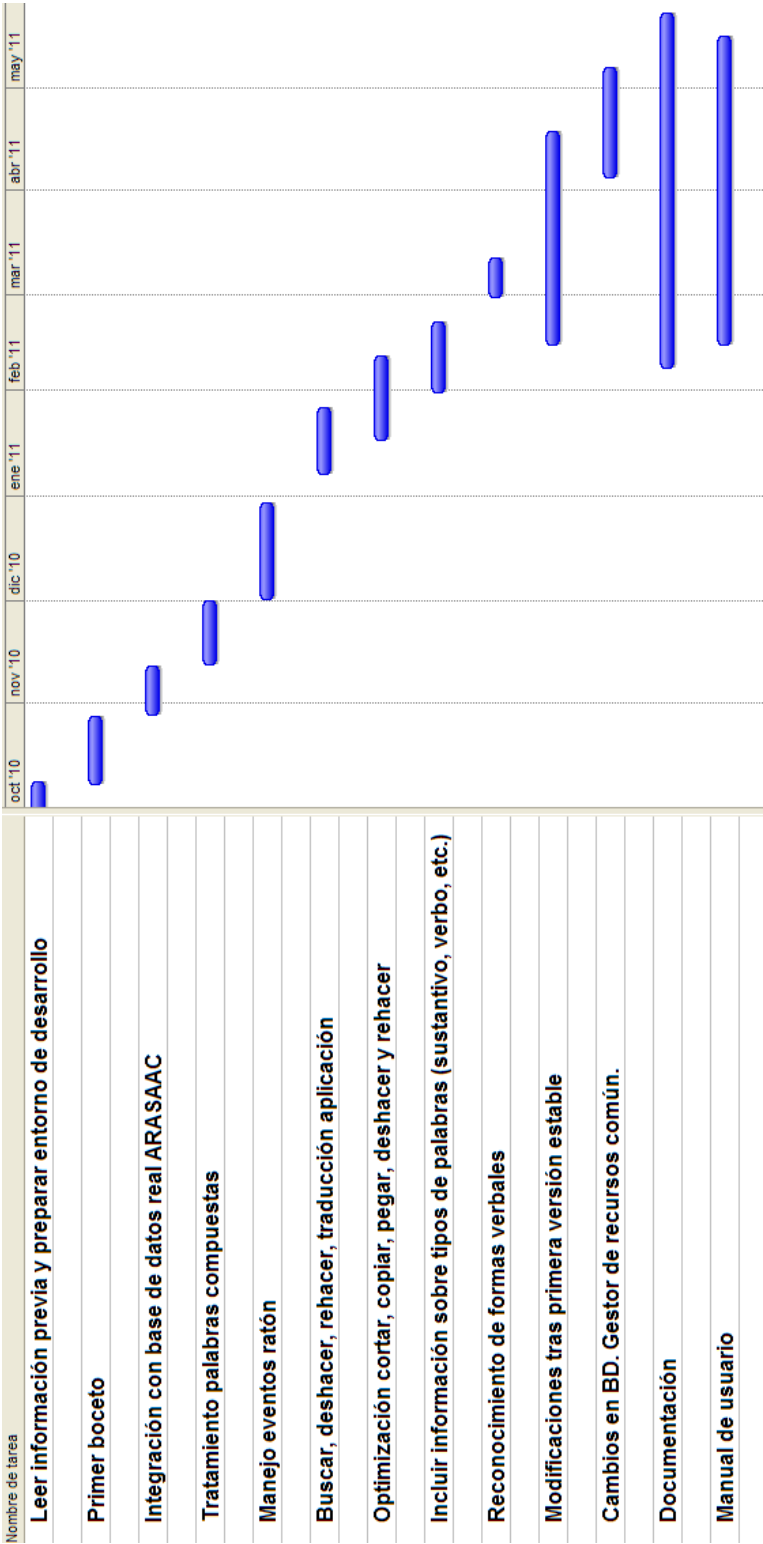


Figura B.1: Diagrama de Gantt de AraWord

Apéndice C

Manual de usuario

Este apéndice recoge el manual de usuario elaborado para la aplicación AraWord. Está publicado bajo licencia Creative Commons Attribution - Non Commercial - Share Alike 3.0. Para más información, se puede consultar <http://creativecommons.org/licenses/by-nc-sa/3.0/es>

C.1. Introducción

Este es el manual de usuario de AraWord. En él se explica el funcionamiento de la aplicación, su aspecto y las formas de realizar cada una de las posibles acciones.

C.2. Instalación de la aplicación

Ejecutar el archivo AraWord.exe y seguir los sencillos pasos de instalación. Es recomendable mantener el directorio por defecto en la instalación.

Para que la aplicación se pueda ejecutar es necesario tener instalada la Máquina Virtual de Java (JVM) versión igual o superior a la 1.6. Se puede obtener desde la dirección web <http://www.java.com/es/download> siendo la descarga gratuita.

Por otra parte es altamente recomendable disponer del paquete de pictogramas de ARA-SAAC, pudiendo utilizarse los mismos archivos que usa TICO en caso de tenerlo también instalado. Si no se dispone de éste, se pueden descargar desde <http://www.catedu.es/arasaac/descargas.php> siendo su descarga también totalmente gratuita.

C.3. Primeros pasos

Tras ejecutar el instalador, se habrá creado una carpeta llamada *AraSuite* (por defecto, en sistemas Windows, en *C:\Archivos de Programa\AraSuite*) que contiene dos carpetas: *AraWord*, *ResourceManager*. También hay dos accesos directos en el escritorio a sendas aplicaciones.

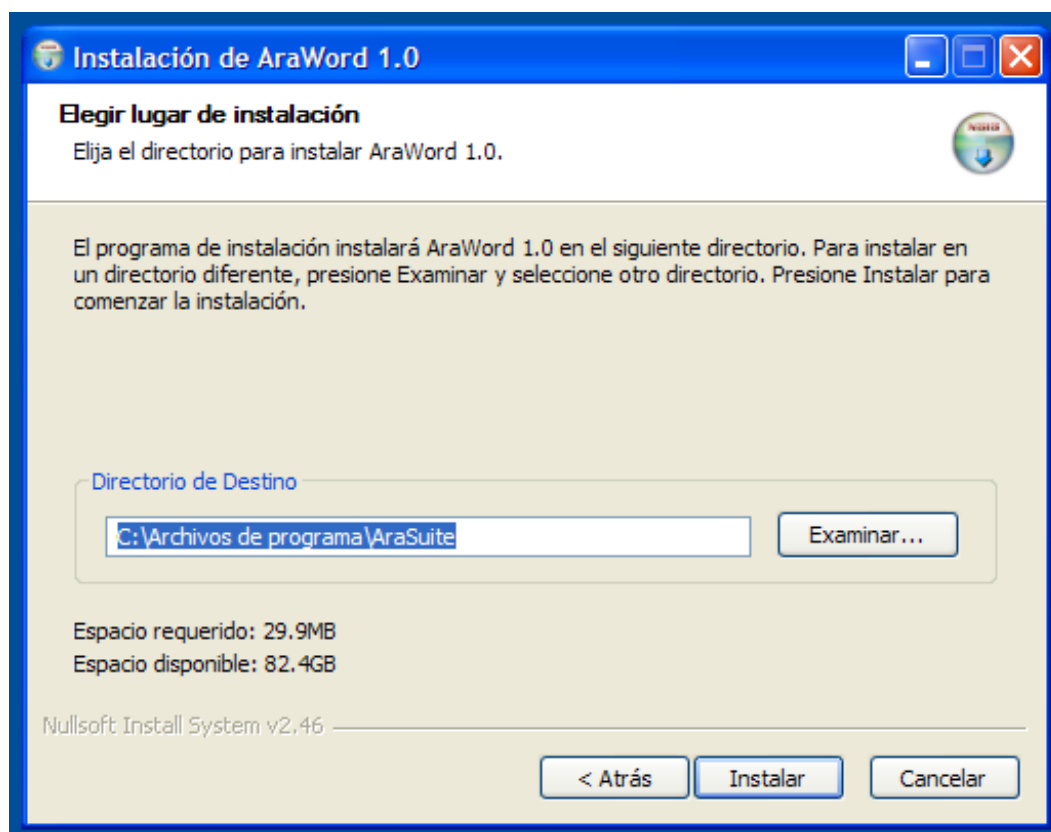


Figura C.1: Instalación de AraWord

Ejecutamos ResourceManager desde el link del escritorio, elegimos opción Importar base de datos. Elegimos carpeta de pictogramas bajados de Internet, así como carpeta en PC (recomendable usar la creada en *C:\Archivos de Programa\AraSuite*) y hacemos clic en aceptar. Tras uno o dos minutos, obtenemos un mensaje diciendo que todo ha ido bien.

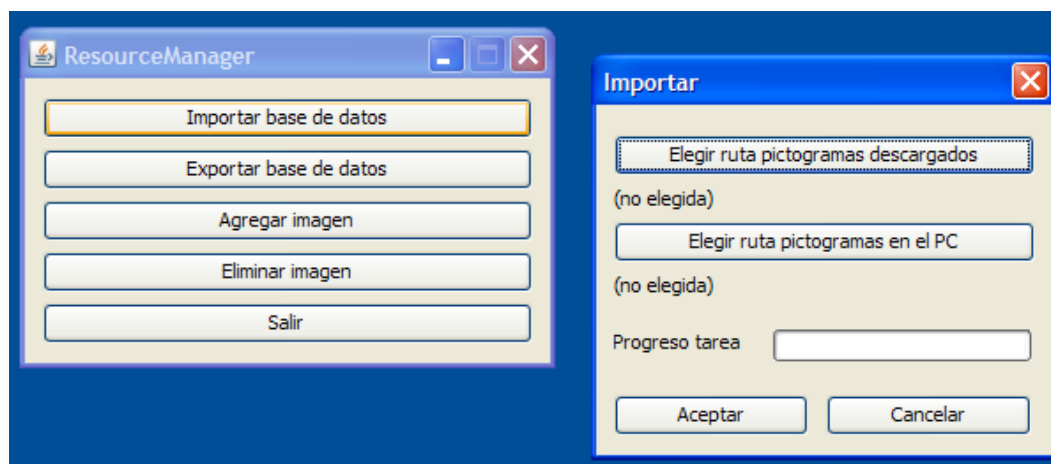


Figura C.2: Importar base de datos en ResourceManager

Tras ello, abrimos AraWord, vamos a Herramientas, Preferencias generales. Pinchamos en Elegir ruta pictogramas... y escogemos la carpeta en PC especificada anteriormente (si se ha seguido la recomendación, será *C:\Archivos de Programa\AraSuite*). Con esto, AraWord sabe dónde se encuentran los pictogramas y ya podemos empezar a trabajar.

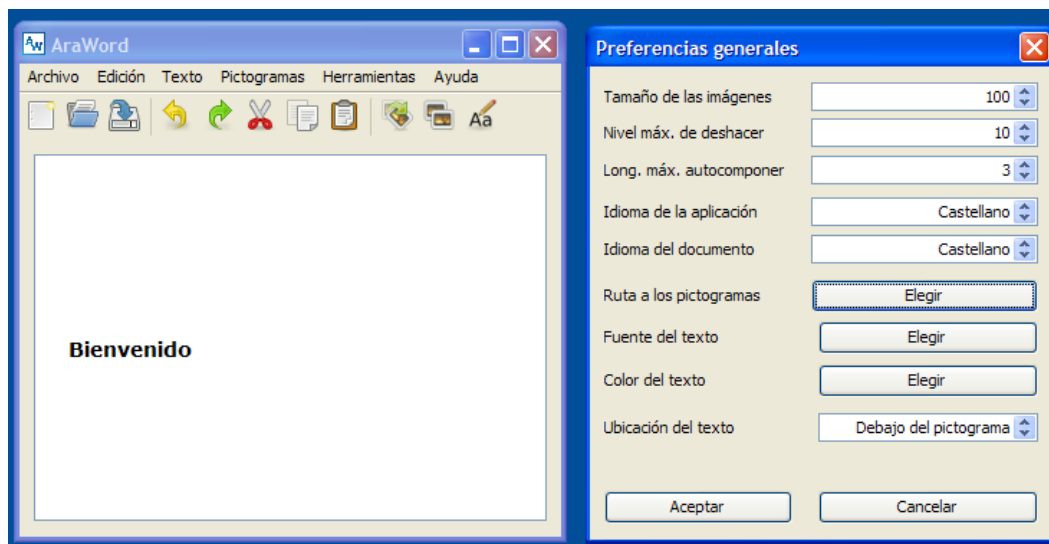


Figura C.3: Elegir base de datos en AraWord

C.4. Manual de uso de AraWord

La aplicación es extremadamente sencilla de utilizar (debido a que fue uno de los requisitos de diseño) y una gran parte de los usuarios no necesitarán consultar este manual salvo tal vez algún detalle puntual. Aquí se ofrece una descripción de las distintas acciones que puede realizar un usuario, se explica el modo de realizarlas y los resultados que cabe esperar de ellas.

Al arrancar la aplicación podemos apreciar tres áreas bien distinguidas. Una de ellas, el área de texto principal, donde editaremos nuestros documentos, una barra de herramientas con atajos para las funciones más habituales, y una barra de menú, que posee los siguientes menús:

- **Archivo:** Acciones a realizar sobre un documento. Permite crear un documento nuevo, cargarlo, guardarlo, exportarlo a diferentes formatos y salir de la aplicación.
- **Edición:** Acciones básicas de edición. Permite deshacer y rehacer, cortar, copiar y pegar, buscar y seleccionar texto dentro del documento.
- **Texto:** Acciones asociadas al texto. Permite cambiar la fuente y el color del texto, así como su ubicación (encima o debajo del pictograma). También permite pasar el texto a mayúsculas o a minúsculas, así como cambiar el idioma del documento.
- **Pictograma:** Acciones asociadas a un pictograma. Permite alterar el tamaño de los pictogramas, cambiar el pictograma asignado a una palabra entre los disponibles, forzar

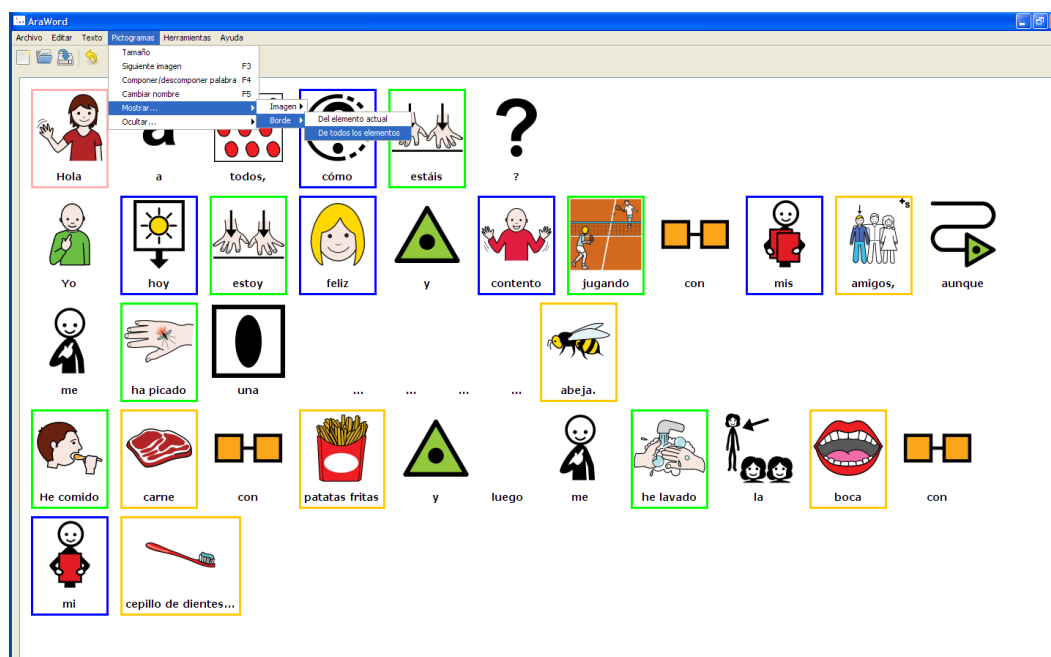


Figura C.4: AraWord en funcionamiento

la descomposición o composición manual de palabras compuestas, ocultar y mostrar borde o imagen de un pictograma concreto o de todos juntos, y cambiar la palabra asociada a un elemento manteniendo fijo su pictograma asociado.

- Herramientas: Preferencias generales del programa (por defecto, nivel máximo de deshacer, longitud máxima de búsqueda automática de palabras compuestas, idioma de la aplicación, tamaño imágenes, idioma, fuente, color y ubicación del texto del documento por defecto, así como elegir la base de datos de imágenes a utilizar) y un enlace al Gestor de Recursos.
- Ayuda: Ayuda de la aplicación, y un cuadro *Acerca de* con información del programa, su versión, créditos y otra información.

Por su parte, el Gestor de Recursos posee cuatro opciones básicas:

- Importar base de datos: Crea la estructura de datos necesaria para AraWord a partir de la carpeta de pictogramas descargada de Internet. Necesario al principio para todos los usuarios.
- Exportar base de datos: Salva la estructura de datos de AraWord en una carpeta de pictogramas con un archivo .xml explicativo. Útil si personalizamos nuestra base de datos de imágenes y deseamos pasarlas a otro ordenador.
- Agregar imagen: Permite añadir un pictograma personalizado a nuestra base de datos. Podemos elegir idioma, tipo de concepto, así como las palabras clave asociadas a ese pictograma.

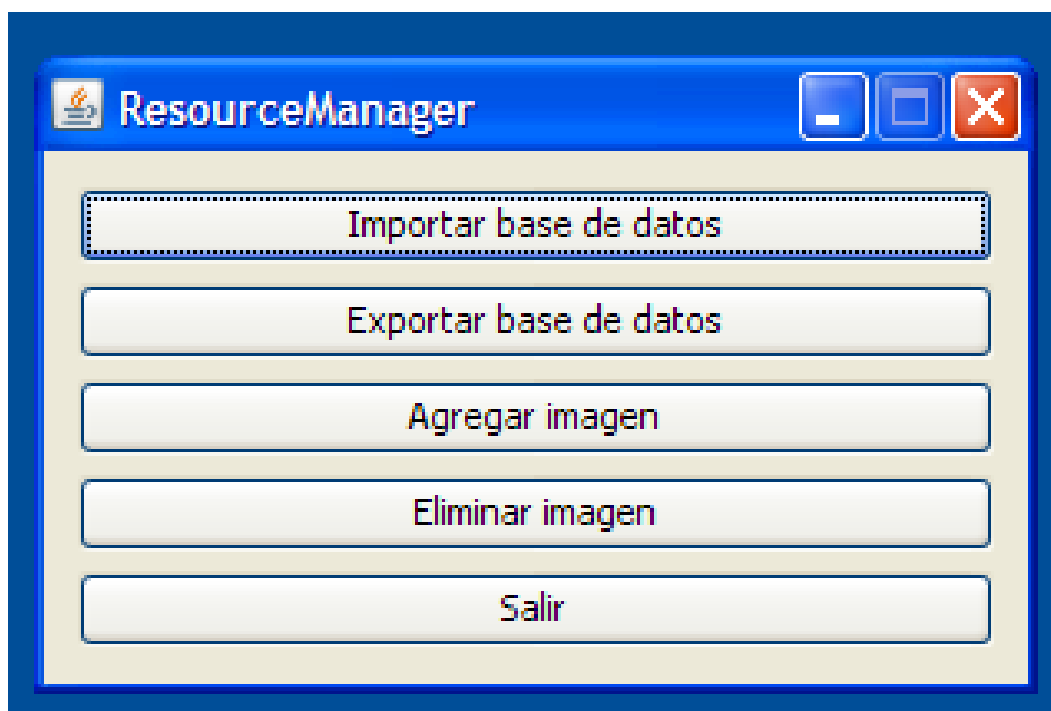


Figura C.5: ResourceManager

- Eliminar imagen: Permite eliminar un pictograma previamente insertado por nosotros, o incluso cualquiera de los descargados de Internet.
- Salir: Cierra el Gestor de Recursos.

Apéndice D

The GNU General Public License

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright

on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered

works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license

the work in any other way, but it does not invalidate such permission if you have separately received it.

- d)* If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation’s users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a)* Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b)* Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
- c)* Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- d)* Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

- e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to

the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically

terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR

PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

<one line to give the program’s name and a brief idea of what it does.>

Copyright (C) <textyear> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
```

```
This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.  
This is free software, and you are welcome to redistribute it  
under certain conditions; type 'show c' for details.
```

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.