



Documentación técnica de Sparebrain



Documentación técnica de Sparebrain

Este documento recoge distintos aspectos relevantes sobre las la implementación de la aplicación Sparebrain, una herramienta de productividad con un doble objetivo: gestionar una programación de tareas y mantener listas de elementos de propósito general.

1	Introducción	5
2	Aspectos relevantes del desarrollo	7
2.1	Características de la interfaz	7
2.2	Preferencias del usuario	10
2.3	Portabilidad	11
2.4	Internacionalización	15
2.5	Validación de campos	16
2.6	Exportación	17
2.7	Copias de seguridad	18
3	Sobre la implementación	19
3.1	Interfaz	19
3.2	Listas	19
3.3	Tareas	19
3.4	Útiles	20
3.5	Excepciones	20
3.6	Recursos	20

“El que no tiene memoria, se hace una de papel”

- Gabriel García Márquez

1 Introducción

Muchos han terminado por acostumbrarse a una libreta repleta de tareas pendientes, a decenas de notas de colores desperdigadas por la mesa, a unos cuantos archivos de texto con anotaciones... a todos esos elementos que, en definitiva, conforman algo que podría considerarse una auténtica *memoria de papel*.

Precisamente de esta idea surgió la aplicación Sparebrain¹, que pretende ser una herramienta de productividad con un doble objetivo: gestionar una programación de tareas y mantener listas de elementos sin un propósito específico. Aspira, de esta manera, a reemplazar en la medida de lo posible a la libreta o las notas a las que nos referíamos anteriormente.

Un taco de notitas amarillas cumple en buena medida lo que promete. Es sencillo, rápido y flexible. Y ese es precisamente el enfoque que se ha buscado aportar en este desarrollo.

¹ Del inglés “spare brain”, que en traducción libre sería “cerebro de repuesto” o “cerebro auxiliar”.

2 Aspectos relevantes del desarrollo

En esta sección se describen algunas particularidades de la aplicación que se consideran especialmente importantes.

2.1 Características de la interfaz

El programa se compone de **dos vistas** separadas, agrupadas mediante un panel tabulado. Una de las vistas sirve para la gestión de las tareas y otra para la de las listas. La ventana puede redimensionarse a voluntad del usuario, aunque hay establecido un tamaño mínimo que garantiza la correcta disposición de los elementos.

2.1.1 Tareas

El panel de tareas reúne los elementos necesarios para la creación, edición y eliminación de tareas. Sobre Mac OS X Tiger tiene el siguiente aspecto:

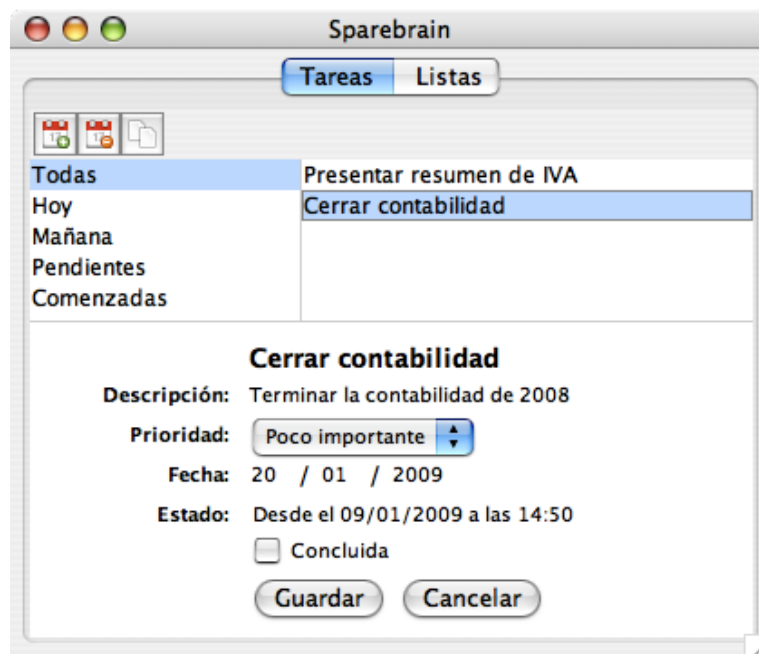


Ilustración 1. Aspecto del panel de tareas

En la parte superior del panel aparecen botones para crear, eliminar y copiar tareas. Los dos últimos sólo permanecen activos mientras hay tareas seleccionadas en el panel.

La parte inferior de la ventana se despliega cuando se selecciona una tarea y muestra los datos de la misma. Los controles admiten edición del contenido,

remarcando el borde cuando el usuario se posiciona en ellos, para dar a entender que el campo se puede editar (Ilustración 2):

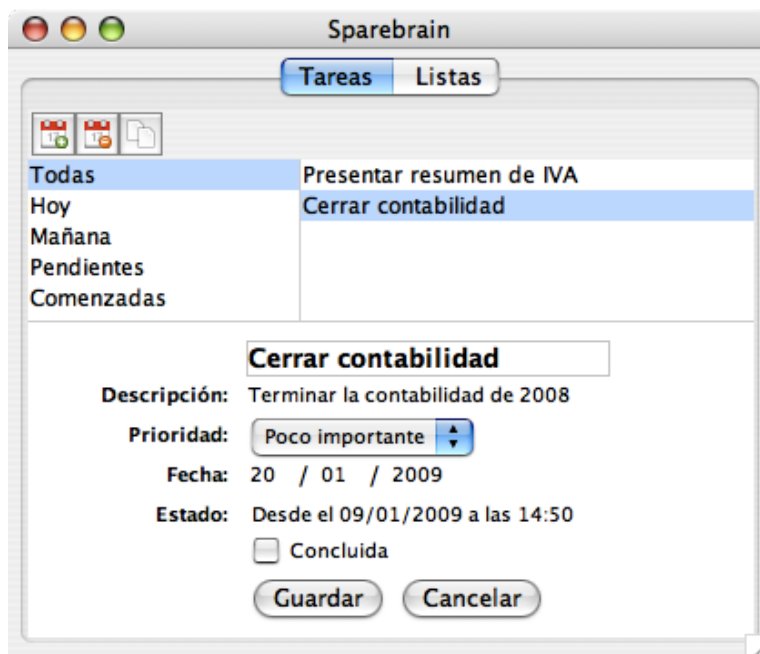


Ilustración 2. Editando un campo

El panel de tareas dispone asimismo de un botón para copiar al portapapeles el volcado de una tarea seleccionada. La tarea anterior quedaría copiada como sigue:

Cerrar contabilidad (20/01/2009)
Terminar la contabilidad de 2008
Poco importante
Comenzada. Desde el 09/01/2009 a las 14:50

El sistema permite seleccionar categorías de tareas. Así, al hacer clic en "hoy" sólo se mostrarán las tareas programadas para hoy y así sucesivamente.

En el menú "Tareas", que permanece activo siempre que lo está este panel, aparecen las mismas opciones de la barra de herramientas más una para borrar todas las tareas. Asimismo, las tareas pueden eliminarse individualmente a través del menú, del botón de eliminar de la barra de herramientas o bien utilizando la tecla "retroceso". El sistema presenta un cuadro de confirmación antes de borrar cada tarea.

El panel que muestra las tareas muestra una barra de desplazamiento si el tamaño de la ventana es insuficiente para mostrar todas las tareas.

2.1.2 Listas

El panel de listas reúne las opciones relativas a la gestión de las listas. La barra de herramientas se encuentra dividida en dos segmentos, uno que tiene efecto sobre las listas y otro sobre los elementos de las listas.

El botón de eliminar una lista, el de copiar y el de agregar un elemento sólo permanecen activos cuando hay una lista seleccionada. Por su lado, el botón de eliminar elemento sólo se activa cuando un elemento está seleccionado. Lo mismo sucede con el menú “Listas”, que aparece en la barra de menús sólo cuando el panel de listas está seleccionado.

Las funciones de edición son muy similares al caso del panel de tareas. Al seleccionar una lista o un elemento se despliega el panel inferior con la descripción del mismo, que el usuario puede editar.

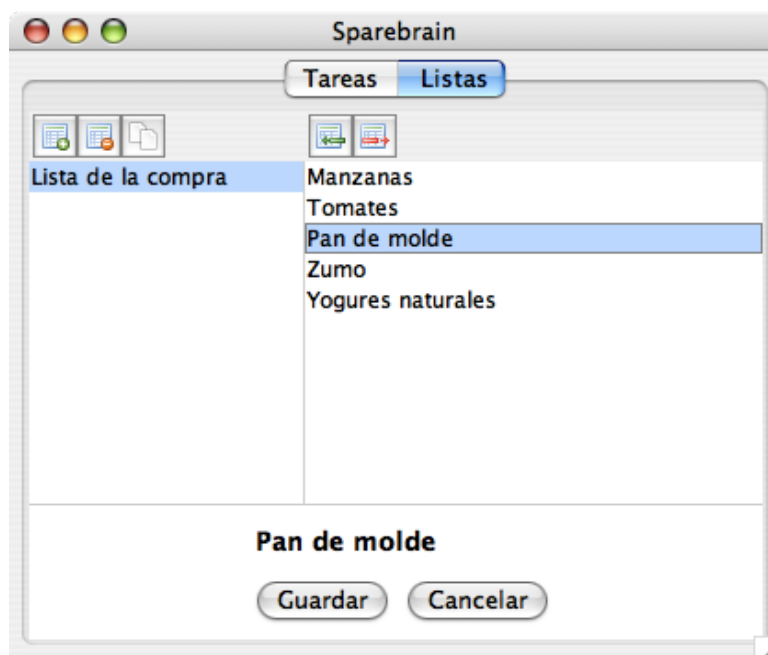


Ilustración 3. Aspecto del panel de listas

El panel de listas dispone también de un botón para copiar al portapapeles el volcado de una lista seleccionada. La lista anterior quedaría copiada como sigue:

Lista de la compra:
Manzanas
Tomates
Pan de molde
Zumos
Yogures naturales

Tanto las listas como los elementos pueden eliminarse individualmente a través del menú, del botón de eliminar de la barra de herramientas o bien utilizando la tecla “retroceso”. El sistema presenta un cuadro de confirmación antes de borrar cada lista, y borra los elementos directamente sin preguntar al usuario. Esto es así porque generalmente son unas pocas palabras y no es crítico perder un elemento de una lista.

El panel que muestra las listas y el que muestra los elementos muestran una barra de desplazamiento si el tamaño de la ventana es insuficiente para mostrar todas las listas o elementos, respectivamente. Dicha circunstancia se muestra en la siguiente imagen (Ilustración 4):

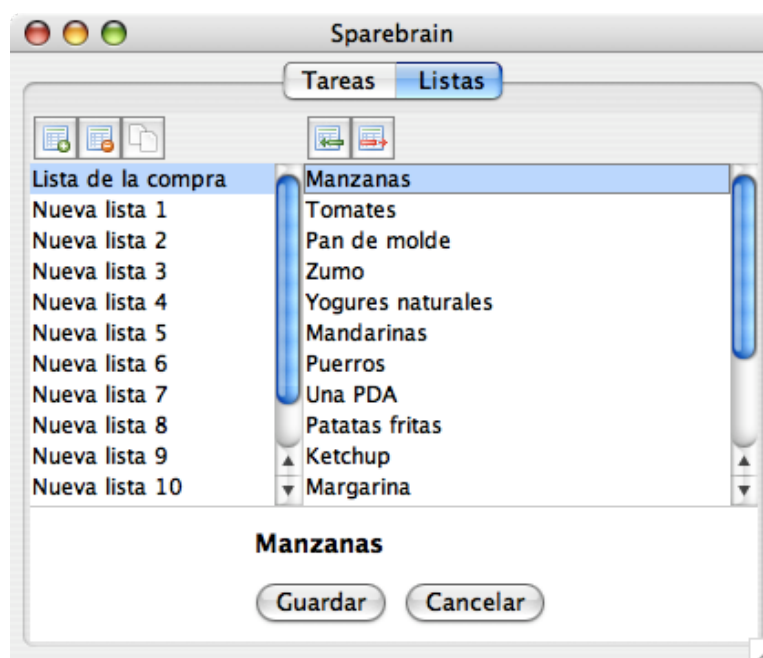


Ilustración 4. Los paneles mostrando barras de desplazamiento

2.2 Preferencias del usuario

Se ha prestado especial atención a hacer del uso de la aplicación una experiencia agradable para el usuario. El sistema graba **automáticamente** diversa información sobre las preferencias del usuario, restaurando el estado al iniciarse. Concretamente se guarda:

- La posición de la ventana en la pantalla cuando el programa se cerró
- El tamaño de la ventana.
- Qué pestaña estaba cargada cuando se cerró el programa (tareas o listas).
- La última categoría de tareas visualizada. Si el usuario estaba consultando la lista de tareas *pendientes* cuando cerró la aplicación, al iniciarse de nuevo estará seleccionada la misma categoría.

- Última lista visualizada. Si el usuario estaba editando, por ejemplo la lista de la compra cuando cerró la aplicación, al iniciarse de nuevo aparecerá seleccionada la misma lista.
- Idioma del usuario (se puede configurar Sparebrain para que aplique siempre un determinado idioma o para que tome los valores del sistema automáticamente).
- Aspecto y sensación preferidos (también en este caso se puede seleccionar un tema manualmente o bien aplicar automáticamente el del sistema operativo).

Las preferencias del usuario se guardan en un archivo en formato XML que se edita haciendo uso de la clase `Properties` de Java. Para facilitar el acceso a los datos y simplificar la sintaxis, se creó una clase `Propiedades` que encapsula la anterior.

2.3 Portabilidad

Una característica prácticamente inherente a cualquier programa en Java es precisamente la posibilidad de ser ejecutado en múltiples plataformas. Sin embargo, a menudo no basta sólo con que un código funcione en un sistema, sino que es necesario que el programa se integre en cierta medida con el sistema operativo. Sparebrain sigue esta filosofía y utiliza diferentes técnicas para adaptarse al entorno en que se utilice.

A continuación se exponen de forma resumida las características del sistema directamente relacionadas con la portabilidad.

2.3.1 Aspectos específicos de Mac OS X

Sparebrain ha sido desarrollado íntegramente sobre Mac OS X, y por ello se ha prestado un especial cuidado a la integración con las particularidades de este sistema, a saber:

- Uso de la barra de menús superior, fuera de la ventana de la aplicación.
- Adaptación al menú del sistema. En la versión para Mac OS X de Sparebrain, no se presenta la opción “Preferencias” ni la opción “Salir” en el menú “Archivo”, sino que se utilizan los botones proporcionados en el menú del sistema, que es donde el usuario espera encontrar estas opciones. La implementación de estas características requiere de bibliotecas específicas de Mac OS X cuyo uso se restringe a una única clase (`VentanaPrincipalMac`) que se liga dinámicamente para permitir la correcta ejecución del programa en todos los sistemas operativos.
- Control de la salida de la aplicación con la pulsación de la combinación `cmd + Q` o el cierre desde el menú del Dock (es necesario capturar el evento de salida para guardar las preferencias). También requiere del uso de bibliotecas específicas.

2.3.2 Teclas de acceso de los menús

En Mac OS X la tecla de función preferente para acceder a los elementos de menú es la tecla comando (cmd). Sin embargo, en Windows y en Linux la tecla utilizada suele ser la de control (ctrl). Esta circunstancia está prevista y el sistema utiliza una u otra según el sistema operativo detectado.

2.3.3 Aspecto y sensación

Mediante las preferencias puede seleccionarse manualmente el tema visual de la aplicación, entre todos los disponibles en el equipo del usuario. Por defecto se aplicará el “aspecto y sensación” correspondiente al sistema operativo en que se ejecute la aplicación. De esta manera la aplicación tendrá la misma apariencia que los programas nativos de cada plataforma.

Ejecutándose sobre Mac OS X Tiger, su sistema operativo “de referencia”, la aplicación presenta, por defecto, la apariencia que se muestra en la siguiente captura de pantalla (Ilustración 5):

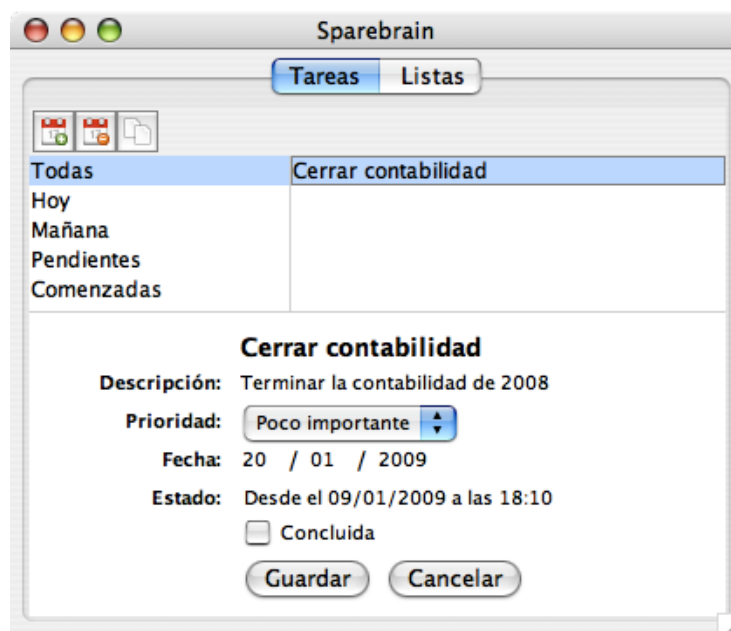


Ilustración 5. Sparebrain sobre Mac OS Tiger

Asimismo, la siguiente ilustración muestra el programa ejecutándose sobre la distribución Ubuntu Linux con el tema GTK aplicado por defecto:

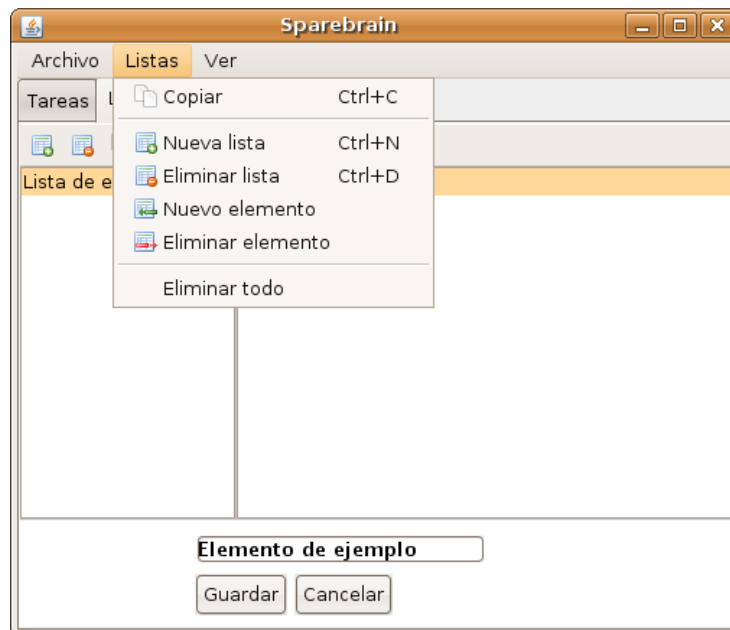


Ilustración 6. Sobre Gnome en Ubuntu Linux

Y la siguiente imagen (Ilustración 7) muestra de nuevo el programa, esta vez ejecutándose sobre Windows XP con el tema predeterminado de este sistema operativo:



Ilustración 7. Sparebrain en Windows XP

A través del diálogo **Preferencias** se puede modificar igualmente el tema visual, pudiendo el usuario seleccionar uno entre todos los aspectos disponibles en el sistema. Por ejemplo, en Mac OS se puede seleccionar el tema “Metal” que confiere al programa la siguiente apariencia:

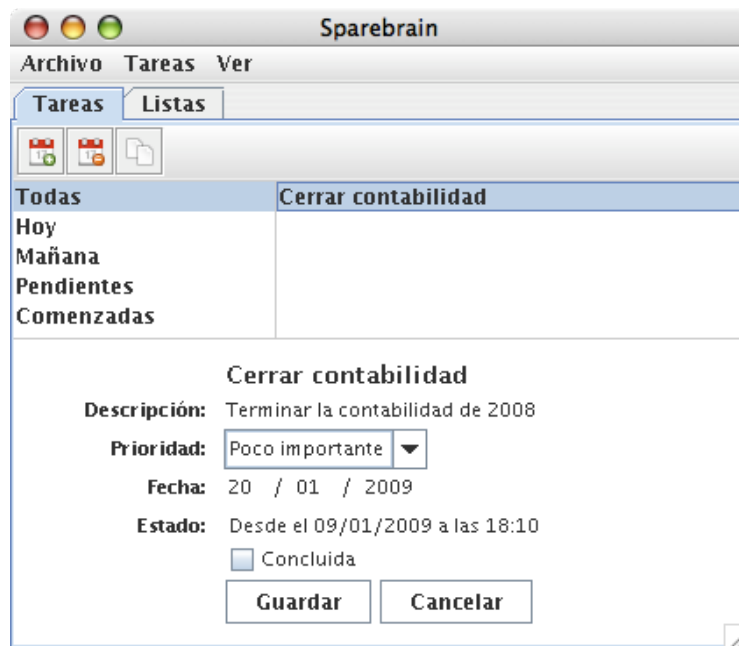


Ilustración 8. Tema "Metal" sobre Mac OS X Tiger

El número de temas disponibles varía con el sistema operativo y la versión del entorno de ejecución.

2.3.4 Separadores y delimitadores

Los ficheros se escriben utilizando saltos de línea y separadores universales para asegurar una correcta presentación de los datos en todos los SO. Estos separadores están incluidos en una única clase auxiliar (`Fichero`) que puede ser accedida desde las clases que resulte conveniente.

2.3.5 Archivos de la aplicación

Los tres principales sistemas operativos del mercado disponen de sus respectivas ubicaciones preferentes para guardar los archivos que necesitan utilizar las aplicaciones. Concretamente, Sparebrain necesita un fichero para almacenar las listas y otro para almacenar las tareas que han de ser guardados en el equipo del cliente.

- En Mac OS los datos del programa para cada usuario deberían guardarse en una subcarpeta del directorio “~/Library/Application Support/”. Cabe destacar que esta ruta es independiente del idioma del sistema². Esta aplicación utiliza el directorio “~/Library/Application Support/Sparebrain”.

² Por ejemplo, un usuario que utilice el sistema en español verá la carpeta “~/Librería/Application Support/” y podrá utilizar indistintamente las dos rutas. Esta circunstancia no está prevista en Windows, donde los nombres de

- En la mayoría de las distribuciones de Linux no existe un directorio específico del sistema dedicado a esto. Generalmente los programas escriben sus datos en un subdirectorio oculto de la carpeta de inicio del usuario, por lo que la ruta con los archivos de la aplicación en Linux es “~/.sparebrain”.
- En Windows, los programas deben guardar los archivos que utilicen en una carpeta denominada “datos de programa”, cuya ubicación es accesible a través de la variable de estado APPDATA. El sistema utiliza esa ruta para almacenar una carpeta de nombre “Sparebrain” con los ficheros que necesita mantener en disco.

El sistema necesita guardar también un archivo XML con las preferencias del usuario. Así:

- En Mac OS las preferencias de las aplicaciones para cada usuario deberían guardarse en un archivo del directorio “~/Library/Preferences/”. Dicho archivo debe tener un nombre único compuesto con el dominio inverso del fabricante (por ejemplo, com.apple) y con el nombre de la aplicación, manteniendo la extensión .plist. Por ejemplo, en este caso el archivo utilizado tiene como nombre “com.nosolopau.sparebrain.plist”.
- En la mayoría de las distribuciones de Linux no existen directrices específicas sobre el almacenamiento de las preferencias, por lo que éstas se guardan en el mismo directorio de los datos de programa con el nombre “sparebrain.xml”.
- En Windows el mecanismo preferente para el almacenamiento de las preferencias es el registro del sistema, pero para no complicar excesivamente el diseño se utilizará también un archivo XML con el nombre “sparebrain.xml”.

La aplicación obtiene en el arranque la plataforma sobre la que se está ejecutando, construye los directorios y archivos necesarios y los hace accesibles al programa. Este tipo de tareas están centralizadas en una única clase denominada `FactoriaArchivos`.

2.4 Internacionalización

La aplicación está preparada para funcionar en varios idiomas. Los archivos de traducción, en formato XML, están debidamente empaquetados dentro del JAR de la distribución, de manera que resulta muy sencillo añadir traducciones al sistema, siendo suficiente añadir un archivo de nombre “*locale_idioma.xml*” al paquete “recursos”, donde *idioma* es el código internacional de la correspondiente lengua (por ejemplo, “en” para el inglés o “es” para el español).

directorios son distintos para cada traducción, obligando al programador a usar exclusivamente las variables de entorno.

El usuario puede, a través de la ventana de preferencias, elegir manualmente un idioma de entre los disponibles o bien seleccionar la detección automática del mismo. Durante el proceso de carga, la aplicación abre el archivo de preferencias y lee si hay un idioma establecido manualmente como preferido. Si no hay ningún idioma preferido o está establecida la opción de detección automática, obtiene el idioma del usuario. Si se dispone de un archivo de traducción para ese idioma, el sistema lo carga y lo aplica. En otro caso, el sistema cargará el archivo de traducción al inglés.

Sobre los menús, las teclas de acceso son las mismas para todos los idiomas. Se ha tomado esta decisión para evitar desconcertar al usuario. Por otro lado, parece que en la industria del software se impone, de un tiempo a esta parte, respetar los atajos de teclado del inglés en vez de tratar de adaptarlos al idioma del usuario.

Todas las cadenas que utiliza el sistema se define mediante una serie de variables accesibles a través de la clase `Traduccion`.

2.4.1 Klingon

En lo que pretende ser un pequeño toque divertido, Sparebrain está traducido al Klingon, la lengua de la raza de humanoides del mismo nombre en la saga Star Trek. Google ya sorprendió hace unos años mostrando su interfaz traducida a este mismo idioma imaginario. El Klingon tiene algunas características interesantes, como la presencia de letras mayúsculas a mitad de palabra, que además puede modificar el significado del término.

La traducción pudo realizarse gracias a un excelente diccionario en línea desarrollado en la academia sueca *Klingonska Akademien*.

2.4.2 Uso de fechas

Asimismo, el formato de fecha es distinto para el inglés y el castellano. En el primer caso se utiliza el esquema `mm/dd/aaaa` y en el segundo, el habitual `dd/mm/aaaa`. Para cada traducción, puede configurarse cuál de estos esquemas de internacionalización se utiliza mediante la modificación del atributo `"locale"`.

2.5 Validación de campos

El enfoque que se ha seguido en general es el de restringir lo menos posible la entrada de datos por parte del usuario, puesto que en general resulta imposible prever el uso que se hará de las características de la aplicación. Hay sólo dos excepciones a este principio, como se indica a continuación:

2.5.1 Tratamiento de fechas

El sistema realiza una comprobación débil cuando el usuario escribe una fecha (en las tareas), verificando que los datos introducidos en los tres campos sean numéricos y representen días, meses y años válidos, para lo cual simplemente considera que el día ha de ser un entero entre 1 y 31 y el mes un entero entre 1 y 12.

Deliberadamente, no se comprueba la existencia del día en el mes señalado. Si el usuario escribe que una tarea tiene su fecha límite el 31 de febrero de 2009, el sistema automáticamente asume que la fecha límite es en realidad el 3 de marzo de 2009.

El mecanismo utilizado en la validación de los campos consiste en un objeto de tipo `InputVerifier`, que retiene el foco sobre el control de texto hasta que contiene un valor correcto. Asimismo, el sistema no admite la inserción de datos incorrectos e ignora la pulsación del botón de aceptar.

2.5.2 Inserción de delimitadores del fichero

Sparebrain usa, en esta versión, dos archivos de texto delimitados para guardar los datos de listas y tareas. El sistema está preparado para asumir como delimitadores cadenas completas de caracteres, con lo que se pueden utilizar separadores de cierta complejidad para no interferir en la actividad del usuario. Aunque pueden ser modificados editando una única línea de código, los separadores que usa el programa por defecto son del tipo “`</F>`” para separar campos en las tareas, o “`</L>`” para separar listas.

Esto hace muy complicado que el usuario utilice los delimitadores como parte del texto introducido, pero si esto sucediera, el sistema mostrará una advertencia y no guardará los datos introducidos.

2.6 Exportación

Sparebrain es capaz de exportar las tareas y las listas a formatos sencillos, con idea de permitir su portabilidad. Por ejemplo, el usuario podría querer guardar las listas y las tareas en un dispositivo móvil. La siguiente fotografía muestra cómo se presenta en un iPod Nano 3G un fichero con las tareas exportadas:

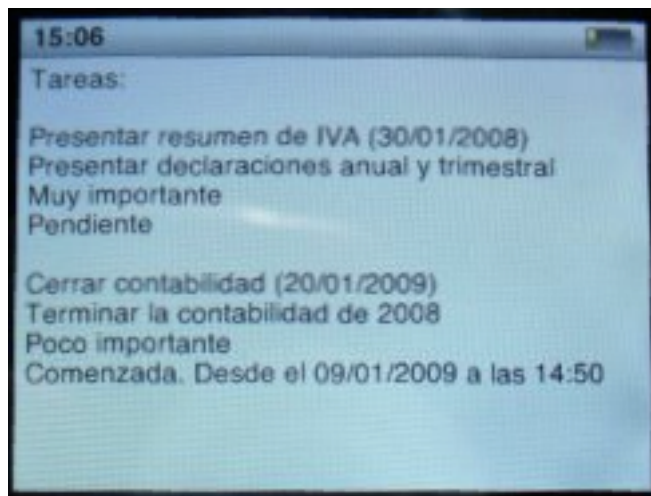


Ilustración 9. Lista de tareas en la pantalla de un iPod

El caso de las listas en análogo. Al exportarse los datos en formato texto son reconocidos por la mayoría de los dispositivos móviles, lo cual aporta posibilidades muy interesantes a esta característica. El formato de la exportación de los datos respeta el idioma en que se ejecute el programa.

2.7 Copias de seguridad

Sparebrain permite guardar copias de seguridad con los datos de la aplicación en un archivo, y utilizar un archivo de copia de seguridad para restaurar el sistema. Los archivos de copia de seguridad son una réplica exacta de los mismos ficheros que utiliza el programa. El proceso de guardar una copia de seguridad puede asimilarse a copiar una copia de los archivos del programa.

2.7.1 Control de formato

Los archivos del programa se guardan anteponiendo una cabecera específica que indica el tipo de archivo que se almacena y la versión de la aplicación. De esta manera queda cubierto un posible cambio de formato en una próxima versión de Sparebrain, pudiendo en ese caso establecer diferentes tratamientos para diferentes formatos de archivo.

Cuando un usuario intentar cargar un archivo de copia de seguridad de listas o tareas, el sistema comprueba que la cabecera del archivo sea correcta, y sólo en ese caso permite la importación de los datos del mismo. Si el archivo no tiene un formato reconocido por el programa, se muestra una advertencia al usuario y se cancela la importación.

3 Sobre la implementación

La aplicación construida consta de 6 paquetes que agrupan 26 clases, como se describe a continuación:

3.1 Interfaz

El paquete `interfaz` contiene las clases de la interfaz de usuario. Existen clases para las ventanas de la aplicación (`VentanaPrincipal` y `VentanaPreferencias`, que heredan de `JFrame`, junto con un derivado de la ventana principal, `VentanaPrincipalMac` que adapta la clase `VentanaPrincipal` a Mac OS X. Existen clases para editar las listas (`EditorListas`) y las tareas (`EditorTareas`), que heredan de la clase abstracta `Editor`, implementando el método `guardar()`.

También existen los dos visores, `VisorTareas` y `VisorListas`, ambos heredando de `JPanel`. Para facilitar el comportamiento especial de algunos controles, hay clases que heredan de los componentes de Swing: `SJComboBox`, `SJTextField` y `SJLabel`. Asimismo, este paquete contiene la clase principal del programa: `Main`.

3.2 Listas

El paquete `listas` contiene clases relacionadas con el manejo de listas. Una de esas clases es la clase `Lista`, que modela una lista compuesta de elementos (objetos de la clase `Elemento`).

Este paquete contiene también la interfaz `ControlListas`, una especificación abstracta de los métodos que deben implementar los controladores para las listas. Uno de esos controladores, el que usa esta versión de `Sparebrain`, es `ControlListasFicheros`, que utiliza archivos de texto para almacenar las listas.

3.3 Tareas

El paquete `tareas` contiene clases relacionadas con el manejo de tareas. Así, la clase `Tarea` modela una tarea con sus correspondientes campos y restricciones.

Este paquete contiene también la interfaz `ControlTareas`, una especificación abstracta de los métodos que deben implementar los controladores para las tareas. Uno de esos controladores es `ControlTareasFicheros`, que utiliza archivos de texto delimitados para almacenar las tareas.

3.4 Útiles

El paquete `util` contiene ciertas clases auxiliares para el conjunto de la aplicación. En primer lugar `FactoriaArchivos` proporciona un acceso uniforme a las rutas de los archivos del equipo, adaptándose al sistema operativo que se ejecuta y encapsulando la resolución de rutas y ficheros. En segundo lugar, la clase `Fichero` proporciona métodos y atributos específicos de la capa de persistencia para proporcionar un acceso uniforme al acceso a datos. Proporciona métodos para la lectura y escritura de cadenas, separadores universales, etc.

La clase `Propiedades` encapsula la clase `Properties`, facilitando la sintaxis en el acceso a propiedades y controlando las excepciones. Por último, la clase `Traducción` reúne el acceso a las cadenas de texto de las traducciones, ahorrando operaciones a la clase `Propiedades` y simplificando la sintaxis.

3.5 Excepciones

El paquete `excepciones` contiene una única clase de excepción personalizada, `CaracterIlegal`, que se lanza cuando se intenta escribir una tarea, lista o elemento cuyo uno de sus campos contiene un carácter que ya se está utilizando como separador en los ficheros.

3.6 Recursos

Finalmente, el paquete `recursos` contiene los archivos de íconos y traducciones en formato XML. Este enfoque permite contener toda la aplicación en un único archivo JAR.