

Escola Universitària Politécnica de Mataró

Centre adscrit a:



UNIVERSITAT POLITÈCNICA
DE CATALUNYA

Enginyeria Tècnica en Informàtica de Gestió

WRITER'S HELP

Memòria

**ALBERTO GUERRERO CORRAL
PONENT: CATALINA JUAN NADAL**

PRIMAVERA 2015



**TecnoCampus
Mataró-Maresme**

Dedicatoria

A mi padre. Siempre serás mi estrella.

A mi hijo. Siempre seré tu estrella.

Resum

L'objectiu del present projecte és la realització d'una aplicació que sigui capaç d'ajudar a l'escriptor en el procés de creació d'un llibre, simplificant els innombrables documents que poden arribar a crear.

L'aplicació Writer's Help s'ha realitzat mitjançant llenguatge Java en un entorn d'escriptori i per a un únic usuari. Com a base de dades s'ha utilitzat MySQL.

Resumen

El objetivo del presente proyecto es la realización de una aplicación que sea capaz de ayudar al escritor en el proceso de creación de un libro, simplificando los innumerables documentos que pueden llegar a crear.

La aplicación Writer's Help se ha realizado mediante lenguaje Java en un entorno de escritorio y para un único. Como base de datos se ha utilizado MySQL.

Abstract

The objective of this project is the development of an application that is able to help the writer in the process of creating a book, simplifying the countless documents that may create .

The Writer's Help application was performed using Java language in a desktop environment and a single user. As database has been used MySQL.

Índice

Índice de Figuras	ix
Índice de tablas	xi
Glosario de términos	xiii
1. Objetivos.....	1
1.1. Propósito.....	1
1.2. Finalidad.....	1
1.3. Objeto	1
1.4. Alcance	1
2. Información sobre la realización del proyecto.....	3
2.1. Antes de iniciar el proyecto.	3
2.2. Definición del alcance del proyecto.....	4
2.2.1. Objetivos y funcionalidades	4
2.2.2. Características de los usuarios.....	5
2.2.3. Restricciones generales	5
2.2.4. Resultados esperados.....	6
2.2.5. Etapas	6
2.3. Definición de los casos de uso.....	6
3. Realización del proyecto.....	25
3.1. Recopilación de información.	25
3.1.1. Herramientas utilizadas	25
3.1.2. Diseño de la base de datos.....	26
3.1.3. Modelo entidad relación.....	27
3.1.4. Modelo del domino	32
3.1.5. Capa aplicación.	35
3.1.6. Capa persistencia.....	36
3.2. Realización de pruebas	38
3.2.1. Pruebas del caso de uso Creación y modificación de un personaje.....	38
3.3. Documentación.....	45
4. Estudio económico.....	47
4.1. Costes de recursos humanos	47
4.2. Amortización de equipos y software	47
4.3. Implementación del prototipo.....	48
5. Conclusiones.....	49

Índice de Figuras

Fig. 1 Modelo de datos	28
Fig. 2 Modelo del dominio	32
Fig. 3 Capa presentación	33
Fig. 4 Capa aplicación	35
Fig. 5 Capa persistencia.....	36
Fig. 6 Método getPersonajesByLi	37
Fig. 7 Captura del árbol vacío	39
Fig. 8 Captura de pantalla Crear Personaje	40
Fig. 9 Captura de pantalla Crear Personaje (aviso)	41
Fig. 10 Captura de pantalla Crear Personaje (rellena)	41
Fig. 11 Captura de pantalla Crear Personaje (creado)	42
Fig. 12 Captura del árbol lleno	43
Fig. 13 Captura de pantalla Modificar Personaje (Localización)	44
Fig. 14 Captura de pantalla Modificar Personaje (Objeto).....	44
Fig. 15 Captura de pantalla Modificar Personaje (Modificado)	45

Índice de tablas

Tabla 3.1 Script de creación de la tabla Libro	30
Tabla 3.2 Script de creación de la tabla Personaje	31
Tabla 4.1 Coste de recursos humanos	47
Tabla 4.2 Amortización de equipos y software	48
Tabla 4.3 Implementación del prototipo.....	48

Glosario de términos

EUPMT	Escola Universitària Politècnica de Mataró
PFC	Proyecto final de carrera
TFC	Trabajo final de carrera
WH	Abreviatura de Writer's Help
ESCRITOR	Usuario de la aplicación
AUTOR	Sinónimo de ESCRITOR
CD	Compact Disc
RFXXX	Estándar seguido para indicar cada uno de los requisitos funcionales.

1. Objetivos

1.1. Propósito

El propósito del PFC es la de diseñar y construir una aplicación informática que facilite al usuario la planificación y creación de una novela. La aplicación será desarrollada en Java con una base de datos local y se ejecutará en entorno de escritorio.

1.2. Finalidad

La finalidad del proyecto es la de ayudar al escritor a planificarse, con el fin de que pueda escribir el libro de la manera más redonda posible.

1.3. Objeto

Encontramos la aplicación, realizada en JAVA, junto con la base de datos utilizada, realizada en MySQL, junto con el diagrama y Script resultantes. También disponemos de la memoria completa de la aplicación.

1.4. Alcance

Con la aplicación realizada en este PFC se pueden crear diferentes fichas que ayudaran al autor a confeccionar su libro basándose en sus necesidades. Queda fuera del proyecto, ya que no era el objetivo, que el autor pueda escribir su libro propiamente en la aplicación.

2. Información sobre la realización del proyecto

A continuación, explicaré las diferentes fases que ha tenido WH, desde la fase inicial de planificación hasta la fase final de pruebas.

2.1. Antes de iniciar el proyecto.

La idea del proyecto Writer's Help surgió a raíz de mi propia experiencia al introducirme en el mundo de la literatura.

Nunca tuve excesivos problemas al escribir relatos cortos. Solían ser sencillos ya que la clave radicaba en la brevedad de la historia, nunca más de 20 páginas, lo que obligaba a la utilización de pocos personajes y localizaciones para evitar que el lector se abrumara en tan pocas páginas.

Los problemas aparecieron cuando traté de abordar mi primera novela. Fue entonces cuando descubrí la importancia de una buena planificación, de tener claro cuáles eran las motivaciones de un personaje determinado o como era exactamente ese edificio donde transcurría el capítulo. Era vital saber hacia dónde conducían cada una de las tramas, como recordar esa idea que había tenido para aquel capítulo. Esta "planificación" solían ser ficheros de texto guardados en la carpeta del libro correspondiente.

Estos ficheros fueron multiplicándose y fue entonces, cuando empecé a olvidar detalles de personajes, tramas o localizaciones.

Hablando con compañeros de afición, como Carlos Sisí (*Los caminantes, La hora del mar*), Juan de Dios Garduño (*Y pese a todo...*) y Alberto Caliani (*El secreto de Boca verde, La conspiración del rey muerto*) me confesaron que a ellos les ocurría lo mismo, que podían describir los ojos de un personaje como azules, para transformarlos capítulos más tarde como verdes. Cosas así ocurrían.

Fue así como empezó a forjarse la idea que acabaría convirtiéndose en este PFC y en la aplicación Writer's Help. Quería facilitar la vida a los escritores y por supuesto a mí mismo.

Para ello empecé el estudio de aplicaciones similares en el mercado, descubriendo que no era especialmente amplio. La más famosa de todas era Scrivener, aunque la mayoría no dejaban de ser procesadores de texto con algunas características concretas.

Quería algo diferente, no tanto un procesador en el que escribir la novela en sí, sino una aplicación que realmente facilitara la faena del escritor. Por experiencia sé que la mayoría de escritores están sumamente habituados a un procesador de texto. El que trabaja con Microsoft Office Word es complicado que se pase a Apache OpenOffice o WordPad y a la inversa pasa lo mismo. Por esa razón descarté incorporar un procesador de textos en la aplicación, cosa que además, en mi opinión, hubiera superado las expectativas de lo que se espera en un PFC.

2.2. Definición del alcance del proyecto

El siguiente paso fue definir el alcance del proyecto. Qué se podría hacer la aplicación y con qué finalidad. Writer's Help realizaría aquellas funciones que un escritor podría realizar en una simple hoja de papel o en un documento de texto. Apuntar ideas, elaborar subtramas, especificar las características de un personaje o las de ese objeto tan importante en su obra.

2.2.1. Objetivos y funcionalidades

En resumen los objetivos y funcionalidades que debería cumplir la aplicación WH se podrían dividir en:

- Creación: El usuario podrá crear diferentes libros, añadiendo a cada uno de ellos los capítulos y las escenas que lo constituyen.

También podrá crear las diferentes fichas de las entidades que forman parte del libro (Personajes, Objetos, Localizaciones, Razas, Ideas y Subtramas).

- Edición: El usuario podrá editar la información, con la finalidad de indicar las características y rasgos destacables, de cada una de estas fichas.
- Eliminación: El usuario podrá eliminar cualquier ficha, así como escenas, capítulos o el propio libro.

- **Importación:** La aplicación debe permitir la importación entre libros de las fichas de Personajes, Objetos, Localizaciones y Razas. Así como de ideas y Subtramas que hayan quedado descartadas en otros libros. La importación no plantea importar tan bien las relaciones, ya que éstas no tienen porque existir en el nuevo libro.

Ejemplo: Queremos importar el personaje A del libro 1 al libro 2. El personaje A, en el libro 1 tiene una relación de archienemigo con el personaje B. En el libro 2 queremos que aparezca el personaje A y lo importamos, pero el personaje B no se importará automáticamente, ya que no tiene por que salir en el libro 2

De igual manera, en la importación del personaje A sí se importaran su raza y su localización, ya que esas sí que son características que tendrá el personaje este en el libro que este.

2.2.2. Características de los usuarios

La aplicación no contempla la necesidad de realizar diferentes tipos de usuarios, ya que se trata de una aplicación dirigida a un único usuario con privilegios totales y sin entorno de red.

Los usuarios de la aplicación están familiarizados con el uso de ordenadores y aplicaciones de tratamiento de textos. Es por ello que la aplicación debe recoger todas sus necesidades a la par que constar con un interfaz sencillo, intuitivo y muy estructurado.

Como curiosidad, creo que la aplicación también podría estar destinada a creadores de cortos o pequeñas películas. Tal como está pensada la aplicación, me hace creer que esta posibilidad existe.

2.2.3. Restricciones generales

No existen restricciones específicas, se puede ejecutar en cualquier equipo de sobremesa o portátil de gama media o baja. Para una óptima visualización de la aplicación, se recomienda una resolución de pantalla de 1280 x 1024.

2.2.4. Resultados esperados

Los resultados y productos que se obtendrán son:

- Una aplicación de escritorio con la que visualizar de forma fácil la información guardada en cada una de la fichas.
- Facilitar la gestión de dicha información.

2.2.5. Etapas

El método de desarrollo del proyecto se basará en las siguientes etapas:

- Definición de los casos de uso.
- Modelo entidad relación de la base de datos.
- La codificación de la aplicación Writer's Help.
- La realización de las pruebas de la aplicación.

2.3. Definición de los casos de uso

En este apartado se especificaran los requisitos que debe satisfacer la aplicación para ser aceptable. Los requisitos se han especificado de manera que sea sencilla la comprobación de que la aplicación los cumple.

- **Mantenimiento y gestión de los libros creados.**

RF001	Crear Libro
Introducción	Mediante una interfaz gráfica se le pedirán al usuario los datos necesarios para la creación de un nuevo libro. Estos datos, en su gran mayoría se podrán rellenar con posterioridad, a excepción del título de la obra y de su estado que son obligatorios.
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto la aplicación mostrará la pantalla Arbol2 del nuevo libro.

RF002	Eliminar Libro
Introducción	Mediante la pantalla Modificación del libro, el usuario podrá eliminar un libro y todas las fichas que cuelguen de él.
Entradas	
Proceso	La aplicación pedirá confirmación al usuario antes de realizar la operación. En caso afirmativo, procederá. En caso negativo, no hará nada.
Salida	La aplicación lanzará el proceso de borrado del libro y volverá a la página Home de la aplicación.

RF003	Modificar Libro
Introducción	Mediante una interfaz gráfica el usuario podrá modificar los datos de un libro. Estos datos, en su gran mayoría se podrán rellenar con posterioridad, a excepción del título de la obra y de su estado que son obligatorios.
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto la aplicación volverá a la pantalla Arbol2 del libro modificando el nodo raíz.

- **Mantenimiento y gestión de las fichas (Personajes, Objetos, Localizaciones y Razas).**

RF004	Crear Ficha (Personajes, Objetos, Localizaciones o Razas)
Introducción	Mediante una interfaz gráfica se le pedirán al usuario los datos necesarios para la creación de una nueva ficha. Estos datos, en su gran mayoría se podrán rellenar con posterioridad a excepción del nombre y de la categoría en el caso de tratarse de un personaje o el tipo si es un objeto, que son obligatorios.
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto la aplicación volverá a la pantalla Arbol2 del libro añadiendo el nuevo nodo a la rama correspondiente.

RF005	Eliminar Ficha (Personajes, Objetos, Localizaciones o Razas)
Introducción	Mediante la pantalla Modificación de la ficha elegida, el usuario podrá eliminarla, eliminando también las relaciones que esta pueda tener.
Entradas	
Proceso	La aplicación pedirá la confirmación al usuario antes de realizar la operación de eliminación. En caso afirmativo, procederá a la eliminación de la ficha junto con las relaciones que esta pueda tener. En caso negativo, no hará nada.
Salida	En el caso de que todo sea correcto la aplicación volverá a la pantalla Arbol2 del libro eliminando el nodo de la rama correspondiente.

RF006	Modificar Ficha (Personajes, Objetos, Localizaciones o Razas)
Introducción	Mediante una interfaz gráfica se le mostraran al usuario los datos de la ficha seleccionada, pudiéndolos modificar. Estos datos, en su gran mayoría se podrán rellenar con posterioridad a excepción del nombre y de la categoría en el caso de tratarse de un personaje o el tipo si es un objeto, que son obligatorios.
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto la aplicación volverá a la pantalla Arbol2 del libro modificando el nodo de la rama correspondiente.

- **Importación de fichas entre libros.**

RF007	Importar Fichas
Introducción	Este requisito hace referencia a la posibilidad de importar fichas de personajes, objetos, localizaciones o razas de otro libro al actual. Este proceso se puede llevar a cabo en cualquier momento desde el nodo importar del árbol del libro.
Entradas	El usuario podrá escoger que fichas quiere importar al libro actual de entre todas las existentes en la base de datos.
Proceso	<p>La aplicación procederá a incorporar dichas fichas al libro actual. El proceso no guardará las posibles relaciones que tenga la ficha a importar con otros objetos o personajes, por entenderse que estos no tienen porque existir en el nuevo libro.</p> <p>Cualquier modificación en alguna de ellas no implica la modificación de la ficha original, pues pasarán a ser independientes las unas de las otras.</p>
Salida	En el caso de que todo sea correcto la aplicación volverá a la pantalla Arbol2 del libro añadiendo las nuevas fichas a sus correspondientes ramas del árbol.

- **Mantenimiento y gestión de ideas.**

RF008	Crear Idea
Introducción	<p>Este requisito hace referencia a la posibilidad de crear una idea que pueda formar parte del libro. Mediante una interfaz gráfica se le pedirán al usuario los datos correspondientes a la idea. Estos datos, en su gran mayoría se podrán rellenar con posterioridad a excepción del nombre y de su estado que son obligatorios.</p> <p>La idea se podrá relacionar con las fichas Personaje, Objeto, Localización y Raza indicando así que tiene que ver con esas fichas.</p>
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	<p>La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos.</p> <p>En caso contrario se mostrará una ventana de aviso en pantalla.</p>
Salida	En el caso de que todo sea correcto se volverá a la pantalla Arbol2, añadiendo el nodo creado a la rama Ideas.

RF009	Eliminar Idea
Introducción	Mediante la pantalla Modificación de la idea, el usuario podrá eliminarla, eliminando las relaciones que esta pueda tener.
Entradas	
Proceso	La aplicación pedirá la confirmación al usuario antes de realizar la operación de eliminación. En caso afirmativo, procederá a la eliminación de la idea junto con las relaciones que esta pueda tener. En caso negativo, no hará nada.
Salida	En el caso de que todo sea correcto aparecerá la pantalla <i>Árbol del libro</i> , eliminando el nodo de la rama Ideas del árbol del libro.

RF010	Modificar Idea
Introducción	Mediante una interfaz gráfica se le mostraran al usuario los datos de la idea seleccionada, pudiéndolos modificar. Estos datos, en su gran mayoría se podrán rellenar con posterioridad a excepción del nombre y de su estado, que son obligatorios
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto se mostrará la pantalla <i>Arbol2 del libro</i> , modificando el nodo correspondiente a la idea.

- **Importación de ideas entre libros.**

RF011	Importar Ideas descartadas
Introducción	Este requisito hace referencia a la posibilidad de importar al libro actual Ideas que hayan sido descartadas en otros libros ya creados. Este proceso se puede llevar a cabo en cualquier momento desde el nodo importar del árbol del libro.
Entradas	El usuario podrá escoger las ideas que desee importar a su libro actual. Las ideas mostradas son únicamente las que han sido descartadas en otro libro.
Proceso	La aplicación procederá a incorporar dichas ideas al libro actual. El proceso no guardará las posibles relaciones que tenga la idea a importar, por entenderse que estos personajes, objetos, localizaciones o razas no tienen por qué existir en el nuevo libro. Cualquier modificación realizada en la idea ya importada no implica la modificación de la idea original, pues pasarán a ser independientes las unas de las otras.
Salida	En el caso de que todo sea correcto la aplicación volverá a la pantalla Arbol2 del libro actual, incorporando las nuevas ideas a su correspondiente subcarpeta.

- **Mantenimiento y gestión de subtramas.**

RF012	Crear Subtrama
Introducción	Este requisito hace referencia a la posibilidad de guardar una subtrama que pueda llegar a formar parte del libro. Mediante una interfaz gráfica se le pedirán al usuario los datos correspondientes a la subtrama. Estos datos, en su gran mayoría se podrán rellenar con posterioridad a excepción del nombre y del estado que serán obligatorios. La subtrama se podrá relacionar con las fichas Personaje, Objeto, Localización y Raza indicando así que estas fichas aparecen en ella.
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto se volverá a la pantalla Arbol2 del libro, añadiendo el nodo creado en la rama Subtramas.

RF013	Eliminar Subtrama
Introducción	Mediante la pantalla Modificación de la subtrama, el usuario podrá eliminarla, eliminando las relaciones que esta pueda tener.
Entradas	
Proceso	La aplicación pedirá la confirmación del usuario antes de realizar la operación de eliminación. En caso afirmativo, procederá a la eliminación, eliminando también las relaciones que tenga con otras fichas. En caso negativo, no hará nada.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, eliminando el nodo seleccionado de la rama Subtramas.

RF014	Modificar Subtrama
Introducción	Mediante una interfaz gráfica se le mostraran al usuario los datos de la subtrama seleccionada, pudiéndolos modificar. Estos datos, en su gran mayoría se podrán rellenar con posterioridad a excepción del nombre y de su estado, que son obligatorios
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, modificando el nodo correspondiente a la subtrama seleccionada.

- **Importación de subtramas entre libros.**

RF015	Importar Subtramas descartadas
Introducción	Este requisito hace referencia a la posibilidad de importar al libro actual Subtramas que hayan sido descartadas en otro libro ya creados. Este proceso se puede llevar a cabo en cualquier momento desde el nodo importar del árbol del libro.
Entradas	El usuario podrá escoger las Subtramas que desee importar a su libro actual. Las Subtramas mostradas son únicamente las que han sido descartadas en otro libro.
Proceso	La aplicación procederá a incorporar dichas Subtramas al libro actual. El proceso no guardará las posibles relaciones que tenga la subtrama a importar, por entenderse que estos personajes, objetos, localizaciones o razas no tienen por qué existir en el nuevo libro. Cualquier modificación realizada en la subtrama ya importada no implica la modificación de la idea original, pues pasarán a ser independientes las unas de las otras.
Salida	En el caso de que todo sea correcto la aplicación volverá a la pantalla Arbol2 del libro actual, incorporando las nuevas ideas a su correspondiente subcarpeta.

- **Mantenimiento y gestión de capítulos.**

RF016	Crear Capítulo
Introducción	Este requisito hace referencia a la posibilidad de crear capítulos en un libro. Mediante una interfaz gráfica se le pedirán al usuario los datos correspondientes al nuevo capítulo. Estos datos, en su gran mayoría se podrán rellenar con posterioridad a excepción del número de capítulo, del título y del estado.
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro añadiendo un nuevo nodo a la rama Capítulos.

RF017	Eliminar Capítulo
Introducción	Mediante la pantalla Modificación de capítulo, el usuario podrá eliminarlo. Eliminando también todas las escenas que estén relacionadas con él.
Entradas	
Proceso	La aplicación pedirá la confirmación del usuario antes de realizar la operación de eliminación. En caso afirmativo, procederá a la eliminación del capítulo junto a las escenas relacionadas. En caso negativo, no hará nada.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, eliminando el nodo correspondiente al capítulo.

RF018	Modificar Capítulo
Introducción	Mediante una interfaz gráfica aparecerán los datos correspondientes al capítulo seleccionado, pudiéndolos modificar, sin olvidar que el número de capítulo, el título y su estado son campos obligatorios.
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, modificando el nodo en la rama capítulos del árbol.

- **Mantenimiento y gestión de escenas.**

RF019	Crear Escena
Introducción	<p>Este requisito hace referencia a la posibilidad de crear una escena perteneciente a un capítulo. Mediante una interfaz gráfica se le pedirán al usuario los datos correspondientes a la nueva escena. Estos datos, en su gran mayoría se podrán rellenar con posterioridad a excepción del número de escena y de su estado.</p> <p>La creación de escenas se hará desde la pantalla Crear o Modificar Capítulo.</p>
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	<p>La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación.</p> <p>En caso contrario se mostrará una ventana de aviso en pantalla.</p> <p>Estos datos se grabaran en base de datos en el momento en el que se grabe el capítulo al que pertenecen.</p>
Salida	En el caso de que todo sea correcto volverá a la ficha de Capítulo a la que pertenecen. Cuando esta ficha se grabe en base de datos, aparecerá un nuevo nodo por cada escena creada bajo la rama Capítulo.

RF020	Eliminar Escena
Introducción	Mediante la pantalla Modificación de la ficha correspondiente a la escena deseada, el usuario podrá eliminarla, eliminando también las relaciones que esta pueda tener.
Entradas	
Proceso	<p>La aplicación pedirá la confirmación del usuario antes de realizar la operación de eliminación.</p> <p>En caso afirmativo, procederá a la eliminación de la escena junto a las relaciones que esta pueda tener.</p> <p>En caso negativo, no hará nada.</p>
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, eliminando el nodo de la rama de su capítulo.

RF021	Modificar Escena
Introducción	Este requisito hace referencia a la posibilidad de modificar los datos de una escena ya existente. Mediante una interfaz gráfica el usuario podrá ver todos los datos y modificar los que desee, sin olvidar que el número de escena y su estado son campos obligatorios.
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a su grabación en base de datos. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, modificando el nodo en la rama del capítulo a la que pertenece.

- **Asignación de fichas a escenas.**

RF022	Añadir fichas a escenas
Introducción	Este requisito hace referencia a la posibilidad de indicar la participación de personajes, objetos, localizaciones y razas en una escena mediante su selección en la pantalla Crear o Modificar escena.
Entradas	
Proceso	La aplicación procederá a grabar las relaciones en el momento en que grabe la escena en base de datos.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, modificando el nodo en la rama del capítulo a la que pertenece.

RF023	Eliminar fichas de escenas
Introducción	Este requisito hace referencia a la posibilidad de eliminar la participación de personajes, objetos, localizaciones y razas de una escena.
Entradas	
Proceso	La aplicación procederá a eliminar las relaciones en el momento en que grabe la escena en base de datos.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, eliminando el nodo de la rama capítulo a la que pertenece.

- **Asignación de fichas a ideas.**

RF024	Añadir fichas a Ideas
Introducción	Este requisito hace referencia a la posibilidad de indicar que una idea está relacionada con determinados personajes, objetos, localizaciones y razas mediante su selección en pantalla.
Entradas	
Proceso	La aplicación procederá a grabar las relaciones en el momento en que grabe la Idea en base de datos.
Salida	En el caso de que todo sea correcto aparecerá la pantalla Arbol2 del libro, modificando el nodo Idea.

RF025	Eliminar fichas de Idea
Introducción	Este requisito hace referencia a la posibilidad de eliminar la relación de personajes, objetos, localizaciones o razas con una idea.
Entradas	
Proceso	La aplicación procederá a eliminar las relaciones en el momento en que grabe la idea en base de datos.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, modificando el nodo Idea en la rama de Ideas.

- **Asignación de fichas a subtramas.**

RF026	Añadir fichas a Subtrama
Introducción	Este requisito hace referencia a la posibilidad de indicar que una Subtrama está relacionada con determinados personajes, objetos, localizaciones y razas mediante su selección en pantalla.
Entradas	
Proceso	La aplicación procederá a grabar las relaciones en el momento en que grabe la Subtrama en base de datos.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, modificando el nodo subtrama.

RF027	Eliminar fichas de Subtrama
Introducción	Este requisito hace referencia a la posibilidad de eliminar la participación de personajes, objetos, localizaciones o razas en una Subtrama.
Entradas	
Proceso	La aplicación procederá a eliminar las relaciones en el momento en que grabe la Subtrama en base de datos.
Salida	En el caso de que todo sea correcto volverá a la pantalla Arbol2 del libro, modificando el nodo Subtrama en la rama de Subtramas.

- **Relación entre personajes.**

RF028	Añadir relación personaje - personaje
Introducción	Este requisito hace referencia a la posibilidad de indicar que un personaje tiene relación con otro. Deberá indicar que tipo de relación tienen. Esta relación se puede añadir a partir de la pantalla de Crear o Modificar Personaje.
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación comprobará si los datos obligatorios están rellenos y se procederá a guardar los datos que se grabaran en base de datos en el momento en el que lo haga el personaje. En caso contrario se mostrará una ventana de aviso en pantalla.
Salida	En el caso de que todo sea correcto aparecerá la pantalla Crear o modificar personaje. Una vez grabado este, se modificará el nodo en el árbol.

RF029	Eliminar relación personaje - personaje
Introducción	<p>Este requisito hace referencia a la posibilidad de eliminar la relación de un personaje con otro.</p> <p>Esta relación se puede eliminar desde la pantalla de Crear o Modificar Personaje.</p>
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación procederá a guardar los datos, que se eliminarán de la base de datos en el momento en el que se grabe el personaje.
Salida	En el caso de que todo sea correcto aparecerá la pantalla Crear o modificar personaje. Una vez grabado este, se modificará el nodo en el árbol.

RF030	Modificar relación personaje - personaje
Introducción	<p>Este requisito hace referencia a la posibilidad de modificar una relación entre personajes ya existente. Deberá indicar que tipo de relación tienen.</p> <p>Esta relación se puede modificar a partir de la pantalla de Crear o Modificar Personaje.</p>
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	<p>La aplicación comprobará si los datos obligatorios están rellenos y se procederá a guardar los datos que se grabarán en base de datos en el momento en el que lo haga el personaje.</p> <p>En caso contrario se mostrará una ventana de aviso en pantalla.</p>
Salida	En el caso de que todo sea correcto aparecerá la pantalla Crear o modificar personaje. Una vez grabado este, se modificará el nodo en el árbol.

- **Relación entre personajes y objetos.**

RF031	Añadir relación personaje - objeto
Introducción	<p>Este requisito hace referencia a la posibilidad de indicar que un personaje tiene relación con un objeto. Deberá indicar que tipo de relación es.</p> <p>Esta relación se puede añadir a partir de la pantalla de Crear o Modificar Personaje.</p>
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	<p>La aplicación comprobará si los datos obligatorios están rellenos y se procederá a guardar los datos que se grabaran en base de datos en el momento en el que lo haga el personaje.</p> <p>En caso contrario se mostrará una ventana de aviso en pantalla.</p>
Salida	En el caso de que todo sea correcto aparecerá la pantalla Crear o modificar personaje. Una vez grabado este, se modificará el nodo en el árbol.

RF032	Eliminar relación personaje - objeto
Introducción	<p>Este requisito hace referencia a la posibilidad de eliminar la relación de un personaje con un objeto.</p> <p>Esta relación se puede eliminar desde la pantalla de Crear o Modificar Personaje.</p>
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	La aplicación procederá a guardar los datos, que se eliminaran de la base de datos en el momento en el que se grabe el personaje.
Salida	En el caso de que todo sea correcto aparecerá la pantalla Crear o modificar personaje. Una vez grabado este, se modificará el nodo en el árbol.

RF033	Modificar relación personaje - objeto
Introducción	<p>Este requisito hace referencia a la posibilidad de modificar una relación entre un personaje y un objeto. Deberá indicar que tipo de relación tienen.</p> <p>Esta relación se puede modificar a partir de la pantalla de Crear o Modificar Personaje.</p>
Entradas	Todos los campos requieren valores alfanuméricos.
Proceso	<p>La aplicación comprobará si los datos obligatorios están rellenos y se procederá a guardar los datos que se grabaran en base de datos en el momento en el que lo haga el personaje.</p> <p>En caso contrario se mostrará una ventana de aviso en pantalla.</p>
Salida	En el caso de que todo sea correcto aparecerá la pantalla Crear o modificar personaje. Una vez grabado este, se modificará el nodo en el árbol.

- **Mantenimiento de fotografías.**

RF034	Añadir imagen a ficha
Introducción	<p>Este requisito hace referencia a la posibilidad de añadir imágenes o bocetos a las diferentes fichas (Personaje, Objeto, Localización, Raza).</p> <p>Estas imágenes se pueden añadir a partir de la pantalla de Crear o Modificar ficha.</p>
Entradas	Las imágenes deben ser de un formato permitido: JPG, JPEG o GIF.
Proceso	La aplicación procederá a guardar las imágenes, que se grabaran en base de datos en el momento en el que lo haga el personaje.
Salida	En el caso de que todo sea correcto aparecerá la pantalla Arbol2 del libro, modificando el nodo.

RF035	Eliminar imagen de ficha
Introducción	Este requisito hace referencia a la posibilidad de eliminar imágenes o bocetos de las diferentes fichas (Personaje, Objeto, Localización, Raza). Estas imágenes se pueden eliminar desde la pantalla Crear o Modificar ficha.
Entradas	
Proceso	La aplicación procederá a eliminar las imágenes, que se eliminarán de la base de datos en el momento en el que se grave el personaje.
Salida	En el caso de que todo sea correcto aparecerá la pantalla Arbol2 del libro, modificando el nodo.

3. Realización del proyecto.

En esta fase del proyecto se abordó la recopilación real de información que se necesitaría para desarrollar la aplicación.

3.1. Recopilación de información.

Aunque la aplicación en principio no debe ser utilizada por usuarios reales, sí que se habló con usuarios potenciales.

Como ya he comentado con anterioridad, se habló con escritores como Juan de Dios Garduño, Carlos Sisí y Alberto Caliani, que tuvieron la amabilidad de darme sus impresiones sobre lo que les gustaría tener en una aplicación de estas características.

Gracias a la información resultante de estas conversaciones, elaboré los casos de uso mostrados en el capítulo anterior y la estructura de la base de datos.

Otro claro ejemplo de la recopilación de información que se realizó fue la de incluir la característica Narrador en el Libro, que indica cómo está escrito el libro y que explico a continuación:

- 3ª Persona (Narrador Omnisciente): El narrador omnisciente es aquel que todo lo sabe. Conoce los hechos y sabe lo que piensan y sienten los personajes.
- 3ª Persona (Narrador Observador): Únicamente cuenta lo que observa. Cuenta la historia de modo parecido a como lo haría una cámara de cine.
- 1ª Persona (Protagonista): A parte de narrador es también el protagonista.
- 1ª Persona (Personaje Secundario): Es un personaje observador de los hechos.
- 2ª Persona: El narrador cuenta la historia en 2ª persona, creando el efecto de estar contándosela a sí mismo.

3.1.1. Herramientas utilizadas

Las herramientas que se han utilizado en la elaboración del PFC son:

- Base de datos: La base de datos utilizada para el proyecto ha sido MySQL (<http://www.mysql.com>). Esta elección se ha debido principalmente, por tratarse de una base de datos relacional, teniendo también en cuenta su

gratuidad. Con anterioridad no había trabajado con MySQL, con lo que además era un valor añadido. Se ha utilizado la herramienta MySQL Workbench 6.3 (<https://www.mysql.com/products/workbench/>) que engloba todo lo necesario para elaborar el diseño, la administración y la realización de las consultas.

- Entorno de desarrollo: Se ha utilizado Eclipse (<http://www.eclipse.org>) para la codificación de la aplicación. Concretamente la versión Indigo. Para implementar la interfaz grafica de usuario se ha utilizado la biblioteca grafica Swing.
- Como Java Runtime Environment se ha utilizado el 1.6.
- Editor de textos: Para la elaboración de la documentación necesaria se ha utilizado Microsoft Office Word.
- Para el soporte tanto en código como en textos, se ha utilizado Notepad++ (<http://notepad-plus-plus.org>).
- Sistema operativo: La aplicación se ha desarrollado bajo Windows, principalmente bajo Windows 8.1 lo cual nos dio algún problema a la hora de instalar MySQL.

Como características comunes a la elección de las herramientas de desarrollo se ha favorecido las que son de libre distribución. También he tenido en cuenta que tuviesen una amplia comunidad de usuarios y una buena integración entre ellas.

Finalmente, señalar, que el no haber trabajado anteriormente con Swing ha hecho que perdiera más tiempo del deseado.

3.1.2. Diseño de la base de datos.

A la hora de la elaboración de la base de datos, ya en sus primeros bocetos, había entidades que estaba claro que iban a aparecer:

- Libro
- Capítulo
- Escena

- Personaje
- Idea
- Subtrama

Con posterioridad aparecieron nuevas, que enriquecían más aún la futura aplicación.

- Objeto
- Localización
- Raza
- Foto personaje
- Foto objeto
- Foto Localización
- Foto raza

Una vez iniciada la elaboración del modelo de base de datos, aparecieron nuevas tablas propias de las relaciones M:N.

- Escena personaje
- Escena objeto
- Escena localización
- Escena raza
- Idea personaje
- Idea objeto
- Idea localización
- Idea raza
- Subtrama personaje
- Subtrama objeto
- Subtrama localización
- Subtrama raza
- Relación personajes
- Relación personaje objeto

3.1.3. Modelo entidad relación.

El modelo obtenido después de revisiones y correcciones fue el siguiente.

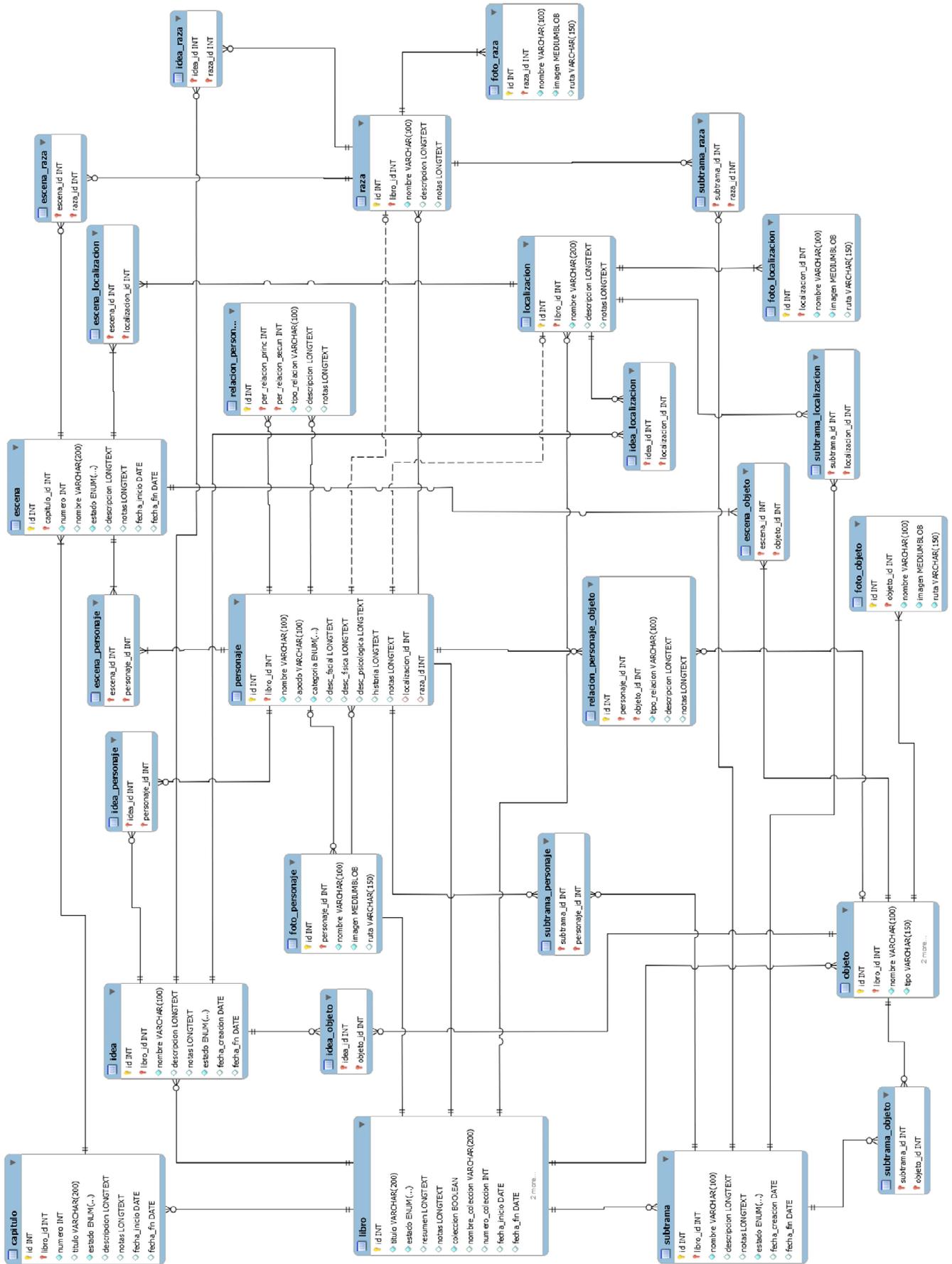


Fig. 1 Modelo de datos

Problemas detectados

Durante el desarrollo del modelo de base de datos, no llegué a detectar dos problemas.

El primero de ellos fue que cometí el error de indicar los atributos obligatorios de las clases como UNIQUE, con lo que a la hora de la verdad por ejemplo solo me dejaba tener un personaje llamado A en todos los libros de la base de datos. Mi intención era que no se repitiesen personajes llamados A en el mismo libro, ya que sería un error por parte del escritor llamar igual a dos personajes. Inevitablemente confundiría a los lectores. La solución que opté fue la de crear índices en las tablas implicadas en las que la clave sería libro_id y nombre (o el atributo necesario). Como ejemplo podemos coger:

```
CREATE UNIQUE INDEX `fk_personaje_unico` ON `writers`.`personaje` (`libro_id` ASC,  
`nombre` ASC);
```

El segundo problema que me encontré no fue un error de definición del modelo de base de datos, sino una limitación por defecto impuesta por el propio MySQL. MySQL tiene una limitación de 4MB a la hora de subir ficheros binarios a la base de datos. Se resolvió modificando el archivo de configuración my.ini o my.cnf (puede encontrarse en diferentes ubicaciones según la versión) donde se cambió la línea que determinaba el valor max_allowed_packet por un valor mayor. También se puede modificar el valor por consola, con la siguiente query que permitirá guardar archivos de hasta 100MB:

```
SET GLOBAL max_allowed_packet=100*1024*1024;
```

Esta query también se ha añadido al script de creación de la base de datos.

Todas las soluciones implican reiniciar el servidor de base de datos.

Script

Una vez resueltos los problemas, la herramienta MySQL Workbench me facilitó la creación del script SQL que acabaría convirtiéndose en la base de datos de la aplicación Writer's Help.

Para evitar alargar este documento innecesariamente (el script de creación de la base de datos se facilitará en el CD de entrega del PFC) solo pondré la parte de creación de la entidad Libro y la entidad Personaje.

```
-----  
-- Table `writers`.`libro`  
-----  
CREATE TABLE IF NOT EXISTS `writers`.`libro` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `titulo` VARCHAR(200) NOT NULL,  
  `estado` ENUM('SIN_EMPEZAR','EN_PROCESO','FINALIZADO') NOT NULL,  
  `resumen` LONGTEXT NULL,  
  `notas` LONGTEXT NULL,  
  `coleccion` TINYINT(1) NOT NULL DEFAULT 0,  
  `nombre_coleccion` VARCHAR(200) NULL,  
  `numero_coleccion` INT NULL,  
  `fecha_inicio` DATE NULL,  
  `fecha_fin` DATE NULL,  
  `genero` VARCHAR(70) NULL,  
  `narrador`  
  ENUM('TERCERA_PERSONA_OMNISCIENTE','TERCERA_PERSONA_OBSERVADOR','PRIM  
ERA_PERSONA_PROTAGONISTA','PRIMERA_PERSONA_SECUNDARIO','SEGUNDA_PERS  
ONA') NULL,  
  PRIMARY KEY (`id`))  
ENGINE = InnoDB;  
  
CREATE UNIQUE INDEX `id_UNIQUE` ON `writers`.`libro` (`id` ASC);  
CREATE UNIQUE INDEX `titulo_UNIQUE` ON `writers`.`libro` (`titulo` ASC);
```

Tabla 3.1 Script de creación de la tabla Libro

A destacar el campo Narrador, un ENUM que indica el estilo de narración en el que está escrito el libro.

```
-----  
-- Table `writers`.`personaje`  
-----  
CREATE TABLE IF NOT EXISTS `writers`.`personaje` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `libro_id` INT NOT NULL,  
  `nombre` VARCHAR(100) NOT NULL,  
  `apodo` VARCHAR(100) NULL,  
  `categoria` ENUM('PRINCIPAL','SECUNDARIO') NOT NULL,  
  `desc_facial` LONGTEXT NULL,  
  `desc_fisica` LONGTEXT NULL,  
  `desc_psicologica` LONGTEXT NULL,  
  `historia` LONGTEXT NULL,  
  `notas` LONGTEXT NULL,  
  `localizacion_id` INT NULL,  
  `raza_id` INT NULL,  
  PRIMARY KEY (`id`, `libro_id`),  
  CONSTRAINT `fk_personaje_libro1`  
    FOREIGN KEY (`libro_id`)  
    REFERENCES `writers`.`libro` (`id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_personaje_localizacion1`  
    FOREIGN KEY (`localizacion_id`)  
    REFERENCES `writers`.`localizacion` (`id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_personaje_raza1`  
    FOREIGN KEY (`raza_id`)  
    REFERENCES `writers`.`raza` (`id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)  
ENGINE = InnoDB;  
  
CREATE INDEX `fk_personaje_libro1_idx` ON `writers`.`personaje` (`libro_id` ASC);  
CREATE INDEX `fk_personaje_localizacion1_idx` ON `writers`.`personaje`  
(`localizacion_id` ASC);  
CREATE UNIQUE INDEX `id_UNIQUE` ON `writers`.`personaje` (`id` ASC);  
CREATE INDEX `fk_personaje_raza1_idx` ON `writers`.`personaje` (`raza_id` ASC);  
CREATE UNIQUE INDEX `fk_personaje_unico` ON `writers`.`personaje` (`libro_id` ASC,  
`nombre` ASC);
```

Tabla 3.2 Script de creación de la tabla Personaje

En este caso cabe destacar, a parte del índice *fk_personaje_unico* ya comentado con anterioridad, los diferentes índices que tiene la tabla con el resto de ellas. Indicando así, que es una de las tablas con mayor importancia.

3.1.4. Modelo del dominio

Una vez creada la base de datos, el siguiente paso era la creación del dominio de la aplicación, con los siguientes términos claros:

- La aplicación debía ser mono usuario.
- La aplicación debía ser una aplicación de escritorio.
- No hacía falta ningún tipo de conexión a la red, ni arquitecturas Servidor Host.

Esta información a tener en cuenta, era importante, ya que de ella dependía la arquitectura interna que tendría la aplicación WH.

El modelo del dominio que se obtuvo es el siguiente (Fig. 2).

A parte de las entidades que ya hemos visto en el modelo de la base de datos, se crearon clases Enum para indicar:

- Estados (Libro, Capítulo, Escena, Idea, Subtrama)
- Tipos (Personaje (Principal o secundario), Libro (Primera persona, Tercera persona...))

Además se crearon Interfaces que indicaran que operaciones se podrían realizar en la aplicación.

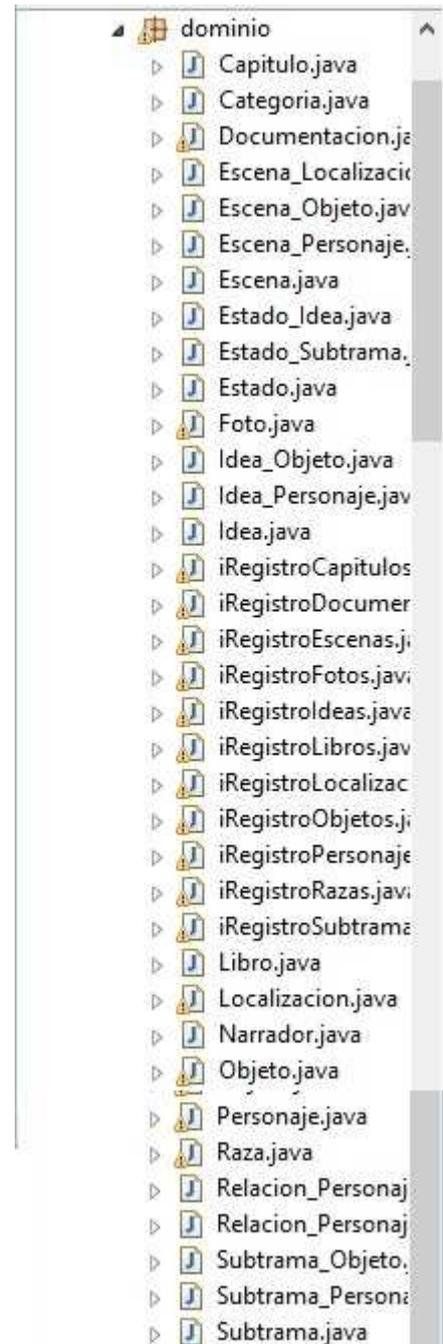


Fig. 2 Modelo del dominio

Capa Presentación.

En la capa presentación se implementaron las pantallas necesarias para una buena interacción entre el usuario y la aplicación.

Es la capa que interactúa directamente con el usuario, mostrándole datos provenientes desde la base de datos e interpretando la información introducida en las diferentes fichas.

La capa presentación está formada por una pantalla Principal que es el *frame* encargado de lanzar la aplicación, una *JInternalFrame* que es la pantalla Home, donde se muestran los diferentes libros que existen en la base de datos divididos en su estado actual (SIN EMPEZAR, EN PROCESO o FINALIZADOS) y la pantalla Arbol2, que es un *JDialog* encargado de mostrar en carpetas y subcarpetas las diferentes fichas de las que se compone el libro.

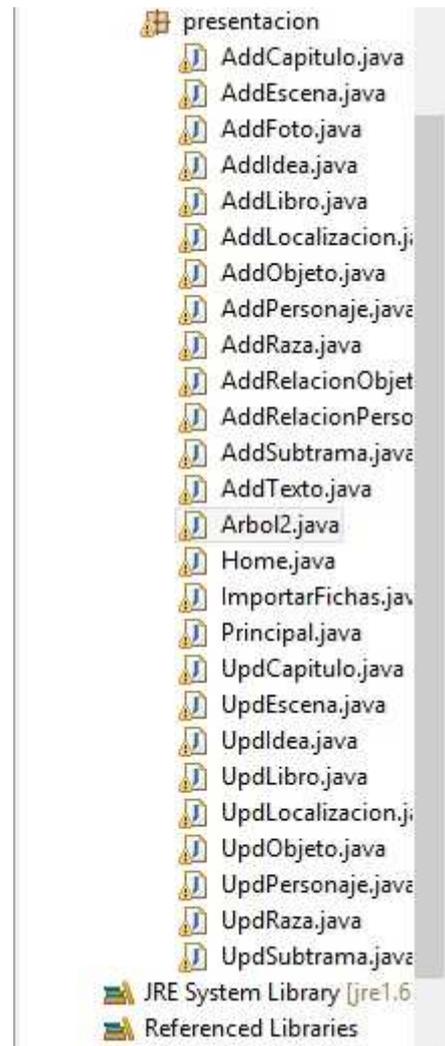


Fig. 3 Capa presentación

Desde este árbol se puede acceder a las diferentes fichas y dependiendo de lo que el usuario realice con ellas, el árbol reaccionará creando, modificando o eliminando nodos en las subcarpetas que le correspondan.

La manera de acceder a estas fichas es sencilla. Si se accede al nodo principal de cada una de las ramas, se abrirá una pantalla ADD, encargadas de la creación de las nuevas fichas. Por ejemplo, si el usuario “pincha” sobre Localizaciones, se abrirá la pantalla para crear una nueva Localización. Pero si por ejemplo “pincha” sobre Personaje_879 se abrirá la pantalla de modificación de dicho personaje.

Para cada ficha existe su pantalla Add (Añadir) y su correspondiente Upd (Modificar).

También se ha intentado no caer en el problema de las múltiples ventanas, por lo que se desea mostrar en cada una de ellas la máxima información posible para que el usuario no esté abriendo una ventana tras otra. De igual forma, en las fichas se ha descartado la utilización de campos muy específicos, para evitar que el usuario se vea abrumado a la hora de rellenarlos. En este aspecto también juega el conocimiento de que cada escritor es distinto a otro y hay quien prefiere detallar al milímetro cualquier rasgo de un personaje y quien prefiere brindarle al lector su nombre y poco más, para que sea él quien lo cree en su mente. Por todos esto, en vez de mostrar un campo color de ojos y otro, color de pelo, se ha optado por uno llamado Descripción facial.

En algunas fichas, como la de Ideas o Subtramas, aparecen unas listas mostrando todos los personajes, objetos, localizaciones y razas que se han creado en el libro. La finalidad de esto, es que el usuario pueda relacionarlos con la idea o la subtrama. Por ejemplo, el usuario está escribiendo un libro en el que se le ha ocurrido que tal vez podría incluir una fuerte discusión entre el personaje A y el B. Como no está seguro de si finalmente la incluirá en el libro la guarda como idea y en su ficha, aparte de desarrollarla, selecciona los personajes A y B. De esta manera la próxima vez que entre a ver esta ficha verá con rapidez que aquella es la idea en la que A y B discutían.

La aplicación también incluye la posibilidad de añadir imágenes o bocetos en algunas fichas (Personajes, Objetos, Localizaciones, Razas) con el fin de que pueda usarlas a modo de inspiración.

Al tratarse de una aplicación de escritorio, decidí dejar de lado el tema del acoplamiento entre esta capa y la capa Dominio. El motivo fue pensar que este “problema” era más propio de aplicaciones Web, las cuales suelen necesitar más actualizaciones de aspecto para no parecer anticuadas. Otro motivo fue evitar tantas consultas a la base de datos, ya que al tener los objetos directamente en el árbol, evitabas tener que consultar a la BBDD sobre que objeto estabas trabajando.

Así pues, cuando vemos en la aplicación el Árbol o las listas, lo que estamos viendo es el propio objeto, no un String representativo de él.

3.1.5. Capa aplicación.

La capa aplicación es la encargada de la comunicación entre la capa presentación y la capa Dominio. En esta capa, el controlador recibe los datos necesarios para construir los objetos o los propios objetos, como también las peticiones a las que dará respuesta.



Fig. 4 Capa aplicación

También se incluye el fichero de conexión a base de datos, para que el controlador pase la conexión y sea él el encargado de decidir cuándo realizar el commit en las transacciones. Ejemplo de esto sería el momento de añadir fotos a un personaje ya creado. El controlador llama al método `iRegistroPersonajes.addFoto(...)` para añadirla, si todo va correcto, posteriormente llama a `iRegistroPersonajes.updatePersonaje(...)` que en el caso de que devuelva algún error será el propio controlador el encargado de realizar el rollback.

Cabe destacar que en nuestro controlador tenemos siempre el libro sobre el que el usuario está trabajando lo que agiliza el rendimiento de la aplicación.

3.1.6. Capa persistencia.

La capa persistencia es la encargada de atacar la base de datos. Sus clases reciben los objetos con los que tendrá que comunicarse con la BBDD, ya sea para grabar el objeto en la base de datos, como para utilizarlos para la realización de búsquedas.

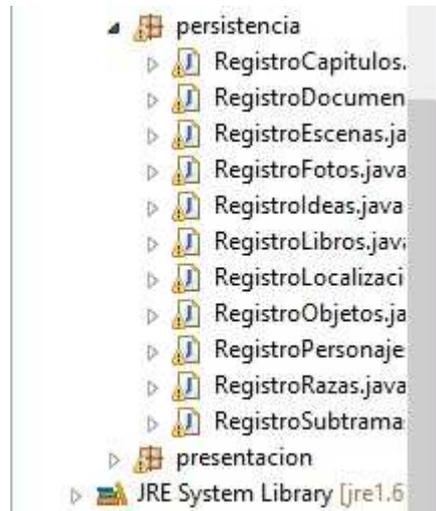


Fig. 5 Capa persistencia

A continuación se muestra un ejemplo de método de la capa persistencia. Concretamente es el encargado de recuperar todos los personajes de un libro. Estos serán guardados en un ArrayList de Personajes.

```
1 public ArrayList getPersonajesByLi(Libro libro, Categoria categoria) throws Exception {
2     ArrayList<Personaje> lista = new ArrayList<Personaje>();
3     RegistroLocalizaciones La = new RegistroLocalizaciones();
4     RegistroRazas Ra = new RegistroRazas();
5     try {
6         Connection conn = ConexionDB.GetConnection();
7         String query = " select * from personaje where libro_id = ? and categoria = ? order by nombre";
8         PreparedStatement preparedStmt = conn.prepareStatement(query);
9         preparedStmt.setInt(1, libro.getId());
10        preparedStmt.setString(2, categoria.getNombreCategoria());
11        ResultSet result = preparedStmt.executeQuery();
12        while (result.next()) {
13            int id = result.getInt("id");
14            int libro_id = result.getInt("libro_id");
15            String nombre = result.getString("nombre");
16            String apodo = result.getString("apodo");
17            Categoria categoria2 = Categoria.valueOf(result
18                .getString("categoria"));
19            String desc_facial = result.getString("desc_facial");
20            String desc_fisica = result.getString("desc_fisica");
21            String desc_psicologica = result.getString("desc_psicologica");
22            String historia = result.getString("historia");
23            String notas = result.getString("notas");
24            Localizacion localizacion = La.getLocalizacionById(result.getInt("localizacion_id"), conn);
25            Raza raza = Ra.getRazaById(result.getInt("raza_id"), conn);
26            localizacion.setLibro(libro);
27            raza.setLibro(libro);
28
29            Personaje personaje = new Personaje(id, libro, nombre, apodo,
30                categoria, desc_facial, desc_fisica, desc_psicologica,
31                historia, notas, localizacion, raza);
32            lista.add(personaje);
33        }
34        conn.close();
35    } catch (Exception e) {
36        System.err.println("Error al devolver los personajes del libro " + libro.getId());
37        System.err.println(e.getMessage());
38    }
39    return lista;
40 }
```

Fig. 6 Método getPersonajesByLi

3.2. Realización de pruebas

No ha habido una fase específica de pruebas, ya que la manera de trabajar ha sido la de centrarse en un caso de uso y desarrollarlo hasta que estuviese finalizado. Esto hacía que constantemente se realizaran pruebas sobre ese caso hasta haberlo finalizado. Una vez acabado, si se realizaba una prueba final para comprobar su correcto funcionamiento.

Al finalizar la aplicación, sí se realizó una prueba exhaustiva de ella donde se comprobó:

- La implementación de todos los casos de uso.
- Su correcto funcionamiento.
- Que la aplicación fuera coherente en el aspecto visual.

Esta manera de trabajar hizo que aunque se avanzara con más lentitud a la hora de programar, se recuperara posteriormente el tiempo, ya que al acabar un caso de uso, prácticamente podías darlo por totalmente finalizado. Además, personalmente creo que anímicamente es mejor para el programador, pues va viendo resultados y evita enfrentarse a unas pruebas que pueden conllevar a descubrir que tienes un código lleno de errores.

También el hecho de que muchos casos de uso sean parecidos, hace que al finalizar uno totalmente todo lo que has aprendido en él te sirva para avanzar con mayor velocidad en el siguiente.

Con todo y con eso, vamos a mostrar en este documento la prueba realizada en la creación y modificación de un personaje.

3.2.1. Pruebas del caso de uso Creación y modificación de un personaje

Como se puede ver en la Fig. 6, partimos de la base de un nuevo libro en el que no existe aún ningún elemento. Nuestro objetivo es crear un nuevo personaje y posteriormente modificarlo.



Fig. 7 Captura del árbol vacío

Lo primero que hacemos es abrir la pantalla Crear Personaje a partir de la carpeta Personajes (Debemos posicionarnos encima y clicar). A continuación aparecerá la pantalla de creación del personaje.

Crear Personaje

Apellido Categoría **Principal** Localización Raza

Nombre

Historia

Descripción facial

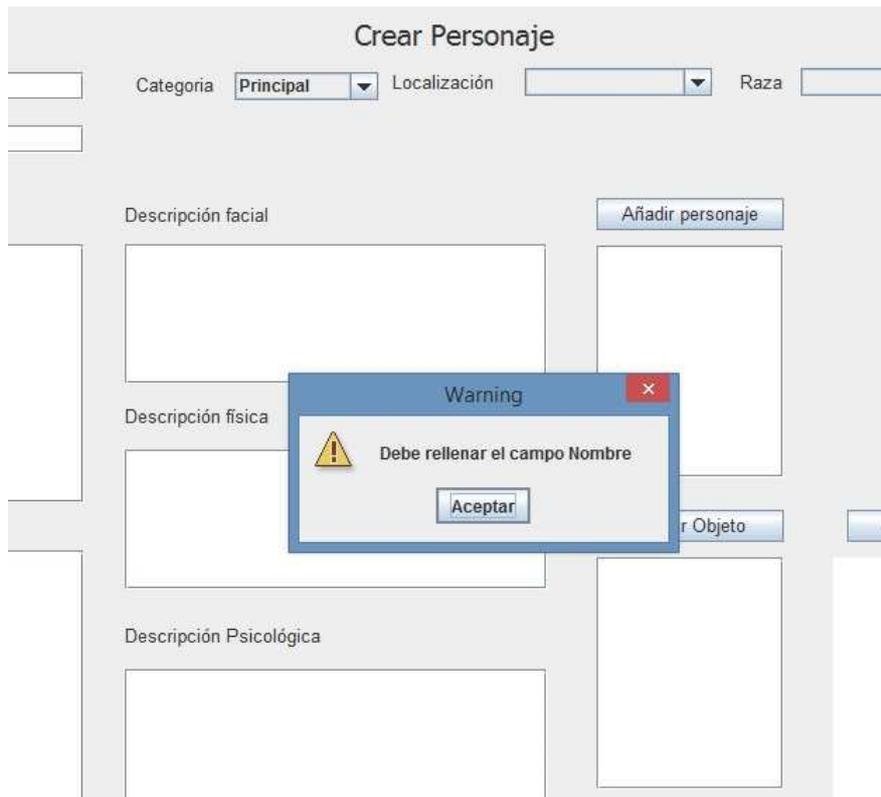
Descripción física

Notas

Descripción Psicológica

Fig. 8 Captura de pantalla Crear Personaje

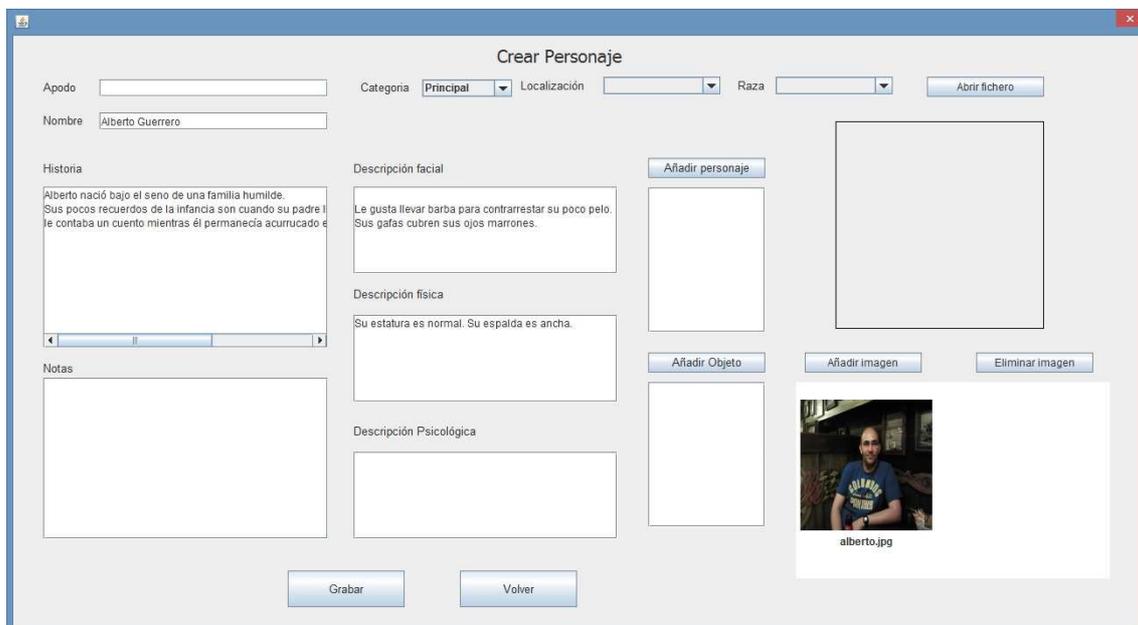
En esta pantalla hay dos campos que son obligatorios. La categoría nos indica si el personaje es Principal o Secundario y aunque es obligatorio al tratarse de un JComboBox el usuario no podrá dejarlo en blanco. El campo nombre también es obligatorio. Si el usuario intenta grabar el personaje sin rellenarlo le aparecerá un aviso en pantalla.



The screenshot shows the 'Crear Personaje' (Create Character) form. At the top, there are dropdown menus for 'Categoria' (set to 'Principal'), 'Localización', and 'Raza'. Below these are three large text input areas: 'Descripción facial', 'Descripción física', and 'Descripción Psicológica'. A 'Añadir personaje' button is visible. A blue warning dialog box is overlaid on the form, containing a yellow warning icon and the text 'Debe rellenar el campo Nombre' (Name field must be filled). An 'Aceptar' button is at the bottom of the dialog.

Fig. 9 Captura de pantalla Crear Personaje (aviso)

Pero en cambio, si después de rellenar los datos convenientes, todo va bien, aparecerá otro aviso indicándole que el personaje se ha creado correctamente.



The screenshot shows the 'Crear Personaje' form with the following filled data: 'Apodo' (empty), 'Nombre' (Alberto Guerrero), 'Historia' (Alberto nació bajo el seno de una familia humilde. Sus pocos recuerdos de la infancia son cuando su padre le contaba un cuento mientras él permanecía acurrucado e...), 'Descripción facial' (Le gusta llevar barba para contrarrestar su poco pelo. Sus gafas cubren sus ojos marrones.), 'Descripción física' (Su estatura es normal. Su espalda es ancha.), and 'Descripción Psicológica' (empty). The 'Añadir personaje' button is visible. A character image is displayed in the bottom right corner, labeled 'alberto.jpg'. Other buttons include 'Abrir fichero', 'Añadir imagen', 'Eliminar imagen', 'Grabar', and 'Volver'.

Fig. 10 Captura de pantalla Crear Personaje (rellena)

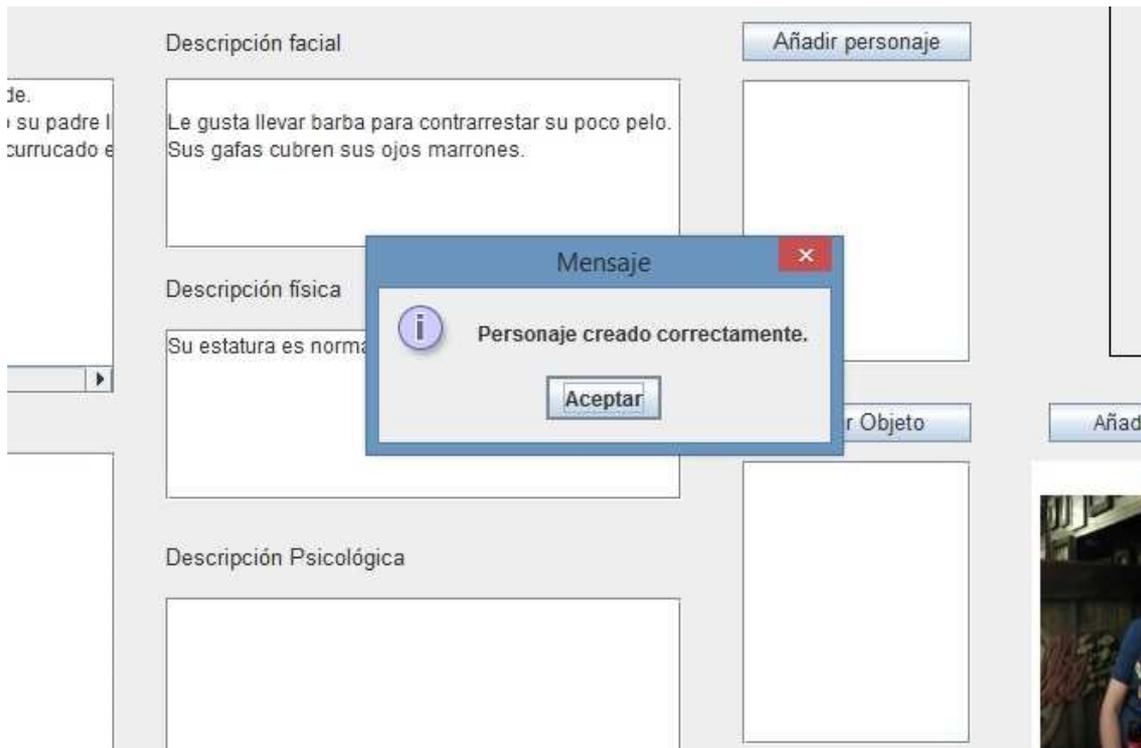


Fig. 11 Captura de pantalla Crear Personaje (creado)

A continuación, el nuevo personaje creado aparecerá dentro de la subcarpeta Principales en la carpeta Personaje. Tal como se puede apreciar en la Fig.11.



Fig. 12 Captura del árbol lleno

A continuación podemos proceder a modificar el personaje, posicionándonos sobre él en el árbol y clicando. Se abrirá la pantalla Modificar Personaje sobre el que podremos realizar las operaciones oportunas. En este ejemplo vamos a crear antes una localización y un objeto, para relacionarlos con el personaje.



The screenshot shows a web form titled "Modificar Personaje". At the top, there are three dropdown menus: "Categoría" with "Principal" selected, "Localización" with "Barcelona" selected and its dropdown menu open showing "Barcelona" as the only option, and "Raza" which is empty. Below these is a blue "Eliminar" button. The "Descripción facial" section contains a text area with the text "Le gusta llevar barba para contrarrestar su poco pelo. Sus gafas cubren sus ojos marrones." and a blue "Añadir personaje" button. The "Descripción física" section is partially visible at the bottom.

Fig. 13 Captura de pantalla Modificar Personaje (Localización)



The screenshot shows a section of the "Modificar Personaje" form. On the left, a text area contains the text "palda es ancha.". In the center, there is a blue "Añadir Objeto" button above a text area containing the word "Ford". On the right, there is a blue "Añadir imagen" button above a small image of a man with glasses and a beard wearing a blue t-shirt.

Fig. 14 Captura de pantalla Modificar Personaje (Objeto)

En el caso de añadir relaciones con objetos, lo mismo que con personajes, se abre una nueva pantalla en la que el usuario deberá rellenar un pequeño formulario.

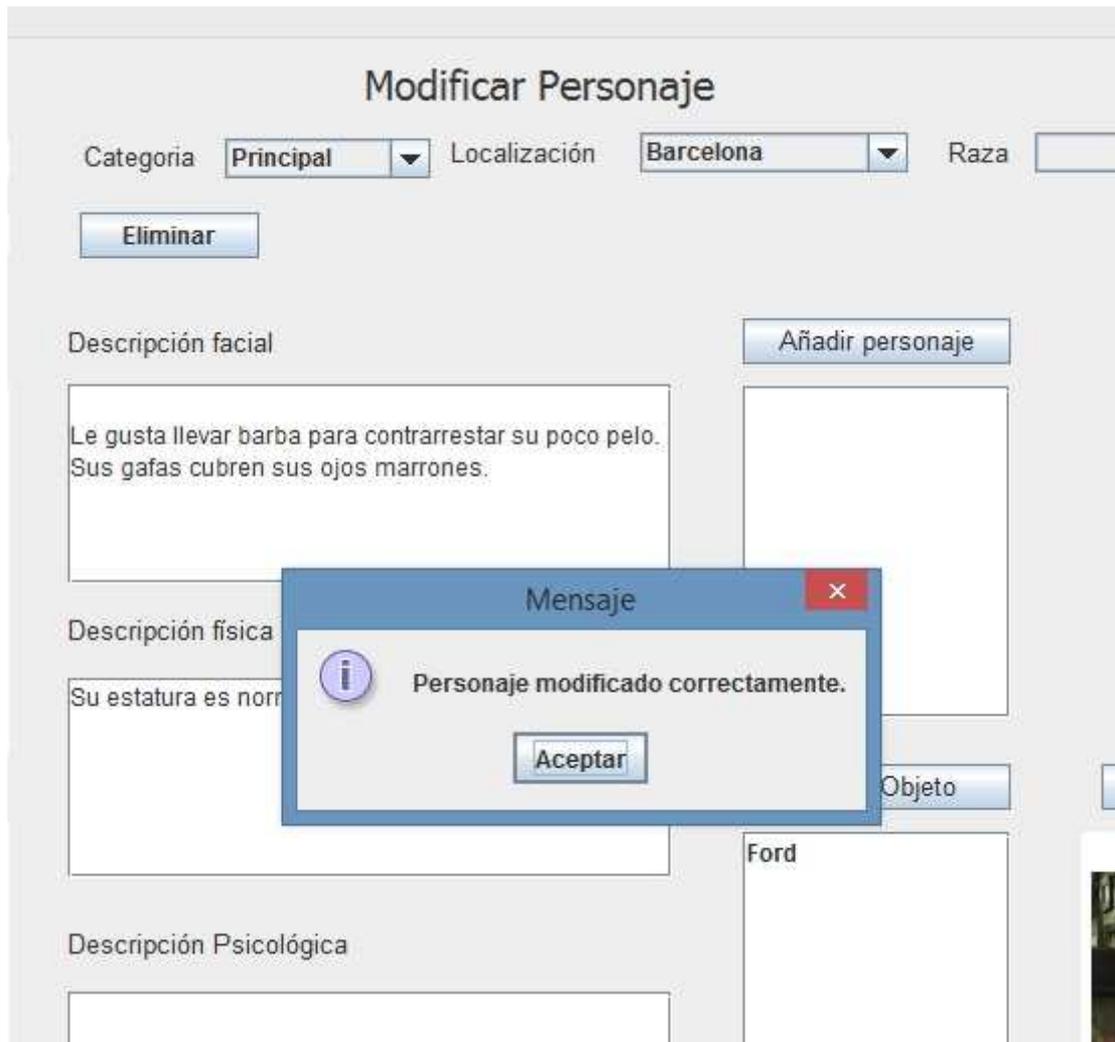


Fig. 15 Captura de pantalla Modificar Personaje (Modificado)

3.3. Documentación.

Aunque el desarrollo de la memoria se ha ido realizando a lo largo del proyecto, es cierto que los momentos en los que más se trabajó en ella fueron al principio y al final.

En un primer momento se dedicó tiempo en el desarrollo de los casos de uso, los cuales quedaron documentados aunque han acabado sufriendo alguna modificación. Es estas últimas semanas también se ha dedicado más tiempo, como por otra parte es lógico, al desarrollo de la memoria.

4. Estudio económico

4.1. Costes de recursos humanos

A continuación se detallan los gastos económicos que supondrían realizar este proyecto para el mercado.

El precio que se ha estipulado por persona varía de la fase de análisis a la fase de desarrollo.

<u>Concepto</u>	<u>Horas</u>	<u>Precio/hora €</u>	<u>Total (€)</u>
Especificaciones y análisis (Ingeniero sénior)	30	25	750
Desarrollo (Ingeniero júnior)	300	12	3600
Correcciones y pruebas (Ingeniero júnior)	150	12	1800
Redacción memoria	40	12	480
TOTAL RECURSOS HUMANOS			6630

Tabla 4.3 Coste de recursos humanos

4.2. Amortización de equipos y software

Se considera que las herramientas serán utilizadas en los próximos proyectos, por lo que únicamente se tiene que calcular la parte proporcional a este. El ordenador y Microsoft Office se renovarían cada año y han sido utilizados durante 520 horas.

El resto de herramientas utilizadas no han supuesto ningún coste.

Equipo y herramientas	Precio	Amortización
Ordenador	700	41,55
Microsoft Office	119	7,06
Total	819	48,61

Tabla 4.4 Amortización de equipos y software

4.3. Implementación del prototipo

Los gastos indirectos propios de la luz, las comunicaciones, el mobiliario, etc. se consideran que serán un 15% del total del proyecto.

Costes de recursos humanos	6630 €
Costes de amortización	48,61 €
Subtotal	6678,61 €
Gastos indirectos (15%)	1001,79 €
TOTAL	7680,40 €

Tabla 4.5 Implementación del prototipo

Así pues, el coste total de la aplicación ascendería a **7680,40 €**

5. Conclusiones

Hay pocas cosas de las que suela arrepentirme, pero una de ellas ha sido no haber realizado este PFC cuando debía haberlo hecho. Entonces, el incorporarme a la vida laboral hizo que lo dejará de lado. Siempre pensaba que en cuanto me estableciera un poco, me pondría con él, que en el siguiente semestre sí que lo haría. Pero la realidad era que el día a día iba absorbiéndome. Y finalmente, paso algo que no debería haber pasado. Deje el proyecto estancado.

Pero la idea de realizarlo no apareció hasta hace poco, cuando visto que mi primer hijo estaba en camino, creí que tenía que ser ahora o nunca. Sin embargo, con lo que no contaba era con lo complicado que me sería compaginar la vida laboral, con la de padre novato, con la de estudiante.

En ese aspecto, estoy orgulloso de poder haber finalizado este PFC.

Sé que hay fallos en este proyecto, pero creo que todos ellos se deben a una mala planificación. Una planificación que se derrumbaba cada vez que se escuchaba un llanto. Soy consciente de que si esta hubiera sido mejor, mejor hubiera sido el resultado. Pero sin embargo, estoy contento y más aún cuando veo las posibilidades de evolucionar que tiene la aplicación. Creo que la idea es original y tiene potencial, y en ese aspecto estoy orgulloso de Writer's Help.

