

Dpto. de Electrónica e Informática
Salesianos Atocha



THE FUTURE OF READING

Fernando García García

Sergio Garrido Escobar

Ana Guillem de la Fuente

Tutor: Carlos Contreras

ÍNDICE

1. Introducción

- ☐ Objetivo
- ☐ Justificación
- ☐ Análisis de lo existente
- ☐ Propuesta detallada

2. Requisitos y manuales de usuario

- ☐ Modelo Entidad/relación
- ☐ Base de datos

3. Diseño

- ☐ Herramientas utilizadas
- ☐ Arquitectura del sitio web
 - Diseño específico por usuario de la página principal
 - Anónimo
 - Usuario
 - Colaborador
 - Administrador

4. Codificación

- ☐ Entornos de programación y herramientas de desarrollo
- ☐ Lenguajes usados
- ☐ Aspectos relevantes de la implementación
 - Validación de datos
 - Control de acceso
 - Encriptación de la información
 - Sistema de carpetas
 - Menú desplegable

5. Configuración del servidor

- ☐ Hardware
- ☐ Seguridad Apache
- ☐ Puesta en funcionamiento en internet mediante NO - IP

6. Pruebas de acceso, creación de carpetas y seguridad

7. Aspectos a mejorar

8. Conclusiones

9. Anexo: Plan de empresa

1. Introducción

1.1 Objetivo

En **eYe ay ay S.L.** tenemos como objetivo principal que la lectura llegue a todo el mundo, de la forma más económica y cómoda posible a través de un portal online. Nuestra idea era crear una aplicación web de venta de libros diferente al resto, centrándonos en un cuidado diseño y en una usabilidad adaptada al usuario con el fin de que la curva de aprendizaje sea alta a la hora de usar nuestra plataforma web, es decir, conseguir que el usuario final se mueva por nuestra web de forma ágil e intuitiva en unos pocos minutos.

1.2 Justificación

Elegimos este proyecto porque reúne en una sola idea la mayoría de conocimientos aprendidos durante el ciclo. Desde nuestro punto de vista, una página web bien desarrollada exige dominar múltiples lenguajes como **HTML, PHP, Javascript, CSS, MySQL e incluso manejo de ficheros en Windows** que, como ya aprendimos a lo largo del curso, se interrelacionan entre sí.

Por otro lado queríamos que fuese un reto para nosotros y que nos ayudase a mejorar nuestros conocimientos y la forma de enfrentarnos a los problemas que fuesen surgiendo. Esto no hubiese tenido mucha importancia si hubiésemos usado herramientas como **Wordpress, Microsoft Frontpage, Google Web Designer o PHPmyAdmin** que facilitan en exceso la tarea de planificación, desarrollo y puesta en funcionamiento de un portal web, por lo que nos propusimos realizar todo el proyecto sólo con **Notepad++** y la consola de **MySQL**.

1.3 Análisis de lo existente

En la actualidad existen varios portales de ventas de libros, un sector dominado por las tres empresas más importantes del sector tecnológico (**Google, Apple y Amazon**). Los tres portales ofrecen muchas posibilidades en cuanto a compras, packs, libros físicos, disponibilidad, etc y disponen de una base de datos de libros enorme, por lo que en ese aspecto es inviable competir contra ellos. Sin embargo, también tienen ciertos puntos negativos.

● Google Books:

Probablemente la más potente de las tres. Ofrece libros electrónicos y cuenta con un diseño muy cuidado y minimalista sin profundizar en

multitud de opciones que podrían confundir al usuario final. Permite leer sus libros tanto con su aplicación nativa como en línea, siendo independiente de cualquier hardware o software específico. Además acaba de implementar el sistema de alquiler de libros.

● Amazon-Kindle:

Con un diseño más clásico, ofrece tanto libros físicos como electrónicos y cuenta con una interfaz algo más compleja y elaborada. Para poder leer los eBooks exige tener ya sea la aplicación de Amazon-Kindle instalada o un Kindle eBook Reader. También acaba de implementar el alquiler de libros pero en este caso sólo para libros físicos.

● iTunes-iBooks:

Al igual que Google, cuenta con un diseño cuidado siguiendo las pautas de diseño de Jonathan Ive. Se basa también en eBooks y es probablemente el sistema más cerrado de los tres ya que sólo se puede usar con software de Apple, ya sea iTunes o iBooks.



1.4 Propuesta detallada

El portal web **EyE ay ay** consiste en una página de distribución y lectura de **eBooks** que engloba gestión de usuarios, permisos, ficheros y archivos tanto de texto como imágenes asentada sobre una base de datos en **MySQL**. Una de las ideas fundamentales a la hora de iniciar este proyecto es que *“Si algo parece bueno probablemente lo sea”*, por lo tanto el aspecto visual es una de las partes más importantes del mismo, conteniendo una alta cantidad de **código CSS** y tomando como referencia el **Material Design** de **Google**.

Nuestro segundo objetivo es ofrecer al usuario la posibilidad tanto de **alquilar** como **comprar** los libros y permitir leer los libros tanto **online como offline** (sólo compra) sin necesidad de depender de ninguna aplicación extra o propietaria más allá del navegador y un lector de PDFs. En cuanto al **alquiler de libros**, pensamos que hay mucha gente que consume literatura rápidamente y a la larga les resulta muy caro comprar una gran cantidad de libros que luego sólo les servirán como decoración o para ocupar disco. Por esto creemos que es una buena alternativa ofrecer este servicio a este tipo de usuario.

Otros puntos a destacar son el sistema de mensajería entre usuarios y de avisos relacionados con los libros.

2. Requisitos y manual de usuario

Para crear un portal web dinámico el requisito fundamental es una base de datos en la que apoyarse, en los siguientes puntos explicaremos su estructura general y su contenido en detalle.

2.1 Modelo Entidad/relación

El primer paso a la hora de estructurar la base de datos es el modelo **entidad relación**. Tras un estudio pormenorizado de lo que queríamos ofrecer, llegamos a la conclusión de que nuestra base de datos debería tener las siguientes entidades o tablas:

Usuario → alberga datos necesarios de cada usuario dado de alta en la página.

Libro → contiene datos necesarios de cada libro subido a la página.

Mensajes → guarda datos básicos de los mensajes (como remitente o destinatario).

Autor → lleva un control de los autores permitidos a la hora de subir libros (Si no existe el autor no se puede insertar un libro con ese autor en la tabla libros).

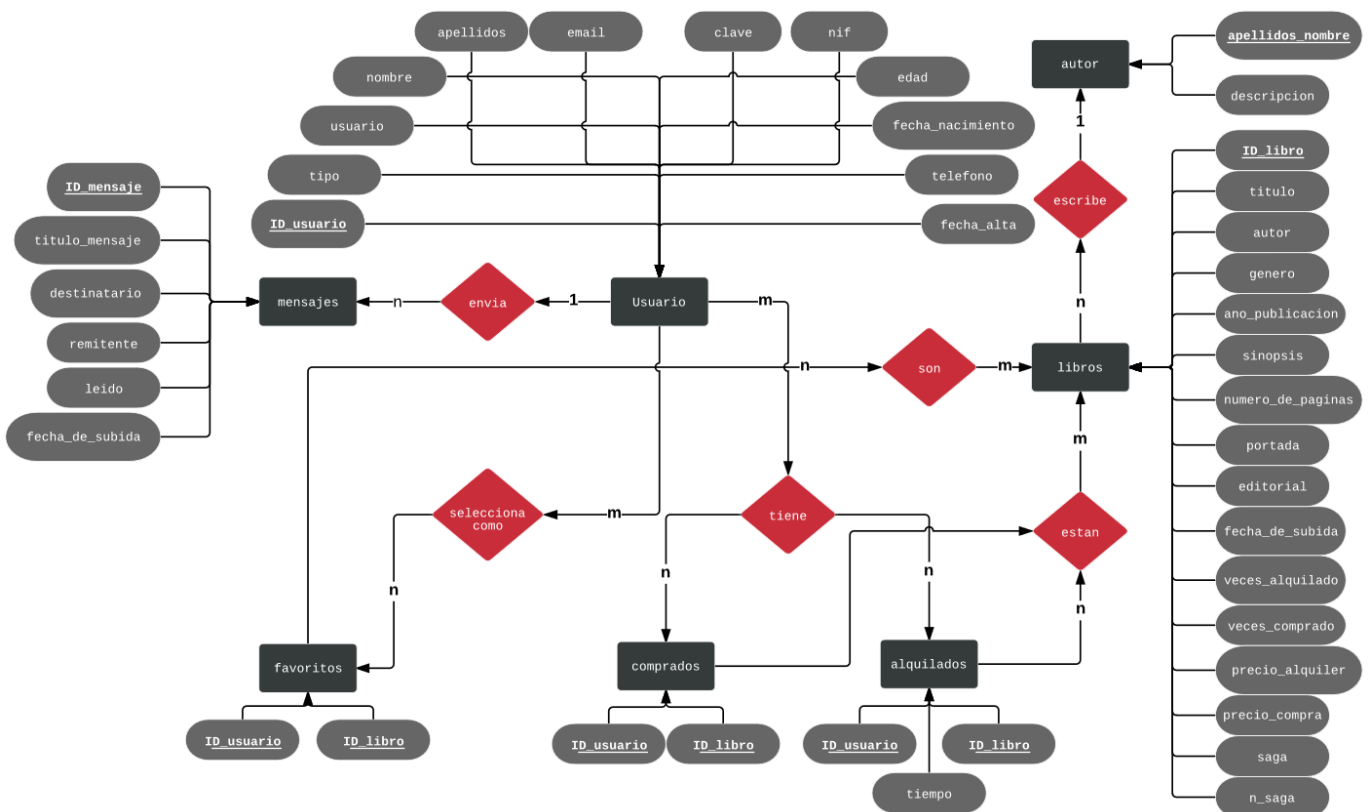
Alquilados → establece qué libro ha sido alquilado y que usuario ha hecho el mismo, también se controlará el tiempo de alquiler mediante esta tabla.

Comprados → guarda información sobre los libros comprados y el usuario que lo ha realizado.

Favoritos → Creada para que el usuario establezca y pueda visualizar una lista de libros favoritos seleccionados por él mismo.

Una vez establecidos nuestras entidades procedimos a otorgarles atributos dependiendo de las necesidades de nuestro proyecto, así como una clave primaria al menos dependiendo del caso.

El esquema entidad-relación quedará de la siguiente manera:



2.2 Base de datos

Tras el análisis del esquema de entidad-relación, procedemos a crear la base de datos en la que sustentaremos nuestra página web. En primer lugar creamos la base de datos a la que llamaremos **Proyecto** y después las diferentes tablas o entidades del esquema entidad-relación:

Tabla Usuario: (Datos con todos los usuarios de la web)

create table usuario (

ID_usuario int primary key auto_increment, → Ponemos un número identificador del usuario el cual será autoincremental, además será clave primaria ya que este campo debe ser único.

tipo varchar(20) not null, → Campo que contiene el tipo de usuario que será (Administrador, usuario, y anónimo) , no puede ser nulo.

usuario varchar(50) not null, → Alias del usuario, no puede ser nulo.

nombre varchar(50) not null, → Nombre del usuario, no puede ser nulo.

apellidos varchar(50) not null, → apellidos del usuario, no puede ser nulo.

email varchar(50) not null, → Email del usuario, no puede ser nulo.

clave varchar(30) not null, → Password del usuario, no puede ser nulo.

nif varchar(9) not null, → Dni del usuario, no puede ser nulo.

edad int not null, → Edad del usuario este campo se autorellena mediante cálculo de la fecha de nacimiento y la fecha actual.

Fecha_nacimiento varchar(10) not null, → Fecha de nacimiento del usuario, no puede ser nulo, se rellena mediante un input type = 'date', por eso le pongo varchar y no date ya que el formato con el que me devuelve un input type = 'date' no es el mismo que con date de mysql, por lo que para evitar problemas le pongo varchar.

telefono varchar(9) default "no", → Teléfono móvil del usuario este campo es opcional si no se rellena se completará automáticamente con "no".

Fecha_alta varchar(10)); → Fecha de alta del usuario, se rellena obteniendo la fecha mediante la función date de php.

Tabla Libros: (Datos con todos los libros de la web)

create table libros (

ID_libro int primary key auto_increment, → Ponemos un número identificador del libro el cual será autoincremental, además será clave primaria ya que este campo debe ser único.

titulo varchar(250) not null, → Campo que contiene el título del libro que ha de introducir, no puede ser nulo.

autor varchar(50) not null, → Campo que contiene el autor del libro que ha de introducir, no puede ser nulo.

genero varchar(20) not null, → Campo que contiene el género del libro que ha de introducir, no puede ser nulo.

ano_publicacion varchar(10) not null, → Campo que contiene el año de publicación del libro, no puede ser nulo.

sinopsis varchar(30000) not null, → Campo que contiene un pequeño resumen del libro a introducir, no puede ser nulo.

numero_de_paginas int not null, → Campo que contiene el número de páginas del libro, no puede ser nulo.

portada varchar(50) not null, → Campo que contiene la ruta de ubicación de la portada del libro, no puede ser nulo.

editorial varchar (40) not null, → Campo que contiene la editorial del libro a introducir, no puede ser nulo.

fecha_de_subida varchar(19) not null, → Campo que contiene la fecha de subida del libro a introducir, no puede ser nulo (Este es un campo que se rellenará automáticamente con la función now()).

veces_alquilado int default 0, → Campo que contiene el número de veces que un libro ha sido alquilado por cualquier usuario, por defecto cuando se introduce un nuevo libro en la base de datos este valor se pondrá a cero cada vez que se alquile un libro se sumará uno a este valor.

veces_comprado int default 0, → Campo que contiene el número de veces que un libro ha sido comprado por cualquier usuario, por defecto cuando se introduce un nuevo libro en la base de datos este valor se pondrá a cero cada vez que se compre un libro se sumará uno a este valor.

precio_alquiler int, → Campo que contiene el precio de alquiler del libro introducido.

precio_compra int, → Campo que contiene el precio de compra del libro introducido.

saga varchar(30) default "", → Campo que contiene la saga del libro si pertenece a una, por defecto este campo se rellena con un espacio en blanco, este valor será opcional de rellenar a la hora de introducir un nuevo libro en la página.

n_saga int); → Campo que contiene la el número de la saga introducida en el campo anterior, por defecto este campo se rellena con un espacio en blanco, este campo como el anterior son opcionales por lo que no se comprobará si se

han rellenado, solo se comprueba si rellenas uno, así no se podrá rellenar uno y dejar el otro en blanco.

Estas dos tablas serán nuestras dos tablas principales del proyecto ya que contendrán la información más importante para el mismo, después hemos creado varias tablas más de apoyo que hacen referencia a estas dos tablas anteriores:

Tabla Mensajes: (Datos de los mensajes enviados)

create table mensajes (

ID_mensaje int primary key auto_increment, → Ponemos un número identificador del mensaje el cual será autoincremental, además será clave primaria ya que este campo debe ser único.

titulo varchar(50) not null, → Campo que contiene el título del mensaje que ha escribir, no puede ser nulo.

destinatario varchar(50) not null, → Campo que contiene el destinatario (usuario destino) del mensaje a enviar, no puede ser nulo.

remitente varchar(50) not null references usuario(usuario), → Campo que contiene el remitente (usuario que envía el mensaje) del mensaje a enviar, no puede ser nulo, hace referencia al campo usuario de la tabla usuario.

fecha_de_subida varchar(19) not null); → Campo que contiene la fecha de envío del mensaje, no puede ser nulo. Esta fecha es un varchar de 19 debido a que es una fecha que contiene la fecha actual de envío de mensaje y también la hora, lo hemos hecho así para cuando el usuario vea todos los mensajes le vengan ordenado por fecha y hora.

Tabla Autor: (Datos de los autores que hace referencia a los libros)

create table autor (

apellidos_nombre varchar(50) references libros(autor),

descripcion varchar(64000) not null, → amplia descripción del autor, no puede estar vacío.

primary key (apellidos_nombre)); → Campo que contiene el apellido y el nombre del autor que se desea habilitar tiene referencia con la tabla libros el campo autor.

Tabla Alquilados: (Datos con los usuarios y los libros que tienen alquilados)

create table alquilados (

ID_usuario int references usuario(ID_usuario), → Referencia a la tabla usuario, cogiendo la id_usuario.

ID_libro int references libros(ID_libro), → Referencia a la tabla usuario, cogiendo la id_libro.

tiempo varchar(19), → Número que contiene el tiempo de alquiler restante para ese libro y ese usuario, es un varchar y no un int debido a que este tiempo contiene la fecha y la hora para hacer la comprobación exacta de cuando se alquiló el libro y así poder deshabilitarlo cuando llegue el momento.

primary key(ID_usuario,ID_libro)); → Establecemos la clave primaria tanto en id_usuario como en id_libro.

Tabla Comprados: (Datos con los usuarios y los libros que tienen comprados)

create table comprados (

ID_usuario int references usuario(ID_usuario), → Referencia a la tabla usuario, cogiendo la id_usuario.

ID_libro int references libros(ID_libro), → Referencia a la tabla usuario, cogiendo la id_libro.

primary key(ID_usuario,ID_libro)); → Establecemos la clave primaria tanto en id_usuario como en id_libro.

Tabla Favoritos: (Datos con los usuarios y los libros que han marcado los mismos como favoritos)

create table favoritos (

ID_usuario int references usuario(ID_usuario),), → Referencia a la tabla usuario, cogiendo la id_usuario.

ID_libro int references libros(ID_libro), → Referencia a la tabla usuario, cogiendo la id_libro.

primary key(ID_usuario,ID_libro)); → Establecemos la clave primaria tanto en id_usuario como en id_libro.

3. Diseño

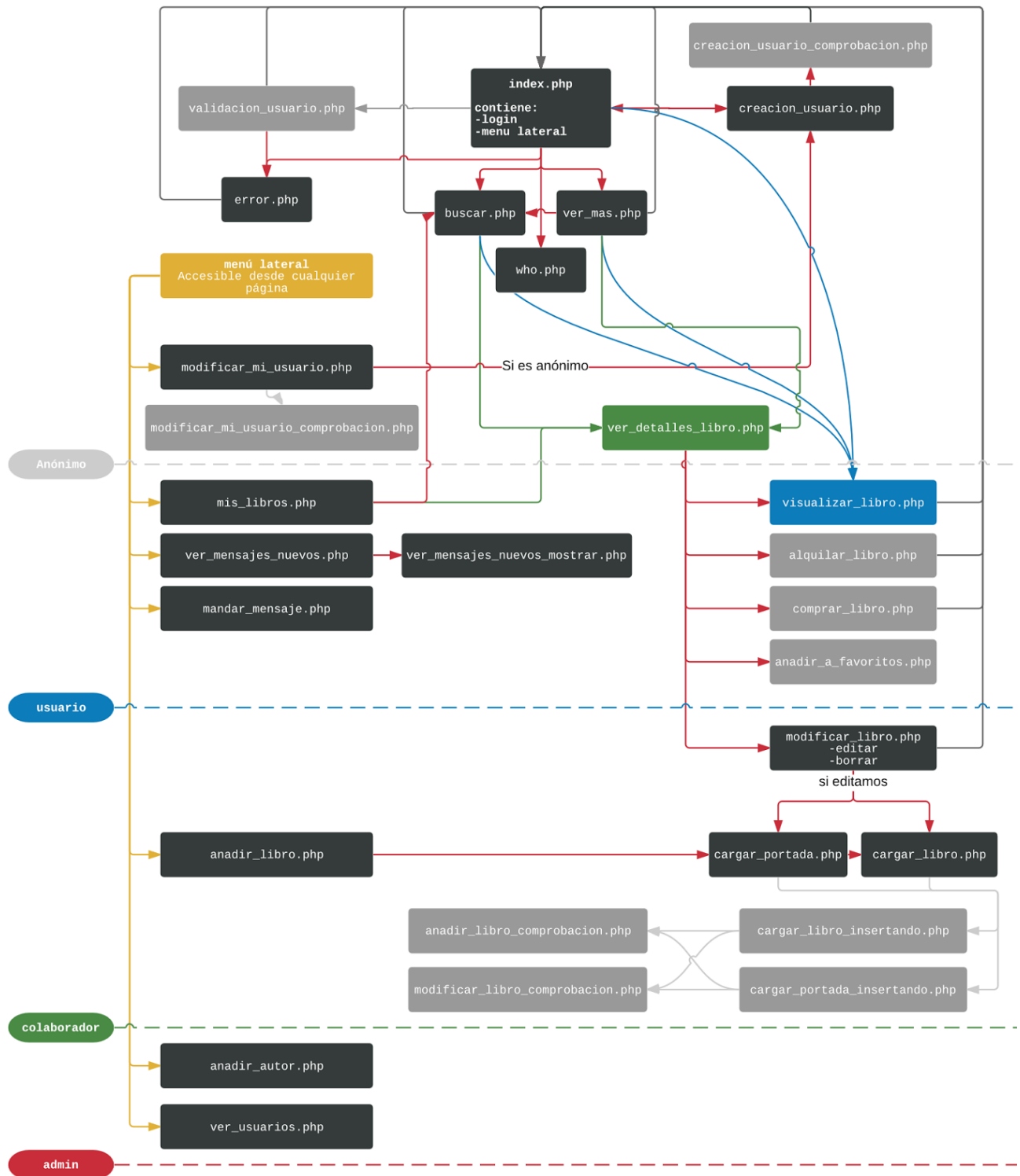
3.1 Herramientas utilizadas

Como hemos mencionado en la propuesta detallada, uno de nuestros objetivos principales a la hora de crear nuestro portal web es usar un **diseño minimalista, cuidado, fluido y sencillo** para el usuario final, para ello hemos usado las siguientes herramientas:



- Como fuente de inspiración para el **CSS** de nuestra página hemos usado el portal **Codrops**, una web con una gran cantidad de tutoriales, plantillas y demos de uso libre.
- Para el diseño previo, la maquetación de la página web, la creación de logos, banners y adaptación de imágenes y vectores hemos usado **Photoshop CS6**.
- Por último hemos usado la aplicación **Lucidchart** para el diseño previo y arquitectura de la página web.

3.2 Arquitectura del sitio web

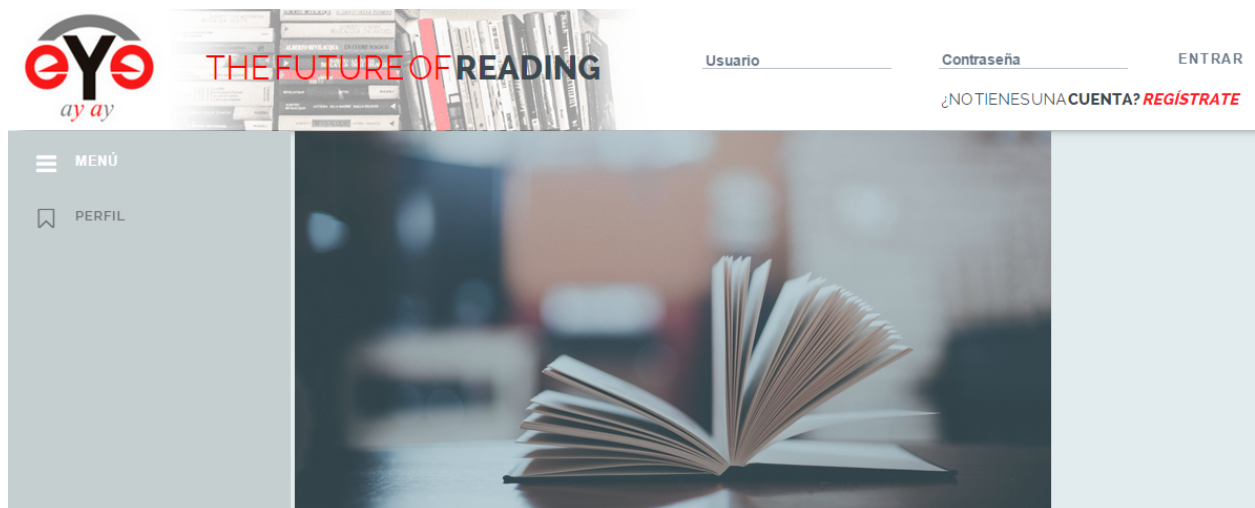


3.3 Diseño específico por usuario

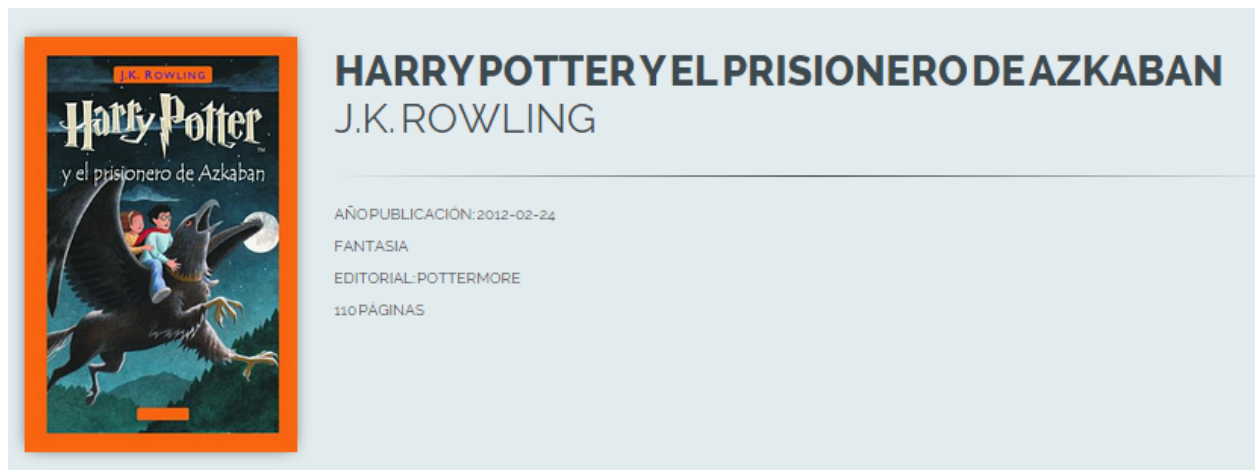
Como hemos observado en el esquema del diseño web, cada usuario tiene acceso hasta una profundidad, en el que entra el usuario siguiente hasta llegar al administrador.

Anónimo:

Este usuario tendrá acceso al Index, donde podrá logearse o registrarse, además de tener acceso a los libros, el buscador, ver más libros de una categoría y el menú lateral que le redireccionará al registro.



Así mismo, dentro de los libros no podrá realizar acción alguna más allá de poder leer la información del mismo:





THE FUTURE OF READING

BIENVENIDO **SERGIO**
TIPO DE USUARIO: USUARIO

[SALIR](#)

≡ MENÚ



NOVEDADES



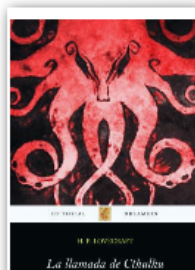
Harry Potter y la Orden del Fénix
J.K. Rowling

[Leer](#)



Harry Potter y el prisionero de Azkaban
J.K. Rowling

[Leer](#)



La llamada de Cthulhu
H.P. Lovecraft

Alquiler 0 €
Compra 0 €



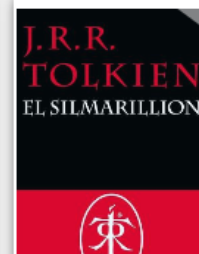
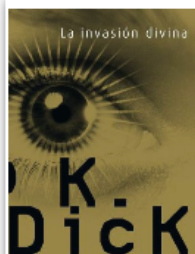
Visiones de robot
Isaac Asimov

[Leer](#)



El hombre bicentenario
Isaac Asimov

Alquiler 0 €
Compra 2 €



Usuario:

Además de todo el acceso del usuario anónimo, podrá acceder a su panel de control, sus mensajes, enviar mensaje, ver sus libros y leerlos. Como detalle, en el buscador aparecen los primeros de varios libros al azar en vez del saludo que aparece al usuario anónimo.



En su perfil podrá editar toda su información:

[Volver a la pagina de inicio](#)

Micuenta

Introduzca sus datos (* Campo no obligatorio *)

Campo	Valor actual	Nuevo valor
Nombre de usuario	sergio	
Nombre	sergio	
Apellidos	Garrido Escobar	
E-mail	segarrido85@gmail.com	
Repita E-mail		
Contraseña		
Repita Contraseña		
N.I.F	50881340-G	
Fecha de nacimiento	1985-07-19	
* Teléfono móvil *	no	

ENVIAR

BORRAR

También puede ver sus mensajes y enviar nuevos:

x	Fecha de envío	Titulo	Enviado por Usuario	Leido	leer
☐	2015/05/31-22:45:06	Prueba de mensaje	spelean	no	Leer
☐	2015/05/31-19:25:22	Bienvenido a EYE.COM	La Dirección.	si	Leer
BORRAR MENSAJES...			VOLVER AL INICIO		

Título del mensaje:

Prueba de mensaje

Nombre del usuario destinatario:

sergio

Escriba su mensaje:

Esto es una prueba del sistema de mensajes. Si el mensaje es nuevo saldrá en verde.

Enviar

VOLVER A **MIS MENSAJES**

Título: Prueba de mensaje Enviado por: spelean El día: 2015/05/31-22:45:06

Esto es una prueba del sistema de mensajes. Si es un mensaje nuevo saldrá en verde.

VOLVER A **MIS MENSAJES**

Y por último puede ver sus libros comprados, alquilados y favoritos, además de leerlos:

THE FUTURE OF READING

BIENVENIDO **SERGIO**

TIPO DE USUARIO: USUARIO

SA LIR

MENU

El intenso perfume de las rosas embalsamaba el estudio y, cuando la ligera brisa agitaba los árboles del jardín, entraba, por la puerta abierta, un intenso olor a lilas o el aroma más delicado de las flores rosadas de los espinos. Lord Henry Wotton, que había consumido ya, según su costumbre, innumerables cigarrillos, vislumbraba, desde el extremo del sofá donde estaba tumbado -tapizado al estilo de las alfombras persas-...

EL RETRATO DE DORIAN GRAY

BÚSCAME

Buscar

MIS LIBROS **COMPRADOS**

J.R.R. TOLKIEN

LAS DOS TORRES

EL SEÑOR DE LOS ANILLOS II

El Señor de los Anillos, II

J.R.R. Tolkien

Leer

J.R.R. TOLKIEN

EL RETORNO DEL REY

EL SEÑOR DE LOS ANILLOS III

El Señor de los Anillos, III

J.R.R. Tolkien

Leer

Harry Potter

Harry Potter y la Orden del Fénix

J.K. Rowling

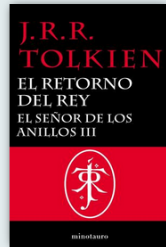
Leer

MIS LIBROS **ALQUILADOS**

La invasión divina

Harry Potter

Harry Potter y el prisionero de Azkaban



EL SEÑOR DE LOS ANILLOS, III. EL RETORNO DEL REY J.R.R. TOLKIEN

AÑO PUBLICACIÓN: 2010-07-14
FANTASIA
EDITORIAL: GRUPO PLANETA SPAIN
619 PÁGINAS

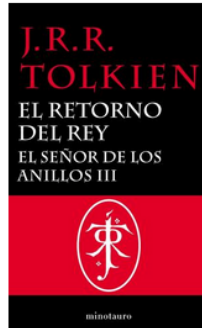
[Eliminar de favoritos](#)

[Leer Libro](#)

SINOPSIS

Los ejércitos del Señor Oscuro van extendiendo cada vez más su maléfica sombra por la Tierra Media. Hombres, elfos y enanos unen sus fuerzas para presentar batalla a Sauron y sus huestes. Ajenos a estos preparativos, Frodo y Sam siguen adentrándose en el país de Mordor en su heroico viaje para destruir el Anillo de Poder en las Grietas del Destino.

AUTOR



EL SEÑOR DE LOS ANILLOS, III. EL RETORNO DEL REY J.R.R. TOLKIEN

AÑO PUBLICACIÓN: 2010-07-14
FANTASIA
EDITORIAL: GRUPO PLANETA SPAIN
619 PÁGINAS

[VOLVER A MIS LIBROS](#)

1

EL SEÑOR DE LOS ANILLOS J.R.R. TOLKIEN

PARTE III:
EL RETORNO DEL REY

Libro 5

1 MINAS TIRITH

Pippin miró fuera amparado en la capa de Gandalf. No sabía si estaba despierto o si dormía, dentro aún de ese sueño vertiginoso que lo había arrebujaado desde el comienzo

Colaborador:

Siguiendo con la jerarquía, el colaborador podrá además administrar los libros, lo que engloba añadir y editar libros:



BIENVENIDO **COLABOR**
TIPO DE USUARIO: COLABORADOR
SALIR

MENÚ

NUEVO LIBRO

INTRODUZCA LOS DATOS (*CAMPO NO OBLIGATORIO*)

VOLVER AL INICIO

Titulo del libro

Autor

Genero

Año publicación

Sinopsis

Numero de páginas



BIENVENIDO **COLABOR**
TIPO DE USUARIO: COLABORADOR
SALIR

MENÚ

PERFIL

MIS MENSAJES

ENVIAR MENSAJE

AÑADIR LIBRO

MIS LIBROS



HARRY POTTER Y EL PRISIONERO DE AZKABAN

J.K. ROWLING

AÑO PUBLICACIÓN: 2012-02-24

FANTASIA

EDITORIAL: POTTERMORE

110 PÁGINAS

Añadir a favoritos

Leer Libro

EDITAR LIBRO

[Volver a la pagina de inicio](#)

MODIFICAR LIBRO

Establezca los cambios

Campo	Valor actual	Nuevo valor
Título del libro:	Libro de Prueba	
Autor	Prueba Autor	
Genero	Ciencia-Ficcion	
Año publicación	1994-12-20	dd/mm/aaaa
Sinopsis	Sobre un libro de prueba	
Numero de páginas	288	
Editorial:	Alfaguara	
'Saga'	Nº	Nº
Precio alquiler	1 €	€
Precio comprar	12 €	€

[Modificar portada »](#)

[Modificar pdf »](#)

Administrador:

Por último, el administrador tiene la posibilidad de eliminar libros, añadir autores a la base de datos y administrar los usuarios, cambiando sus permisos y roles:



THE FUTURE OF READING

BIENVENIDO **SPELEAN**
TIPO DE USUARIO: ADMINISTRADOR
SALIR

MENÚ

Tiene 1 mensaje sin leer

PERFIL

MIS MENSAJES

ENVIAR MENSAJE

ADMINISTRAR USUARIOS

AÑADIR AUTOR

AÑADIR LIBRO



THE FUTURE OF READING

BIENVENIDO **SPELEAN**
TIPO DE USUARIO: ADMINISTRADOR
SALIR

MENÚ

Nombre del autor

Isaac Asimov

Escriba una breve descripción del autor:

Isaac Asimov, escritor norteamericano de origen ruso, nació en Petrovich en 1920 y falleció en 1992. Doctor en ciencias por la Universidad de Columbia, fue también profesor de

Añadir autor

Autores ya existentes:

J.K. Rowling	Eliminar
J.R.R. Tolkien	Eliminar
Philip K. Dick	Eliminar

VOLVER AL INICIO

BÚSQUEDA DE USUARIOS

Buscar por:

☒ Fecha de alta
 ☐ Usuario
 ☐ ID
 ☐ Edad

BUSCAR

Id	Tipo	Usuario	Nombre	Apellido	Email	NIF	Edad	Nacimiento	Teléfono	Fecha de alta	Rol de usuario
6	Usuario	AnaG94	Ana	Guillem	asd@asdes	53903255H	20	1994-12-20	no	2015-06-08	▼ Cambiar
7	Usuario	Mec95	Mec	mec	mec@meces	53903255H	20	1994-12-20	no	2015-06-08	▼ Cambiar
5	Usuario	usu1	usuario	de prueba	test@test.com	50881340G	0	2015-06-01	no	2015-06-07	▼ Cambiar
3	Usuario	sergio	sergio	Garrido Escobar	segarrido85@gmail.com	50881340G	29	1985-07-19	no	2015-05-31	▼ Cambiar
4	Colaborador	colabor	cola	borator	s@s.com	50881340G	715	1300-03-12	no	2015-05-31	▼ Cambiar
1	Administrador	spelean	Fernando	Garcia	ferferfer321@hotmail.com	50880608p	30	1984-06-20	665685652		▼ Cambiar

4. Codificación

4.1 Entornos de programación y herramientas de desarrollo

Los entornos de desarrollo utilizados para este proyecto han sido **Windows 8.1, Windows 7, Ubuntu 14 y Raspbian**. En cuanto a las herramientas, aunque hemos usado en los primeros compases algunas como Sublime Text o Xampp, las que han sido utilizadas la mayor parte del desarrollo han sido las siguientes:



- **LAMP** como paquete de servicios para el alojamiento y soporte de la aplicación web. Este paquete gratuito contiene **Apache, MySQL y PHP, Perl y Python**.

- La web ha sido diseñada por completo con **Notepad++**, un editor sencillo pero versátil que aporta las guías necesarias para que la codificación de la web no sea demasiado complicada.
- **Normalize** permite a los navegadores renderizar todos los elementos de una forma más consistente y de acuerdo a los estándares actuales, evitando bugs y añadiendo una mayor compatibilidad del CSS con la mayoría de navegadores.

4.2 Lenguajes usados

A lo largo del desarrollo de nuestra plataforma web hemos usado **HTML5**, **CSS3**, **MySQL**, **Javascript** y **PHP**.

4.3 Aspectos relevantes de la implementación

Validación de datos:

Mediante el uso de PHP hemos aplicado validación de datos en los siguientes apartados:

creación_usuario.php la cual mediante un formulario me hará introducir los siguientes datos: nombre de usuario, nombre, apellidos, Email, repetición del email, contraseña, repetición de contraseña, NIF, fecha de nacimiento y opcional el teléfono móvil, todos estos datos serán llevados a la página **creación_usuario_comprobacion.php** en la cual se comprueba cada uno de los datos introducidos de la siguiente manera:

nombre de usuario: Se comprueba que el campo no esté vacío, que el nombre introducido tenga al menos cuatro caracteres y que no exista un usuario con el mismo nombre de usuario.

- **nombre** : Se comprueba que el campo no esté vacío.
- **apellidos** : Se comprueba que el campo no esté vacío.
- **email** : Se comprueba que el campo no esté vacío, que lo introducido sea un email (se comprueba que tenga al menos una @) y que no exista previamente el email introducido en la base de datos.
- **repetición del email**: Se comprueba que el campo no esté vacío y que lo introducido coincida con el campo email.
- **contraseña**: Se comprueba que el campo no esté vacío, que sean seis caracteres al menos y que esos caracteres sean letras y números.
- **repetición del email**: Se comprueba que el campo no esté vacío y que lo introducido coincida con el campo contraseña.

- **NIF:** Se comprueba que el campo no esté vacío, que en la parte de los números solo se introduzcan números y que sean ocho números y por último que la letra escogida coincida mediante un algoritmo de validación con el número de NIF introducido.
- **fecha de nacimiento:** Se comprueba que el campo no esté vacío, una vez introducida se obtiene la fecha actual y se calcula la edad del usuario, dato que será introducido junto al resto de datos del usuario al final de la comprobación de errores.
- **Teléfono:** Se comprueba solamente si se han introducido datos que estos sean solo números y que estos sean exactamente nueve.

Si todo este proceso de comprobación de errores no es correcto se mostrará en la página [creación_usuario.php](#) con el error que haya ocurrido, en caso contrario los datos en la tabla “usuario” de la base de datos y se creará dentro de la carpeta “carpetas_usuarios” una carpeta con el nombre del nuevo usuario y dentro de esta otras dos carpetas “leídos” y “nuevos” las cuales contendrán todos los mensajes recibidos por el usuario (este proceso está explicado con detalle más adelante), además se enviará un mensaje de bienvenida a la página web, mensaje que se almacenará en la carpeta “nuevos”. Una vez hecho esto se lanza automáticamente la página [index.php](#) con la conexión del nuevo usuario ya establecida.

[modificar_mi_usuario.php](#), la cual nos mostrará todos los datos introducidos en mi perfil y además nos proporciona un formulario para cambiar los datos que deseemos, una vez introducidos se enviarán a la página [modificar_mi_usuario_comprobacion.php](#) la cual comprobará posibles errores de la siguiente manera: (Todos los campos si están vacíos se otorgan los valores que ya tenían en la base de datos)

- **nombre de usuario:** Se comprueba que el nombre introducido tenga al menos cuatro caracteres y que no exista un usuario con el mismo nombre de usuario.
- **Nombre:** No se comprueba.
- **Apellidos:** No se comprueba.
- **email :** Se comprueba que lo introducido sea un email (se comprueba que tenga al menos una @) y que no exista previamente el email introducido en la base de datos.
- **repetición del email:** Se comprueba (solamente si se ha introducido algo en el campo email) que lo introducido coincida con el campo email.

- **contraseña:** Se comprueba que sean seis caracteres al menos y que esos caracteres sean letras y números.
- **repetición de contraseña:** Se comprueba (solamente si se ha introducido algo en el campo contraseña) que lo introducido coincida con el campo contraseña.
- **NIF:** Se comprueba que en la parte de los números solo se introduzcan números y que sean ocho números y por último que la letra escogida coincida mediante un algoritmo de validación con el número de NIF introducido.
- **fecha de nacimiento:** Si se cambian los datos del campo se obtiene la fecha actual y se calcula la edad del usuario, dato que será introducido junto al resto de datos del usuario al final de la comprobación de errores.
- **Teléfono:** Se comprueba que los datos introducidos sean solo números y que estos sean exactamente nueve.

Si todo este proceso de comprobación de errores no es correcto se mostrará en la página [modificar_mi_usuario.php](#) con el error que haya ocurrido, si todo es correcto se actualizarán los datos en la tabla “usuario” de la base de datos y se modificará el nombre de la carpeta del usuario la cual se encuentra dentro de la carpeta “carpetas_usuarios” esto se hará solamente si se está cambiando el nombre del usuario. Una vez hecho esto se lanza automáticamente la página [modificar_mi_usuario.php](#) con los nuevos datos ya actualizados.

[anadir_libro.php](#), en esta página tendremos que rellenar un formulario, cuyos datos serán reenviados a la página [anadir_libro_comprobacion.php](#) en esta página se comprobarán dichos datos de la siguiente manera:

- **Título:** Se comprueba que el título no esté vacío y que no exista un título con el mismo nombre en la base de datos.
- **Autor:** Se comprueba que no esté vacío (Hay que seleccionar uno de una lista desplegable cuyos componentes se introducen en la página [anadir_autor.php](#) (como ya explicamos más arriba)).
- **Género:** Se comprueba que no esté vacío.
- **Año publicación:** Se comprueba que no esté vacío.
- **Sinopsis:** Se comprueba que no esté vacío.
- **Número de páginas:** Se comprueba que no esté vacío y que lo introducido sean sólo números.
- **Género:** Se comprueba que no esté vacío.
- **Editorial:** Se comprueba que no esté vacío.
- **Saga:** Se comprueba que no esté vacío solamente si se ha rellenado el campo Número de saga.

- **Número de saga:** Se comprueba que no esté vacío solamente si se ha rellenado el campo Saga, además se comprueba en este caso que lo introducido sean solo números.
- **Editorial:** Se comprueba que no esté vacío.
- **Precio alquiler:** Se comprueba que no esté vacío y que lo introducido no esté vacío.
- **Precio compra:** Se comprueba que no esté vacío y que lo introducido no esté vacío.

Si aparece algún error se muestra de nuevo la página [anadir_libro.php](#) con su mensaje de error correspondiente, si todo es correcto se inserta una nueva entrada en la tabla “libros” con los datos insertados y se creará una carpeta dentro de la carpeta del autor del libro, esta carpeta se nombrará con el nombre que le hayamos puesto al título del libro, después se lanza la página [cargar_portada.php](#) donde seleccionaremos el archivo jpeg que deseamos para el libro a subir, el archivo se guarda en la carpeta ebooks/nombre del autor/título del libro y se renombra como cover.jpeg, y a continuación lanza la página [cargar_libro.php](#) donde seleccionaremos el archivo pdf del libro propiamente dicho, como la portada este archivo se guarda en ebooks/nombre del autor/título del libro y se renombra como ebook.pdf.

[anadir_libro.php](#): en esta página tendremos que rellenar un formulario, cuyos datos serán reenviados a la página [anadir_libro_comprobacion.php](#) en esta página se comprobarán dichos datos de la siguiente manera:

- **Título:** Se comprueba que el título no esté vacío y que no exista un título con el mismo nombre en la base de datos.
- **Autor:** Se comprueba que no esté vacío (Hay que seleccionar uno de una lista desplegable cuyos componentes se introducen en la página [anadir_autor.php](#) (como ya explicamos más arriba)).
- **Género:** Se comprueba que no esté vacío.
- **Año publicación:** Se comprueba que no esté vacío.
- **Sinopsis:** Se comprueba que no esté vacío.
- **Número de páginas:** Se comprueba que no esté vacío y que lo introducido sean sólo números.
- **Género:** Se comprueba que no esté vacío.
- **Editorial:** Se comprueba que no esté vacío.
- **Saga:** Se comprueba que no esté vacío solamente si se ha rellenado el campo Número de saga.

- **Número de saga:** Se comprueba que no esté vacío solamente si se ha rellenado el campo Saga, además se comprueba en este caso que lo introducido sean solo números.
- **Editorial:** Se comprueba que no esté vacío.
- **Precio alquiler:** Se comprueba que no esté vacío y que lo introducido no esté vacío.
- **Precio compra:** Se comprueba que no esté vacío y que lo introducido no esté vacío.

Si aparece algún error se muestra de nuevo la página [anadir_libro.php](#) con su mensaje de error correspondiente, si todo es correcto se inserta una nueva entrada en la tabla “libros” con los datos insertados y se creará una carpeta dentro de la carpeta del autor del libro, esta carpeta se nombrará con el nombre que le hayamos puesto al título del libro, después se lanza la página [cargar_portada.php](#) donde seleccionaremos el archivo jpeg que deseamos para el libro a subir, el archivo se guarda en la carpeta ebooks/nombre del autor/título del libro y se renombra como cover.jpeg, y a continuación lanza la página [cargar_libro.php](#) donde seleccionaremos el archivo pdf del libro propiamente dicho, como la portada este archivo se guarda en ebooks/nombre del autor/título del libro y se renombra como ebook.pdf.

Control de acceso:

El control de acceso a las diferentes secciones de la web se ha realizado mediante variables de sesión y condiciones en PHP, posibilitando en según qué caso y como se muestra en el apartado de diseño qué parte puede ver cada usuario. Un ejemplo de este funcionamiento sería el menú lateral:

```

<?php
#contenido del menu lateral segun usuario
if ($SESSION['tipo'] == "anonimo") {
echo '<li><a href="creacion_usuario.php"><i class="fa fa-fw fa-bookmark-q"></i><span>Perfil</span></a></li>';
}

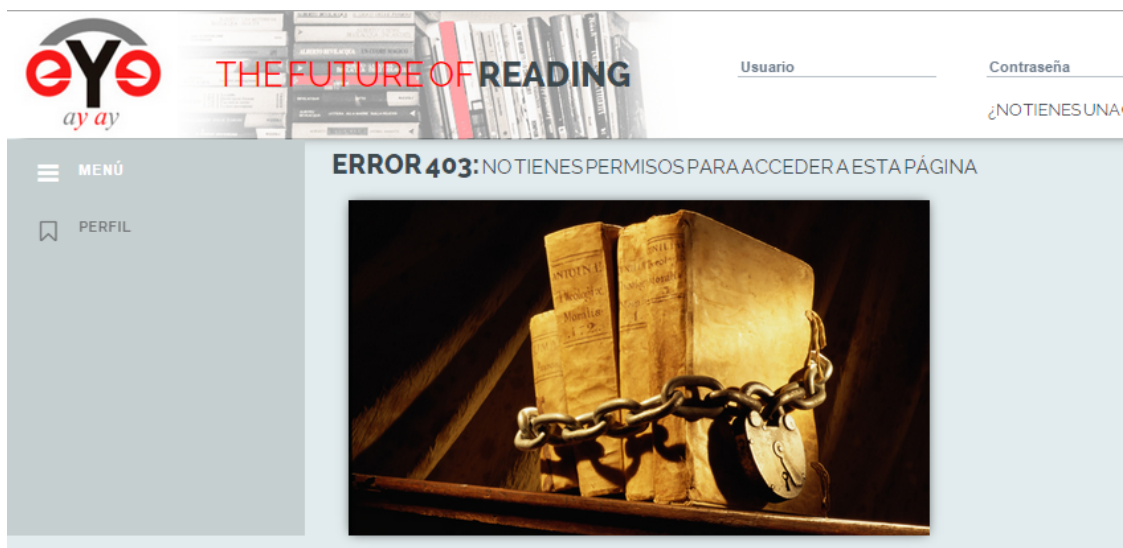
if ($SESSION['tipo'] == "Administrador") {
echo '
<li><a href="modificar_mi_usuario.php"><i class="fa fa-fw fa-bookmark-q"></i><span>Perfil</span></a></li>
<li><a href="ver_mensajes_nuevos.php"><i class="fa fa-fw fa-envelope-q"></i><span>Mis mensajes</span></a></li>
<li><a href="mandar_mensaje.php"><i class="fa fa-fw fa-bell-q"></i><span>Enviar Mensaje</span></a></li>
<li><a href="ver_usuarios.php"><i class="fa fa-fw fa-user-q"></i><span>Administrar usuarios</span></a></li>
<li><a href="anadir_autor.php"><i class="fa fa-fw fa-image"></i><span>Añadir autor</span></a></li>
<li><a href="anadir_libro.php"><i class="fa fa-fw fa-image"></i><span>Administrar libros</span></a></li>
';
}

if ($SESSION['tipo'] == "Colaborador") {
echo '
<li><a href="modificar_mi_usuario.php"><i class="fa fa-fw fa-bookmark-q"></i><span>Perfil</span></a></li>
<li><a href="ver_mensajes_nuevos.php"><i class="fa fa-fw fa-envelope-q"></i><span>Mis mensajes</span></a></li>
<li><a href="mandar_mensaje.php"><i class="fa fa-fw fa-bell-q"></i><span>Enviar Mensaje</span></a></li>
<li><a href="anadir_libro.php"><i class="fa fa-fw fa-image"></i><span>Administrar libros</span></a></li>
<li><a href="mis_libros.php"><i class="fa fa-fw fa-bell-q"></i><span>Mis libros</span></a></li>
';
}

if ($SESSION['tipo'] == "Usuario") {
echo '
<li><a href="modificar_mi_usuario.php"><i class="fa fa-fw fa-bookmark-q"></i><span>Perfil</span></a></li>
<li><a href="ver_mensajes_nuevos.php"><i class="fa fa-fw fa-envelope-q"></i><span>Mis mensajes</span></a></li>
<li><a href="mandar_mensaje.php"><i class="fa fa-fw fa-bell-q"></i><span>Enviar Mensaje</span></a></li>
<li><a href="mis_libros.php"><i class="fa fa-fw fa-bell-q"></i><span>Mis libros</span></a></li>
';
}
??

```

También se realizará dicho control según usuario para evitar que acceda a páginas no deseadas mostrando el siguiente mensaje:



Encriptación de la información:

En aras de proteger los datos de inicio de sesión de nuestros usuarios hemos utilizado la encriptación md5:

`$password = md5($password);`

Sistema para evitar la copia de los documentos PDF en alquiler:

Para evitar el copiado de los libros que tenemos en alquiler, hemos recurrido a la creación de un DIV invisible que se sitúe por encima de lector pdf de la siguiente manera y con el estilo que se muestra:

```
<div class="anticopia" id="anticopia"></div>
```

```
.anticopia{  
  position:absolute;  
  top:500px;  
  left:310px;  
  font-size: 0.9em;  
  opacity:0;  
  color: #3c4a50;  
  width:780px;  
  height:110%;  
  padding-top:6px;  
}
```



Si coloreamos dicho DIV su aspecto será el siguiente, en donde se puede observar que tapa los botones del lector PDF pero deja libre la barra de navegación o scroll:



Sistema de carpetas:

Cada vez que se crea/elimina un usuario, o un autor o un libro lleva detrás la creación/eliminación de todo un sistema de carpetas asociado.

En la carpeta raíz de nuestro proyecto (la cual se llama proyecto) se encontrarán ya creadas por defecto dos carpetas **carpetas_usuarios** y **ebooks**, estas carpetas contendrán los mensajes de los futuros usuarios y los ebooks respectivamente. Una vez aclarado donde se crearán los futuros archivos y carpetas expliquemos como lo hemos realizado:

Creación carpetas usuarios:

Lo primero que hicimos (una vez comprobados errores e insertado el nuevo usuario en la base de datos) fue establecer dentro de una variable (\$directorio) la ruta de la carpeta del nuevo usuario, nombrándola con su nombre de usuario (\$usuario):

```
$directorio = "./carpetas_usuarios/$usuario";
```

Después se crea la carpeta en la ruta establecida con los permisos pertinentes mediante (0777) mediante la función mkdir:

```
mkdir($directorio,0777);
```

Una vez creada esta carpeta y otorgados los permisos se procede a crear de la misma manera las subcarpetas leídos y nuevos las cuales irán dentro de la carpeta del nuevo usuario:

```
$directorio = "./carpetas_usuarios/$usuario/nuevos";  
mkdir($directorio,0777);  
$directorio = "./carpetas_usuarios/$usuario/leidos";  
mkdir($directorio,0777);
```

Para la creación de la carpeta autor se crea igual pero en vez de en la carpeta “carpetas_usuarios” en la carpeta “ebooks”.

Este sistema será el mismo para la creación de las carpetas de los libros, se obtendrá en una variable directorio la ruta del libro seleccionando de la lista desplegable el nombre de un autor ya existente y después el de el libro a insertar y será esta la carpeta que se cree, como ya hemos mencionado con mkdir.

De igual manera que cuando se inserta un autor, un libro o un nuevo usuario se crean las carpetas tenemos la opción (siendo administrador) de eliminar uno de los antes citados.

Eliminación carpetas autores:

Para ello obtenemos primero la ruta de la carpeta del autor

```
$carpeta = "./ebooks/$autor_mayusc";
```

Después utilizando una función a la que llamaremos más tarde creamos el sistema de borrado:

```
function eliminar_recurso($carpeta){
```

Mediante la función `opendir` la cual abre una carpeta y la función `readdir` la cual lee todos los elementos de una carpeta (esta última la establezco como opción de salida de el siguiente bucle):

```
$directorio = opendir($carpeta);  
while($archivo = readdir($directorio)){
```

Salto, dentro del bucle que va leyendo los archivos de la carpeta indicada, los archivos `.` y `..` para que no los considere:

```
if($archivo != '.' && $archivo != '..') {
```

Si es un directorio, volvemos a llamar a la función para que elimine el contenido del mismo:

```
if(is_dir($carpeta.'/'.$archivo)) {  
  
    eliminar_recurso($carpeta.'/'.$archivo);  
}
```

Si no es un directorio, lo borramos con la función `unlink`:

```
else{  
  
    unlink($carpeta.'/'.$archivo);  
}  
}  
}
```

Después se cierra el directorio con la función `closedir`:

```
closedir($directorio);
```

Y a continuación borramos el directorio base indicado antes(`$carpeta = "/ebooks/$autor_mayusc";`)

```
rmdir($carpeta);  
}
```

Por último llamamos a la función:

```
eliminar_recurso($carpeta);
```

Para eliminar las carpetas correspondientes a archivos de usuario, o si estamos eliminando un libro, cuando se ha borrado un usuario o un libro, utilizamos una función casi idéntica adaptada a los valores que necesitamos para ese caso.

Sistema de envío de mensajes:

Después de pulsar el botón de enviar mensaje (y de realizar todas las comprobaciones pertinentes), los datos de este (título, remitente, destinatario y fecha de envío) se envían a la tabla mensajes de la base de datos, la cual será consultada para mostrar dicha información por pantalla cuando sea requerida, pero el contenido del mensaje se guardará en la carpeta “nuevos” del usuario destinatario de la siguiente manera.

Guardamos en una variable (\$file) la instrucción fopen (la cual indicará donde tiene que escribir el nuevo archivo de texto) y a continuación le pondremos la ruta del usuario remitente (en este caso \$destinatario), como ya hemos comentado en la carpeta “nuevos” ira el archivo de texto el cual nombraremos con el valor que se le haya otorgado a ID_mensaje (\$fila2[0]) que es un número autoincremental. Esta ID_mensaje se ha creado al insertar los valores (titulo,remite,destinatario y fecha de envio) antes mencionados en la base de datos:

```
$file=fopen("./carpetas_usuarios/$destinatario/nuevos/$fila2[0].txt","a");
```

Una vez determinada esta variable procedemos a escribir el mensaje propiamente dicho(\$mensaje):

```
fwrite($file,"$mensaje".PHP_EOL);
```

Y cerramos el mensaje:

```
fclose($file);
```

Como hemos mencionado este mensaje se guarda en la carpeta “nuevos” del usuario, pero una vez que ha sido leído será borrado de esta carpeta y copiado a

la carpeta “leídos” del nuevo usuario, esto lo hemos realizado de la siguiente manera:

Una vez que está abierto el archivo y mostrado por pantalla el contenido del mensaje, realizamos una copia del mismo a la carpeta antes mencionada mediante el comando copy:

```
copy("./carpetas_usuarios/$usuario/nuevos/$archivo.txt","./carpetas_usuarios/$usuario/leidos/$archivo.txt");
```

Y borramos la anterior mediante el comando unlink:

```
unlink("./carpetas_usuarios/$usuario/nuevos/$archivo.txt");
```

A la hora de borrar mensajes hay que tener en cuenta que el sistema está diseñado como un formulario el cual contiene todos los mensajes y se pueden ir seleccionando los que se desea eliminar y dentro de este formulario otro el cual sólo afectará a cada mensaje individualmente y servirá para visualizar el mensaje indicado. Así cada checkbox llevará asociado un input como este:

```
<input type='checkbox' name='mensaje_a_borrar[]' value='$fila[0]'>
```

El cual si nos fijamos tiene el valor de \$fila[0] que es el valor del ID_mensaje de ese mensaje en concreto, el cual es el valor autoincremental único que explicábamos más arriba, esto lo metemos en el array mensaje_a_borra[]. Después contamos el número de variables que ha recogido el array mensaje_a_borra[]:

```
$contar = count($mensaje_a_borrar);
```

Y nos metemos ese número de veces en el siguiente bucle:

```
for ($i=0;$i<$contar;$i++) {
```

Dentro del bucle borramos el cada mensaje por separado y de la base de datos también:

```
$borrar = $mensaje_a_borrar[$i];
mysql_query("delete      from      mensajes      where
ID_mensaje='$borrar'", $conexion);
```

Borramos el mensaje .txt dicho de cualquiera de las dos carpetas (nuevos o leídos):

```
if (unlink("./carpetas_usuarios/$usuario/leidos/$borrar.txt"));
if (unlink("./carpetas_usuarios/$usuario/nuevos/$borrar.txt"));
}
```

Alerta Mensajes sin leer:

Este sistema lo hemos desarrollado para cuando hay algún mensaje nuevo sin leer, para ello podríamos haber leído dentro de la tabla “mensajes” la columna leídos que por defecto cuando se envía un mensaje es “no” (cuando se lee un mensaje el valor cambia a si). Lo que hemos hecho ya que como tenemos el sistema de carpetas que cuando lees los mensajes pasen de la carpeta “nuevos” a “leídos”, por lo tanto sólo tenemos que ver si existe algún mensaje en la carpeta “nuevos”, para ello:

Cuento el número de archivos existentes en dicha carpeta e insertamos el dato dentro de una variable:

```
$mensajes_nuevos=count(glob("./carpetas_usuarios/$usuario/nuevos/{*.txt}", GLOB_BRACE));
```

Si ese número es mayor que cero mostraremos la alerta de que hay mensajes sin leer:

```
if ($mensajes_nuevos > 0) {
echo      "<a      href='ver_mensajes_nuevos.php'>Tiene      "
.$mensajes_nuevos. " mensaje sin leer</a><br>";
}
```

Menú desplegable

Para animar los SVGs (Scalable Vector Graphics), utilizaremos la librería JavaScript SVG para navegadores modernos. En este menú deberemos dejar algo de espacio entre sus laterales para poder ver el efecto elástico que se anima.

El vector está insertado después del *inner menu* y utilizamos 2 atributos data para almacenar las rutas por defecto en las que utilizaremos la animación (data-morph-open), y gracias a Snap.svg podemos transformarlo de una forma a otra.

5. Configuración del servidor

5.1 Hardware

Como servidor utilizaremos un hardware con el que hemos trabajado a lo largo del curso, nos referimos a la Raspberry Pi B+. Nos hemos decidido por este modelo debido a que sus características son suficientes para desempeñar las funciones y servicios que necesitaremos a la hora de alojar nuestro portal web y tiene un coste tanto energético como económico mínimo. Las especificaciones son las siguientes:

Procesador: Broadcom BCM2835 SoC full HD (igual que el modelo B)

RAM: 512 MB SDRAM 400 MHz (igual que el modelo B)

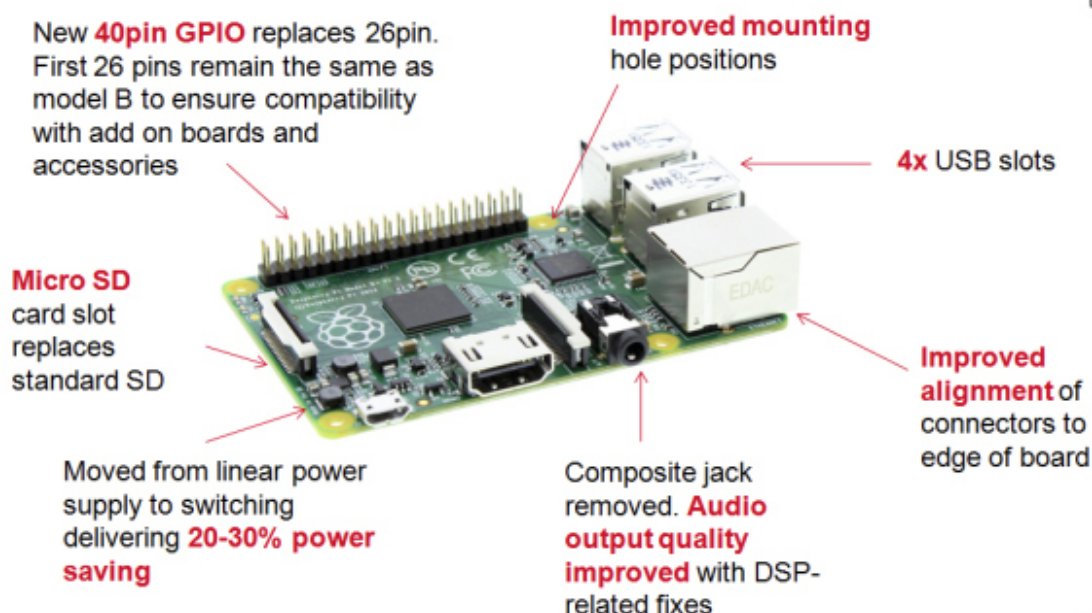
Almacenamiento: tarjeta microSD (en el modelo B era una SD)

USB: cuatro puertos USB 2.0 (en el modelo B sólo había 2)

Energía: 600mA hasta 1.8A a 5V (en el modelo B, 750mA hasta 1.2A a 5V)

Pines GPIO: 40 (en el modelo B, 26)

Tarjeta de memoria MicroSD: Kingston clase 10 de 16GB.



El hardware se sustenta sobre el **sistema operativo Raspbian**. Para la puesta a punto del servidor, hemos instalado el servicio LAMP: **Apache** (Servidor Web), **PHP5**, **MySQL server** (Servidor de Bases de Datos); y el servicio **ProFTPd** (Servidor FTP).

Para la instalación de los servicios nos hemos valido del terminal del sistema operativo Raspbian, y hemos ejecutado los siguientes comandos:

Apache

- `sudo aptitude update`
- `sudo aptitude install apache2`

MySQL Server

- `sudo aptitude install mysql-server`

PHP5

- `sudo aptitude install php5`
- `sudo aptitude install php5-mysql`

5.2 Seguridad Apache

Los puntos de seguridad que hemos tenido en cuenta para nuestra página web han sido, entre otros:

- **Seguridad para las carpetas del sitio web**

DocumentRoot /var/www/proyecto ← Directorio base

<Directory />

Options FollowSymLinks

AllowOverride None

</Directory>

<Directory /var/www/proyecto>

Options -Indexes FollowSymLinks MultiViews ← Prohibición de la lectura

de subcarpetas

AllowOverride None

Order allow,deny

allow from all

</Directory>

A nivel de configuración del servidor Apache, hemos establecido como directorio raíz la carpeta *proyecto* que se halla dentro de la carpeta */var/www*. Además, para asegurarnos que cualquier usuario que utilice la web no tenga acceso a las subcarpetas del sitio web, hemos modificado el archivo *default* de la carpeta de configuración de Apache */etc/apache2/sites-available* y hemos modificado la línea mostrada arriba con *Options -Indexes* para que no muestre las subcarpetas del sitio.

5.3 Puesta en funcionamiento en internet mediante NO - IP

Para hacer que nuestro servidor Raspberry Pi sea un servidor completamente funcional y accesible desde cualquier punto de internet utilizaremos el sistema No - IP. Este sistema hace que cada vez que se reinicia el router, o cambia la IP pública por algún motivo, lo envíe a unos DNS dinámicos para que redirijan nuestra IP.

El primer paso que debemos hacer es registrarnos en <http://www.noip.com>.

Create Your No-IP Account

ana_guillem@gmail.com	ana_guillem@gmail.com
ayeyay	
ayeyay	.ddns.net ▼

El siguiente paso es activar la cuenta, y loguearse. Una vez dentro, debemos descargar el cliente en nuestro servidor, en este caso una Raspberry Pi. Ahora podremos descargar nuestro cliente no-ip en el servidor. Para ello ejecutaremos el comando:

wget http://www.no-ip.com/client/linux/noip-duc-linux.tar.gz

```
pi@raspberrypi ~ $ wget http://www.no-ip.com/client/linux/noip-duc-linux.tar.gz
--2015-06-07 21:01:12-- http://www.no-ip.com/client/linux/noip-duc-linux.tar.gz
Resolviendo www.no-ip.com (www.no-ip.com)... 8.23.224.110
Conectando con www.no-ip.com (www.no-ip.com) [8.23.224.110]:80... conectado.
Petición HTTP enviada, esperando respuesta... 301 Moved Permanently
Localización: http://www.noip.com/client/linux/noip-duc-linux.tar.gz [siguiendo]
--2015-06-07 21:01:13-- http://www.noip.com/client/linux/noip-duc-linux.tar.gz
Resolviendo www.noip.com (www.noip.com)... 8.23.224.107
Conectando con www.noip.com (www.noip.com) [8.23.224.107]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 134188 (131K) [application/x-gzip]
Grabando a: "noip-duc-linux.tar.gz"

100%[=====>] 134.188      121K/s   en 1,1s

2015-06-07 21:01:15 (121 KB/s) - "noip-duc-linux.tar.gz" guardado [134188/134188]
]
```

Ahora procederemos a la descompresión del archivo mediante el comando:

tar -zxvf noip-duc-linux.tar.gz

Acto seguido, entramos en la carpeta y ejecutamos el comando make seguido de sudo make install y rellenamos los datos del script con nuestra cuenta y contraseña de no IP, establecemos el intervalo de actualización predefinido (30s) y elegimos si queremos que se ejecute algo cada vez que se actualiza:

```

pi@raspberrypi ~/noip-2.1.9-1 $ sudo make install
if [ ! -d /usr/local/bin ]; then mkdir -p /usr/local/bin;fi
if [ ! -d /usr/local/etc ]; then mkdir -p /usr/local/etc;fi
cp noip2 /usr/local/bin/noip2
/usr/local/bin/noip2 -C -c /tmp/no-ip2.conf

Auto configuration for Linux client of no-ip.com.

Please enter the login/email string for no-ip.com ana.guillem.dlf@gmail.com
Please enter the password for user 'ana.guillem.dlf@gmail.com' *****

Only one host [ayeyeay.ddns.net] is registered to this account.
It will be used.
Please enter an update interval:[30]
Do you wish to run something at successful update?[N] (y/N) n

New configuration file '/tmp/no-ip2.conf' created.
mv /tmp/no-ip2.conf /usr/local/etc/no-ip2.conf

```

Ahora configuraremos un fichero de arranque para que cada vez que se inicialice nuestro servidor Raspberry Pi se auto-inicie el programa. Para ello creamos un fichero llamado noip2 en la carpeta /etc/init.d:

sudo nano /etc/init.d/noip2

Y copiamos lo siguiente:

```

GNU nano 2.2.6          Fichero: /etc/init.d/noip2

#!/bin/bash
### BEGIN INIT INFO
# Provides: Servicio No-IP
# Required-Start: $syslog
# Required-Stop: $syslog
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Arranque automatico para noip
# Description:
#
### END INIT INFO
sudo /usr/local/bin/noip2

```

Guardamos el archivo, le damos permiso de ejecución, y lo colocamos en la cola de arranque:

sudo chmod +x /etc/init.d/noip2
sudo update-rc.d noip2 defaults

Por último debemos asegurarnos que tenemos abiertos los puertos 80 para HTTP y 22 para poder gestionarla de forma remota en caso de necesidad mediante SSH.

Una vez hecho esto, configuraremos nuestro host en la cuenta No-IP. Para realizar este paso, nos logueamos y vamos a la sección *Hosts / Redirects*, y hacemos clic en *Add Host*, y rellenamos el formulario:

Hostname Information	
Hostname:	ayeyeay ddns.net 
Host Type:	☒ DNS Host (A) ☐ DNS Host (Round Robin) ☐ DNS Alias (CNAME)  ☐ Port 80 Redirect ☐ Web Redirect ☐ AAAA (IPv6)
IP Address:	79.153.137.235 
Assign to Group:	- No Group -  Configure Groups
Enable Wildcard:	Wildcards are a Plus / Enhanced feature. Upgrade Now! 

Esta configuración nos guarda y actualiza nuestra IP pública del router (79.153.137.235 en este caso), y le damos un nombre.

Una vez realizadas estas configuraciones, procederemos a configurar el router para que el que intente acceder, lo haga por el puerto 80 (*HTTP*) y la petición llegue al servidor. Para ello entraremos en nuestro router y vamos a la sección (en el caso de nuestro router) *Port Forwarding*, y configuramos los puertos que queramos, en nuestro caso el 80:

Remaining number of entries that can be configured: 79

WAN Connection(s) : PVC:8/36 ▼

Server Name :

☒ Select a Service : Web Server (HTTP) ▼

☐ Custom Server :

Schedule : always ▼ [View Available Schedules](#)

Server IP Address(Host Name) : 192.168.1.45

External Port Start	External Port End	Protocol	Internal Port Start	Internal Port End	Remote Ip
80	80	TCP ▼	80	80	

6. Pruebas de acceso, creación de carpetas y seguridad

- Login

Intentamos logearnos de forma errónea y nos aparecerá el siguiente mensaje:

USUARIO O CONTRASEÑA INCORRECTOS

Usuario
spelean

Contraseña

VOLVER AL **INICIO**

ENTRAR

En caso de logearnos correctamente nos redireccionará a **index.php**



ay ay

THE FUTURE OF READING

BIENVENIDO **SPELEAN**
TIPO DE USUARIO: ADMINISTRADOR

SALIR

MENÚ

- **Añadir autor desde el panel del Administrador:**

En caso de dejar los campos vacíos:

Nombre del autor

Escriba una breve descripción del autor:

Añadir autor

Error campo autor vacio
Error campo descripcion vacio
Autores ya existentes:

Si lo hacemos correctamente con un autor llamado Prueba Autor:

Autores ya existentes:

Ana	Eliminar
H. P. Lovecraft	Eliminar
Isaac Asimov	Eliminar
J.K. Rowling	Eliminar
J.R.R. Tolkien	Eliminar
Philip K. Dick	Eliminar
Prueba Autor	Eliminar

Comprobamos que se haya creado su carpeta para los libros:

```
pi@raspberrypi /var/www/proyecto $ ls ebooks
Ana          Isaac Asimov  J.R.R. Tolkien Prueba Autor
H. P. Lovecraft J.K. Rowling  Philip K. Dick
```

- **Añadir libro desde el panel del Administrador:**

Si se hace incorrectamente:

Enviar

INICIALIZAR

ERROR CAMPO TÍTULO VACÍO
ERROR CAMPO AUTOR VACÍO
ERROR DEBE SELECCIONAR UN GENERO PARA EL LIBRO A INTRODUCIR
ERROR CAMPO AÑO DE PUBLICACIÓN VACÍO
ERROR CAMPO SINOPSIS VACÍO
ERROR CAMPO NÚMERO DE PÁGINAS VACÍO
ERROR CAMPO EDITORIAL VACÍO
ERROR CAMPO PRECIO ALQUILER VACÍO
ERROR CAMPO PRECIO COMPRA VACÍO

Si se hace correctamente:

NUEVO LIBRO

INTRODUZCALOSDATOS(*CAMPO NO OBLIGATORIO*)

VOLVER AL INICIO

Titulo del libro

Libro de Prueba

Autor

Prueba Autor

Genero

Ciencia-Ficcion

Año publicación

20/12/1994

Sinopsis

Sobre un libro de prueba

Numero de páginas

288

Editorial

Alfaguara

* Saga *

Nº

Precio alquiler

1

Precio comprar

12

Nos pedirá un archivo con formato de imagen para la portada

SELECCIONE UN ARCHIVO PARA LA PORTADA

imagen-prueba-nivel.jpg

Nos pedirá también el archivo en PDF del libro:

INTRODUZCA EL ARCHIVO DEL LIBRO

ebook.pdf

Y comprobamos que se crea el sistema de carpetas correspondiente al autor y su libro con los ficheros ya renombrados que hemos subido.

```
pi@raspberrypi /var/www/proyecto $ ls ebooks/Prueba\ Autor/Libro\ de\ Prueba/
cover.jpeg  ebook.pdf
```

- Prueba de que funcionan los mensajes

Si lo haces incorrectamente

Título del mensaje:

Nombre del usuario destinatario:

Escriba su mensaje:

Error campo destinatario vacío

VOLVER A MIS MENSAJES

Si lo haces correctamente

Título del mensaje:	Nombre del usuario destinatario:
Mensaje de Prueba	sergio
Escriba su mensaje:	
Este es un mensaje de prueba	Enviar

VOLVER A MIS MENSAJES

El mensaje ha llegado correctamente y se añade el archivo correspondiente que contiene el mensaje a la carpeta nuevos

x	Fecha de envío	Título	Enviado por Usuario	Leído	leer
☐	2015/06/08-19:56:29	Mensaje de Prueba	AnaG94	no	Leer
☐	2015/06/07-19:34:48	esto es una prueba	spelean	si	Leer
☐	2015/06/07-17:33:58	ae	sergio	si	Leer

```
pi@raspberrypi /var/www/proyecto $ ls carpetas_usuarios/sergio/nuevos/17.txt
```

Los mensajes ya leídos se muestran correctamente en la carpeta leídos:

```
pi@raspberrypi /var/www/proyecto $ ls carpetas_usuarios/sergio/leidos/2.txt 7.txt
```

- **Administrador de usuarios**

BÚSQUEDA DE USUARIOS

Buscar por:

☐ Fecha de alta
☒ Usuario
☐ ID
☐ Edad

BUSCAR

Id	Tipo	Usuario	Nombre	Apellido	Email	N.I.F	Edad	Nacimiento	Teléfono	Fecha de alta	Rol de usuario
6	Usuario	AnaG94	Ana	Guillem	asd@asd.es	53903255H	20	1994-12-20	no	2015-06-08	▼ Cambiar
7	Usuario	Mec95	Mec	mec	mec@mec.es	53903255H	20	1994-12-20	no	2015-06-08	▼ Cambiar
5	Usuario	usu1	usuario	deprueba	test@test.com	50881340G	0	2015-06-01	no	2015-06-07	▼ Cambiar
3	Usuario	sergio	sergio	Garrido Escobar	segarrido85@gmail.com	50881340G	29	1985-07-19	no	2015-05-31	▼ Cambiar

- Buscamos, por ejemplo, un *Usuario* que empieza por *ana*:

BÚSQUEDA DE USUARIOS

Buscar por:

☐ Fecha de alta
☒ Usuario
☐ ID
☐ Edad

BUSCAR

Id	Tipo	Usuario	Nombre	Apellido	Email	N.I.F	Edad	Nacimiento	Teléfono	Fecha de alta
6	Usuario	AnaG94	Ana	Guillem	asd@asd.es	53903255H	20	1994-12-20	no	2015-06-08

- **Comprobación de seguridad para entrar a las subcarpetas del sitio:**



7. Aspectos a mejorar

Durante la realización de este proyecto hemos encontrado diversas dificultades y errores que por falta de tiempo o dinero no hemos podido llevar a cabo y podrían ser aspectos a mejorar en futuras versiones.

En primer lugar, conseguimos solucionar en parte el **evitar que los usuarios pudiesen descargarse los libros** teniéndolos alquilados **mediante el uso de DIVs**. Esto para usuarios comunes funcionaría correctamente, pero para un usuario experimentado no es suficiente y tendríamos que usar otras herramientas. Durante nuestra investigación al respecto llegamos a la conclusión de que la única manera para evitar que apareciese el botón de descargar documento era mediante la creación específica de archivos PDF con **protección de descarga y contraseña**, aunque esta solución podría dar problemas a la hora de mostrar los libros por conflictos con los permisos, a parte de ser costosa ya que requiere la licencia del **Adobe Writer**.

En segundo lugar, creemos que, debido al auge de la telefonía móvil, podríamos haber desarrollado una aplicación para **Android e IOS** similar a las de **Apple, Google o Amazon** y así evitar cierta dependencia del navegador web a la hora de comprar libros.

Opinamos también que en un futuro una sola Raspberry no será suficiente a la hora de sustentar una plataforma más grande y con un tráfico mayor, por lo tanto

tendríamos que realizar una infraestructura de red más compleja y añadirle la seguridad pertinente.

Por otro lado, observamos que en muchas ocasiones a lo largo de todo nuestro código hemos tenido que realizar un bucle y dentro de este un formulario dentro de otro formulario. Esto generaba un error al que no le hemos podido encontrar la causa. Dicho error surgía al mostrar más de una entrada en esos formularios, donde el primer dato que mostraba (solamente los primeros datos del primer formulario) no permitía interactuar con el mismo y esos datos o no se enviaban o si era un botón no realizaba acción alguna. La solución que encontramos fue desarrollar un sistema mediante el cual la primera vez que se entra en el bucle (cuando $x=1$) se muestre antes que nada un formulario que no ejecuta acción alguna (el formulario fantasma), formulario que asimila dicho error y que nos permitiría después mostrarnos el resto de formularios deseados de forma completamente funcional.

Además, la primera vez que se introduce un libro este se genera su entrada en la base de datos normalmente, pero al contrario que el resto de libros introducidos, no crea la carpeta del libro y los archivos ebook y cover se sitúan directamente en la carpeta del autor generando un error, ya que estos archivos no siguen la jerarquía de nuestro árbol de carpetas y la misma los busca dentro de la carpeta del libro. Una vez introducido un segundo libro y aún siendo de otro autor diferente el error no se vuelve a producir.

Por último y siguiendo con la seguridad, mientras escribimos estas líneas estamos desarrollando métodos para evitar posibles ataques de seguridad mediante SQL injection, pero desconocemos si una vez finalizado el proyecto conseguiremos implementarlos satisfactoriamente.

8. Conclusiones

A lo largo del desarrollo de nuestro proyecto nos hemos dado cuenta de que, incluso habiendo aprendido mucho durante el curso, **el mundo de la informática es ilimitado** y siempre hay algo nuevo que descubrir. Gracias a no haber utilizado herramientas como Wordpress o Frontpage, hemos podido “pelearnos” con el código, ver como los elementos de nuestra página web no quedaban como queríamos o incluso casi rehacer el proyecto tras implementar el PHP debido a los problemas de posición en los DIV que causó. **Hemos profundizado en otros**

campos en los que sólo habíamos rozado la superficie, descubriendo cosas nuevas como la gestión de carpetas y archivos mediante PHP o las infinitas posibilidades que aporta el CSS.

También hemos apreciado que el tiempo es limitado, y que es muy complicado abarcar todo lo que pretendíamos en un primer momento, dándonos cuenta de que nos han faltado muchas ideas que teníamos previstas para este proyecto. Nos hubiese encantado haber incluido más contenido de las asignaturas de **redes, seguridad o administración de sistemas** y siempre encuentras una nueva idea o mejora que añadir pero para ello hubiésemos necesitado unos plazos mayores, algo que por otro lado entendemos que se establece para que aprendamos a medir el tiempo del que disponemos.

En conclusión, estamos felices por el resultado obtenido, por ver que el esfuerzo durante estos meses se ha visto reflejado en el proyecto desarrollado y que a pesar del tiempo usado en rehacer apartados del mismo al final han servido para imprimirle nuestro toque personal. Por todo ello creemos que ha sido una experiencia bonita e importante para el nuestro desarrollo tanto profesional como personal.

9. Anexo: Plan de empresa

Índice

1. Presentación de la empresa y resumen del proyecto

2. Presentación de los promotores

3. Análisis de la idea de negocio

4. Plan de marketing

5. Plan de producción y calidad

6. Plan de organización

7. Área jurídica

8. Estudio económico y financiero

9. Anexos

1.- PRESENTACIÓN DE LA EMPRESA

En **eYe ay ay S.L.** tenemos como objetivo principal que la lectura llegue a todo el mundo, de la forma más económica y cómoda posible a través de un portal online. La distribución de los libros se realizaría mediante el alquiler o préstamo por tiempo o la compra permanente de los mismos. Para ello pediremos al cliente cierta información personal como su número de tarjeta de crédito así como su DNI, con el fin de comprobar su autenticidad, todo ello a través de una conexión cifrada para garantizar la seguridad de los datos del cliente. Además el cliente recibirá un correo con el comprobante de cada alquiler efectuado, con el fin de evitar posibles errores en los mismos.

Nuestra empresa tendrá como sede las instalaciones COWORKING de IBERCENTER en la Plaza de Carlos Trías Bertrán 4 (inmueble Holiday Inn). Al usar una empresa que nos permite asentarnos en el mismo complejo que otras, podemos abaratar costes en el alquiler y tenemos la posibilidad de contactar con facilidad con otras empresas. Por último, la sede está situada en el centro financiero de Madrid, por lo que las posibilidades de negocios son mayores.

eYe ay ay se centrará en la realización de servicios propios de una biblioteca prestando los mismos a través de la red. El modelo de negocio se centra en la previsualización y alquiler de libros electrónicos durante un tiempo limitado con el fin de, por un lado, reducir costes y por otro, bajar los precios prestando el mismo servicio que una biblioteca normal pero más económico para el cliente.

Nuestra razón de ser es llegar a todos los amantes de la lectura, sin importar su poder adquisitivo ya que, gracias al alquiler de libros, todo el mundo será capaz de acceder a nuestro amplio contenido por un precio muy asequible.

Nuestras necesidades financieras totales para poner en marcha nuestro proyecto ascienden a 100.000€, que englobarán los gastos iniciales del alquiler del local y los contratos con los proveedores.

Contamos con el apoyo de varios proveedores que nos proporcionarán los equipos y las correspondientes editoriales. En cuanto al departamento de informática, destaca **Raspberry Pi Corporation** para nuestros equipos informáticos y amplio grupo de editoriales que nos ofrecen sus libros en versión digital, como por ejemplo **Alfaguara, Salamandra, Planeta,...** En cuanto al software, utilizaremos software libre como **Ubuntu, Wordpress, mySQL**, etc

Nuestra política de calidad se basa en la norma ISO 9001-2008 . Para más información sobre **eYe ay ay** puede visitar nuestra web corporativa www.eyeyay.es/info

2. Presentación de los promotores

Nuestra empresa contará en un principio con tres promotores siendo a su vez socios iniciales.

Fernando García García, diplomado en Enfermería por la Universidad de Alcalá de Henares, ha trabajado en Prosegur y en el sector servicios(el 73) durante 1 y 10 años respectivamente. Actualmente está cursando segundo curso de Administración de Sistemas Informáticos en Red. Dispone del certificado Cisco CCNA y CCNAS además de certificados Oracle MySQL. Desempeñará funciones de responsable de seguridad y desarrollo de software y administración de Bases de datos.

Sergio Garrido Escobar dispone de dos años completos de Ingeniería Técnica Informática de Sistemas en la UPSAM(Universidad Pontificia de Salamanca en Madrid) y actualmente cursa segundo curso de Administración de Sistemas Informáticos en Red. Cuenta con los certificados MCITP y MCSA de Administración de Sistemas Windows Server 2003 y 2008 además de los certificados Cisco CCNA y CCNAS además de certificados Oracle MySQL. Además cuenta con el título de inglés CAE(Certificate in Advanced English) de la Universidad de Cambridge. Se responsabilizará de la administración de sistemas en entornos Linux y Windows, además de relaciones comerciales e internacionales.

Ana Guillem de la Fuente está cursando actualmente segundo curso de Administración de Sistemas Informáticos en Red. Cuenta con el certificado Cisco CCNA y CCNAS además de certificados Oracle MySQL y amplios conocimientos de programación en java. Será la encargada del desarrollo y soporte del entorno web de **eYe ay ay**.

Además, los tres socios poseen experiencia en el mundo de la informática debido a los tres meses de prácticas en empresas de desarrollo de software y mantenimiento de hardware y a un proyecto en conjunto de desarrollo de páginas web a medida.

Los tres socios aportaremos un 33,33% de capital a la empresa, por tanto, el grado de implicación en el proyecto es equitativo.

3. Análisis de la idea de negocio

DEBILIDADES:

1. No disponemos de experiencia en el sector de la venta y/o alquiler de libros, ya sea en formato físico o digital.
2. Posibles fallos de seguridad en la infraestructura ya que el mundo de la seguridad informática está en constante evolución por lo que somos susceptibles a probables ataques.

FORTALEZAS:

1. Los tres socios tenemos una amplia experiencia en el mundo de la informática por lo que el desarrollo de las aplicaciones y el soporte informático de la empresa no supondrán ninguna dificultad y podremos plasmar nuestras ideas tal y como han sido concebidas, así como añadir futuras modificaciones a nuestro proyecto.
2. El servicio de alquiler online aporta comodidad y facilidades a nuestros clientes, ya que mediante nuestra web pueden acceder al contenido desde cualquier lugar y cualquier momento.
3. Alta competitividad gracias a los bajos precios ya que, al contrario que **Amazon, Google Books y iTunes**, nosotros ofrecemos alquileres en vez de la compra del producto.

AMENAZAS:

1. Existe una fuerte competencia en el sector que ofrecen un producto similar, sobre todo por el oligopolio formado por **Amazon, iTunes y Google Books**.
2. Posibles pérdidas económicas provocadas por el aumento de piratería en nuestro país como queda reflejado en diversos estudios y artículos como los siguientes:
 - [El sector del libro dejó de ganar 350 millones por la piratería](#)
 - [El 84% de los contenidos culturales que se adquieren en España son piratas](#)
 - [BSA alerta del aumento de la piratería en España](#)

OPORTUNIDADES:

1. Amplio rango de clientes debido a que la lectura es una afición que abarca todo tipo de personas gracias a su gran variedad de temas.
2. Gracias a la aprobación de la nueva **Ley de Propiedad intelectual** y la nueva Directiva (2008/8/EC) por la cual **cambian los impuestos aplicados sobre los bienes digitales**, estaremos en igualdad de condiciones frente a **Amazon, iTunes y Google Books**, ya que estas estarán obligadas a pagar impuestos de sus productos en el país en el que venden al contrario de pagar dónde más facilidades tenían. Por tanto, pasarán del 15% al 21%, viéndose reflejado en sus precios finales.
3. Gracias a la Comunidad de Madrid disponemos de una línea de financiación destinada a la Modernización y promoción del comercio por **AVALMADRID**, con un 1,5% de interés, un

importe mínimo de 6000€ y máximo de 300000€ y una comisión de apertura del 0,5%.
Existen más detalles en el siguiente enlace:

http://www.avalmadrid.es/productos/plan-modernizacion-promocion-comercio_2_39.html

4. Plan de Marketing

-Marketing estratégico:

Se ha realizado un estudio de mercado con el objetivo de determinar en qué punto se encuentra nuestra empresa en el mercado y la estrategia a seguir. El estudio completo se incluye en el apartado anexo 1 de este documento.

Al ser una empresa de nueva creación nuestro objetivo sería alcanzar 7000 préstamos o alquileres en primer trimestre y una cuota de mercado del 5-7% para finales del 2015

-Marketing operativo:

PRODUCTO:

El servicio ofrecido a los clientes provendrá de un grupo de editoriales que nos aportarán los libros digitales, tanto clásicos como los más actuales. Ofreceremos a los usuarios un portal elegante, funcional, rápido e intuitivo para que estos puedan encontrar y disfrutar de sus libros fácil y cómodamente. Nuestro objetivo es que el cliente tenga a su disposición un gran catálogo de libros a un precio razonable mediante el alquiler y ofertas especiales (packs, sagas, best sellers,...).

PRECIO:

El precio final de nuestros libros viene se establece teniendo como referencia los precios de la competencia dando la sensación de que los libros son mucho más baratos que los de nuestros competidores. A continuación, mostramos ejemplos de servicio que ofrecemos cuyos precios estarán establecidos por debajo de los de la competencia con el fin de entender nuestra estrategia de precios:

Google Books:

Mundodisco(La Luz Fantástica): 5,49€

Mundodisco(Ritos Iguales): 4,99€

Mundodisco(Mort): 5,99€

Mundodisco(Brujerías): 5,99€

Mundodisco(Me vestiré de medianoche):5,99€

eYe ay ay (estimando unas 12 horas de lectura por libro):

Mundodisco(La Luz Fantástica): 0,3€/hora → 3,1€

Mundodisco(Ritos Iguales): 0.2€/hora → 3,72€

Mundodisco(Mort): 0,3€/hora → 3,1€

Mundodisco(Brujerías): 0,3€/hora → 3,1€

Mundodisco(Me vestiré de medianoche): 0,3€/hora → 3,1€

En cuanto al hardware necesario para la red de la empresa necesitaremos:

-3 Raspberrys 35€ = 105€

-1 switch (Cisco 2950 24) = 146,562€

-1 Router Linksys WRT1900AC= 235,95€

-3 monitores Samsung = 100€

-Cableado = 10€

-Teclados y ratones=90€

PROMOCIÓN:

Como gancho usaremos principalmente nuestros precios competitivos, lo que a su vez conducirá a una publicidad gratuita gracias al boca a boca de nuestros clientes. A esta estrategia sumaremos el lanzamiento de varias ofertas de libros muy demandados, así como packs de sagas o género a un precio muy atractivo para fomentar el interés de nuestros clientes. Por último realizaremos una campaña de marketing agresivo a través de redes sociales (Facebook, twitter, Instagram) además de anuncios en otras webs de tráfico alto y foros.

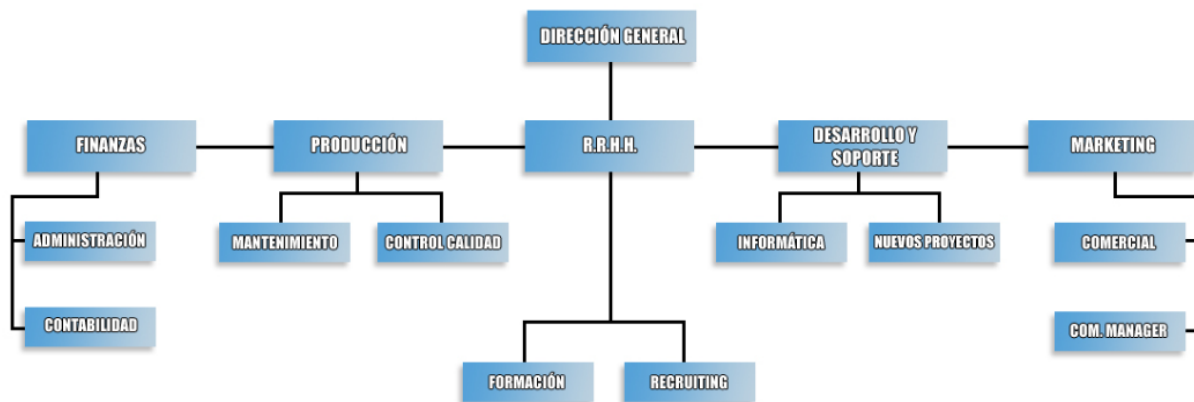
DISTRIBUCIÓN:

El servicio de atención al cliente será rápido y cercano, buscando una respuesta inmediata (máximo 5 minutos) ante cualquier problema que les surja ya sea a través de correo electrónico, como chat, llamada a su propio teléfono para que no les acarree coste alguno, etc.

El servicio de venta funcionará las 24h gracias a usar un sistema automatizado, mientras que la atención al cliente se realizará de 10:00h a 21:00h de Lunes a Viernes, y los Sábados de 10:00h a 14:00h.

Sergio Garrido será el responsable del departamento comercial de la empresa. Por último, al usar el formato digital no existen intermediarios entre la empresa y las editoriales.

5.- Plan de organización



Los tres promotores de la empresa se encargarán de la dirección de la misma.

El departamento de finanzas estará supervisado por Fernando García García y se encargará de manejar el capital de la empresa, realizar la contabilidad, la venta de productos y las relaciones con las entidades bancarias. Este departamento constará de tres trabajadores para cuyo puesto se requerirá la posesión del título de Grado en Administración y Dirección de Empresas. Dos de ellos estarán en la sección de Contabilidad, mientras que el tercero pertenecerá a la sección de Administración.

El departamento de producción estará supervisado por Sergio Garrido Escobar y se encargará de manejar la supervisión de los proyectos, así como el control de calidad de las aplicaciones. Para el puesto de jefe, serán requeridos conocimientos de desarrollo de software así como experiencia en análisis y optimización del código. Dicho departamento se dividirá en dos secciones (mantenimiento y calidad) que contarán con una persona en cada uno, ambos con experiencia en desarrollo de software, especialmente php, html y mysql.

El departamento de RRHH estará supervisado por Ana Guillem de la Fuente y se encargará de las contrataciones, el control de las altas y bajas y la formación de los empleados. Dos personas estarán en este departamento, uno para la sección de Formación, que deberá tener el Grado en Ingeniería Informática y otro en Recruiting que deberá tener el Grado en Relaciones Laborales y Recursos Humanos

El departamento de desarrollo y soporte estará supervisado por Ana Guillem de la Fuente y se encargará de las innovaciones que se quieran realizar en la empresa, así como la atención al cliente. Este departamento constará de una plantilla de tres trabajadores, en la sección de Informática habrá dos trabajadores los cuales tendrán una titulación de Grado Superior en Administración de Sistemas en Red, y en Nuevos Proyectos tendremos un trabajador con una titulación de Grado Superior en Desarrollo y Aplicaciones Web.

El departamento de marketing estará supervisado por Sergio Garrido Escobar y se encargará de mostrar al público una imagen de la empresa que atraiga clientes. Llevará a cabo la gestión publicitaria en redes sociales. Será necesaria una titulación de grado superior en Marketing y Ventas y dos personas se encontrarán en dicho puesto, una como Comercial y la segunda como Community Manager.

8.- Estudio Económico

CAPITAL SOCIAL	100.000 €
ALQUILER OFICINA 12 meses	24.000 €
HARDWARE	1067,512 €
SOFTWARE	5.000 €
DEUDA EDITORIAL 1 a 2 años	70.000 €
DEUDA EDITORIAL 2 a 6 meses	8.000 €
DEUDA EDITORIAL 3 a 2 meses	5.000 €
CUENTA ABIERTA EN BANCO A	40.000 €
CUENTA ABIERTA EN BANCO B	20.000 €

BALANCE SITUACIÓN 27/02/2015 ACTIVOS: 90067,512 € PASIVOS: 83.000 €

PREVISIÓN ANUAL:

Ingresos:

1. **Beneficios obtenidos por alquiler de libros (cifra de negocio): 650.000€**
2. **fondo de inversión: 100.000 €**

Costes:

1. **Gastos de personal y Seguridad Social: 250.000 €**
2. **Gastos en Luz, Teléfono e Internet: 3000 €**
3. **Impuesto de sociedades: 30%**

BALANCE:

ACTIVO NO CORRIENTE:

- Alquiler Oficina
- Hardware
- Software

ACTIVO CORRIENTE:

- Cuenta Abierta en Banco A
- Cuenta Abierta en Banco B

PATRIMONIO NETO:

- Capital Social

PASIVO NO CORRIENTE:

- Deuda Editorial 1 a 2 años

PASIVO CORRIENTE:

- Deuda Editorial 2 a 6 meses
- Deuda editorial 3 a 2 meses

CUENTA DE RESULTADOS:

- a. Resultado de explotación: $650.000 - 250.000 = 400.000 \text{ €}$
- b. Resultado Financiero: 100.000 €
 - i. Ingresos Financieros: 100.000 €
 - ii. Gastos Financieros: 0 €
- c. Resultado antes de impuestos: $400.000 + 100.000 = 500.000 \text{ €}$
 - i. Con impuesto de sociedades: $500.000 * 0,3 = 150.000 \text{ €}$

RENTABILIDAD FINANCIERA: $150.000 / 100.000 = 1,5\%$

RENTABILIDAD ECONÓMICA: $400.000 / 500.000 = 0,8\%$