**Universidad Nacional Autónoma de México**

**Facultad de Contaduría y Administración**

**“Sitio dinámico Biblioteca”**

**Proyecto Final**

**Presenta:**

**Martínez Rodríguez Oscar Mauricio**

**Profesora:**

**Angie Aguilar**

México, Ciudad de México Junio,2021

Contenido

[Introducción 3](#_Toc75212730)

[Pantallas del programa módulos 5](#_Toc75212731)

[Hoja de estilos CSS 13](#_Toc75212732)

[Javascript 15](#_Toc75212733)

[Pantallas de PostgreSQL 15](#_Toc75212734)

[Pagina en funcionamiento 17](#_Toc75212735)

[Conclusiones 22](#_Toc75212736)

## Introducción

**Planteamiento del problema**

El problema o hipótesis principal del proyecto es hacer un sitio dinámico, con PHP e interacción con base de datos. La hipótesis o problema principal es un sitio dinámico con una conexión a base de datos, al saber esto se empezó trabajando en los módulos para satisfacer la necesidad del cliente. Por su parte el cliente nos dio los siguientes requerimientos:

|  |  |  |  |
| --- | --- | --- | --- |
| Lista de requerimientos | | | |
| ID | **Tipo** | **Descripción** | **Fecha** |
| 1 | Calidad | Configurar un nuevo VirtualHost: biblioteca.conf | 28/05/2021 |
| 2 | Calidad | Nombre del sitio: www.biblioteca.unam.mx | 28/05/2021 |
| 3 | Calidad | Directorio raíz: /var/www/biblioteca | 28/05/2021 |
| 4 | Calidad | Bitácora: biblio-error.log, biblio-access.log | 28/05/2021 |
| 5 | Calidad | Dar de alta un nuevo repositorio público: ProyectoFinal. | 28/05/2021 |
| 6 | Calidad | Incluir el archivo: readme.md - Indicar que es un proyecto escolar. | 28/05/2021 |
| 7 | Calidad | Agregar todos los archivos del proyecto al repositorio y enviar los cambios a GitHub. | 28/05/2021 |
| 8 | Calidad | Enviar el enlace del repositorio a: angie.aguilar.dominguez@gmail.com | 28/05/2021 |
| 9 | Funcionalidad | index.php: nombre de la página, tabla con los libros ordenados por autor, empezando por apellido paterno, con enlace para editar (form\_edita.php) o eliminar (form\_elimina.php) los libros. | 28/05/2021 |
| 10 | Funcionalidad | creditos.php: nombre de quién elaboró. Ver ejemplo: https://www.unam.mx/creditos | 28/05/2021 |
| 11 | Funcionalidad | form\_libros.php: formulario de alta de un libro, considerar los siguientes elementos: nombre del libro, edición (primera edición, segunda, etc), editorial, autor (nombre, apellido paterno, apellido materno). | 28/05/2021 |
| 12 | Funcionalidad | alta.php: archivo en PHP que recibe los datos y los guarda en la BD. | 28/05/2021 |
| 13 | Funcionalidad | form\_edita.php: formulario donde se pueden visualizar los datos del libro para cambiarlos y volver a guardar, envía los datos a guarda\_edicion.php. | 28/05/2021 |
| 14 | Funcionalidad | guarda\_edicion.php: archivo en PHP que recibe los datos y los guarda en la BD. | 28/05/2021 |
| 15 | Funcionalidad | form\_elimina.php: recibe el ID del libro que se desea dar de baja y pide confirmación para eliminar el registro, envía la confirmación a elimina\_libro.php | 28/05/2021 |
| 16 | Funcionalidad | elimina\_libro.php: elimina el registro de la base de datos.  Código fuente bajo control de versiones en el repositorio. | 28/05/2021 |
| 17 | Funcionalidad | Tabla para: autores, libros, editoriales. Recordar: un autor puede escribir varios libros. Un libro puede tener uno o más autores. Una editorial puede editar varios libros. | 28/05/2021 |
| 18 | Funcionalidad | DBO: bibliotecario (Insertar, actualizar, borrar, consultar).  El script de creación e inserción de registros en la BD también se debe incluir en el repositorio. | 28/05/2021 |
| 19 | Calidad | Documentación:   * Portada, tabla de contenidos * Planteamiento del problema (Introducción) * Herramientas empleadas (Desarrollo, incluir versiones empleadas de: Apache, PostgreSQL, etc.) * Solución del problema (Desarrollo, incluir capturas de pantalla de flujo para: alta de autor, alta de libro, etc.) * Conclusiones, mejoras a futuro | 28/05/2021 |
| 20 | Calidad | La documentación también se debe incluir en el repositorio. | 28/05/2021 |
| 21 | Usuario | Inicio y cierre de sesión. | 28/05/2021 |
| 22 | Calidad | Hoja de estilos (responsivo). | 28/05/2021 |
| 23 | Funcionalidad | JavaScript en validaciones (formularios). | 28/05/2021 |

**Herramientas empleadas**

**Versiones**

Sistema operativo Debian 10

Apache2: Apache 2.4.38 (Debian)

SQl: Debian (11.10.0)

PHP 8.0.7

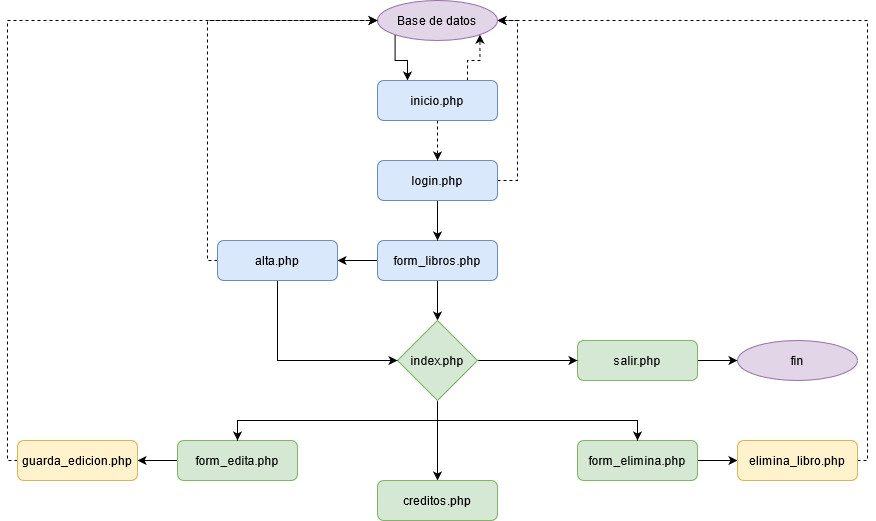
HTML5

CSS3

Javascript 1ª edición – ECMAScript 2020

**Diagrama de flujo del programa**

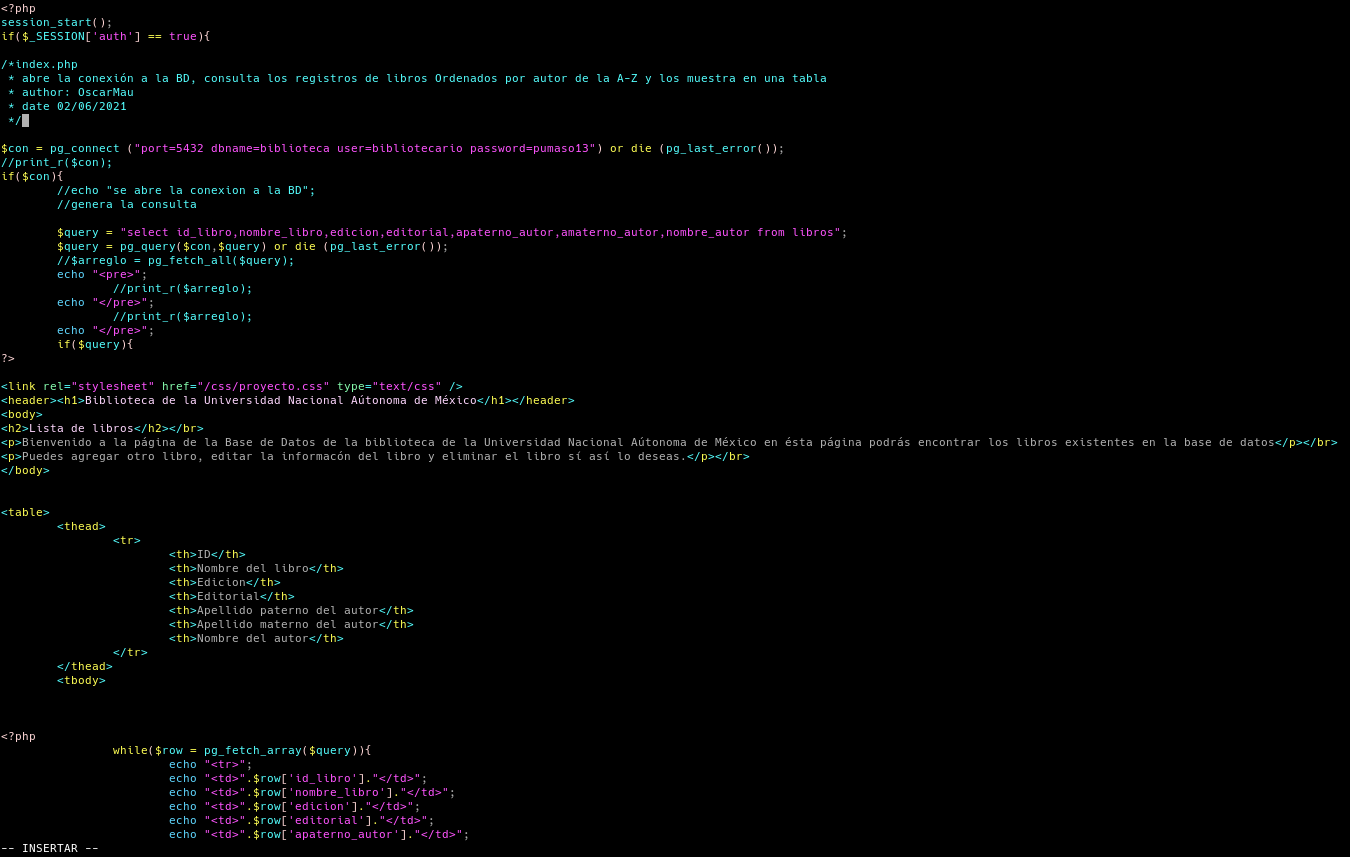
El siguiente programa expresa de una manera gráfica como es que trabaja el sitio web de una forma general. Se puede observar cual es la dirección y orden del sitio con las flechas sólidas, y con las flechas punteadas podemos observar los módulos que interactúan con la base de datos.



# Pantallas del programa módulos

**Módulo Index.php**

El primer módulo en crearse fue el módulo de index.php ya que esté muestra los libros de la base de datos y también en esté podemos ir a los módulos de edición o eliminación, y también al módulo de registrar más libros. En este módulo lo primero que se hace es conectarse a la BD de Biblioteca con el usuario bibliotecario con su respectivo password. Después de esto se hace una selección de los valores en la tabla libros e imprime en una tabla los valores seleccionados de la tabla libro, en una tabla hecha con HTML para visualizar los datos que hay en la BD



En la segunda pantalla podemos observar, que lo que hace es llamar los valores seleccionados de la BD para acomodarlos en la tabla HTML creada previamente. Además, agrega dos enlaces en la misma tabla ya sea para editar o eliminar registro. Un poco, más debajo de puede observar también que se agregan dos enlaces para registrar otro libro o para salir de la sesión, también se agrega un if por sí es que no se puede conectar a la BD o si hubo un error en una sentencia SQL. Hasta abajo se encuentra una referencia a inicio.php ya que para entra a esta parte se necesita iniciar sesión primero.



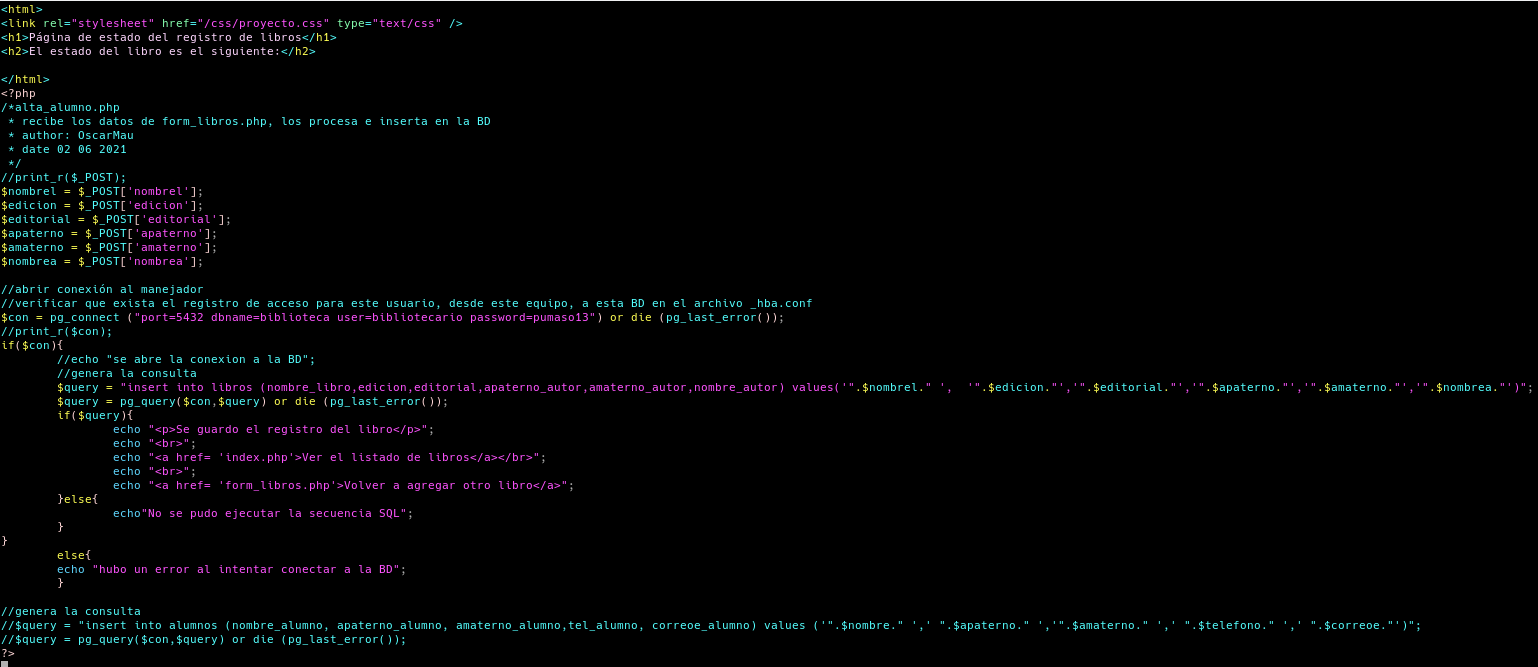
**Módulo form\_libros.php**

En este modulo es donde se abre sesión para agregar un libro y de aquí se encuentra un enlace para ir a index.php. En la primera parte esta el apartado para iniciar sesión que nos redirige a inicio.php, y si el inicio resulta exitoso podremos visualizar la parte HTML del código la cual es un pequeño formulario donde se podrán agregar los datos del libro que tiene la BD. Estos datos del formulario son enviados a alta.php para que los datos escritos o agregados en este apartado sean agregados a la BD en el modulo de alta.php. Al final se encuentra un enlace a index.php para regresar al inicio de la página.



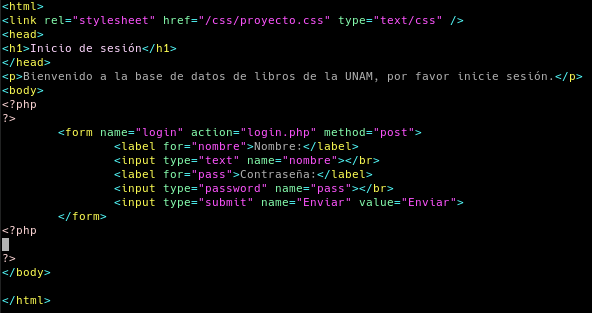
**Módulo alta.php**

En este modulo se reciben los datos del formulario para que se guarden directamente en la base de datos acuerdo a su orden, y en la BD se le asigna un id a cada libro o cadena de tos registrada. En la primera parte de este módulo se manda a llamar a la hoja de estilos del proyecto la cual es proyecto.css, se coloca el titulo del modulo y una breve descripción de él estado del libro. Continuando abajo con el código se mandan a llamar los campos en donde están los valores para pasarlos por el método POST a la BD. Y se abre la conexión a la base de datos para que esta reciba los datos capturados por el método POST. Se hace in INSTERT para agregar o insertar los valores a la tabla de libros en la BD y así llenar los campos y en automático incremental se les va a asignar su id a cada registro o conjunto de datos registrados. Un poco mas abajo se encuentran los links a index.php y form\_libros, así como los ELSE que actúan en caso de que no funcione la conexión a la base de datos.



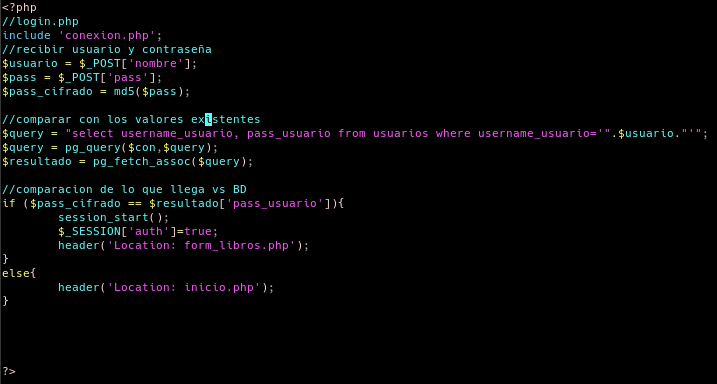
**Módulo Inicio.php**

El propósito de este modulo es principalmente hacer la pantalla de inicio de sesión con un pequeño formulario que recibirá los datos de usuario y contraseña para validarlos y compararlos con los que se encuentran registrados en la BD. La mayoría de este módulo esta hecho en HTML; y como se pude ver en un principio se le hace un llamado a la hoja de estilos proyecto.css en la carpeta css para que se apliquen los estilos y color al módulo, después encontramos el titulo y una pequeña leyenda. Continuando abajo encontramos el formulario que recibe los datos y los envía al módulo login.php, para que valide los datos enviados.



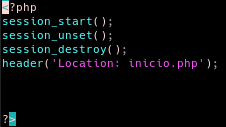
**Módulo login.php**

Este modulo se encarga de dar o denegar el acceso a la sesión acorde a los datos que recibe por el método POST del formulario de inicio.php. En la primera parte podemos encontrar que se incluye el módulo conexión ya que este modulo es necesario para conectarse a la BD, pero por otro lado se puede hacer la conexión en les mismo modulo como lo vimos en los módulos padsados. Continuando se reciben los datos del formulario de inicio.php por el método POST, posteriormente se hace una selección en los campos de user\_name y pass\_usuario de la tabla de usuarios del usuario que recibe por el método POST. Después se hace una comparación y asociación con el dato recibido de usuario para posteriormente comparar el pasword cifrado en la BD, cabe mencionar que es un cifrado md5, y posteriormente se compara con el pasword recibido el formulario, esto atraves de un if que nos dará el acceso o nos lo negara depende si los password son iguales. Si es que son iguales nos da el acceso al modulo de form\_libros.php y si no lo son nos regresa a inicio.php.



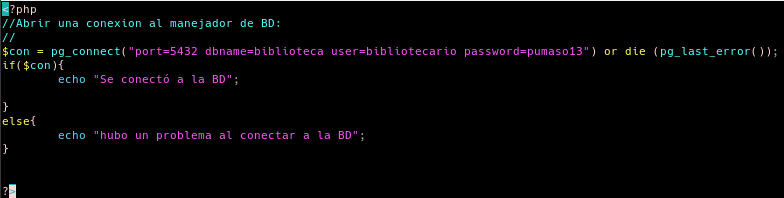
**Módulo salir.php**

Este modulo es llamado desde index.php y se utiliza para salir de la sesión abierta actualmente.



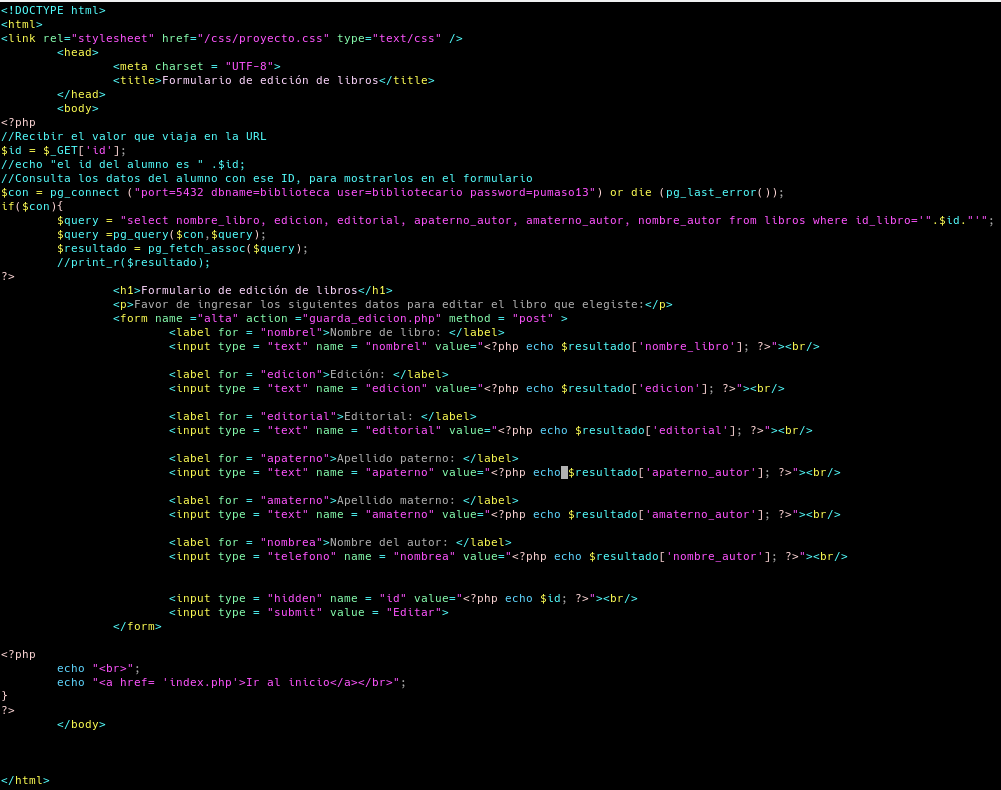
**Módulo conexión.php**

Este modulo es llamado por login.php para abrir una conexión a la base de datos por medio de un if el cual nos dará acceso a la misma o nos notificara que no se pudo conectar a la BD.



**Módulo form\_edita.php**

Lo que hace este modulo en una forma resumida es editar cualquier registro seleccionado que ya se encuentre en la BD. En la primera parte podemos observar la referencia para aplicar la hoja de estilos proyecto.css y debajo de esto se aplica la codificación UTF-8 para que se puedan interpretar los acentos y símbolos del teclado. Se puede ver en la parte de abajo del código que se utiliza el método GET para enviar la información a través de la propia URL, para después hacer una conexión con la BD y seleccionar los datos del libro o registro que queramos editar. Para editar los campos estos se muestran en un formulario HTML para editarlos campo por campo, al final se puede ver la referencia a index.php.



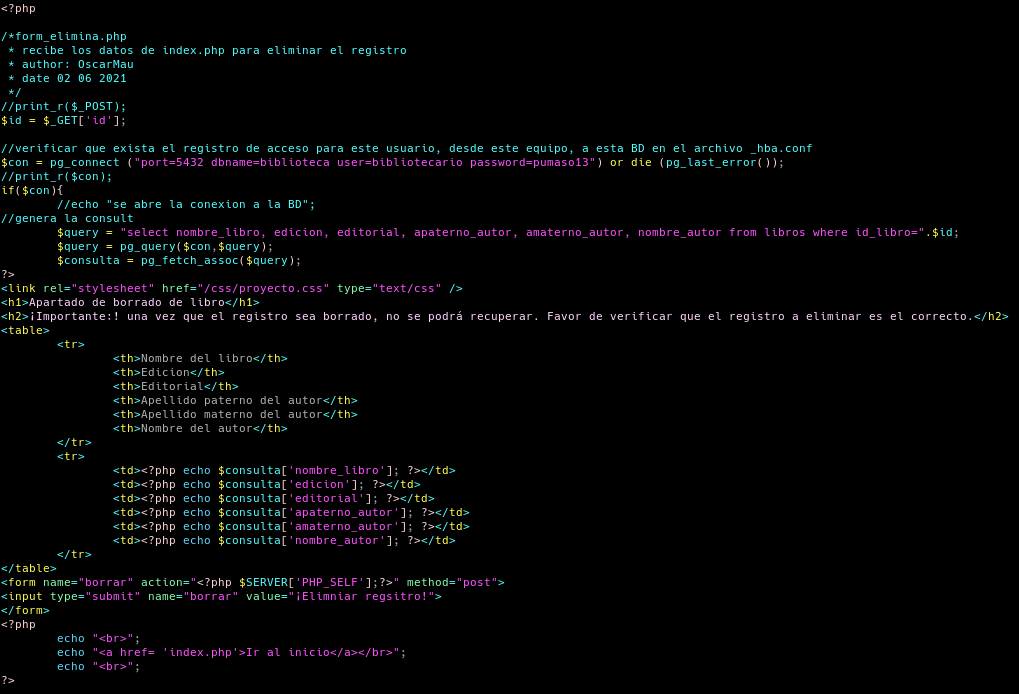
**Módulo guarda\_edición.php**

Lo que hace este modulo en pocas palabras es recibir los parámetros de form\_edita.php y los actualiza en la BD. En la parte superior se puede observar la referencia a la hoja de estilos proyecto.css para que aplique el estilo en el módulo. Para después abrir un apartado de código PHP y se envían por el método POST los cambios escritos en los campos de form\_edita.php. Después de esto se abre la conexión a la BD y después un if que hace un UPDATE en los campos de la tabla libros y sean efectivos los cambios. Si es que se pueden actualizar los cambios dentro de la BD se hace referencia a index.php y a form\_libros.php para salir del módulo. Si no se puede manda un aviso de error.

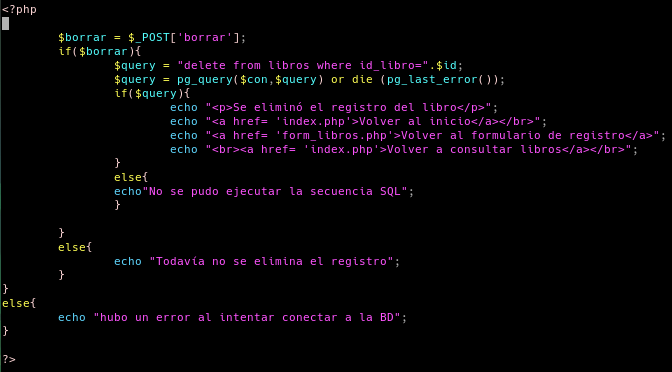


**Módulo form\_elimina.php**

La función de este modulo en resumen es eliminar el registro del libro seleccionado en la BD. En la primera parte del código se mando por el método GET el ID para identificar el registro. Poco después se hace una conexión a la BD para con un if hacer un SELECT sobre los campos con la referencia del ID que se recibe por el método GET, el ID recibido y sus campos son seleccionados de la tabla libros y se muestran en una pequeña tabla HTML para visualizar el registro que va a ser eliminado. Debajo de las consultas se manda a llamar una acción de borrado con el método POST que enviara los valores seleccionados a que sean eliminados y al final del código en la primera parte se hacer referencia a index.php para volver al inicio.



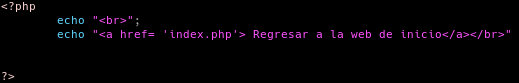
Continuando con el modulo de form\_elimina en esta segunda parte del código por el método POST se llama form borrar y se mete a un if que hace una query para borrar de la tabla libros el registro que se identifica por el ID, Si es que se borra el libro o registro del mismo sale un mensaje que fue borrado el libro junto con la referencia a index.php y form\_libros.php. Si es que no se pudo salen mensajes notificando el error.



**Módulo créditos.php**

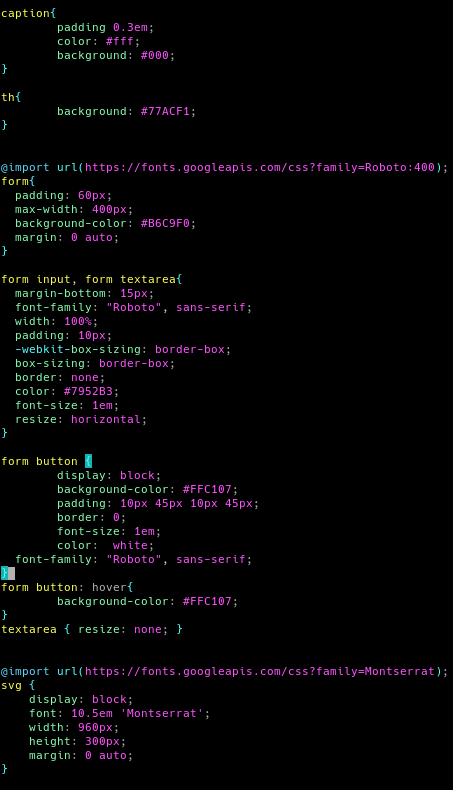
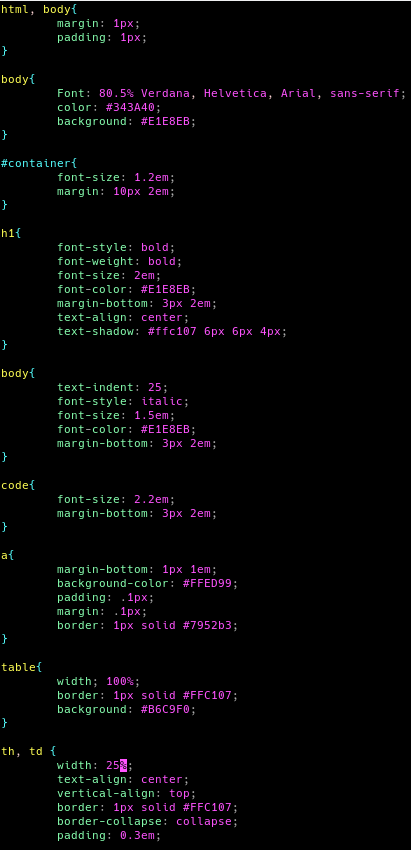
En este modulo

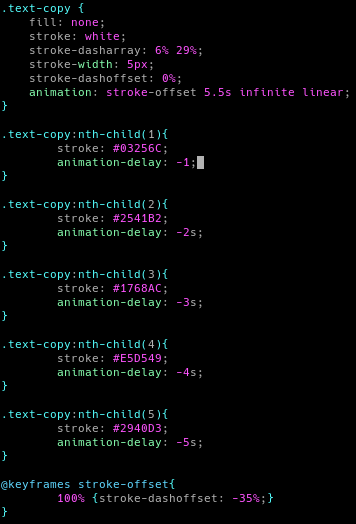




# Hoja de estilos CSS

En esta hoja de estilos





# Javascript

Primero.js



Alta.js



Edita.js



Elimina.js



# Pantallas de PostgreSQL

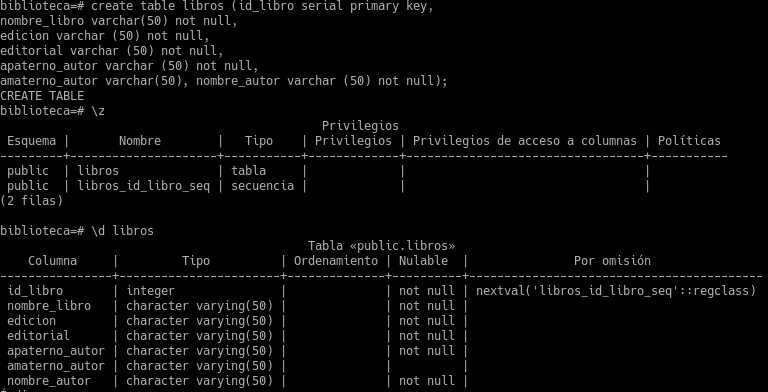
Pantallas de creación del usuario bibliotecario con su contraseña



Pantalla de creación de la base de datos Biblioteca con el usuario bibliotecario como dueño de esta. Además, se hace la conexión a la nueva base de datos creada.



Pantalla de la creación de la tabla libros. Con todos sus valores. Además, se revisan los privilegios las tablas que hay en la base de datos con el comando \z. Y se hace una revisión de los campos de la tabla libros.



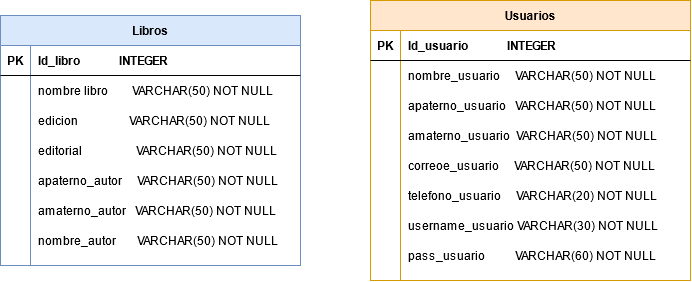
Pantalla de la creación de la tabla Usuarios junto con todos sus valores en la base de datos Biblioteca. Se hace una revisión del listado de relaciones de las tablas en la Base de datos y también una revisión de los privilegios de usuario sobre estas.



Se le asignan los privilegios de insertar, seleccionar, actualizar y borrar en la tabla de libros y usuarios al usuario bibliotecario. Se le asignan el permiso de uso a la tabla de usuarios\_id\_usuario\_seq al usuario bibliotecario. Y finalmente se revisa con el comando \z que los privilegios se hallan aplicado.



Diagramas entidad relación de las tablas en PostgreSQL

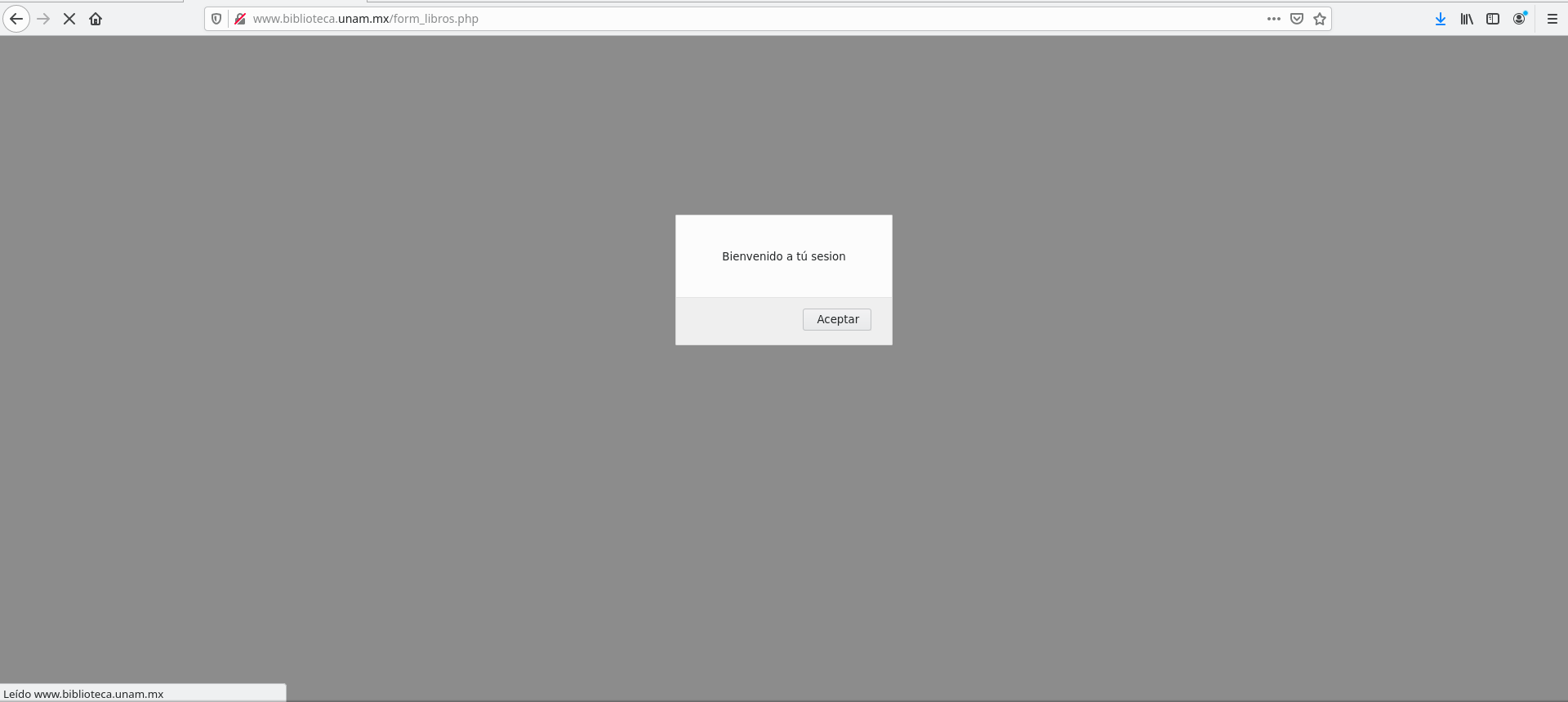


# Pagina en funcionamiento

**Página inicio.php**



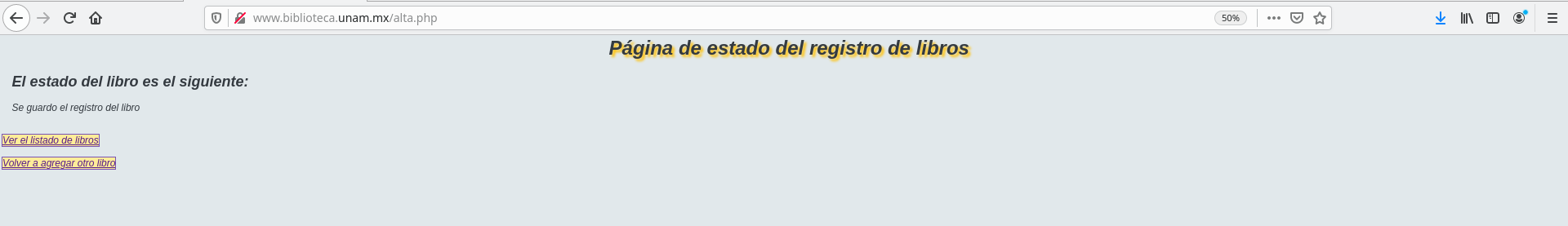
**Página form\_libros.php**





**Página alta.php**

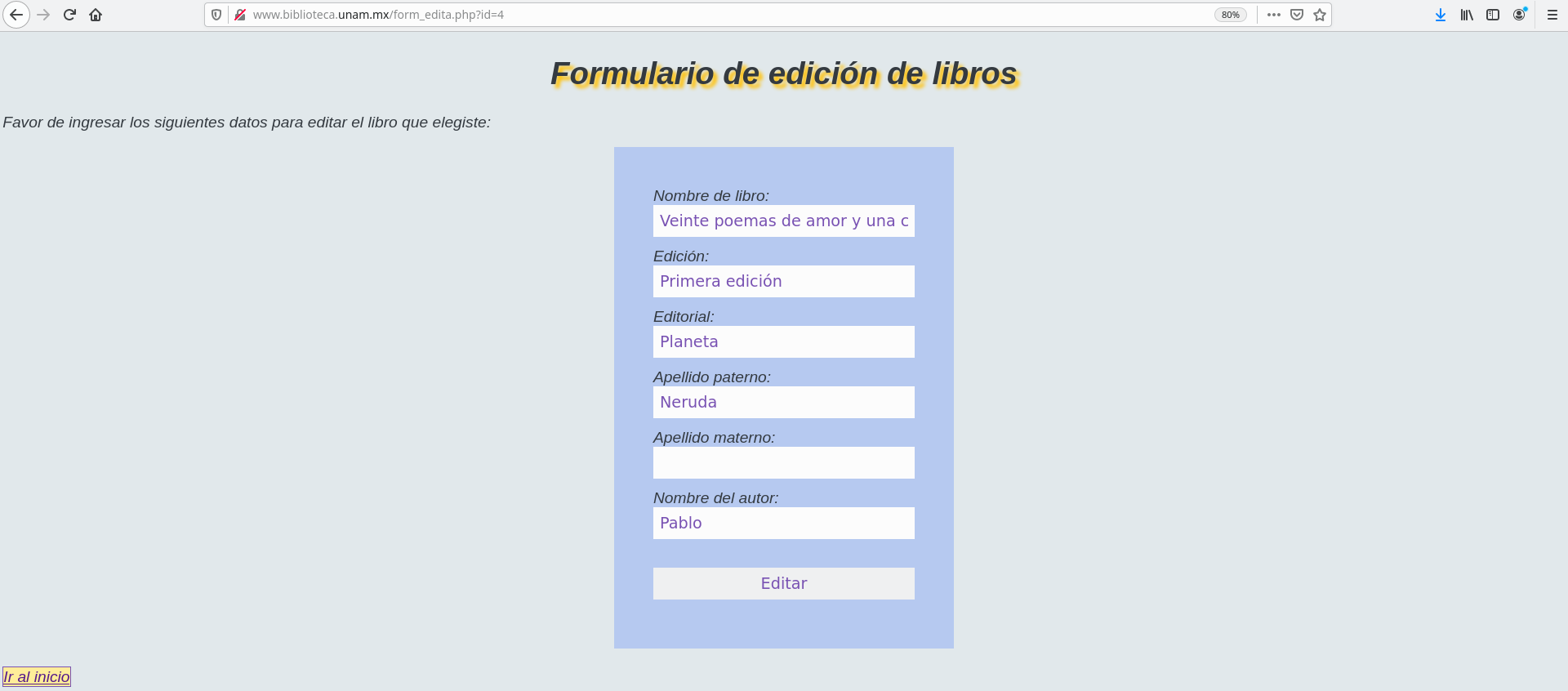




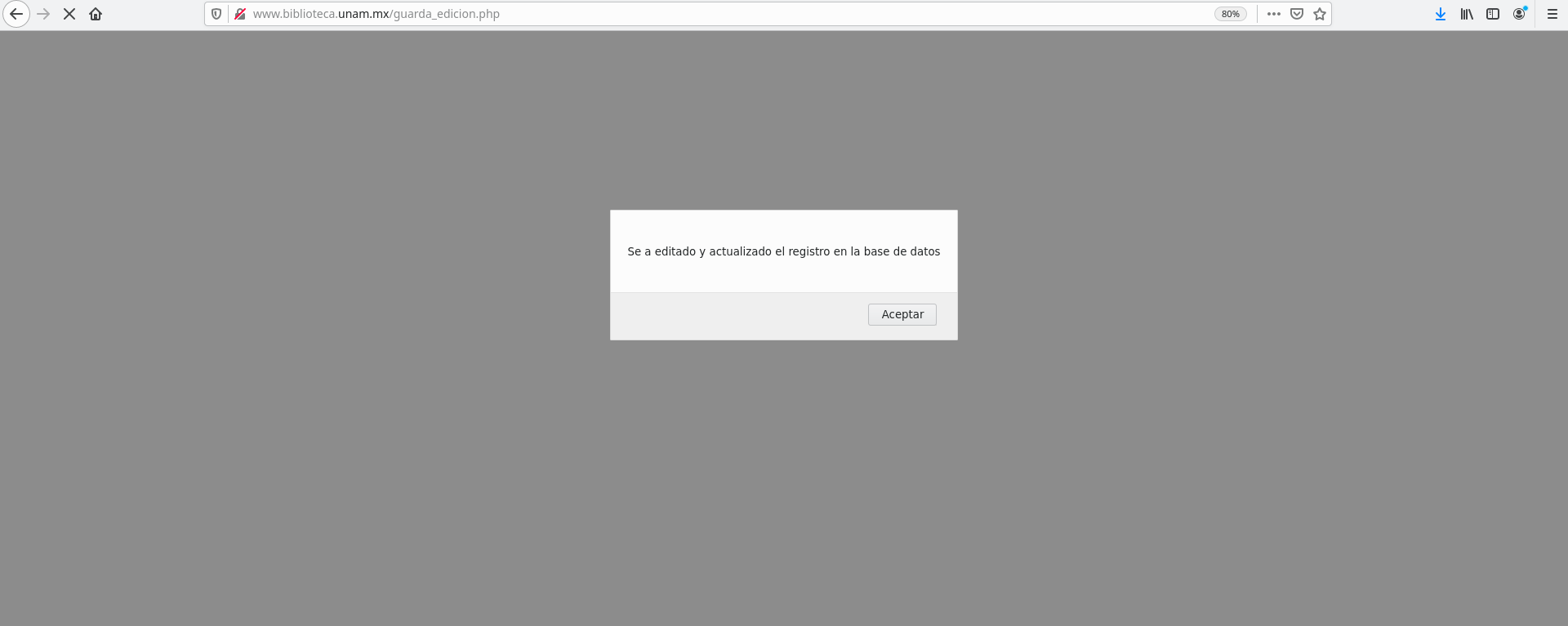
**Página index.php**

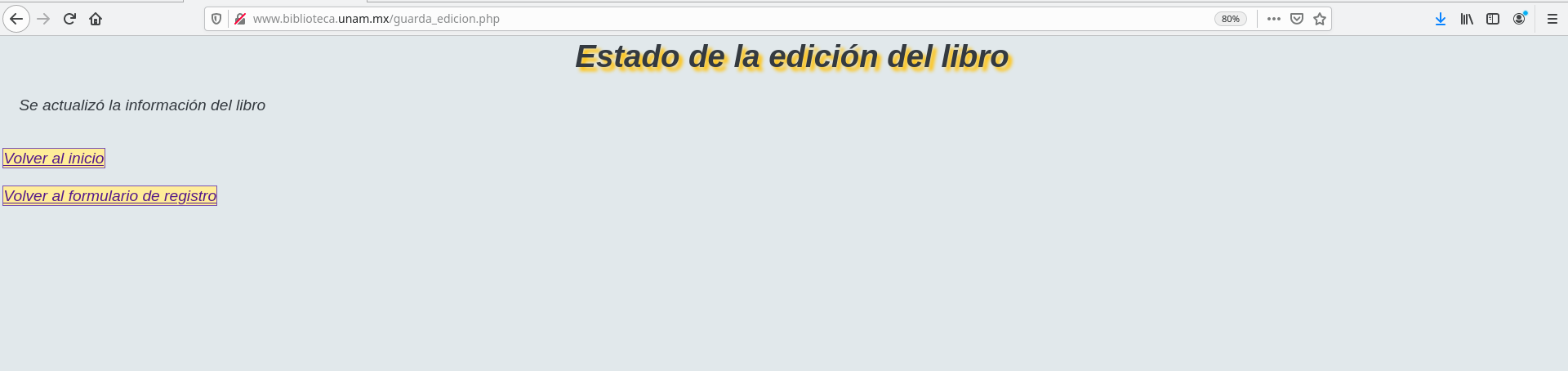


**Página form\_edita.php**



**Página guarda edicion.php**



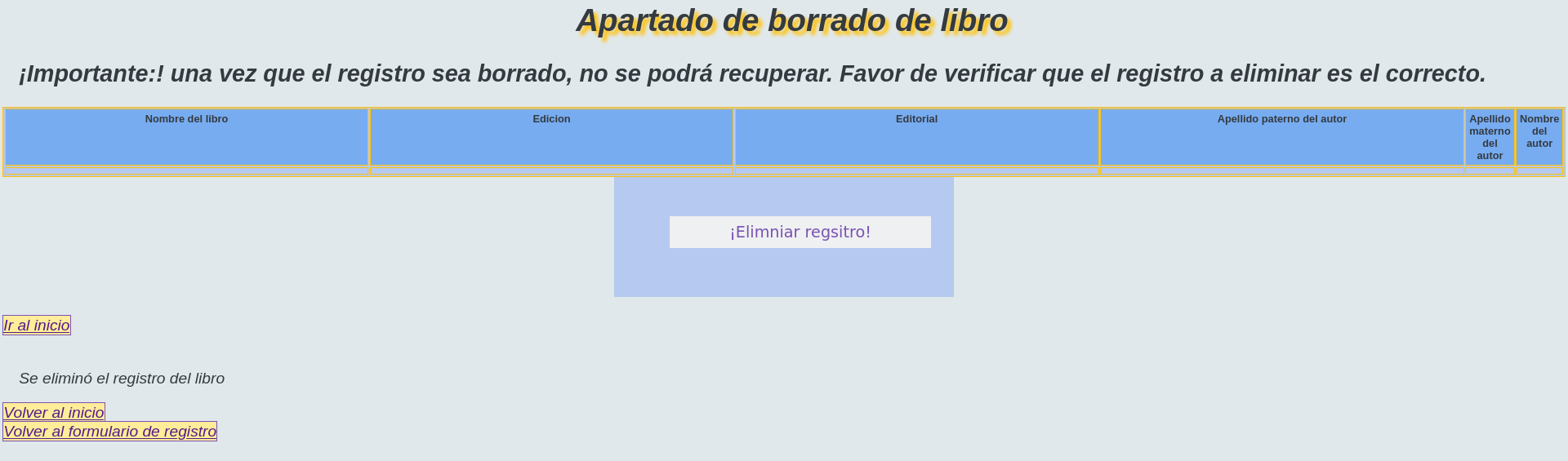


**Página form\_elimina.php**

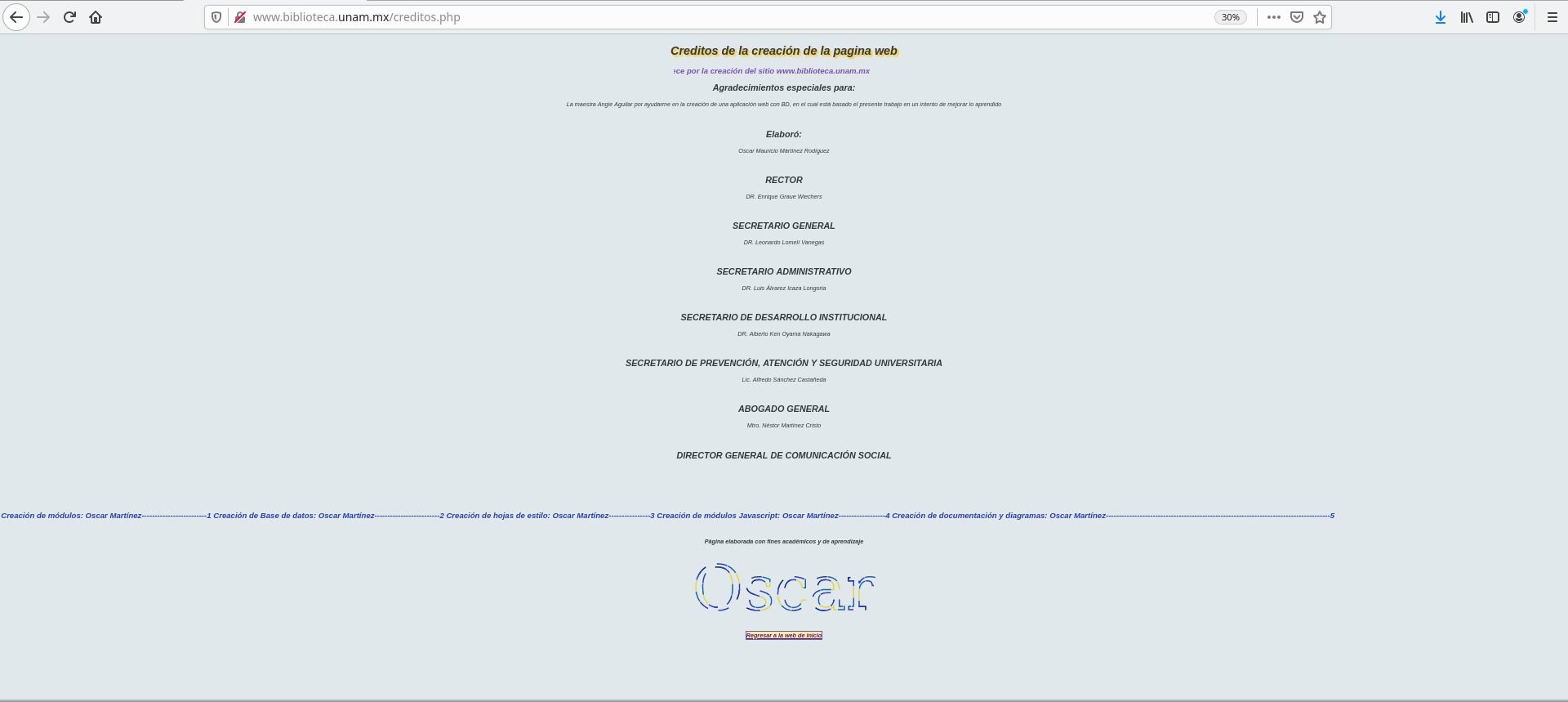








**Página creditos.php**



# Conclusiones

En la entrega final del proyecto se puede afirmar que se entrega un proyecto funcional apegado a los estándares del cliente y buscando siempre la satisfacción del mismo. Se entrega el sitio [www.biblioteca.unam.mx](http://www.biblioteca.unam.mx) completamente funcional en todos sus módulos, así como con los requisitos solicitados por el cliente, así como algunas pequeñas mejoras en diseño, funcionalidad e interactividad con el usuario.

**Mejoras a futuro**

El presente proyecto podría mejorarse de varias formas dependiendo de la necesidad del cliente, la institución y la biblioteca, a partir de estos puntos podemos analizar los diferentes puntos de vista de los usuarios y clientes que pueden o van a utilizar la plataforma de la biblioteca de la UNAM.

Podríamos decir que podemos mejorar los siguientes aspectos con base a los diferentes puntos de vista analizados anteriormente se sugiere:

* Implementación de un buscador de libros por: autor, nombre del libro y editorial.
* Modulo de ubicación de libro: se puede implementar un apartado entro del sitio para que te arroje la ubicación del libro dentro de la misma biblioteca
* Creación de un usuario alumno para que pueda consultar o buscar un libro en la BD.
* Mejora de front-end del sitio a través de le implementación de carruseles de imágenes, encabezados y pies de paginas con enlace a los contactos de la biblioteca.
* Crear un apartado con la descripción del libro para saber más de este a través de su consulta
* Mejorar los protocolos de seguridad en credenciales de acceso, vista de módulos y control de acceso.
* Establecer un protocolo de registro de usuarios, así como de baja de usuarios.
* Establecer un protocolo de alta de libros, así como de baja de los mismos.
* Creación de un apartado de género del libro en la base de datos.
* Establecer un modulo automatizado para prestar libros a través de un protocolo de seguridad para la mayor optimización en el préstamo de los mismos.

Las mejoras a futuro van a depender de las necesidades del cliente y la institución principalmente a la cual se le enviaran estas sugerencias para seguir mejorando la funcionalidad de la página web, así como optimizar los servicios que esta otorga, a través de la mejora de sus procesos con aplicaciones informáticas y digitales que le ayudara a la institución a mantenerse en la vanguardia de la era digital como una de las universidades de primer nivel en México.