



Benemérita Universidad Autónoma de Puebla

“Pensar bien, para vivir mejor”

Facultad de Ciencias de la Computación



Aplicaciones web

Profesor: Mario Rossainz

Integrantes:

Alejandro Coyopol Cocolotl 2020650888

Carlos Antonio Arenas Bretón 202027778

Jesús Noé Antonio Ambrosio 202062848

Angel David Salas Mendoza 202056170

Proyecto final

19/05/2025

Sistema web para petición de asesorías

Objetivo: Diseñar e Implementar un Sistema Web que permita a un alumno solicitar una asesoría a uno o varios profesores, de tal forma que el profesor por su parte pueda ver, haciendo uso del mismo sistema, las asesorías que tiene agendadas. El sistema Web mostrará al alumno un formulario en donde deberá proporcionar la siguiente información:

1. Nombre completo
2. Matrícula
3. Programa Educativo (ICC, LCC, ITI)
4. Materia/Asignatura de la cual solicita asesoría
5. Elección del profesor con el cual solicita la asesoría
6. Si es alumno o no del profesor elegido
7. Elección de la fecha de la asesoría
8. Elección de la hora de la asesoría
9. El asunto

A continuación, el alumno enviará los datos a través de un botón de envío. Posteriormente el alumno podrá visualizar a través de un logueo proporcionando su matrícula y contraseña, todas las citas que ha agendado en el sistema para sus asesorías y el estatus de ésta (en proceso, aceptada, denegada) junto con el comentario del profesor.

El profesor por su parte podrá ver todas las solicitudes de asesorías que tiene por parte de los alumnos al loguearse con su ID de profesor y contraseña. Por cada asesoría solicitada el profesor podrá ver la siguiente información:

1. Nombre del alumno
2. Matrícula
3. Programa Educativo
4. Fecha de solicitud de asesoría
5. Hora de solicitud de asesoría
6. Asunto (tipo de asesoría)
7. Si es alumno suyo o no

Y con esa información el profesor podrá responder cada asesoría aceptando o denegando la solicitud además de un comentario para justificar dicha respuesta.

- a) Genere el diseño del sitio Web (pensando en el modelo tradicional de cajas) el cual consta de: el mapa del sitio y el diseño de las páginas web que lo conforman. Pueden diseñar en Balsamiq o en UWE-MagicDraw.
- b) La codificación del proyecto debe considerar el uso del modelo tradicional de cajas y el empleo de HTML5, JavaScript, CSS, Servlets y/o JSPs y acceso a base de datos relacional usando Querys en SQL. La base de datos deberá ser realizada en MySQL y será relacional.

Diseño en Balsamiq




A Web Page

← → ✕ 🏠

https://

🔍



Mentorizate
PROFESOR-ALUMNO

BIENVENIDO ALUMNO:

Usuario:

Contraseña:

[Registrarse](#)


INICIAR SESIÓN

A Web Page

← → ✕ 🏠

https://

🔍



Mentorizate
PROFESOR-ALUMNO

REGISTRARSE:

Nombre:

Matricula:

Contraseña:

PUESTO:

PROFESOR
ALUMNO

REGISTRARSE

A Web Page

https://

BIENVENIDO: _____

 HOME SOLICITAR ASESORIAS ASESORIAS APROBADAS CERRAR SESIÓN

Nombre completo del alumno:

Matricula:



A Web Page

https://

Formulario de petición

 HOME SOLICITAR ASESORIAS ASESORIAS APROBADAS CERRAR SESIÓN

Nombre completo:

Matricula:

Programa:

Materia:

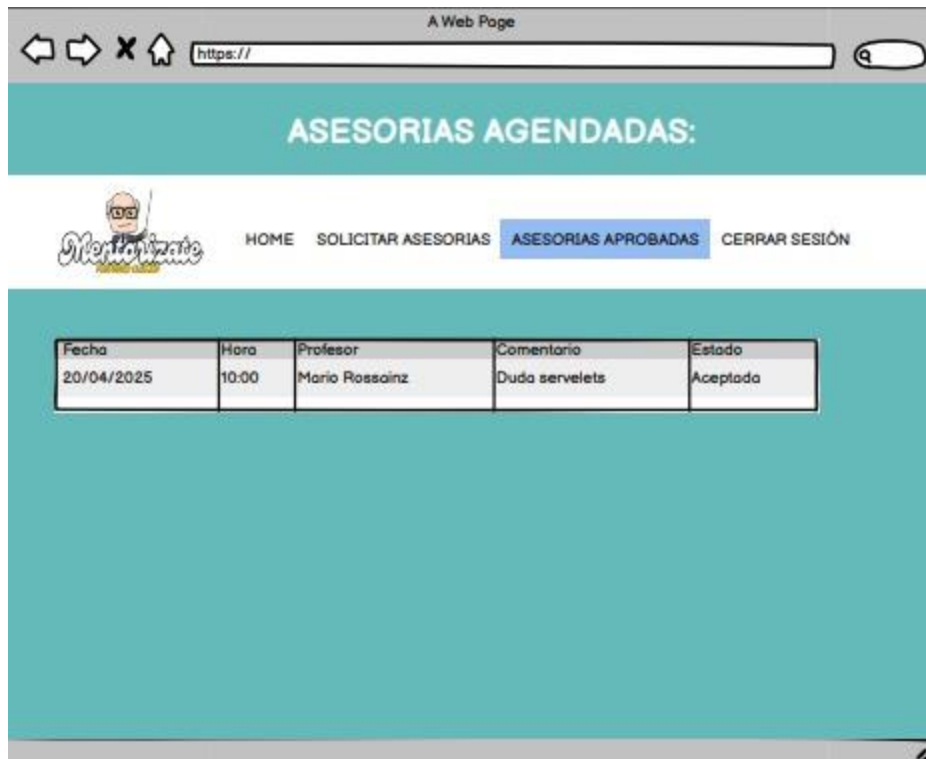
Profesor:

¿Es mi profesor?

Elección hora de asesoria:

Elección hora de asesoria:

Asunto:



A Web Page

X

https://

BIENVENIDO: _____



HOME

VER SOLICITUDES

SOLICITUDES PROXIMAS

CERRAR SESIÓN

Alumno	Fecha	Hora	Materia	Asunto	Acción
Antonio Arenas	22/01	2025	10:00	duda servelets	<div>Aceptar</div> <div>Denega</div>
Comentario:					

A Web Page

X

https://

BIENVENIDO: _____



HOME

VER SOLICITUDES

SOLICITUDES PROXIMAS

CERRAR SESIÓN

Fecha	Hora	Profesor	Comentario	Estado
20/04/2025	10:00	Mario Rossainz	Duda servelets	Aceptada

Ejecución del sitio web

¿Qué es Mentorízate?

Es una plataforma educativa con el fin de dar asesorías, diseñada para conectar a **profesores y alumnos**. Permite acceso personalizado según el tipo de usuario, ya sea **alumno o profesor**.

ALUMNO


Paso 1. Pantalla de bienvenida



Instrucciones:

1. Al ingresar al sitio web, verás la pantalla con el mensaje:
"BIENVENIDO A: Mentorízate"
2. Aparecen dos botones:
 - a. **Soy Alumno**
 - b. **Soy Profesor**
3. Haz clic según tu rol:
 - a. Si eres estudiante, elige **"Soy Alumno"**.
 - b. Si eres docente, elige **"Soy Profesor"**.

PASO 2: Inicio de sesión como Alumno



BIENVENIDO ALUMNO:

Usuario:

Contraseña:

INICIAR SESIÓN


[Registrarse](#)

Instrucciones:

Después de seleccionar “Soy Alumno”:

1. Verás el mensaje: **“BIENVENIDO ALUMNO”**
2. Se mostrarán dos campos:
 - a. **Usuario:** Escribe tu **matrícula**.
 - b. **Contraseña:** Escribe tu **contraseña personal**.
3. Haz clic en el botón amarillo **“INICIAR SESIÓN”** para entrar a la plataforma.

Si aún no tienes cuenta, puedes hacer clic en el enlace **“Registrarse”** para crear una.



REGISTRARSE:

Nombre:

Matrícula o Clave:

Contraseña:

Programa:

Tipo de usuario:

Registrarse

[Iniciar Sesión](#)

Rellenar el formulario y dar clic en enviar



The image shows a registration form for 'Mentorizate'. At the top is a cartoon character of a man with glasses and a pointer. Below it is the logo 'Mentorizate' with 'PROFESOR-ALUMNO' underneath. The main heading is 'REGISTRARSE:'. Below that is a green message: '¡Registro exitoso! Ahora puedes iniciar sesión.' There are two input fields: 'Nombre:' and 'Matricula o Clave:'.

Paso 3. Panel principal del alumno



The image shows a student dashboard. At the top, a teal banner says 'BIENVENIDO: Carlos Antonio'. Below this is a navigation bar with the Mentorizate logo and four buttons: 'HOME', 'SOLICITAR ASESORIAS', 'ASESORIAS APROBADAS', and 'CERRAR SESIÓN'. The main content area has a teal background and displays the student's name 'Nombre completo del alumno: Carlos Antonio' and matriculation number 'Matricula: 202027778'. To the right is a profile icon of a student in a graduation cap. At the bottom, a small footer reads '© 2025 Asesorías Web | Mentorizate'.

Instrucciones:

Tras iniciar sesión correctamente.

- Se verá la información (nombre y matrícula del alumno)
- Barra de navegación superior con opciones:
 - a. Home: Página principal
 - b. Solicitar asesorías: Para crear una solicitud de asesoría
 - c. Asesorías aprobadas: Solicitudes confirmadas
 - d. Cerrar sesión: Salir del sistema

Paso 4: Solicitud de asesorías

BIENVENIDO: Juan Pérez



HOME

SOLICITAR ASESORIAS

ASESORIAS APROBADAS

CERRAR SESIÓN

Nombres: Juan Pérez

Matricula: 12245678

Carrera: ICC

Asesoría: Tipo de asesoría

Profesor: Seleccione un profesor

Eres su Alumno?: ☐ Sí ☐ No

Fecha: dd/mm/aaaa

Hora: --:--

Asunto: Descripción de la asesoría

Enviar

© 2025 Asesorías Web | Mentorizate

Nombre: Carlos Antonio

Matricula: 202027778

Carrera: ICC

Asesoría: Tipo de asesoría

Profesor: Seleccione un profesor

Eres su Alumno?: ☐ Sí ☐ No

Fecha: dd/mm/aaaa

Hora: --:--

Asunto: Descripción de la asesoría

Enviar

¡Solicitud de asesoría enviada correctamente!

Instrucciones:

En esta sección se tendrá que rellenar unos campos:

Nombre
Matrícula
Carrera
Asesoría
Profesor

Fecha
Hora
Asunto

Una vez que se rellene, se realiza la solicitud a espera de su revisión.

Paso 5: Gestión de asesorías

BIENVENIDO: Carlos Antonio



HOME

SOLICITAR ASESORIAS

ASESORIAS APROBADAS

CERRAR SESIÓN

Tus citas de asesoría

Materia	Fecha	Hora	Estado	Comentario
Programación Web	2025-05-20	10:00:00	aceptada	CLARO
Dudas	2025-05-20	10:00:00	aceptada	SI SI
Dudas de ProgramaciA'n	2025-05-20	10:00:00	denegada	LAB OCUPADO
Duda servelets	2025-06-26	11:00:00	en proceso	

© 2025 Asesorías Web | Mentorizate

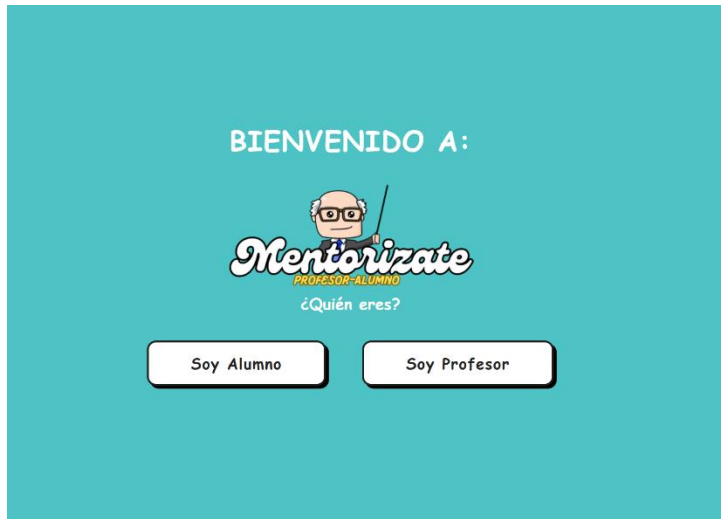
Instrucciones:

En esta sección se verá la solicitud de asesorías solicitadas.

Paso 6.

Cerrar sesión

Al hacer clic en “cerrar sesión” será redirigido a la página de inicio.



PROFESOR

Paso 1. Pantalla de bienvenida



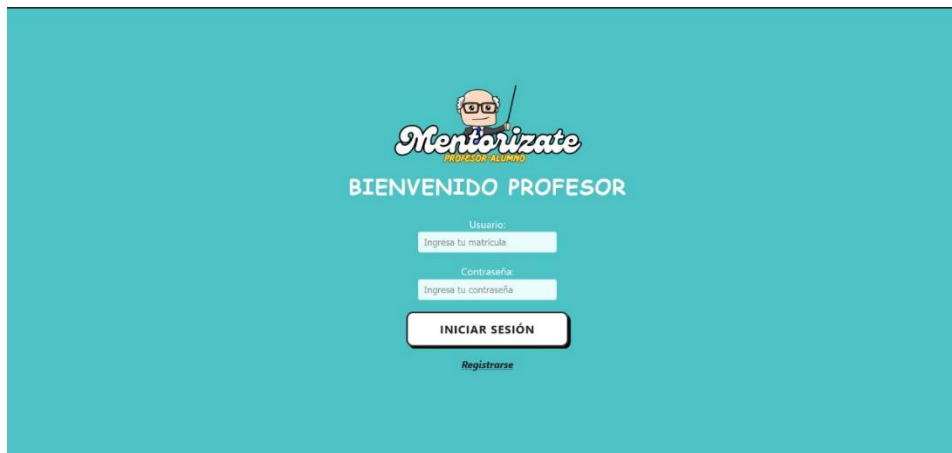
Instrucciones:

Al ingresar al sitio web, verás la pantalla con el mensaje:

“BIENVENIDO A: Mentorízate”

4. Aparecen dos botones:
 - a. **Soy Alumno**
 - b. **B Soy Profesor**
5. Haz clic según tu rol:
 - a. Si eres estudiante, elige **“Soy Alumno”**.
 - b. Si eres docente, elige **“Soy Profesor”**.

Paso 2. Inicio de sesión como profesor




Instrucciones:

Después de seleccionar “Soy Profesor”:

4. Verás el mensaje: **“BIENVENIDO ALUMNO”**
5. Se mostrarán dos campos:
 - a. **Usuario:** Escribe tu **matrícula**.
 - b. **Contraseña:** Escribe tu **contraseña personal**.
6. Haz clic en el botón amarillo **“INICIAR SESIÓN”** para entrar a la plataforma.

Si aún no tienes cuenta, puedes hacer clic en el enlace **“Regístrate”** para crear una.



REGISTRARSE:

Nombre:

Matrícula o Clave:

Contraseña:

Programa:

Tipo de usuario:


Registrarse

[Iniciar Sesión](#)

Rellenar los campos y enviar

Paso 3. Panel principal del profesor


BIENVENIDO: Dr. Mario Rossainz



[HOME](#) [VER SOLICITUDES](#) [SOLICITUDES PROXIMAS](#) [CERRAR SESIÓN](#)

Nombre completo:
Dr. Mario Rossainz

Clave:
202022



© 2025 Asesorías Web | Mentorizate

Instrucciones:

Tras iniciar sesión correctamente (ejemplo con Mario Rossainz)

- Se verá la información (nombre y matrícula del profesor)
- Barra de navegación superior con opciones:
- e. Home: Página principal
- f. Ver solicitudes: Solicitud recibida, por aprobar/denegar
- g. Solicitudes próximas: Solicitudes a futuro
- h. Cerrar sesión: Salir del sistema

Paso 4. Panel de solicitud

BIENVENIDO: Dr. Mario Rossainz



[HOME](#)
[VER SOLICITUDES](#)
[SOLICITUDES PROXIMAS](#)
[CERRAR SESIÓN](#)

Solicitudes por Aprobar

Alumno	Matricula	Programa	Materia	Fecha	Hora	Asunto	Es su alumno	Responder
Carlos Antonio	202027778	ICC	Duda servelets	2025-06-26	11:00:00	Duda sobre servelets	Sí	<div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between;"> Seleccione ▼ </div> <div style="border-bottom: 1px solid #ccc; height: 15px; margin-top: 5px;"></div> <div style="margin-top: 5px;">Comentario...</div> </div> <div style="text-align: right; margin-top: 5px;"> Enviar </div>

© 2025 Asesorías Web | Mentorizate

En esta sección se verá la solicitud recibida por el alumno, con un espacio para agregar comentarios, si lo desea.

Paso 5. Panel de solicitud a futuro

BIENVENIDO: Dr. Mario Rossainz

HOME VER SOLICITUDES SOLICITUDES PRÓXIMAS CERRAR SESIÓN

Solicitudes Aceptadas

Fecha	Hora	Alumno	Materia	Asunto	Estado
2025-05-18	14:30:00	Jose Isaias Bautista	Estructura de Datos	Repaso de árboles binarios	aceptada

© 2025 Asesorías Web | Mentorizate

En esta sección verá las solicitudes que tiene programada para impartir.

CODIGOS

Index.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>

  <title>TODO supply a title</title>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <meta http-equiv="refresh" content="0;
URL=http://localhost:8080/AsesoriasWeb/login/iniciarsesion.jsp">

</head>

<body>

  <div>TODO write content</div>

</body>

</html>
```

Estilos.css

```
/* estilos.css - Estilos base reutilizables */
```

```
:root {

  --color-primario: #6cc3c3;

  --color-secundario: #1e3c72;

  --color-acento: #f9b233;

  --color-fondo: #f4f6fa;

  --color-texto: #222;

  --color-blanco: #fff;

  --color-gris: #e0e0e0;

}
```

```
body {
```

```
margin: 0;

font-family: 'Segoe UI', Arial, sans-serif;

background: var(--color-fondo);

color: var(--color-texto);

min-height: 100vh;

display: flex;

flex-direction: column;
}
```

```
.header {

background: var(--color-blanco);

color: var(--color-texto);

display: flex;

align-items: center;

justify-content: space-between;

padding: 0.5rem 2rem;

box-shadow: 0 2px 8px rgba(0,0,0,0.04);
}
```

```
.logo {

height: 48px;
}
```

```
.nav {

display: flex;

gap: 1.5rem;
```

```
}
```

```
.nav-link{
```

```
  color: var(--color-texto);
```

```
  background: transparent;
```

```
  text-decoration: none;
```

```
  font-weight: 600;
```

```
  font-size: 1.08rem;
```

```
  padding: 0.4rem 1rem;
```

```
  border-radius: 6px;
```

```
  transition: background 0.2s, color 0.2s;
```

```
  letter-spacing: 0.5px;
```

```
  display: inline-block;
```

```
}
```

```
.nav-link-activo, .nav-link:hover {
```

```
  background: var(--color-primario);
```

```
  color: var(--color-blanco) !important;
```

```
}
```

```
.cerrar-sesion {
```

```
  font-weight: bold;
```

```
  color: var(--color-secundario);
```

```
}
```

```
.main-content {
```

```
  flex: 1;
```

```
  display: flex;
```

```
flex-direction: column;

align-items: center;

justify-content: flex-start;

padding: 2rem 1rem;

background: var(--color-primario);

min-height: 60vh;
}

.main-content h1 {

margin-bottom: 2rem;

font-size: 2.2rem;

color: var(--color-primario);
}

.titulo-bienvenida {

width: 100%;

background: var(--color-primario);

color: var(--color-blanco);

text-align: center;

font-size: 2.5rem;

font-weight: bold;

padding: 1.5rem 0 1rem 0;

letter-spacing: 1px;
}

.info-alumno {

color: var(--color-blanco);
```

```
font-size: 1.2rem;

font-weight: 600;

margin-top: 2rem;

margin-left: 2rem;
}

.foto-alumno {

width: 200px;

height: 200px;

background: var(--color-blanco);

border: 2px solid #444;

display: flex;

align-items: center;

justify-content: center;

margin-left: 4rem;
}

.contenedor-alumno {

display: flex;

flex-direction: row;

align-items: flex-start;

margin-top: 2rem;
}

.acciones {

display: flex;

gap: 2rem;
```

```
margin-bottom: 2rem;
}

.btn-principal, .btn-secundario {
padding: 1rem 2.5rem;
border: none;
border-radius: 8px;
font-size: 1.1rem;
font-weight: 600;
cursor: pointer;
text-decoration: none;
transition: background 0.2s, color 0.2s;
}

.btn-principal {
background: var(--color-primario);
color: var(--color-blanco);
}

.btn-principal:hover {
background: var(--color-acento);
color: var(--color-texto);
}

.btn-secundario {
background: var(--color-blanco);
color: var(--color-primario);
border: 2px solid var(--color-primario);
```



```
}  
  
.btn-secundario:hover {  
    background: var(--color-primario);  
    color: var(--color-blanco);  
}
```

```
.footer {  
    background: var(--color-gris);  
    color: #555;  
    text-align: center;  
    padding: 1rem 0;  
    font-size: 1rem;  
    margin-top: auto;  
}
```

```
/* Utilidades */
```

```
.text-center {  
    text-align: center;  
}
```

```
.mt-2 {  
    margin-top: 2rem;  
}
```

```
.mb-2 {  
    margin-bottom: 2rem;  
}
```

```
table {  
    width: 100%;  
    border-collapse: collapse;  
    background-color: white;  
    border: 2px solid black;  
}  
  
/  
th, td {  
    border: 1px solid black;  
    padding: 8px;  
    text-align: left;  
}  
  
th {  
    background-color: #f2f2f2; /  
    font-weight: bold;  
}  
  
button {  
    background-color: #4CAF50;  
    color: white;  
    padding: 10px 20px;
```

```
border: none;

cursor: pointer;

border-radius: 5px;

margin-top: 10px;
}

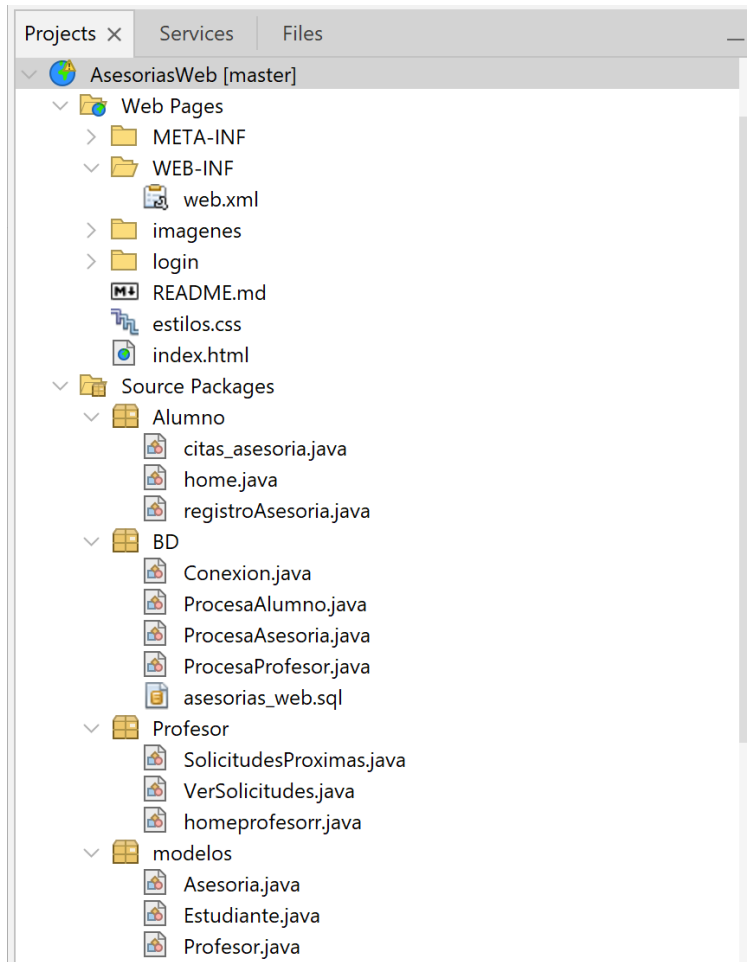
button:hover {

background-color: #45a049;
}

form {

display: flex;

flex-direction: column;
}
```



Alumno

Citas_asesoria.java

/*

* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

* Click nbfs://nbhost/SystemFileSystem/Templates/JSP_Servlet/Servlet.java to edit this template

*/

package Alumno;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.List;

import java.util.Map;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import BD.ProcesaAsesoria;

import modelos.Estudiante;

/**

*

* @author aleja

*/

public class citas_asesoria extends HttpServlet {

```

/**
 * Processes requests for both HTTP GET and POST
 * methods.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    response.setContentType("text/html;charset=UTF-8");
    try (PrintWriter out = response.getWriter()) {
        out.println("<!DOCTYPE html>");
        out.println("<html lang='es'>");
        out.println("<head>");
        out.println("    <meta charset='UTF-8'>");
        out.println("    <meta name='viewport' content='width=device-width, initial-
scale=1.0'>");
        out.println("    <title>Solicitar Asesoría</title>");
        out.println("    <link rel='stylesheet' href='estilos.css'>");
        out.println("</head>");
        out.println("<body>");

        Estudiante alumno = (Estudiante) request.getSession().getAttribute("alumno");

```

```

    if (alumno == null) {

        response.sendRedirect("login/iniciarsesion.jsp");

        return;

    }

    out.println("    <div class='titulo-bienvenida'>BIENVENIDO: <span style='border-
bottom:2px solid #fff; padding:0 2rem;'>" + alumno.getNombre() + "</span></div>");

    out.println("    <header class='header'>");

    out.println("        <img src='imagenes/logo.png' alt='Logo' class='logo'
style='height:60px;'>");

    out.println("        <nav class='nav'>");

    out.println("            <a href='home' class='nav-link'>HOME</a>");

    out.println("            <a href='registroAsesoria' class='nav-link'>SOLICITAR
ASESORIAS</a>");

    out.println("            <a href='citas_asesoria' class='nav-link nav-link-
activo'>ASESORIAS APROBADAS</a>");

    out.println("            <a href='login/iniciarsesion.jsp' class='nav-link cerrar-
sesion'>CERRAR SESIÓN</a>");

    out.println("        </nav>");

    out.println("    </header>");

    out.println("    <main class='main-content'>");

    out.println("        <h2>Tus citas de asesoría</h2>");

    out.println("        <table
border='1'><tr><th>Materia</th><th>Fecha</th><th>Hora</th><th>Estado</th><th>Com
entario</th></tr>");

    int idAlumno = alumno.getIdEstudiante();

    List<Map<String, Object>> citas =
ProcesaAsesoria.getAsesoriasPorAlumno(idAlumno);

    for (Map<String, Object> cita : citas) {

```

```

        out.println("<tr>");
        out.println("<td>" + cita.get("materia") + "</td>");
        out.println("<td>" + cita.get("fecha_solicitud") + "</td>");
        out.println("<td>" + cita.get("hora_solicitud") + "</td>");
        out.println("<td>" + cita.get("estado") + "</td>");
        out.println("<td>" + (cita.get("comentario") != null ? cita.get("comentario") : "") +
"</td>");
        out.println("</tr>");
    }
    out.println("    </table>");
    out.println(" </main>");
    out.println(" <footer class='footer'>");
    out.println("    &copy; 2025 Asesorías Web | Mentorízate");
    out.println(" </footer>");
    out.println("</body>");
    out.println("</html>");
}
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```

/**
 * Handles the HTTP <code>GET</code> method.
 *
 * @param request servlet request
 * @param response servlet response

```


* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

* Handles the HTTP <code>POST</code> method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

* Returns a short description of the servlet.

*

```
* @return a String containing servlet description
*/
@Override
public String getServletInfo() {
    return "Short description";
} // </editor-fold>

}
```

Home.java

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/JSP_Servlet/Servlet.java to edit this
template
*/
package Alumno;

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import modelos.Estudiante;
```

```
/**
 *
 * @author aleja
 */
public class home extends HttpServlet {

    /**
     * Processes requests for both HTTP GET and POST
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            out.println("<!DOCTYPE html>");
            out.println("<html lang='es'>");
            out.println("<head>");
            out.println("    <meta charset='UTF-8'>");
```

```
        out.println("    <meta name='viewport' content='width=device-width, initial-
scale=1.0'>");

        out.println("    <title>Inicio Alumno</title>");

        out.println("    <link rel='stylesheet' href='estilos.css'>");

        out.println("</head>");

        out.println("<body>");

        Estudiante alumno = (Estudiante) request.getSession().getAttribute("alumno");

        if (alumno == null) {

            response.sendRedirect("login/iniciarsesion.jsp");

            return;

        }

        out.println("    <div class='titulo-bienvenida'>BIENVENIDO: <span style='border-
bottom:2px solid #fff; padding:0 2rem;'>" + alumno.getNombre() + "</span></div>");

        out.println("    <header class='header'>");

        out.println("        <img src='imagenes/logo.png' alt='Logo' class='logo'
style='height:60px;'>");

        out.println("        <nav class='nav'>");

        out.println("            <a href='home' class='nav-link nav-link-activo'>HOME</a>");

        out.println("            <a href='registroAsesoria' class='nav-link'>SOLICITAR
ASESORIAS</a>");

        out.println("            <a href='citas_asesoria' class='nav-link'>ASESORIAS
APROBADAS</a>");

        out.println("            <a href='login/iniciarsesion.jsp' class='nav-link cerrar-
sesion'>CERRAR SESIÓN</a>");

        out.println("        </nav>");

        out.println("    </header>");

        out.println("    <main class='main-content'>");

        out.println("        <div class='contenedor-alumno'>");
```

```

        out.println("        <div class='info-alumno'>");
        out.println("        Nombre completo del alumno: <strong>" + alumno.getNombre()
+ "</strong><br><br>");
        out.println("        Matrícula: <strong>" + alumno.getMatricula() + "</strong>");
        out.println("        </div>");
        out.println("        <div class='foto-alumno'>");
        out.println("        <img src='imagenes/estudiante.png' alt='Foto predeterminada
del alumno' width='180' height='180' '>");
        out.println("        </div>");
        out.println("    </div>");
        out.println(" </main>");
        out.println(" <footer class='footer'>");
        out.println("    &copy; 2025 Asesorías Web | Mentorízate");
        out.println(" </footer>");
        out.println("</body>");
        out.println("</html>");

    }
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

/**

* Handles the HTTP <code>GET</code> method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

* Handles the HTTP <code>POST</code> method.

*

* @param request servlet request

* @param response servlet response

* @throws ServletException if a servlet-specific error occurs

* @throws IOException if an I/O error occurs

*/

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

/**

* Returns a short description of the servlet.

*

* @return a String containing servlet description

```
*/  
  
@Override  
public String getServletInfo() {  
    return "Short description";  
} // </editor-fold>  
  
}
```

Registroasesoria.java

```
package Alumno;  
  
/*  
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to  
change this license  
 * Click nbfs://nbhost/SystemFileSystem/Templates/JSP_Servlet/Servlet.java to edit this  
template  
*/  
  
import BD.ProcesaProfesor;  
import java.io.IOException;  
import java.io.PrintWriter;  
import java.sql.Connection;  
import java.sql.PreparedStatement;  
import java.util.List;  
import javax.servlet.ServletException;  
import javax.servlet.http.HttpServlet;
```

```
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import modelos.Estudiante;
import modelos.Profesor;

/**
 *
 * @author aleja
 */
public class registroAsesoría extends HttpServlet {

    /**
     * Processes requests for both HTTP GET and POST
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request, HttpServletResponse
    response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            out.println("<!DOCTYPE html>");
        }
    }
}
```



```
out.println("<html lang='es'>");
out.println("<head>");
out.println("  <meta charset='UTF-8'>");
out.println("  <meta name='viewport' content='width=device-width, initial-
scale=1.0'>");
out.println("  <title>Solicitar Asesoría</title>");
out.println("  <link rel='stylesheet' href='estilos.css'>");
out.println("</head>");
out.println("<body>");
Estudiante alumno = (Estudiante) request.getSession().getAttribute("alumno");
if (alumno == null) {
    response.sendRedirect("login/iniciarsesion.jsp");
    return;
}
out.println("  <div class='titulo-bienvenida'>BIENVENIDO: <span style='border-
bottom:2px solid #fff; padding:0 2rem;'>" + alumno.getNombre() + "</span></div>");
out.println("  <header class='header'>");
out.println("    <img src='imagenes/logo.png' alt='Logo' class='logo'
style='height:60px;'>");
out.println("    <nav class='nav'>");
out.println("      <a href='home' class='nav-link'>HOME</a>");
out.println("      <a href='registroAsesoria' class='nav-link nav-link-
activo'>SOLICITAR ASESORIAS</a>");
out.println("      <a href='citas_asesoria' class='nav-link nav-link'>ASESORIAS
APROBADAS</a>");
out.println("      <a href='login/iniciarsesion.jsp' class='nav-link cerrar-
sesion'>CERRAR SESIÓN</a>");
out.println("    </nav>");
```

```

        out.println("    </header>");

        out.println("    <main class='main-content'>");

        out.println("        <form class='form-solicitud' action="" method='post'>");

        out.println("            Nombre: <input type='text' name='nombre' value="" +
alumno.getNombre() + "" class='input-alumno' readonly><br>");

        out.println("            Matricula: <input type='number' name='matricula' value="" +
alumno.getMatricula() + "" class='input-alumno' readonly><br>");

        out.println("            Carrera: <select name=\"Programa_educativo\" class='input-
alumno' required disabled>");

        out.println("                <option value="" + alumno.getPrograma() + "" selected>" +
alumno.getPrograma() + "</option>");

        out.println("            </select><br>");

        out.println("            Asesoría: <input type='text' name='asesoria' placeholder='Tipo de
asesoría' class='input-alumno' required><br>");

        out.println("            Profesor: <select name=\"select-profesor\" class='input-alumno'
required>");

        out.println("                <option value="">Seleccione un profesor</option>");

        List<Profesor> profesores = ProcesaProfesor.obtenerProfesores();

        for (Profesor prof : profesores) {

            out.println("                <option value="" + prof.getIdProfesor() + "">" + prof.getNombre()
+ " (" + prof.getPrograma() + ")</option>");

        }

        out.println("            </select><br>");

        out.println("            Eres su Alumno?: ");

        out.println("            <input type='radio' name='pertenencia' value='si' class='input-
alumno' required> Sí");

        out.println("            <input type='radio' name='pertenencia' value='no' class='input-
alumno' required> No<br>");

```

```

        out.println("        Fecha: <input type='date' name='fecha' placeholder='Fecha de la
asesoría' class='input-alumno' required><br>");

        out.println("        Hora: <input type='time' name='hora' placeholder='Hora de la
asesoría' class='input-alumno' required><br>");

        out.println("        Asunto: <textarea name='descripcion' placeholder='Descripción
de la asesoría' class='input-alumno' required></textarea><br>");

        out.println("        <button type='submit' class='input-alumno'>Enviar</button>");
        out.println("    </form>");

        boolean exito = false;

        String mensaje = null;

        if ("POST".equalsIgnoreCase(request.getMethod())) {

            String nombre = request.getParameter("nombre");

            String matricula = request.getParameter("matricula");

            String programa = request.getParameter("Programa_educativo");

            String asesoria = request.getParameter("asesoria");

            int profesorId = Integer.parseInt(request.getParameter("select-profesor"));

            String pertenencia = request.getParameter("pertenencia");

            boolean esTuDocente = "si".equalsIgnoreCase(pertenencia);

            String fecha = request.getParameter("fecha");

            String hora = request.getParameter("hora");

            String asunto = request.getParameter("descripcion");

            try (Connection con = BD.Conexion.getConexion()) {

                String sql = "INSERT INTO asesorias (alumno_id, profesor_id, materia,
es_tu_docente, fecha_solicitud, hora_solicitud, asunto, estado) VALUES (?, ?, ?, ?, ?, ?, ?,
'en proceso')";

                try (PreparedStatement ps = con.prepareStatement(sql)) {

                    ps.setInt(1, alumno.getIdEstudiante());

```

```
        ps.setInt(2, profesorId);
        ps.setString(3, asesoria);
        ps.setBoolean(4, esTuDocente);
        ps.setDate(5, java.sql.Date.valueOf(fecha));
        ps.setTime(6, java.sql.Time.valueOf(hora + ":00"));
        ps.setString(7, asunto);
        ps.executeUpdate();
        exito = true;
    }
} catch (Exception e) {
    e.printStackTrace();

    mensaje = "<div style='color:red;'>Error al registrar la asesoría: " + e.getMessage()
+ "</div>";
}

if (exito) {
    mensaje = "<div id='msg-exito' style='color:green;'>¡Solicitud de asesoría enviada
correctamente!</div>";
}
}

if (mensaje != null) {
    out.println(mensaje);

    out.println("<script>setTimeout(function(){ var m=document.getElementById('msg-
exito'); if(m){ m.style.display='none'; } }, 2500);</script>");
}

out.println(" </main>");
out.println(" <footer class='footer'>");
out.println("    &copy; 2025 Asesorías Web | Mentorízate");
```

```
        out.println(" </footer>");  
        out.println("</body>");  
        out.println("</html>");  
    }  
}
```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```
/**  
 * Handles the HTTP <code>GET</code> method.  
 *  
 * @param request servlet request  
 * @param response servlet response  
 * @throws ServletException if a servlet-specific error occurs  
 * @throws IOException if an I/O error occurs  
 */  
  
@Override  
protected void doGet(HttpServletRequest request, HttpServletResponse response)  
    throws ServletException, IOException {  
    processRequest(request, response);  
}  
  
/**  
 * Handles the HTTP <code>POST</code> method.  
 *  
 * @param request servlet request
```

```
* @param response servlet response
* @throws ServletException if a servlet-specific error occurs
* @throws IOException if an I/O error occurs
*/
```

```
@Override
```

```
protected void doPost(HttpServletRequest request, HttpServletResponse response)
```

```
    throws ServletException, IOException {
    processRequest(request, response);
}
```

```
/**
```

```
* Returns a short description of the servlet.
```

```
*
```

```
* @return a String containing servlet description
```

```
*/
```

```
@Override
```

```
public String getServletInfo() {
```

```
    return "Short description";
```

```
}// </editor-fold>
```

```
}
```

Profesor

Homeprofesorr.java

```
package Profesor;
```

```
import java.io.IOException;
```

```
import java.io.PrintWriter;
```

```
import javax.servlet.ServletException;
```

```
import javax.servlet.http.HttpServlet;
```

```
import javax.servlet.http.HttpServletRequest;
```

```
import javax.servlet.http.HttpServletResponse;
```

```
import modelos.Profesor;
```

```
/**
```

```
 * @author aleja
```

```
 */
```

```
public class homeprofesorr extends HttpServlet {
```

```
    /**
```

```
     * Processes requests for both HTTP GET and POST
```

```
     * methods.
```

```
     *
```

```
     * @param request servlet request
```

```
     * @param response servlet response
```

```
     * @throws ServletException if a servlet-specific error occurs
```

```
     * @throws IOException if an I/O error occurs
```

```

*/

protected void processRequest(HttpServletRequest request, HttpServletResponse
response)

    throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

    //Validar sesión

    Profesor profesor = (Profesor) request.getSession().getAttribute("profesor");

    if (profesor == null) {

        response.sendRedirect("login/iniciarsesion.jsp");

        return;

    }

    //HTML Completo

    out.println("<!DOCTYPE html>");

    out.println("<html lang='es'>");

    out.println("<head>");

    out.println("    <meta charset='UTF-8'>");

    out.println("    <meta name='viewport' content='width=device-width, initial-
scale=1.0'>");

    out.println("    <title>Inicio Profesor</title>");

    out.println("    <link rel='stylesheet' href='estilos.css'>");

    out.println("</head>");

    out.println("<body>");

    //Bienvenida

    out.println("    <div class='titulo-bienvenida'>BIENVENIDO: <span style='border-
bottom:2px solid #fff; padding:0 2rem;'> + profesor.getNombre() + "</span></div>");

    //Menu

```



```
        out.println("    <header class='header'>");

        out.println("        <img src='imagenes/logo.png' alt='Logo' class='logo'
style='height:60px;'>");

        out.println("        <nav class='nav'>");

        out.println("            <a href='homeprofesor' class='nav-link nav-link-
activo'>HOME</a>");

        out.println("            <a href='VerSolicitudes' class='nav-link'>VER SOLICITUDES</a>");

        out.println("            <a href='SolicitudesProximas' class='nav-link'> SOLICITUDES
PROXIMAS</a>");

        out.println("            <a href='login/iniciarsesion.jsp' class='nav-link cerrar-
sesion'>CERRAR SESIÓN</a>");

        out.println("        </nav>");

        out.println("    </header>");

    //Cuerpo

    out.println("    <main class='main-content'>");

    out.println("        <div class='contenedor-alumno'>");

    out.println("            <div class='info-alumno'>");

    out.println("                <strong>Nombre completo:</strong><br>" + profesor.getNombre()
+ "<br><br>");

    out.println("                <strong>Clave:</strong><br>" + profesor.getClaveProfesor());

    out.println("            </div>");

    out.println("            <div class='foto-alumno'>");

    out.println("                <img src='imagenes/profesor.png' alt='Foto predeterminada del
alumno' width='180' height='180' >");

    out.println("            </div>");

    out.println("        </div>");

    out.println("    </main>");

    //Footer
```

```
        out.println("    <footer class='footer'>");
        out.println("        &copy; 2025 Asesorías Web | Mentorízate");
        out.println("    </footer>");
        out.println("</body>");
        out.println("</html>");
    }
}
```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```
/**
 * Handles the HTTP <code>GET</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}

/**
 * Handles the HTTP <code>POST</code> method.
```

```
*  
* @param request servlet request  
* @param response servlet response  
* @throws ServletException if a servlet-specific error occurs  
* @throws IOException if an I/O error occurs  
*/
```

```
@Override
```

```
protected void doPost(HttpServletRequest request, HttpServletResponse response)  
    throws ServletException, IOException {  
    processRequest(request, response);  
}
```

```
/**
```

```
* Returns a short description of the servlet.  
*  
* @return a String containing servlet description  
*/
```

```
@Override
```

```
public String getServletInfo() {  
    return "Página de inicio del profesor con navegación y datos básicos";  
} // </editor-fold>
```

```
}
```

Solicitudesproximas.java

```
package Profesor;

import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;
import java.util.Map;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import modelos.Profesor;

public class SolicitudesProximas extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {

            Profesor profesor = (Profesor) request.getSession().getAttribute("profesor");
            if (profesor == null) {
                response.sendRedirect("login/iniciarsesion.jsp");
                return;
            }
        }
    }
}
```

```
}
```

```
List<Map<String, Object>> solicitudes =  
BD.ProcesaAsesoria.getAsesoriasPorProfesor(profesor.getIdProfesor());
```

```
out.println("<!DOCTYPE html>");  
out.println("<html lang='es'>");  
out.println("<head>");  
out.println("  <meta charset='UTF-8'>");  
out.println("  <meta name='viewport' content='width=device-width, initial-  
scale=1.0'>");  
out.println("  <title>Solicitudes Aceptadas</title>");  
out.println("  <link rel='stylesheet' href='estilos.css'>");  
out.println("<style>");  
out.println("table {");  
out.println("  width: 100%;");  
out.println("  border-collapse: collapse;");  
out.println("  background-color: white;");  
out.println("  border: 2px solid black;");  
out.println("}");  
out.println("th, td {");  
out.println("  border: 1px solid black;");  
out.println("  padding: 8px;");  
out.println("  text-align: left;");  
out.println("}");  
out.println("th {");
```

```
out.println(" background-color: #f2f2f2;");
out.println(" font-weight: bold;");
out.println("");
out.println("button {");
out.println(" background-color: #4CAF50;");
out.println(" color: white;");
out.println(" padding: 10px 20px;");
out.println(" border: none;");
out.println(" cursor: pointer;");
out.println(" border-radius: 5px;");
out.println(" margin-top: 10px;");
out.println("");
out.println("button:hover {");
out.println(" background-color: #45a049;");
out.println("");
out.println("form {");
out.println(" display: flex;");
out.println(" flex-direction: column;");
out.println("");
out.println("</style>");
    out.println("</head>");
    out.println("<body>");

    out.println(" <div class='titulo-bienvenida'>BIENVENIDO: <span style='border-
bottom:2px solid #fff; padding:0 2rem;'>" + profesor.getNombre() + "</span></div>");
    out.println(" <header class='header'>");
```

```

        out.println("    <img src='imagenes/logo.png' alt='Logo' class='logo'
style='height:60px;'>");

        out.println("    <nav class='nav'>");

        out.println("        <a href='homeprofesor' class='nav-link'>HOME</a>");

        out.println("        <a href='VerSolicitudes' class='nav-link'>VER SOLICITUDES</a>");

        out.println("        <a href='SolicitudesProximas' class='nav-link nav-link-
activo'>SOLICITUDES PRÓXIMAS</a>");

        out.println("        <a href='login/iniciarsesion.jsp' class='nav-link cerrar-
sesion'>CERRAR SESIÓN</a>");

        out.println("    </nav>");

        out.println(" </header>");


        out.println(" <main class='main-content'>");

        out.println("    <div class='contenedor-profesor'>");

        out.println("        <h2>Solicitudes Aceptadas</h2>");

        out.println("        <table border='1' cellpadding='8'>");

        out.println("            <tr><th>Fecha</th><th>Hora</th><th>Alumno</th><th>Materia</th><th>Asunto</th><th>
>Estado</th></tr>");

        for (Map<String, Object> s : solicitudes) {

            if (!"aceptada".equalsIgnoreCase(String.valueOf(s.get("estado")))) {

                continue;

            }

            out.println("                <tr>");

            out.println("                    <td>" + s.get("fecha_solicitud") + "</td>");

```

```

        out.println("        <td>" + s.get("hora_solicitud") + "</td>");
        out.println("        <td>" + s.get("nombre") + "</td>");
        out.println("        <td>" + s.get("materia") + "</td>");
        out.println("        <td>" + s.get("asunto") + "</td>");
        out.println("        <td>" + s.get("estado") + "</td>");
        out.println("    </tr>");
    }

    out.println("    </table>");
    out.println("</div>");
    out.println("</main>");

    out.println("    <footer class='footer'>");
    out.println("        &copy; 2025 Asesorías Web | Mentorízate");
    out.println("    </footer>");
    out.println("</body>");
    out.println("</html>");
}
}
}

```

VerSolicitudes.java

```
package Profesor;
```

```
import java.io.IOException;
```

```
import java.io.PrintWriter;
```



```
import java.util.List;
import java.util.Map;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import modelos.Profesor;

/**
 * @author carlos y Angel
 */
public class VerSolicitudes extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        Profesor profesor = (Profesor) request.getSession().getAttribute("profesor");
        if (profesor == null) {
            response.sendRedirect("login/iniciarsesion.jsp");
            return;
        }

        List<Map<String, Object>> solicitudes =
        BD.ProcesaAsesorias.getAsesoriasPorProfesor(profesor.getIdProfesor());
```

```
response.setContentType("text/html;charset=UTF-8");
try (PrintWriter out = response.getWriter()) {
    out.println("<!DOCTYPE html>");
    out.println("<html lang='es'>");
    out.println("<head>");
    out.println("<head><title>Solicitudes</title><meta charset='UTF-8'>");
    out.println("<meta name='viewport' content='width=device-width, initial-
scale=1.0'>");
    out.println("<link rel='stylesheet' href='estilos.css'></head><body>");
    out.println("<style>");
out.println("table {");
out.println("  width: 100%;");
out.println("  border-collapse: collapse;");
out.println("  background-color: white;");
out.println("  border: 2px solid black;");
out.println("}");
out.println("th, td {");
out.println("  border: 1px solid black;");
out.println("  padding: 8px;");
out.println("  text-align: left;");
out.println("}");
out.println("th {");
out.println("  background-color: #f2f2f2;");
out.println("  font-weight: bold;");
out.println("}");
out.println("button {");
```

```
out.println(" background-color: #4CAF50;");
out.println(" color: white;");
out.println(" padding: 10px 20px;");
out.println(" border: none;");
out.println(" cursor: pointer;");
out.println(" border-radius: 5px;");
out.println(" margin-top: 10px;");
out.println("}");
out.println("button:hover {");
out.println(" background-color: #45a049;");
out.println("}");
out.println("form {");
out.println(" display: flex;");
out.println(" flex-direction: column;");
out.println("}");
out.println("</style>");

//Bienvenida

out.println(" <div class='titulo-bienvenida'>BIENVENIDO: <span style='border-
bottom:2px solid #fff; padding:0 2rem;'>" + profesor.getNombre() + "</span></div>");

//Menu

out.println(" <header class='header'>");

out.println(" <img src='imagenes/logo.png' alt='Logo' class='logo'
style='height:60px;'>");

out.println(" <nav class='nav'>");

out.println(" <a href='homeprofesorrr' class='nav-link'>HOME</a>");

out.println(" <a href='VerSolicitudes' class='nav-link nav-link-activo'>VER
SOLICITUDES</a>");
```

```

        out.println("        <a href='SolicitudesProximas' class='nav-link'> SOLICITUDES
PROXIMAS</a>");

        out.println("        <a href='login/iniciarsesion.jsp' class='nav-link cerrar-
sesion'>CERRAR SESIÓN</a>");

        out.println("    </nav>");

        out.println(" </header>");

        //Cuerpo

        out.println(" <main class='main-content'>");

        out.println("    <div class='contenedor-profesor'>");

        out.println("<h2>Solicitudes por Aprobar</h2>");

        out.println("<table style='width: 100%; border-collapse: collapse; background-color:
white; border: 2px solid black;'>");

        out.println("<tr><th>Alumno</th><th>Matrícula</th><th>Programa</th><th>Materia</th>
<th>Fecha</th><th>Hora</th><th>Asunto</th><th>Es su
alumno</th><th>Responder</th></tr>");

        for (Map<String, Object> s : solicitudes) {

            if (!"en proceso".equals(String.valueOf(s.get("estado")))) {

                continue;

            }

            out.println("<tr>");

            out.println("<td>" + s.get("nombre") + "</td>");

            out.println("<td>" + s.get("matricula") + "</td>");

            out.println("<td>" + s.get("programa") + "</td>");

            out.println("<td>" + s.get("materia") + "</td>");

            out.println("<td>" + s.get("fecha_solicitud") + "</td>");

            out.println("<td>" + s.get("hora_solicitud") + "</td>");

```

```
        out.println("<td>" + s.get("asunto") + "</td>");

        out.println("<td>" + ((Boolean.TRUE.equals(s.get("es_tu_docente")) ? "Sí" : "No") +
"</td>");

        out.println("<td>");

        out.println("<form method='post' action='VerSolicitudes'>");

        out.println("<input type='hidden' name='id_asesoria' value='" + s.get("id_asesoria")
+ "'/>");

        out.println("<select name='estado' required>");

        out.println("<option value=''>Seleccione</option>");

        out.println("<option value='aceptada'>Aceptar</option>");

        out.println("<option value='denegada'>Denegar</option>");

        out.println("</select><br>");

        out.println("<textarea name='comentario' required placeholder='Comentario...'
rows='2' cols='20'></textarea><br>");

        out.println("<button type='submit'>Enviar</button>");

        out.println("</form>");

        out.println("</td>");

        out.println("</tr>");
    }

    out.println("</table>");

    out.println("    </div>");

    out.println("</main>");

    //Footer

    out.println("    <footer class='footer'>");

    out.println("        &copy; 2025 Asesorías Web | Mentorízate");

    out.println("    </footer>");
```

```
        out.println("</body>");  
        out.println("</html>");  
    }  
}
```

@Override

```
protected void doPost(HttpServletRequest request, HttpServletResponse response)  
    throws ServletException, IOException {  
    int id = Integer.parseInt(request.getParameter("id_asesoria"));  
    String estado = request.getParameter("estado");  
    String comentario = request.getParameter("comentario");  
  
    BD.ProcesaAsesoria.responderSolicitud(id, estado, comentario);  
    response.sendRedirect("VerSolicitudes");  
}  
}
```

Modelos.

Asesorías.java

```
package modelos;
```

```
/**
```

```
 *
```

```
 * @author aleja
```

```
 */
```

```
public class Asesoria {
```

```
    private int idAsesoria;
```

```
    private int alumnold;
```

```
    private int profesorId;
```

```
    private String materia;
```

```
    private boolean esTuDocente;
```

```
    private java.sql.Date fechaSolicitud;
```

```
    private java.sql.Time horaSolicitud;
```

```
    private String asunto;
```

```
    private String estado;
```

```
    private String comentario;
```

```
    public Asesoria() {}
```

```
    public Asesoria(int idAsesoria, int alumnold, int profesorId, String materia, boolean  
esTuDocente, java.sql.Date fechaSolicitud, java.sql.Time horaSolicitud, String asunto,  
String estado, String comentario) {
```

```
this.idAsesoria = idAsesoria;  
  
this.alumnold = alumnold;  
  
this.profesorId = profesorId;  
  
this.materia = materia;  
  
this.esTuDocente = esTuDocente;  
  
this.fechaSolicitud = fechaSolicitud;  
  
this.horaSolicitud = horaSolicitud;  
  
this.asunto = asunto;  
  
this.estado = estado;  
  
this.comentario = comentario;  
  
}
```

```
public int getIdAsesoria() { return idAsesoria; }
```

```
public void setIdAsesoria(int idAsesoria) { this.idAsesoria = idAsesoria; }
```

```
public int getAlumnold() { return alumnold; }
```

```
public void setAlumnold(int alumnold) { this.alumnold = alumnold; }
```

```
public int getProfesorId() { return profesorId; }
```

```
public void setProfesorId(int profesorId) { this.profesorId = profesorId; }
```

```
public String getMateria() { return materia; }
```

```
public void setMateria(String materia) { this.materia = materia; }
```



```
public boolean isEsTuDocente() { return esTuDocente; }

public void setEsTuDocente(boolean esTuDocente) { this.esTuDocente = esTuDocente; }


public java.sql.Date getFechaSolicitud() { return fechaSolicitud; }

public void setFechaSolicitud(java.sql.Date fechaSolicitud) { this.fechaSolicitud =
fechaSolicitud; }


public java.sql.Time getHoraSolicitud() { return horaSolicitud; }

public void setHoraSolicitud(java.sql.Time horaSolicitud) { this.horaSolicitud =
horaSolicitud; }


public String getAsunto() { return asunto; }

public void setAsunto(String asunto) { this.asunto = asunto; }


public String getEstado() { return estado; }

public void setEstado(String estado) { this.estado = estado; }


public String getComentario() { return comentario; }

public void setComentario(String comentario) { this.comentario = comentario; }
}
```

Estudiante.java

/*

* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
change this license

* Click <nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java> to edit this template

*/

package modelos;

/**

*

* @author aleja

*/

public class Estudiante {

private int idEstudiante;

private String nombre;

private String matricula;

private String contrasena;

private String programa;

public Estudiante() {

}

public Estudiante(int idEstudiante, String nombre, String matricula, String contrasena, String programa) {

 this.idEstudiante = idEstudiante;

 this.nombre = nombre;

 this.matricula = matricula;

 this.contrasena = contrasena;

```
        this.programa = programa;
    }

```

```
public int getIdEstudiante() {
    return idEstudiante;
}

```

```
public void setIdEstudiante(int idEstudiante) {
    this.idEstudiante = idEstudiante;
}

```

```
public String getNombre() {
    return nombre;
}

```

```
public void setNombre(String nombre) {
    this.nombre = nombre;
}

```

```
public String getMatricula() {
    return matricula;
}

```

```
public void setMatricula(String matricula) {

```

```
        this.matricula = matricula;
    }
}
```

```
public String getContrasena() {
    return contrasena;
}
```

```
public void setContrasena(String contrasena) {
    this.contrasena = contrasena;
}
```

```
public String getPrograma() {
    return programa;
}
```

```
public void setPrograma(String programa) {
    this.programa = programa;
}
}
```

Profesor.java

/*

* Click <nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt> to
change this license

* Click <nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java> to edit this template

*/

package modelos;

/**

*

* @author aleja

*/

public class Profesor {

private int idProfesor;

private String nombre;

private String claveProfesor;

private String contrasena;

private String programa;

public Profesor() {

}

public Profesor(int idProfesor, String nombre, String claveProfesor, String contrasena, String programa) {

this.idProfesor = idProfesor;

this.nombre = nombre;

this.claveProfesor = claveProfesor;

this.contrasena = contrasena;

```
        this.programa = programa;
    }
}
```

```
public int getIdProfesor() {
    return idProfesor;
}
```

```
public void setIdProfesor(int idProfesor) {
    this.idProfesor = idProfesor;
}
```

```
public String getNombre() {
    return nombre;
}
```

```
public void setNombre(String nombre) {
    this.nombre = nombre;
}
```

```
public String getClaveProfesor() {
    return claveProfesor;
}
```

```
public void setClaveProfesor(String claveProfesor) {
```

```
    this.claveProfesor = claveProfesor;  
}
```

```
public String getContrasena() {  
    return contrasena;  
}
```

```
public void setContrasena(String contrasena) {  
    this.contrasena = contrasena;  
}
```

```
public String getPrograma() {  
    return programa;  
}
```

```
public void setPrograma(String programa) {  
    this.programa = programa;  
}  
}
```

BDD

Procesaalumno.java

```
package BD;
```

```
import modelos.Estudiante;
```

```
import java.sql.*;
```

```
import java.util.*;
```

```
public class ProcesaAlumno {
```

```
    // Método para obtener todos los alumnos
```

```
    public static LinkedList<Estudiante> obtenerAlumnos() {
```

```
        LinkedList<Estudiante> listaAlumnos = new LinkedList<>();
```

```
        try {
```

```
            // Establece la conexión con la base de datos
```

```
            Connection conexion = Conexion.getConexion();
```

```
            Statement sentenciaSQL = conexion.createStatement();
```

```
            ResultSet rs = sentenciaSQL.executeQuery("SELECT * FROM alumnos");
```

```
            // Procesa los resultados
```

```
            while (rs.next()) {
```

```
                Estudiante alumno = new Estudiante();
```

```
                alumno.setIdEstudiante(rs.getInt("id_alumno"));
```

```
                alumno.setNombre(rs.getString("nombre"));
```



```
        alumno.setMatricula(rs.getString("matricula"));

        alumno.setContrasena(rs.getString("contrasena"));

        alumno.setPrograma(rs.getString("programa"));

        listaAlumnos.add(alumno);
    }

    // Cierra los recursos

    rs.close();

    sentenciaSQL.close();

    conexion.close();
} catch (SQLException e) {

    e.printStackTrace(); // Si hay error, lo imprime
}

return listaAlumnos;
}

// Método para autenticar un alumno con matrícula y contraseña
public static Estudiante autenticarAlumno(String matricula, String contrasena) {

    Estudiante alumno = null;

    try {

        // Establece la conexión con la base de datos

        Connection conexion = Conexion.getConexion();

        String sql = "SELECT * FROM alumnos WHERE matricula = ? AND contrasena = ?";

        PreparedStatement ps = conexion.prepareStatement(sql);
```

```
ps.setString(1, matricula);

ps.setString(2, contrasena);


ResultSet rs = ps.executeQuery();


// Si encuentra un alumno, lo asigna
if (rs.next()) {

    alumno = new Estudiante();

    alumno.setIdEstudiante(rs.getInt("id_alumno"));

    alumno.setNombre(rs.getString("nombre"));

    alumno.setMatricula(rs.getString("matricula"));

    alumno.setContrasena(rs.getString("contrasena"));

    alumno.setPrograma(rs.getString("programa"));

}


// Cierra los recursos

rs.close();

ps.close();

conexion.close();

} catch (SQLException e) {

    e.printStackTrace(); // Si hay error, lo imprime

}

return alumno;

}
```

```
}
```

ProcesaAsesoria.java

```
package BD;
```

```
import java.sql.*;
```

```
import java.util.*;
```

```
public class ProcesaAsesoria {
```

```
    // Asesorías de un alumno por su id
```

```
    public static List<Map<String, Object>> getAsesoriasPorAlumno(int idAlumno) {
```

```
        List<Map<String, Object>> asesorias = new ArrayList<>();
```

```
        String sql = "SELECT * FROM asesorias WHERE alumno_id = ?";
```

```
        try (Connection con = Conexion.getConexion();
```

```
            PreparedStatement ps = con.prepareStatement(sql)) {
```

```
            ps.setInt(1, idAlumno);
```

```
            ResultSet rs = ps.executeQuery();
```

```
            ResultSetMetaData meta = rs.getMetaData();
```

```
            int cols = meta.getColumnCount();
```

```
            while (rs.next()) {
```

```
                Map<String, Object> fila = new HashMap<>();
```

```
                for (int i = 1; i <= cols; i++) {
```

```
                    fila.put(meta.getColumnName(i), rs.getObject(i));
```

```
                }
```

```
        asesorias.add(fila);  
    }  
    } catch (SQLException e) {  
        e.printStackTrace();  
    }  
    return asesorias;  
}
```

// Asesorías de un profesor por su id

```
public static List<Map<String, Object>> getAsesoriasPorProfesor(int idProfesor) {  
    List<Map<String, Object>> asesorias = new ArrayList<>();  
    //String sql = "SELECT * FROM asesorias WHERE profesor_id = ?";  
    String sql = "SELECT a.*, al.nombre, al.matricula, al.programa "  
        + "FROM asesorias a "  
        + "JOIN alumnos al ON a.alumno_id = al.id_alumno "  
        + "WHERE a.profesor_id = ?";  
    try (Connection con = Conexion.getConexion(); PreparedStatement ps =  
con.prepareStatement(sql)) {  
        ps.setInt(1, idProfesor);  
        ResultSet rs = ps.executeQuery();  
        ResultSetMetaData meta = rs.getMetaData();  
        int cols = meta.getColumnCount();  
        while (rs.next()) {  
            Map<String, Object> fila = new HashMap<>();  
            for (int i = 1; i <= cols; i++) {
```

```
        fila.put(meta.getColumnName(i), rs.getObject(i));
    }
    asesorias.add(fila);
}
} catch (SQLException e) {
    e.printStackTrace();
}
return asesorias;
}
```

```
public static void responderSolicitud(int idAsesoria, String estado, String comentario) {
    String sql = "UPDATE asesorias SET estado = ?, comentario = ? WHERE id_asesoria = ?";
    try (Connection con = Conexion.getConnection();
        PreparedStatement ps = con.prepareStatement(sql)) {

        ps.setString(1, estado); // 'aceptada' o 'denegada'
        ps.setString(2, comentario);
        ps.setInt(3, idAsesoria);

        ps.executeUpdate();

    } catch (SQLException e) {
        e.printStackTrace(); // Puedes reemplazar con un logger si estás en producción
    }
}
```

```
}  
}
```

ProcesaProfesor.java

```
package BD;
```

```
import modelos.Profesor;
```

```
import java.sql.*;
```

```
import java.util.*;
```

```
public class ProcesaProfesor {
```

```
    public static Profesor autenticarProfesor(String clave_profesor, String contrasena) {
```

```
        Profesor profesor = null;
```

```
        try {
```

```
            Connection conexion = Conexion.getConexion();
```

```
            String sql = "SELECT * FROM profesores WHERE clave_profesor = ? AND contrasena  
= ?";
```

```
            PreparedStatement ps = conexion.prepareStatement(sql);
```

```
            ps.setString(1, clave_profesor);
```

```
            ps.setString(2, contrasena);
```

```
            ResultSet rs = ps.executeQuery();
```

```
            if (rs.next()) {
```

```
                profesor = new Profesor();
```

```
                profesor.setIdProfesor(rs.getInt("id_profesor"));
```

```
                profesor.setNombre(rs.getString("nombre"));
```

```
        profesor.setClaveProfesor(rs.getString("clave_profesor"));

        profesor.setContrasena(rs.getString("contrasena"));

        profesor.setPrograma(rs.getString("programa"));
    }

    rs.close();

    ps.close();

    conexion.close();
} catch (SQLException e) {

    e.printStackTrace();

}

return profesor;
}
```

```
public static List<Profesor> obtenerProfesores() {

    List<Profesor> listaProfesores = new ArrayList<>();

    try{

        Connection conexion = Conexion.getConexion();

        Statement stmt = conexion.createStatement();

        ResultSet rs = stmt.executeQuery("SELECT * FROM profesores");

        while (rs.next()) {

            Profesor profesor = new Profesor();

            profesor.setIdProfesor(rs.getInt("id_profesor"));

            profesor.setNombre(rs.getString("nombre"));

            profesor.setClaveProfesor(rs.getString("clave_profesor"));

        }

    } catch (SQLException e) {

        e.printStackTrace();

    }

    return listaProfesores;
}
```

```
        profesor.setContrasena(rs.getString("contrasena"));

        profesor.setPrograma(rs.getString("programa"));

        listaProfesores.add(profesor);
    }

    rs.close();

    stmt.close();

    conexion.close();
} catch (SQLException e) {

    e.printStackTrace();

}

return listaProfesores;

}

}
```


BASE DE DATOS:

-- phpMyAdmin SQL Dump

-- version 5.2.1

-- <https://www.phpmyadmin.net/>

--

-- Servidor: 127.0.0.1

-- Tiempo de generación: 20-05-2025 a las 07:24:53

-- Versión del servidor: 10.4.32-MariaDB

-- Versión de PHP: 8.2.12

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";

START TRANSACTION;

SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;

/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;

/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;

/*!40101 SET NAMES utf8mb4 */;

--

-- Base de datos: `asesorias_web`

--

--

-- Estructura de tabla para la tabla `alumnos`

--

CREATE TABLE `alumnos` (

`id_alumno` int(11) NOT NULL,

`nombre` varchar(100) NOT NULL,

`matricula` varchar(20) NOT NULL,

`contrasena` varchar(100) NOT NULL,

`programa` enum('ICC','LCC','ITI') NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

--

-- Volcado de datos para la tabla `alumnos`

--

INSERT INTO `alumnos` (`id_alumno`, `nombre`, `matricula`, `contrasena`,
 `programa`) VALUES

(1, 'Carlos Antonio', '202027778', '202027778', 'ICC'),

(2, 'Jose Isaias Bautista', '202028833', '202028833', 'LCC'),

(3, 'Liam Payne', '202021610', '202021610', 'ITI'),

(4, 'Noe Antonio', '202022020', '202022020', 'ICC'),

```
(5, 'Alejandro Cocolotl', '202021111', '202021111', 'ICC'),  
(6, 'Angel David Salas', '202022222', '202022222', 'ICC');
```

```
-----
```

```
--
```

```
-- Estructura de tabla para la tabla `asesorias`
```

```
--
```

```
CREATE TABLE `asesorias` (  
  `id_asesoria` int(11) NOT NULL,  
  `alumno_id` int(11) NOT NULL,  
  `profesor_id` int(11) NOT NULL,  
  `materia` varchar(100) NOT NULL,  
  `es_tu_docente` tinyint(1) NOT NULL DEFAULT 0,  
  `fecha_solicitud` date NOT NULL,  
  `hora_solicitud` time NOT NULL,  
  `asunto` text NOT NULL,  
  `estado` enum('en proceso','aceptada','denegada') NOT NULL DEFAULT 'en proceso',  
  `comentario` text DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
--
```

```
-- Volcado de datos para la tabla `asesorias`
```

--

```
INSERT INTO `asesorias` (`id_asesoria`, `alumno_id`, `profesor_id`, `materia`,  
`es_tu_docente`, `fecha_solicitud`, `hora_solicitud`, `asunto`, `estado`,  
`comentario`) VALUES
```

```
(1, 1, 1, 'Programación Web', 1, '2025-05-20', '10:00:00', 'Dudas sobre Servlets', 'aceptada',  
'CLARO'),
```

```
(2, 2, 2, 'Estructura de Datos', 0, '2025-05-18', '14:30:00', 'Repaso de árboles binarios',  
'aceptada', 'Nos vemos en el laboratorio 3'),
```

```
(3, 3, 1, 'Bases de Datos', 1, '2025-05-19', '09:00:00', 'Consultas JOIN complejas',  
'denegada', 'Horario ocupado, intente otra fecha'),
```

```
(4, 1, 1, 'Dudas', 1, '2025-05-20', '10:00:00', 'Duda de c', 'aceptada', 'SI SI'),
```

```
(5, 1, 1, 'Dudas de ProgramaciÃ³n', 1, '2025-05-20', '10:00:00', 'dudas sobre programar',  
'denegada', 'LAB OCUPADO'),
```

```
(6, 1, 2, 'Duda servelets', 1, '2025-06-26', '11:00:00', 'Duda sobre servlets', 'aceptada',  
'Claro, nos vemos'),
```

```
(7, 1, 2, 'De servlets', 1, '2025-05-20', '00:00:00', 'Duda de servlets', 'en proceso', NULL),
```

```
(8, 1, 1, 'Duda de programar', 1, '2025-06-25', '11:00:00', 'Dudas de C', 'aceptada',  
'Claro :D');
```

--

-- Estructura de tabla para la tabla `profesores`

--

```
CREATE TABLE `profesores` (
```

```
  `id_profesor` int(11) NOT NULL,
```

```
`nombre` varchar(100) NOT NULL,  
`clave_profesor` varchar(20) NOT NULL,  
`contrasena` varchar(100) NOT NULL DEFAULT 'cambiar123',  
`programa` enum('ICC','LCC','ITI') NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;  
  
--  
  
-- Volcado de datos para la tabla `profesores`  
  
--  
  
INSERT INTO `profesores` (`id_profesor`, `nombre`, `clave_profesor`, `contrasena`,  
`programa`) VALUES  
  
(1, 'Dra. Beatriz Beltrán', '202021', '123456789', 'ICC'),  
(2, 'Dr. Mario Rossainz', '202022', '123456789', 'LCC'),  
(3, 'Garnica Ceron', '202023', '1234567890', 'ICC'),  
(4, 'Ceron Garnica', '2020235', '1234567890', 'LCC');  
  
--  
  
-- Índices para tablas volcadas  
  
--  
  
--  
  
-- Indices de la tabla `alumnos`  
  
--  
  
ALTER TABLE `alumnos`
```

```
ADD PRIMARY KEY (`id_alumno`),  
ADD UNIQUE KEY `matricula` (`matricula`);  
  
--  
-- Indices de la tabla `asesorias`  
--  
ALTER TABLE `asesorias`  
  ADD PRIMARY KEY (`id_asesoria`),  
  ADD KEY `alumno_id` (`alumno_id`),  
  ADD KEY `profesor_id` (`profesor_id`);  
  
--  
-- Indices de la tabla `profesores`  
--  
ALTER TABLE `profesores`  
  ADD PRIMARY KEY (`id_profesor`),  
  ADD UNIQUE KEY `clave_profesor` (`clave_profesor`);  
  
--  
-- AUTO_INCREMENT de las tablas volcadas  
--  
--  
-- AUTO_INCREMENT de la tabla `alumnos`
```

--

ALTER TABLE `alumnos`

MODIFY `id_alumno` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=7;

--

-- AUTO_INCREMENT de la tabla `asesorias`

--

ALTER TABLE `asesorias`

MODIFY `id_asesoria` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=9;

--

-- AUTO_INCREMENT de la tabla `profesores`

--

ALTER TABLE `profesores`

MODIFY `id_profesor` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;

--

-- Restricciones para tablas volcadas

--

--

-- Filtros para la tabla `asesorias`

--

ALTER TABLE `asesorias`

```
ADD CONSTRAINT `asesorias_ibfk_1` FOREIGN KEY (`alumno_id`) REFERENCES
`alumnos` (`id_alumno`) ON UPDATE CASCADE,
```

```
ADD CONSTRAINT `asesorias_ibfk_2` FOREIGN KEY (`profesor_id`) REFERENCES
`profesores` (`id_profesor`) ON UPDATE CASCADE;
```

```
COMMIT;
```

```
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
```

```
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

☐ Mostrar todo | Número de filas: 25 | Filtrar filas: | Ordenar según la clave: Ninguna

Opciones extra

		id_alumno	nombre	matricula	contrasena	programa
<input type="checkbox"/>	Editar Copiar Borrar	1	Carlos Antonio	202027778	202027778	ICC
<input type="checkbox"/>	Editar Copiar Borrar	2	Jose Isaias Bautista	202028833	202028833	LCC
<input type="checkbox"/>	Editar Copiar Borrar	3	Liam Payne	202021610	202021610	ITI
<input type="checkbox"/>	Editar Copiar Borrar	4	Noe Antonio	202022020	202022020	ICC
<input type="checkbox"/>	Editar Copiar Borrar	5	Alejandro Cocolotl	202021111	202021111	ICC
<input type="checkbox"/>	Editar Copiar Borrar	6	Angel David Salas	202022222	202022222	ICC

☐ Seleccionar todo Para los elementos que están marcados: Editar Copiar Borrar Exportar

		id_asesoria	alumno_id	profesor_id	materia	es_tu_docente	fecha_solicitud	hora_solicitud	asunto	estado	comentario
<input type="checkbox"/>	Editar Copiar Borrar	1	1	1	Programación Web	1	2025-05-20	10:00:00	Dudas sobre Servlets	aceptada	CLARO
<input type="checkbox"/>	Editar Copiar Borrar	2	2	2	Estructura de Datos	0	2025-05-18	14:30:00	Repaso de árboles binarios	aceptada	Nos vemos en el laboratorio 3
<input type="checkbox"/>	Editar Copiar Borrar	3	3	1	Bases de Datos	1	2025-05-19	09:00:00	Consultas JOIN complejas	denegada	Horario ocupado, intente otra fecha
<input type="checkbox"/>	Editar Copiar Borrar	4	1	1	Dudas	1	2025-05-20	10:00:00	Duda de c	aceptada	SI SI
<input type="checkbox"/>	Editar Copiar Borrar	5	1	1	Dudas de ProgramaciÃ³n	1	2025-05-20	10:00:00	dudas sobre programar	denegada	LAB OCUPADO
<input type="checkbox"/>	Editar Copiar Borrar	6	1	2	Duda servelets	1	2025-06-26	11:00:00	Duda sobre servlets	aceptada	Claro, nos vemos
<input type="checkbox"/>	Editar Copiar Borrar	7	1	2	De servlets	1	2025-05-20	00:00:00	Duda de servlets	en proceso	NULL
<input type="checkbox"/>	Editar Copiar Borrar	8	1	1	Duda de programar	1	2025-06-25	11:00:00	Dudas de C	aceptada	Claro :D

				id_profesor	nombre	clave_profesor	contrasena	programa
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	1	Dra. Beatriz Beltrán	202021	123456789	ICC
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	2	Dr. Mario Rossainz	202022	123456789	LCC
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	3	Garnica Ceron	202023	1234567890	ICC
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	4	Ceron Garnica	2020235	1234567890	LCC