**Pasos a seguir**

1. **Registro:**
   * Explica el proceso de registro para los usuarios.
   * Indica que todos los campos del formulario de registro son obligatorios.
2. **Inicio de sesión:**
   * El inicio de sesión permitirá a los usuarios acceder al contenido del <main>.
   * Los usuarios podrán iniciar sesión tanto con su nombre de usuario como con su correo electrónico.
   * Deberás conectar las API correspondientes para permitir el inicio de sesión a través de Google, GitHub y Twitter.

Recuerda mencionar que el contenido del <main> solo se mostrará si el usuario ha iniciado sesión correctamente; de lo contrario, no se mostrará el contenido.

1. **Configuración de la base de datos:**
   * Crea una base de datos en MySQL llamada "tu\_base\_de\_datos".
   * Crea una tabla llamada "Pais" con las siguientes columnas:
     + id (clave primaria, autoincremental)
     + nombre\_pais (cadena de caracteres)
   * Inserta los países en la tabla "Pais":
     + INSERT INTO Pais (nombre\_pais) VALUES ('Haiti'), ('Puerto Rico'), ('Venezuela'), ('República Dominicana'), ('Mexico'), ('Peru'), ('Chile'), ('Colombia'), ('Argentina'), ('España'), ('Cuba');
   * Crea una tabla llamada "estudiantes" con las siguientes columnas:
     + id (clave primaria, autoincremental)
     + nombre (cadena de caracteres)
     + apellido (cadena de caracteres)
     + pais (cadena de caracteres)
     + identificacion (cadena de caracteres)
     + telefono (cadena de caracteres)
     + direccion (cadena de caracteres)
     + email (cadena de caracteres)
   * Asegúrate de que la tabla y las columnas tengan los tipos de datos adecuados.
2. **Validación de datos en el formulario de registro:**
   * Antes de enviar el formulario, valida que todos los campos obligatorios estén completos.
   * Realiza validaciones adicionales para los campos nombre, apellido, identificación y email, asegurándote de que no estén vacíos y cumplan con un formato específico.
   * Puedes utilizar funciones de validación en JavaScript o bibliotecas como "validator.js" para facilitar esta tarea.
3. **Funcionalidad de registro de estudiantes:**
   * Asegúrate de que todos los campos requeridos se completen correctamente antes de enviar el formulario.
   * Implementa una función en JavaScript que se ejecute cuando se envíe el formulario y capture los valores ingresados en los campos.
   * Utiliza una solicitud AJAX para enviar los datos del estudiante al servidor y guardarlos en la base de datos.
   * Valida los campos del formulario antes de enviar los datos al servidor para asegurarte de que están completos y cumplen con los criterios requeridos.
   * En el servidor, implementa una ruta y función manejadora correspondiente para recibir la solicitud de registro de estudiantes y realizar la inserción de los datos en la base de datos.
4. **Funcionalidad de búsqueda de estudiantes:**
   * Cuando se presione el botón "Buscar", realiza una solicitud al servidor para buscar un estudiante en la base de datos según el código proporcionado.
   * En el servidor, crea una ruta (endpoint) que maneje la solicitud de búsqueda de estudiantes.
   * En la función manejadora de la ruta, obtén el código proporcionado en la solicitud y realiza una consulta a la base de datos para obtener los datos del estudiante correspondiente.
   * Rellena los campos del formulario de búsqueda con los datos obtenidos de la base de datos.
   * Si no se encuentra ningún estudiante con el código proporcionado, muestra un mensaje indicando que no se encontró ningún resultado.
5. **Funcionalidad de edición de estudiantes:**
   * Deshabilita los campos de edición del formulario (excepto el campo de código) inicialmente.
   * Al presionar el botón "Buscar", realiza la búsqueda del estudiante y, si se encuentra, habilita los campos de edición.
   * Permite al estudiante editar los campos habilitados.
   * Cuando se presione el botón "Guardar" en el formulario de edición, realiza una solicitud al servidor para actualizar los datos del estudiante en la base de datos.
   * En el servidor, crea una ruta que maneje la solicitud de actualización de estudiantes.
   * En la función manejadora de la ruta, obtén los datos del estudiante proporcionados en la solicitud y realiza una consulta a la base de datos para actualizar los datos correspondientes del estudiante.
6. **Funcionalidad de eliminación de estudiantes:**
   * Al presionar el botón "Buscar", realiza la búsqueda del estudiante y, si se encuentra, muestra los datos del estudiante en los campos correspondientes.
   * Deshabilita los campos de edición del formulario (excepto el campo de código) inicialmente.
   * Al presionar el botón "Eliminar Estudiante", realiza una solicitud al servidor para eliminar el estudiante de la base de datos.
   * En el servidor, crea una ruta que maneje la solicitud de eliminación de estudiantes.
   * En la función manejadora de la ruta, obtén el código del estudiante proporcionado en la solicitud y realiza una consulta a la base de datos para eliminar el estudiante correspondiente.