



Prueba de desempeño del módulo JavaScript

Clan: Jeff Bezos

Reglas de la prueba

- **Comunicación:** No está permitido hablar o comunicarse de cualquier forma con otros estudiantes durante la prueba.
- **Integridad Académica:** Cualquier forma de trampa, incluido el plagio, copia o uso de material no autorizado (autogeneradores de código o inteligencia artificial), resultará en una calificación de cero en el examen y puede llevar a sanciones adicionales según las políticas de RIWI.
- **Material Permitido:** Solo se permite ver material de apoyo como lo son diapositivas o ejercicios realizados en clase, y notas tomadas en el cuaderno. Están permitidas las siguientes páginas para acceso:
 - <https://www.w3schools.com/>
 - <https://developer.mozilla.org/es/>
 - <https://getbootstrap.com/>
 - <https://boxicons.com/>
- **Permanencia en el Aula:** Una vez iniciado el examen, no se permite salir del aula hasta haber entregado el examen y descanso cada 3 horas.
- **Entrega:** Una vez finalizada la prueba se debe subir en una carpeta comprimida, con el siguiente nombre: Prueba3_NombreApellido del coder al Moodle, en la carpeta comprimida deben estar el ejercicio, la carpeta con el contenido. La hora máxima de entrega es a las 10:00pm las pruebas enviadas después de esta hora no se tendrán en cuenta.

Esta prueba está diseñada para evaluar tus habilidades en desarrollo web mediante la creación de una aplicación completa que incorpora varias funcionalidades clave.



www.riwi.io



301 732 53 27



Cl. 16 # 55 - 129



Se espera que construyas un sistema de autenticación, que permita el registro y login de usuarios, así como la protección de rutas sensibles mediante un guardián. Además, deberás implementar la persistencia de sesión utilizando Local Storage para asegurar que los usuarios permanezcan autenticados entre sesiones del navegador.

Para la gestión de datos, se utilizará json-server, simulando una base de datos real para realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) y garantizando la consistencia de los datos en la aplicación.

Este proyecto te permitirá demostrar tu competencia en integrar diversas tecnologías y prácticas de desarrollo modernas en un entorno de desarrollo web completo.

En resumen la aplicación web debe contar con las siguientes funcionalidades:

1. Sistema de Autenticación:

- Registro de usuarios (UNICAMENTE el rol administrador).
- Login de usuarios.
- Protección de rutas mediante un guardián.

2. Persistencia de Sesión:

- Uso del Local Storage para mantener la sesión iniciada.

3. Persistencia de Datos:

- Uso de **json-server** para simular una base de datos y manejar operaciones CRUD.

4. Encarpetado de Proyecto

- Generar una organización a nivel estructural de los archivos, para generar una separación de responsabilidades y que sea más sencillo navegar en el proyecto. Genere los archivos que considere necesarios.

Ejemplo:

Carpetas de **assets**, **componentes**, **utilidades**, **servicios**, entre otros.



www.rivi.io



301 732 53 27



Cl. 16 # 55 - 129



```
</Riwi>
project-root/
├── assets/
│   └── (archivos estáticos como imágenes y estilos)
├── data/
│   └── db.js
├── services/
│   ├── hooks.js
│   ├── register.js
│   └── (otros servicios)
├── views/
│   ├── login.html
│   ├── register.html
│   ├── admin.html
│   └── visitor.html
├── index.html
├── styles.css
└── db.json
```

5. Sistema de Administración

- Apartado para usuario administrador que permita la gestión completa de una entidad. (Crear, Actualizar, Eliminar, Obtener).



www.rivi.io



301 732 53 27



Cl. 16 # 55 - 129



Descripción del problema

Se desea crear una página de administración de reserva de vuelos, en la que el agente (admin) de vuelo tendrá la capacidad de generar una serie de reservas a partir de los vuelos disponibles según su flota disponible.

Debes generar un usuario con rol administrador en el archivo db.json que permitirá al usuario administrador loguearse con este rol.

Adicional el usuario administrador podrá editar, y eliminar reservas.

Además, el usuario admin podrá reservar vuelos siempre y cuando estos no hayan superado la capacidad del vuelo. También en esta misma vista podrá visualizar las reservas por vuelo.

En el archivo db.json encontrarás las entidades: **flights** con información pre-establecida, **users** (vacío) y las respectivas **reservations** (vacío).

El proyecto tendrá consigo una carpeta de "assets" para utilizar en el proyecto:

En la vista de registro se le pedirá al usuario

- nombre
- correo
- fecha de nacimiento
- contraseña

Nota Por defecto, los usuarios creados desde esta vista deben tener el rol:*

Visitante

****Se facilitará un proyecto base, el cual se espera se complemente con todos los archivos necesarios para generar una solución completa y separada por responsabilidad. Complementar con estilos propios dará puntos extra!.***

¡Éxitos!



www.riwi.io



301 732 53 27



Cl. 16 # 55 - 129