



# Prueba de desempeño del módulo 3 – JavaScript

Team Leader: Javier Cómbita Téllez

Grupo: Dell

## Reglas de la prueba

- **Comunicación:** Está prohibido hablar o comunicarse de cualquier forma con otros estudiantes durante el examen.
- **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.
- **Idioma:** Toda la interfaz visual y el código debe ser **100%** en inglés, lo único que se permite en español son los comentarios de ayuda y la información en sí de los productos
- **Tecnologías Permitidas**
  - **Frontend:** HTML, CSS, JavaScript
  - **Backend:** json-server
  - **Persistencia:** LocalStorage
- **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://boxicons.com/>
- <https://getbootstrap.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: **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 **(14:00 pm)** las pruebas enviadas después de esta hora no se tendrán en cuenta.
- **Código:** en el código el estudiante debe comentar sus datos al inicio de todo el archivo HTML de esta manera para garantizar la transparencia.  
en las etiquetas meta "organization" deben colocar "RIWI\_CLAN":

```
<!--  
Nombre Completo: Nombre Completo Del Coder  
Clan: Nombre clan  
fecha: fecha del evaluacion de desempeño  
-->  
<!DOCTYPE html>
```

...

y en el encabezado del HTML, Author, y descripción deben ser llenados tambien.

## ACTIVIDAD

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

Se espera que construyas un sistema de autenticación de usuarios, 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.

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

**1. Sistema de Autenticación:**

- Registro de usuarios(admin)
- Login de usuarios(admin)
- Protección de rutas mediante un guardián

**2. Persistencia de Sesión:**

- Uso de Localstorage para mantener la sesión iniciada

**3. Consistencia de Datos:**

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

## Descripción del problema





## Sistema de gestión de Construpro

### Descripción General:

El proyecto consiste en desarrollar un sistema de gestión para la compañía Construpro S.A, accesible únicamente para el administrador. Su objetivo principal es gestionar y publicar el inventario del almacén principal. El sistema permitirá crear, leer, actualizar y eliminar información sobre los productos y sus servicios, utilizando un servidor JSON para almacenar y manejar los datos.

La plataforma debe permitir ver la lista de productos sin estar logueado **pero** para poder administrar los productos debe haber iniciado sesión primero

Por lo tanto un sistema de autenticación con un guardián que me proteja la ruta y me guarde la sesión es requerido

### Requisitos del Sistema:

para la gestión de usuarios administrador, cada usuario debe tener :

- Nombre Completo
- Nombre de usuario (nombre clave sin espacios)
- Correo electrónico
- Contraseña

Cada producto debe incluir como mínimo el nombre, imagen(url), tipo de producto, el precio por unidad, la cantidad de unidades y una pequeña descripción del producto

Tipos de productos:

- Material
- Maquinaria
- Herramientas

### Funcionalidades GUARDIAN:

- la vista de dashboard debe ser protegida y solo se puede acceder a ella después de iniciar sesión
- Si existe una sesión iniciada **NO** debo tener acceso a la vista del login, me debe redirigir a la vista del dashboard

### Funcionalidades CRUD:

1. **Crear:**



- Agregar nuevos productos con sus respectivos tipos, como mínimo debemos crear 2 productos de cada tipo.

## 2. Leer:

- Mostrar los productos disponibles en el almacén.
- Únicamente en la sección de dashboard debo ver el valor total de cada producto (precio unidad \* valor total)

## 3. Modificar:

- Actualizar el estado de productos (**agotado-disponible**), además de la información básica de cada producto

## 4. Eliminar:

- Quitar productos del sistema.