

Filtro Modulo 3 – JavaScript

Nombre: _____

Trainer: Kevin Wilfred Mejia Torres

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, copiado o uso de material no autorizado, 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, si se requiere ver estos materiales de apoyo se debe comunicar al Trainer.
- **Permanencia en el Aula:** Una vez iniciado el examen, no se permite salir del aula hasta haber entregado el examen y, de preferencia, hasta que haya transcurrido al menos la mitad del tiempo asignado.
- **Entrega:** Una vez finalizada la prueba se debe subir los archivos a Moodle en su respectivo apartado la hora máxima de entrega es a las 2:00 pm, las pruebas enviadas después de esta hora se tendrán en cuenta

Objetivo general

Desarrollar una aplicación web llamada "Riwi Jobs" utilizando JSON Server como backend falso y JavaScript Vanilla para el frontend. La aplicación tendrá cuatro vistas principales: una pública para listar puestos de empleo, una para el inicio de sesión, una para el registro de empresas y otra para la administración de vacantes por empresa.

Herramientas Necesarias

- JSON Server
- HTML
- CSS
- JavaScript Vanilla

Especificaciones

Vista Pública: Listado de Vacantes

Objetivo: Mostrar todas las vacantes disponibles.

Funcionalidad:

- Al cargar la página, se deben realizar peticiones GET al JSON Server para obtener las vacantes registradas sin importar la empresa.
- Mostrar cada vacante con su imagen, título, descripción, locación y fecha.
- El buscador debe ser funcional, se debe filtrar la vacante que su título coincida con la búsqueda del usuario, y si también por la modalidad, los filtros deben ser acumulables.

Vista de Login

Objetivo: Permitir a las empresas iniciar sesión.

Funcionalidad:

- Al enviar el formulario, hacer una petición GET con el parámetro de búsqueda por email al JSON Server para validar las credenciales.

- Si las credenciales son correctas, redirigir a la vista de administración.
- Una vez autenticado se debe guardar la sesión de la empresa en el localStorage.

Puntos adicionales

- La implementación de un guardián o middleware para proteger la vista de administración contará como puntos adicionales.

Vista de registro

Objetivo: Dar la funcionalidad a las empresas para registrarse en el sistema.

Funcionalidad

- Registrar una compañía
- Antes de registrar la compañía se debe validar que no esté registrada en el JSON-Server, por lo tanto, el email y Nit deben ser únicos.
- Validar que la contraseña tenga como mínimo 6 caracteres

Vista de Administración

- **Objetivo:** Permitir a las empresas gestionar sus vacantes.

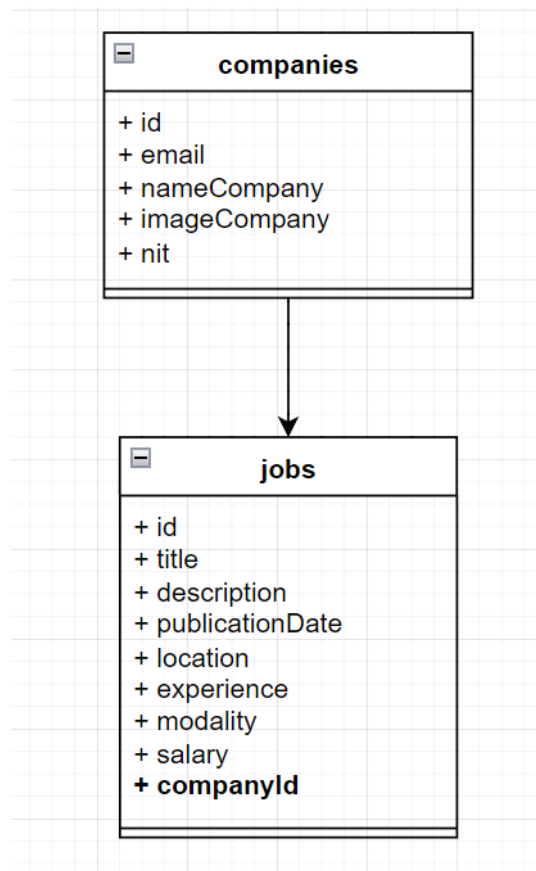
Funcionalidades:

Gestión de Vacantes:

- Listar todas las vacantes con opciones para editar o eliminar (listar solo las vacantes de la empresa autenticada).
- Formulario para añadir una nueva Vacante.
- Al editar o añadir una noticia, hacer una petición PUT o POST respectivamente al JSON Server.

Tener en cuenta que existen relaciones entre las dos entidades nombradas hasta el momento, Donde una compañía puede tener muchas vacantes a su nombre.

Diagrama de relaciones.



Rubrica de evaluación

Criterios de Evaluación	Ponderación
Sustentación (40%)	
1. Claridad y Coherencia	10%
2. Dominio del Contenido	20%
3. Argumentación y Justificación	10%
Código (60%)	
Funcionalidad (50%)	
1. Aplicación de temas vistos (DOM, JSON o APIs,...)	15%
2. Modulacion	5%
3. Funcionalidad en la solución del problema planteado	30%
Calidad del Código (10%)	
1. Buenas Prácticas (reutilización, accesibilidad)	5%
2. Eficiencia y Rendimiento(no redundancia)	5%
Total	100%