

Guía de la Prueba de desempeño JavaScript

Nombre de la prueba: Crudtask

Módulo de formación al que corresponde: JavaScript

Duración de la prueba: la prueba tendrá un calendario semanal con la siguiente estructura para las jornadas AM y PM:

| Hora | Día 1 | Día 2 | Día 3 | Día 4 | Día 5 |
|-------------------|----------------|--|--|--|---|
| 06:00 – 07:00 | Prueba técnica | Sustentación de la solución entregada (individual)* <i>Con previa cita asignada</i> | Sustentación de la solución entregada (individual)* <i>Con previa cita asignada</i> | Feedback personalizado por parte de tu TL* | Carga de resultados y feedback a Moodle por parte de tu TL* |
| 07:00 – 08:00 | | | | | |
| 08:00 – 08:20 | Break 1 | | | | |
| 08:20 – 09:00 | Prueba técnica | | | | |
| 09:00 – 10:00 | | | | | |
| 10:00 – 10:20 | Break 2 | | | | |
| 10:20 – 11:00 | Prueba técnica | | | | |
| 11:00 – 12:00 | | | | | |
| 12:00 – 12:20 | Break 3 | | | | |
| 12:20 – 13:00 | Prueba técnica | | | | |
| 13:00 – 14:00 | | | | | |
| Cambio de jornada | | | | | |
| 14:00 – 15:00 | Prueba técnica | Sustentación de la solución entregada (individual)* <i>Con previa cita asignada</i> | Sustentación de la solución entregada (individual)* <i>Con previa cita asignada</i> | Feedback personalizado por parte de tu TL* | Carga de resultados y feedback a Moodle por parte de tu TL* |
| 15:00 – 16:00 | | | | | |
| 16:00 – 16:20 | Break 1 | | | | |
| 16:20 – 17:00 | Prueba técnica | | | | |
| 17:00 – 18:00 | | | | | |
| 18:00 – 18:20 | Break 2 | | | | |
| 18:20 – 19:00 | Prueba técnica | | | | |
| 19:00 – 20:00 | | | | | |
| 20:00 – 20:20 | Break 3 | | | | |
| 20:20 – 21:00 | Prueba técnica | | | | |
| 21:00 – 22:00 | | | | | |

Notas: *Durante **toda la jornada** tendrás trabajo **individual** asignado en el cual deberás estar avanzando, mientras eres llamado a tu cita de sustentación o feedback en la sala de entrenamiento habitual. Durante el **día 5** tu TL no estará disponible para resolver dudas, pero cuentas con tu tutor oficial.

Temas abordados:

- Manipulación avanzada del **DOM**.
- **Persistencia de datos** mediante Local Storage.
- **Consumo de APIs** con Fetch y manejo de promesas.
- Creación de **rutas dinámicas** en una SPA.
- **Autenticación de usuarios** y gestión de roles (admin y visitante).
- Uso de **json-server** para simular una base de datos.

- **Validaciones**, manejo de errores y consistencia de datos.
- Aplicación de **buenas prácticas de programación** en proyectos JavaScript.

Herramientas y recursos:

Hardware requerido:

Laptop o computadora con conexión estable a internet (otorgada por Riwi).

Software requerido:

- Navegador web (Google Chrome o Mozilla Firefox).
- Visual Studio Code (VS Code).
- Node.js versión 18 o superior.
- json-server.

Objetivo de la prueba: Identificar las capacidades del coder para **aplicar los conocimientos teóricos y prácticos** adquiridos en JavaScript, desarrollando una aplicación **SPA (Single Page Application)** que implemente autenticación, rutas protegidas, persistencia de sesión y consumo de APIs.

La prueba evalúa la habilidad para:

- Estructurar y desarrollar una aplicación modular en JavaScript.
- Simular una base de datos con json-server y realizar operaciones CRUD.
- Implementar autenticación con persistencia en Local Storage.
- Aplicar validaciones, manejo de errores y buenas prácticas.
- Documentar el proyecto mediante un **README.md profesional**.
- Defender técnicamente la solución ante el TL, explicando decisiones de diseño y desarrollo.

Estructura de la prueba: Esta prueba está basada en un caso de uso empresarial: un **Sistema de Gestión de Eventos**, que representa escenarios reales donde los usuarios pueden autenticarse, crear y gestionar eventos.

Parte 1 – Análisis de requisitos

- Identificación de entidades principales (usuario, evento).
- Diseño del flujo de autenticación y roles.
- Diagrama de rutas y estructura base de la SPA.

Parte 2 – Modelado y persistencia de datos

- Creación del archivo db.json simulando la base de datos.
- Configuración de json-server y endpoints CRUD.
- Uso de Local Storage para persistir la sesión del usuario.

Parte 3 – Implementación de funcionalidades

- Registro y login de usuarios con roles.
- Protección de rutas y redirecciones.
- Creación, edición, eliminación y visualización de eventos.



- Validación de capacidad máxima en registros de eventos.

Parte 4 – Manejo de errores y validaciones

- Implementación de try...catch en las operaciones Fetch.
- Validación de campos vacíos y credenciales incorrectas.
- Mensajes descriptivos para el usuario en errores y confirmaciones.

Parte 5 – Documentación y entrega final

- Elaboración de README.md profesional con:
 - Descripción del sistema.
 - Instrucciones para ejecutar el proyecto.
 - Capturas de pantalla y datos del coder.
- Preparación de la sustentación individual.

Formato de las preguntas: el formato determinado para esta prueba será **Caso de uso**. La prueba bajo este formato mide las habilidades de programación y resolución de problemas.

Permisos y limitaciones:

Se permite el uso de la IA bajo los términos establecidos en los lineamientos y, además, se permite ver material de apoyo como el contenido E-learning o ejercicios realizados en sesiones de entrenamiento y notas tomadas durante las sesiones de entrenamiento, ya sea en el cuaderno o de forma digital, acordando con el TL cuáles son los recursos que va a usar, previo inicio 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, resultará en una calificación de cero en el examen y puede llevar a sanciones adicionales según las políticas de RIWI.

Idioma: Todo el código debe ser **100% en inglés**; eso incluye el README y diagramas; lo único que se permite en español son los datos en sí y la interfaz visual de la prueba

Implicaciones:

- Esta prueba determina tu continuidad en el programa de entrenamiento, basado en el **Capítulo IV - Evaluación y calificación** del *Reglamento interno de convivencia Riwi*.
- Esta prueba tiene como objetivo identificar y evaluar el progreso del coder y su capacidad para resolver problemas prácticos.
- Se calificará tanto la solución técnica como la calidad de la sustentación.

Evaluación:



- **Rúbrica de evaluación:** en la plataforma Moodle podrás consultar los criterios de evaluación asociados a esta prueba; te sugerimos revisarlos detenidamente para que enfoques tus esfuerzos en lo importante. Se evaluarán aspectos como funcionalidad, dominio del coder en JavaScript, documentación y uso de git.
- **Ejecución de la entrega:** mediante la plataforma oficial Moodle en el respectivo espacio de entrega asignado por Riwi.
- **Visualización de los resultados:** en la plataforma oficial Moodle.
 - Fecha de visualización: al final del quinto día de la semana de evaluación.
 - Lugar de la retroalimentación: en la sala habitual de entrenamiento durante el cuarto día de la semana de pruebas.

¡Éxitos en este evento evaluativo!