

## Notas:

- TEST:
  - a. Las respuestas a las preguntas se realizarán en el documento TEST-HOJA DE RESPUESTAS indicando vuestro nombre dentro del documento
  
- Ejercicio 1: LISTADO DE TAREAS EN Vue
  - a. Un campo de texto para introducir la descripción de una nueva tarea y un botón para agregarla a la lista.
  - b. Botones para filtrar las tareas visualizadas: todos, pendientes y completados.
  - c. Visualización de la lista de tareas, mostrando la descripción de cada tarea.
  - d. Cada tarea debe poder marcarse como completada al hacer clic sobre ella.
  - e. Mostrar dinámicamente el número de tareas pendientes.
  - f. Para cada tarea, **mostrar la fecha de registro y, si está completada, la fecha de completado.**
  - g. Utilizar Vue.js para la gestión de la aplicación.
  - h. Implementar métodos en Vue para agregar tareas, cambiar su estado de completado, y filtrarlas.
  - i. Implementar la lógica para registrar automáticamente la fecha y hora actuales cuando se agrega una nueva tarea y cuando se completa.
  
  - j. Criterios de Evaluación:  
Funcionalidad: La aplicación debe cumplir con todos los requisitos funcionales descritos.  
Calidad del Código: El código debe ser limpio, bien organizado, y fácil de leer.
  
- Ejercicio 2: VALIDACIÓN FECHA
  - a. Implementar la lógica para validar si la fecha introducida es correcta considerando el número de días de cada mes y si el año es bisiesto.
  - b. Si la fecha es inválida, mostrar un mensaje indicando que la fecha introducida no es válida.
  - c. Si la fecha es válida, mostrar tres bloques de información, uno encima del otro y con un tamaño fijo e igual para los tres:
  - d. El primero indica si la fecha es correcta.
  - e. El segundo indica si el año es bisiesto.
  - f. El tercero muestra la fecha completa, incluyendo el día de la semana.
  - g. Usa JavaScript puro para manejar la validación del formulario y la lógica para calcular si una fecha es correcta y si el año es bisiesto.
  
  - h. Criterios de Evaluación:  
Funcionalidad: La página debe cumplir con todas las funcionalidades descritas.  
Validación: Debe implementarse correctamente la lógica de validación de fechas.  
Calidad del Código: El código debe estar bien organizado, ser legible y comentado adecuadamente.
  
- Ejercicio 3: VALIDACIONES DE FORMULARIO
  - a. Implementar validaciones específicas para asegurar que la información suministrada por el usuario sea correcta y completa antes de ser enviada

- b. Mostrar mensajes de error específicos cuando la información proporcionada no cumpla con las validaciones.
- c. Impedir el envío del formulario hasta que todos los campos obligatorios sean llenados correctamente y las validaciones sean superadas.

d. Criterios de Evaluación:

Funcionalidad: Cumplimiento de todos los requisitos funcionales y validaciones.

Calidad del Código: Claridad, eficiencia y organización del código.

- La entrega del examen se realiza en la misma carpeta que descargasteis. Poniendo vuestro nombre y comprimiéndola