

Validación y Verificación de Software (INF780)

Práctica No 1

Fecha de emisión: Lunes, 6 de octubre de 2025

Fecha de presentación: Lunes, 13 de octubre de 2025

Desarrolla una **API REST de Eventos/Recordatorios** usando **NestJS + TypeORM + MySQL** que permita:

- CRUD completo de eventos.
- Consultar **eventos próximos** (a partir de “ahora”).
- Validar que fecha + hora no esté en el pasado al **crear** (y opcionalmente al **actualizar**).
- Ordenar por fecha y hora ascendente.

Modelo sugerido (referencial, puedes variarlo)

Evento { id, titulo, descripcion?, fecha(YYYY-MM-DD), hora(HH:mm o HH:mm:ss), createdAt, updatedAt }

Reglas mínimas

- título requerido (3–120 caracteres).
- fecha formato válido YYYY-MM-DD.
- hora formato válido HH:mm o HH:mm:ss.
- Al crear: fecha+hora >= now() (requerido).
- Al listar: orden por fecha, luego hora (ASC).
- Respuestas con **códigos HTTP** adecuados (201, 200, 204, 400, 404, etc.).

Endpoints requeridos

Base de recursos: /eventos

1) Crear evento

- **POST** /eventos
- **Body (JSON):**
- { "titulo": "Texto", "descripcion": "Opcional", "fecha": "YYYY-MM-DD", "hora": "HH:mm" }
- **Respuestas:**
 - 201 Created (objeto creado)
 - 400 Bad Request (validación o fecha/hora pasada)

2) Listar eventos

- **GET** /eventos
- **Respuestas:**
 - 200 OK (lista)

3) Obtener evento por id

- **GET** /eventos/:id
- **Respuestas:**
 - 200 OK (objeto)
 - 404 Not Found

4) Actualizar evento (parcial o total)

- **PATCH** /eventos/:id
 - **Body (JSON, parcial):** mismos campos que crear, todos opcionales.
 - Validación opcional: si se cambia fecha u hora, verificar que fecha+hora >= now().
- **Respuestas:**
 - 200 OK (objeto actualizado)
 - 400 Bad Request (validación)
 - 404 Not Found

5) Eliminar evento

- **DELETE** /eventos/:id
- **Respuestas:**
 - 204 No Content
 - 404 Not Found