

# Pre-Enunciado de Parcial 2

---

**Objetivo:** Registro de Partidas de Juegos de Mesa

---

Fecha: Martes 24 de junio 15:00 h

## Contexto General

Se desea construir una aplicación web para registrar partidas jugadas de distintos **juegos de mesa**, utilizando una base de datos con juegos precargados y permitiendo gestionar libremente la información relacionada con cada partida jugada.

La aplicación deberá estar compuesta por:

- Una base de datos con información precargada de **juegos**.
- Funcionalidad completa de ABM sobre **partidas**.
- Listado y filtrado de partidas según criterios establecidos.
- Backend implementado con Node.js, Express y Sequelize.
- Frontend desarrollado con React (Vite) y Bootstrap.

## Modelo de Datos

Tabla **JUEGOS** (no editable por el alumno)

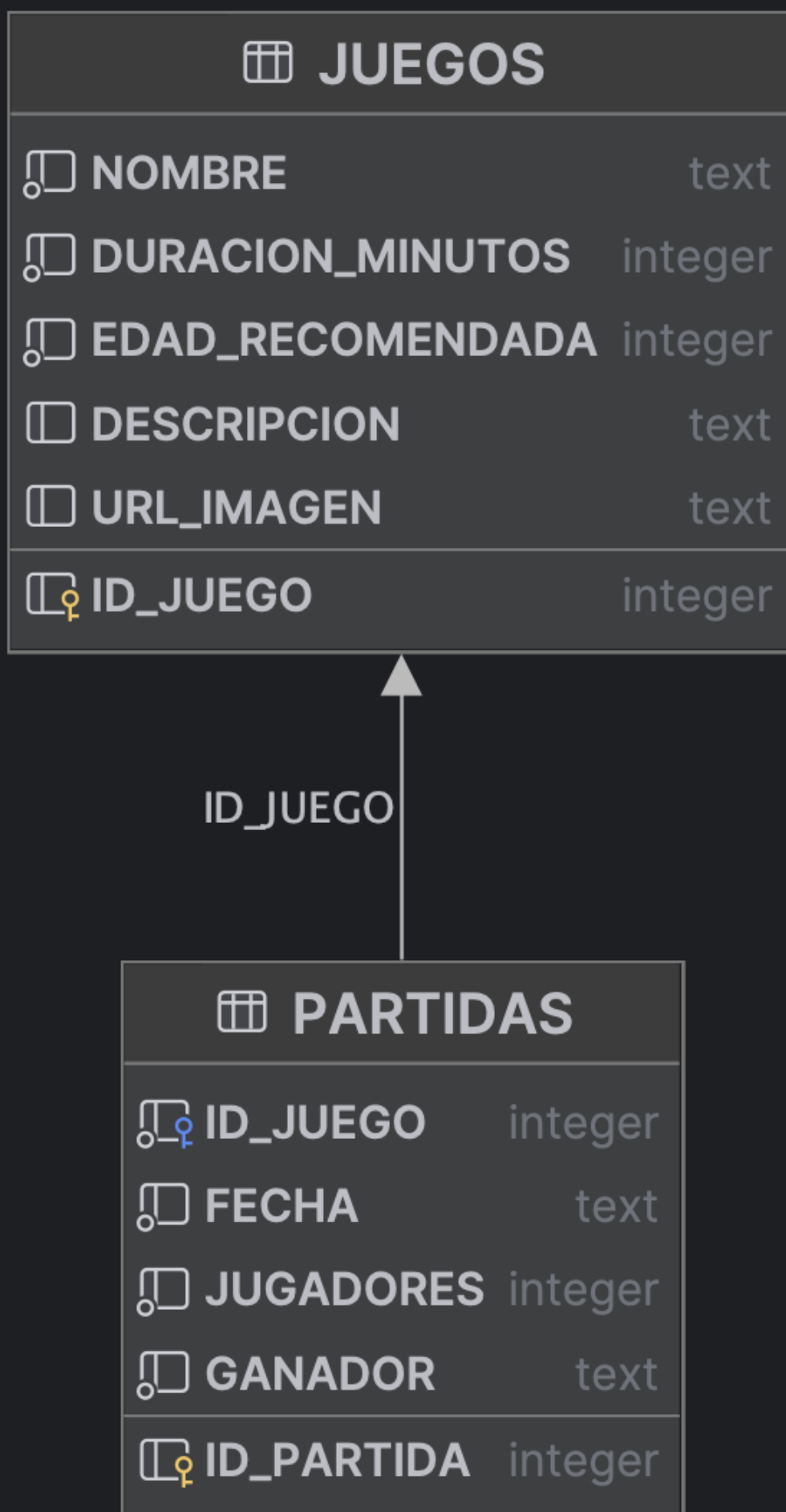
- **ID\_JUEGO** INTEGER PRIMARY KEY AUTOINCREMENT
- **NOMBRE** TEXT NOT NULL
- **DURACION\_MINUTOS** INTEGER NOT NULL
- **EDAD\_RECOMENDADA** INTEGER NOT NULL
- **DESCRIPCION** TEXT
- **URL\_IMAGEN** TEXT

Tabla **PARTIDAS**

- **ID\_PARTIDA** INTEGER PRIMARY KEY AUTOINCREMENT
- **ID\_JUEGO** INTEGER NOT NULL (clave foránea a **JUEGOS**)
- **FECHA** TEXT NOT NULL (formato **YYYY-MM-DD**)
- **JUGADORES** INTEGER NOT NULL
- **GANADOR** TEXT NOT NULL

La relación entre juegos y partidas es de uno a muchos (un juego puede tener muchas partidas registradas).

DER de la Base de Datos



## Requisitos del Backend

- ABM completo sobre **partidas** con validaciones:
  - La cantidad de jugadores debe ser **mayor o igual a 2**.
  - El nombre del ganador no puede estar vacío.
  - La fecha debe estar en un formato válido.
  - Debe validarse que el juego asociado exista.
- Listado y búsqueda de partidas:
  - Filtro por **juego**.
  - Filtro por **fecha límite (hasta)** (si está vacío mostrar siempre desde las más actuales).
  - Límite de 25 resultados, ordenados por fecha descendente.
- Los endpoints deben devolver mensajes adecuados y respetar códigos HTTP.

## Requisitos del Frontend

- Pantalla de listado y filtro de partidas:
  - Mostrar la información de cada partida, incluyendo el nombre del juego.
  - Agregar además la posibilidad de hacer click en el nombre del juego para ver un cuadro desplegable con el resto de la información del juego.
  - Formularios para aplicar filtros.
  - Botones para editar o eliminar cada partida.
- Formulario de alta y modificación de partidas:
  - Inputs validados para todos los campos.
  - Select para elegir el juego (mostrando el nombre).
  - Mensajes claros ante errores o validaciones.
- Carga inicial: el sistema mostrará por defecto las **últimas 10 partidas registradas**.
- ✨ Se espera una visualización enriquecida: en el listado, el alumno podrá implementar opcionalmente un "popup" (ventana emergente) o tooltip con la descripción y detalles del juego asociado., el popup puede ser mediante bootstrap o mediante un alert de Javascript pero debe contener, además del Nombre del juego, al menos:
  - 🖼 Imagen (si el link de la imagen está roto debe ser reemplazado con una imagen de carátula por defecto)
  - 🧠 Descripción
  - ⌚ Duración
  - 🧑 Edad mínima

## Importante

El día del parcial se agregará una **funcionalidad adicional vinculada a la tabla de partidas**. La base de datos original provista podrá sufrir modificaciones que deberán ser integradas a la solución existente.

**Fin del pre-enunciado.**