

2da EVALUACIÓN PARCIAL

Aplicación de Gestión de Ciudades

**Objetivo:**  
Desarrollar el frontend de una aplicación full stack de gestión de ciudades de la Argentina.

- Tecnologías:**
- **Backend (Provisto):** Node.js, Express.js, Sequelize con SQLite.
  - **Frontend:** React.js, Bootstrap, Axios, react-hook-form, react-router-dom.
  - **IDE recomendado:** Visual Studio Code.

Base de datos

Tabla: ciudades

Campo	Tipo	Descripción	Restricciones
id	INTEGER	Identificador único de la ciudad	PRIMARY KEY , AUTO_INCREMENT , NOT NULL
ciudad	STRING	Nombre de la ciudad	NOT NULL
provincia	STRING	Provincia en donde está ubicada la ciudad	NOT NULL
poblacion	INTEGER	Cantidad de habitantes	NOT NULL

Backend: provisto

El backend del proyecto se incluye y está funcional. Para ejecutar el backend podés hacerlo con el script `dev`

```
npm run dev
```

Los endpoints implementados son los siguientes

sa		
GET	/api/ciudades/	Obtiene todas las ciudades
GET	/api/ciudades? nombre=chi&provincia=ri&orden=poblacion	Obtiene todas las ciudades filtradas por nombre y provincia y ordenadas por un campo específico
GET	/api/ciudades/{id}	Obtiene una ciudad por Id
POST	/api/ciudades/	Crea una ciudad
PUT	/api/ciudades/{id}	Modifica los datos de una ciudad
DELETE	/api/ciudades/{id}	Elimina una ciudad

Requisitos Funcionales que debés implementar

1. **Opción de la barra de navegación**  
En la barra de navegación de la aplicación debe incluir una nueva opción: Ciudades. Esta opción debe navegar hacia el componente
2. **Listado de ciudades**  
Debe haber un campo que permita filtrar los utilitarios por el año Debe haber una lista desplegable de estados ("todos", "activo", "mantenimiento", "baja") que de permita filtrar los utilitarios por estado. Los filtros tienen que actuar en conjunto si el año está en blanco y el estado seleccionado es "todos" debe mostrar todos los utilitarios. Si uno de los dos filtros está en blanco o "todos" solamente va a filtrar por el criterio que tiene un valor.

### 3. Formato del listado

Al lado de cada utilitario exhibido en la tabla debe haber un botón que permita eliminar el utilitario de la tabla de la base de datos. Preferentemente debe haber un dialogo de confirmación previo a la eliminación (podés utilizar `confirm()`). Una vez eliminado se tiene que cargar la tabla con el listado actualizado de utilitarios.

### 4. Botón eliminar

Al lado de cada ciudad en la tabla debe haber un botón que permita eliminar la ciudad de la tabla de la base de datos. Preferentemente debe haber un dialogo de confirmación previo a la eliminación (podés utilizar `confirm()`). Una vez eliminada la ciudad se tiene que cargar la tabla con el listado actualizado de ciudades sin interacción del usuario.

## Requisitos Técnicos

---

- Frontend:
  - Implementado con React.js
  - Estilos y apariencia con: Bootstrap, Tailwind, css
  - Interacción con el backend usando Axios
  - Forms (en el caso de utilizarse) con react-hook-form
  - Navegación y rutas con react-router-dom

## Tiempo de resolución

---

Tenés un tiempo de 75 minutos para resolver el ejercicio

## Entrega

---

- Comprimir la carpeta `frontend/` en un `.zip` sin incluir `node_modules`.
- Nombrar el archivo como: `parcial_2_3k5_<legajo>_<apellido>.zip`. Remplazá `<legajo>` y `<apellido>` por tu legajo y tu apellido.
- Subir el archivo comprimido a la UV (moodle) en el parcial
- Permanecer en el lugar *con la aplicación funcionando* hasta que se realice el coloquio.

## Evaluación

---

### Rúbrica (10 puntos):

1. Opción de barra de navegación y navegación – 1 punto
2. Componente con el listado de ciudades – 4 puntos (\*)
3. Botón eliminar en cada fila – 1 punto
4. Eliminación del la ciudad – 3 puntos (\*)
5. Organización del código y uso de Bootstrap – 1 punto

Los items marcados con (\*) deben estar completamente funcionales para aprobar el parcial.

**Aprobación:** 6 puntos y la impmentación

**Coloquio (10 puntos)** Si el caso práctico resulta aprobado debés dar un coloquio consistente en el agregado de nuevos requisitos al frontend que deberás implementar sin ayuda de herramientas de AI. Es posible que el éste también incluya

**Nota del parcial** La nota del parcial resultará del promedio entre las dos notas. Para aprobar **es requisito que ambas instancias** de aprobación estén aprobadas.