

## PROYECTO PARCIAL 1

**Contexto del Proyecto:** Desarrolle una aplicación web que implemente el caso de estudio asignado (revisar asignación de casos), por ejemplo:

1. Sistema de gestión de **estudiantes y cursos**.
2. Plataforma de administración de **productos y órdenes**.
3. Sistema de gestión de **libros y autores**.
4. Sistema de red social con **usuarios y grupos**.
5. Plataforma de música que administre **playlists y canciones**.

La aplicación debe utilizar **Custom Elements**, **Shadow DOM**, **API REST**, **slots**, y un **menú de navegación** para moverse entre las funcionalidades principales. Debe ser completamente desarrollado con **arrow functions** para las definiciones de funciones y debe incluir templates reutilizables en caso de ser necesario.

### Requisitos del Proyecto:

1. **Estructura del Proyecto:**
  - Cree una página principal que permita navegar entre las secciones del sistema utilizando un menú de navegación.
  - Cada sección debe representar una entidad (por ejemplo, Estudiantes, Cursos, Playlists, Canciones, etc.).
  - Implementar componentes reutilizables para listas, formularios, y mensajes de retroalimentación (como confirmaciones o errores).
2. **API REST:**
  - Desarrolle o consuma una API REST que permita realizar las siguientes operaciones para cada entidad involucrada en la relación m a n:
    - **Listar:** Mostrar los datos existentes.
    - **Crear:** Permitir agregar nuevos registros.
    - **Actualizar:** Modificar registros existentes.
    - **Eliminar:** Borrar registros.
  - Los datos deben manejarse mediante fetch con peticiones a la API (GET, POST, PUT y DELETE).
3. **Custom Elements:**
  - Cree **elementos personalizados** para representar las entidades principales (por ejemplo, <student-list>, <course-form>, <author-list>, etc.).
  - Cada elemento debe:
    - Usar el **Shadow DOM** para encapsular su contenido y estilo.
    - Implementar funcionalidad específica como manejo de eventos, renderizado dinámico y consumo de la API REST.
  - **Header:** Una barra de navegación que incluye el título de la aplicación.
  - **Footer:** Información de pie de página que se muestra en la parte inferior de la aplicación.
4. **Uso de Slots:**
  - Utilice **slots** en los componentes personalizados para permitir la inserción dinámica de contenido, como encabezados, acciones adicionales o mensajes.
5. **Templates:**
  - Diseñe templates reutilizables para el renderizado de listas, formularios y tablas.

- Los templates deben ser dinámicos y actualizables según los datos obtenidos desde la API REST.
- 6. **Menú de Navegación:**
  - Implemente un **menú de navegación** con un componente personalizado (<nav-bar>), que permita cambiar entre las vistas de las entidades.
  - El menú debe ser funcional utilizando Custom Elements, con al menos las siguientes opciones:
    - Inicio
    - Entidad 1: Opciones para registrar, listar y eliminar
    - Entidad 2: Opciones para registrar, listar y eliminar
    - Entidad intermedia: Listado de elementos
    - Acerca de: Datos del estudiante.
- 7. **Estilo y Responsividad:**
  - Aplique estilos encapsulados dentro del Shadow DOM para garantizar la cohesión visual.
  - Garantice que la aplicación sea completamente responsiva.
- 8. **Uso de Arrow Functions:**
  - Todas las funciones, ya sean de manipulación del DOM, gestión de eventos o consumo de datos de la API, deben estar definidas como **arrow functions**.

#### **FORMA DE EVALUACIÓN:**

- Subir el código fuente a un repositorio público de GitHub.
- Cada estudiante deberá presentar una exposición del trabajo realizado.

### Asignación de Casos

Nota: En los casos de estudio se incluyen los atributos mínimos, el estudiante debe implementar los atributos que hagan falta.

ESTUDIANTES	CASO
ALVAREZ CORDOVA IAN JOEL	Caso 1
AMAGUA CASTAÑEDA CHRISTIAN ISMAEL	Caso 2
ARROYO VIZUETE SANTIAGO XAVIER	Caso 3
CHAFLA CARGUACUNDO KAREM ESTEFANIA	Caso 4
COLLAGUAZO AMAGUA LUIS ELIAN	Caso 5
CORONADO ACHIG MICHAEL ANDRES	Caso 6
CUASQUER CHISAGUANO MISHHELL ESTEFANIA	Caso 7
CUEVA CANGAS WENDY NICOLE	Caso 8
GAHONA PATIÑO JORDAN ROBERTO	Caso 9
MARTINEZ GUERRERO IRVING WILLIAM	Caso 10
OLALLA ROMERO KRIS SEBASTIAN	Caso 11
PEREZ MERO ARIEL MATEO	Caso 12
PERUGACHI CACOANGO KARLA MICHELLE	Caso 13
RAMOS PUJOTA KEVIN JESUS	Caso 14
RIVERA ESPINOSA STEPHANI JAMILETTE	Caso 15
TRUJILLO CAÑADAS NICOLE ESTEFANIA	Caso 16
VEGA GAROFALO BRAYAN JASMANY	Caso 17
VELASQUEZ MALES TUPAC YUPANQUI	Caso 18
VINUEZA FLORES MELANY ANAHI	Caso 19

## CASOS DE ESTUDIO

### 1. Sistema de Gestión de Cursos y Estudiantes

- **Descripción:** Un estudiante puede inscribirse en múltiples cursos, y un curso puede tener múltiples estudiantes.
- **Entidades:**
  - **Estudiantes:** id\_estudiante, nombre, apellido, correo
  - **Cursos:** id\_curso, nombre\_curso, descripcion
- **Relación:**
  - Tabla intermedia: Inscripciones con los atributos id\_estudiante y id\_curso.

### 2. Sistema de Gestión de Proyectos y Empleados

- **Descripción:** Un empleado puede trabajar en varios proyectos, y un proyecto puede tener múltiples empleados asignados.
- **Entidades:**
  - **Empleados:** id\_empleado, nombre, puesto, salario
  - **Proyectos:** id\_proyecto, nombre\_proyecto, fecha\_inicio, fecha\_fin
- **Relación:**
  - Tabla intermedia: Participaciones con los atributos id\_empleado, id\_proyecto y rol.

### 3. Sistema de Biblioteca: Libros y Autores

- **Descripción:** Un libro puede tener múltiples autores, y un autor puede haber escrito múltiples libros.
- **Entidades:**
  - **Libros:** id\_libro, titulo, genero, fecha\_publicacion
  - **Autores:** id\_autor, nombre, nacionalidad, fecha\_nacimiento
- **Relación:**
  - Tabla intermedia: AutoresLibros con los atributos id\_libro y id\_autor.

### 4. Plataforma de Música: Canciones y Playlists

- **Descripción:** Una canción puede estar en múltiples playlists, y una playlist puede contener múltiples canciones.
- **Entidades:**
  - **Canciones:** id\_cancion, titulo, artista, duracion
  - **Playlists:** id\_playlist, nombre, usuario\_creador, fecha\_creacion
- **Relación:**
  - Tabla intermedia: CancionesPlaylists con los atributos id\_cancion y id\_playlist.

### 5. Sistema de E-commerce: Productos y Órdenes

- **Descripción:** Una orden puede incluir múltiples productos, y un producto puede pertenecer a múltiples órdenes.
- **Entidades:**
  - **Productos:** id\_producto, nombre, precio, stock
  - **Órdenes:** id\_orden, fecha, cliente

- **Relación:**
  - Tabla intermedia: DetallesOrden con los atributos id\_orden, id\_producto, cantidad y precio\_unitario.

## 6. Sistema de Cine: Películas y Actores

- **Descripción:** Una película puede tener múltiples actores, y un actor puede participar en múltiples películas.
- **Entidades:**
  - **Películas:** id\_pelicula, titulo, año, género
  - **Actores:** id\_actor, nombre, nacionalidad, edad
- **Relación:**
  - Tabla intermedia: ParticipacionesPelicula con los atributos id\_pelicula, id\_actor y rol.

## 7. Sistema Educativo: Profesores y Asignaturas

- **Descripción:** Un profesor puede enseñar varias asignaturas, y una asignatura puede ser enseñada por varios profesores.
- **Entidades:**
  - **Profesores:** id\_profesor, nombre, especialidad
  - **Asignaturas:** id\_asignatura, nombre, horas
- **Relación:**
  - Tabla intermedia: ProfesoresAsignaturas con los atributos id\_profesor y id\_asignatura.

## 8. Sistema de Red Social: Usuarios y Grupos

- **Descripción:** Un usuario puede pertenecer a varios grupos, y un grupo puede tener múltiples usuarios.
- **Entidades:**
  - **Usuarios:** id\_usuario, nombre, correo, fecha\_registro
  - **Grupos:** id\_grupo, nombre\_grupo, descripcion
- **Relación:**
  - Tabla intermedia: MiembrosGrupo con los atributos id\_usuario y id\_grupo.

## 9. Sistema de Hospital: Doctores y Pacientes

- **Descripción:** Un doctor puede atender a varios pacientes, y un paciente puede ser atendido por varios doctores.
- **Entidades:**
  - **Doctores:** id\_doctor, nombre, especialidad, horario
  - **Pacientes:** id\_paciente, nombre, edad, historial\_medico
- **Relación:**
  - Tabla intermedia: Consultas con los atributos id\_doctor, id\_paciente y fecha\_consulta.

## 10. Sistema de Eventos: Participantes y Actividades

- **Descripción:** Un participante puede inscribirse en varias actividades, y una actividad puede tener múltiples participantes.

- **Entidades:**
  - **Participantes:** id\_participante, nombre, edad, email
  - **Actividades:** id\_actividad, nombre\_actividad, descripcion
- **Relación:**
  - Tabla intermedia: Inscripciones con los atributos id\_participante y id\_actividad.

#### 11. Sistema Escolar: Estudiantes y Clubs

- **Descripción:** Un estudiante puede ser miembro de varios clubs, y un club puede tener múltiples estudiantes.
- **Entidades:**
  - **Estudiantes:** id\_estudiante, nombre, grado, correo
  - **Clubs:** id\_club, nombre, actividad\_principal
- **Relación:**
  - Tabla intermedia: MiembrosClub con los atributos id\_estudiante y id\_club.

#### 12. Plataforma de Viajes: Usuarios y Destinos Favoritos

- **Descripción:** Un usuario puede tener varios destinos favoritos, y un destino puede ser favorito de múltiples usuarios.
- **Entidades:**
  - **Usuarios:** id\_usuario, nombre, correo
  - **Destinos:** id\_destino, nombre, ubicacion
- **Relación:**
  - Tabla intermedia: Favoritos con los atributos id\_usuario y id\_destino.

#### 13. Sistema de Aprendizaje en Línea: Estudiantes y Cursos Completados

- **Descripción:** Un estudiante puede completar varios cursos, y un curso puede ser completado por varios estudiantes.
- **Entidades:**
  - **Estudiantes:** id\_estudiante, nombre, correo, nivel
  - **Cursos:** id\_curso, nombre, duracion, nivel
- **Relación:**
  - Tabla intermedia: CursosCompletados con los atributos id\_estudiante, id\_curso y fecha\_completado.

#### 14. Sistema de Producción: Máquinas y Proyectos

- **Descripción:** Una máquina puede participar en varios proyectos, y un proyecto puede requerir varias máquinas.
- **Entidades:**
  - **Máquinas:** id\_maquina, nombre, capacidad
  - **Proyectos:** id\_proyecto, nombre, fecha\_inicio, fecha\_fin
- **Relación:**
  - Tabla intermedia: AsignacionesMaquinas con los atributos id\_maquina, id\_proyecto.

#### 15. Sistema de Restaurantes: Platos y Ingredientes

- **Descripción:** Un plato puede contener múltiples ingredientes, y un ingrediente puede ser usado en varios platos.
- **Entidades:**
  - **Platos:** id\_plato, nombre, precio
  - **Ingredientes:** id\_ingrediente, nombre, cantidad\_disponible
- **Relación:**
  - Tabla intermedia: IngredientesPlatos con los atributos id\_plato, id\_ingrediente y cantidad\_usada.

#### 16. Plataforma de Compras: Usuarios y Métodos de Pago

- **Descripción:** Un usuario puede tener varios métodos de pago, y un método de pago puede ser usado por varios usuarios.
- **Entidades:**
  - **Usuarios:** id\_usuario, nombre, correo
  - **Métodos de Pago:** id\_metodo, tipo, entidad
- **Relación:**
  - Tabla intermedia: PagosUsuarios con los atributos id\_usuario y id\_metodo.

#### 17. Sistema de Contratación: Empleados y Habilidades

- **Descripción:** Un empleado puede tener varias habilidades, y una habilidad puede ser compartida por varios empleados.
- **Entidades:**
  - **Empleados:** id\_empleado, nombre, puesto
  - **Habilidades:** id\_habilidad, nombre\_habilidad
- **Relación:**
  - Tabla intermedia: HabilidadesEmpleados con los atributos id\_empleado y id\_habilidad.

#### 18. Plataforma de Streaming: Series y Géneros

- **Descripción:** Una serie puede pertenecer a varios géneros, y un género puede incluir varias series.
- **Entidades:**
  - **Series:** id\_serie, nombre, año
  - **Géneros:** id\_genero, nombre\_genero
- **Relación:**
  - Tabla intermedia: SeriesGeneros con los atributos id\_serie y id\_genero.

#### 19. Sistema de Reservas: Clientes y Habitaciones de Hotel

- **Descripción:** Un cliente puede reservar múltiples habitaciones en un hotel, y una habitación puede ser reservada por múltiples clientes en diferentes fechas.
- **Entidades:**
  - **Clientes:**
    - id\_cliente: Identificador único del cliente.
    - nombre: Nombre completo.
    - correo: Correo electrónico.
    - telefono: Número de contacto.
  - **Habitaciones:**

- id\_habitacion: Identificador único de la habitación.
  - tipo: Tipo de habitación (individual, doble, suite).
  - precio\_noche: Precio por noche.
- **Relación:**
  - **Tabla intermedia:** Reservas
    - id\_reserva: Identificador único de la reserva.
    - id\_cliente: Clave foránea hacia Clientes.
    - id\_habitacion: Clave foránea hacia Habitaciones.
    - fecha\_inicio: Fecha de inicio de la reserva.
    - fecha\_fin: Fecha de finalización de la reserva.