

Gestor de Trenes y Simulador de Viajes

El IES Comercio quiere crear una aplicación web llamada "**Thomas y sus Amigos: Gestor de Trenes**". La aplicación permitirá registrar trenes con su nombre, tipo y velocidad, consultarlos desde un listado, filtrarlos mediante búsqueda y **simular viajes entre estaciones** con eventos aleatorios.

Tu tarea es **diseñar e implementar la aplicación web** usando **HTML, CSS y JavaScript**, siguiendo las especificaciones a continuación.

Thomas y sus Amigos: Gestor de Trenes

Registrar Nuevo Tren

Nombre: Tipo:

Velocidad (km/h): Añadir Tren

Simulador de Viajes

Estación Origen: Estación Destino: Selecciona Tren: Simular Viaje

Listado de Trenes

Buscar por nombre o tipo..

rafa Tipo: Vapor Velocidad: 12 km/h Fecha de alta: 7/11/2025	123 Tipo: Eléctrico Velocidad: 12 km/h Fecha de alta: 7/11/2025
---	--

Estructura General

La web constará de una sola página (`index.html`) con tres secciones principales:

1. Registro de Trenes

- Permite añadir nuevos trenes introduciendo:
 - **Nombre** (texto, obligatorio)
 - **Tipo** (selector: Vapor, Diésel, Eléctrico, Maglev)
 - **Velocidad máxima** (número entre 10 y 300 km/h)
- Al pulsar "**Añadir Tren**", se agrega al listado.
- La **fecha de alta** se añade automáticamente (ejemplo: "07/11/2025").
- Los trenes se guardan en **LocalStorage**.

2. Simulador de Viajes

- El usuario selecciona **dos estaciones** (de una lista predefinida: Sodor Central, Knapford, Vicarstown y Tidmouth Sheds).
- Selecciona un **tren registrado** para el viaje.
- Al pulsar "**Simular Viaje**", la app calcula:
 - **Distancia**: explicada más adelante.
 - **Tiempo estimado** (distancia/velocidad del tren).
 - **Evento aleatorio** (ejemplo: "¡Retraso por ovejas en la vía!", "¡Avería en el motor!", "Viaje sin incidentes").
- Muestra los resultados en una tarjeta con emojis y colores.

3. Listado de Trenes

- Muestra todos los trenes en una cuadrícula.
- Cada tarjeta incluye:
 - Nombre
 - Tipo
 - Velocidad máxima
 - Fecha de alta
- Campo de búsqueda para filtrar por **nombre o tipo**.

Funcionalidades a Implementar

Añadir Tren

Al completar el formulario y pulsar "**Añadir Tren**", se crea un objeto:

```
{  
  nombre: "Thomas",  
  tipo: "Vapor",  
  velocidad: 80,  
  fechaAlta: "07/11/2025"  
}
```

- Se guarda en un array y en **LocalStorage**.
- El listado se actualiza automáticamente.

Simular Viaje

Al seleccionar dos estaciones y un tren, la aplicación calculará la distancia entre ellas usando una **matriz de distancias predefinidas** entre las siguientes cuatro estaciones de Sodor:

- Sodor Central
- Knapford
- Vicarstown
- Tidmouth Sheds

La matriz de distancias predefinidas será la siguiente:

- Sodor Central a Knapford: 25 km
- Sodor Central a Vicarstown: 15 km
- Knapford a Vicarstown: 10 km
- Vicarstown a Tidmouth Sheds: 12 km

Si la distancia entre las estaciones seleccionadas no está definida directamente en la matriz, se calculará como la suma de las distancias de las estaciones intermedias. Por ejemplo, la distancia de Sodor Central a Tidmouth Sheds se calculará como:

- Sodor Central → Vicarstown → Tidmouth Sheds (15 km + 12 km = 27 km)

La simulación mostrará:

- **Distancia total** (calculada según la matriz y las estaciones intermedias).
- **Tiempo estimado** (redondeado a minutos, basado en la velocidad del tren seleccionado y la distancia).
- **Evento aleatorio** (3 opciones, una de las cuales implica un retraso).

Ejemplo de salida:

Viaje de Sodor Central a Knapford

Tren: Thomas (Vapor, 80 km/h)

Distancia: 25 km

Tiempo estimado: 19 minutos

¡Retraso por ovejas en la vía! (+10 min)

Buscar Tren

Campo de búsqueda para filtrar por **nombre o tipo** (no sensible a mayúsculas). Usad un botón para buscar.

Borrar Todos los Trenes

Botón que vacía el **LocalStorage**.

Requisitos de Diseño

Usa **flexbox o grid** para organizar los elementos.

No os preocupéis por los colores.

No uséis frameworks CSS.

Evaluación

Criterio	Descripción	Puntuación
HTML semántico	Uso correcto de etiquetas, formularios, secciones	0.5 pts
Estilos CSS y diseño temático	Organización de los elementos.	0,5 pt
Uso de objetos y arrays	Estructura de datos y manipulación con JS	2,5 pts
LocalStorage	Persistencia correcta	1.5 pts
Filtrado de trenes	Buscador funcional	2 pts
Simulador de viajes	Lógica de cálculo, eventos aleatorios, salida clara	3 pts
Total		10 pts

Entrega

Un **.zip** con:

- `index.html`
- `style.css`
- `script.js`