

Módulo 2 – App de Clima (Frontend)

1) Propósito

Desarrollar la **primera versión (MVP)** del frontend de una **aplicación de clima**, aplicando **HTML5 semántico**, **Bootstrap** para estilos y diseño responsivo, y **JavaScript** básico para la interacción. El trabajo se versiona en **Git/GitHub** e incluye un **README** descriptivo.

Este es tu proyecto: puedes darle la temática que quieras (clima **galáctico**, **ciudades icónicas**, o tu **región**). **Explora, crea y divírtete** mientras practicas lo aprendido. **Sé autodidacta**: la curiosidad y la constancia distinguen a todo gran desarrollador.

2) Objetivos de aprendizaje

- Estructurar páginas con HTML5 semántico.
- Aplicar Bootstrap (grid, utilidades, componentes) para un layout responsivo (mobile-first).
- Manipular el DOM con JavaScript (eventos y estados simples).
- Gestionar el proyecto con Git/GitHub (commits descriptivos, ramas y README).

3) Alcance (MVP)

Páginas mínimas:

- **Inicio (Home)**: grilla de **localidades** (mínimo **10**) presentadas como **cards de Bootstrap** que muestran clima actual (ícono, temperatura, estado).
- **Detalle de localidad**: vista con datos ampliados de la ciudad seleccionada (temperatura, humedad, viento) y **pronóstico de la semana actual** (lista o cards por día).

Navegación:

- **Navbar** con enlaces a **Home**, **Detalle** (cuando corresponda) y (opcional) **Acerca de**.
- **Footer con información básica del proyecto**.

4) Requisitos funcionales mínimos

- Listado de **≥10 localidades** en Home con cards informativas.
- Al seleccionar una localidad, navegar a Detalle y mostrar el pronóstico semanal.
- **Responsive**: se debe ver bien en **móvil (≤420 px)** y **escritorio (≥1024 px)**.
- Interacciones básicas con **JS** (click en card/botón para ver detalle).

5) Requisitos técnicos

- **HTML5 semántico** (header, nav, main, section, article, footer).
- **Bootstrap**:
 - Grid (contenedores, filas, columnas) y **utilidades** (espaciados, tipografía, display).
 - **Componentes**: navbar, card, badge/list-group/button según diseño.
 - Inclusión por **CDN** (CSS y JS de Bootstrap).
- **Git/GitHub**: repo público **weather-frontend-m2**, mínimo 3 commit con mensaje descriptivos.

6) Entregables

- **Proyecto comprimido** en un único archivo **.zip**.
- **README.md** dentro del proyecto, que incluya **enlace al repositorio público de GitHub**.

Rubrica de evaluación

Criterio	Excelente (3 pts)	Adecuado (2 pts)	Básico (1 pt)	Insuficiente (0 pts)
HTML5 semántico y estructura	Etiquetas semánticas correctas en todas las páginas; jerarquía clara; sin “divitis”.	Semántica mayormente correcta; 1–2 desajustes menores.	Semántica limitada o confusa.	No usa semántica; estructura desordenada.
Bootstrap y diseño responsive	Usa grid/containers, utilidades y componentes; mobile-first estable en móvil ($\leq 420\text{px}$) y desktop ($\geq 1024\text{px}$).	Usa grid y algunas utilidades; responde con pequeños quiebres.	Poco uso de Bootstrap; responsive parcial con cortes notables.	No aplica Bootstrap o el sitio no es responsive.
JS/DOM (interacción, detalle y pronóstico)	Cards de ≥ 10 localidades; click navega a Detalle ; muestra pronóstico semanal correctamente; código legible.	Listado y navegación a detalle funcionan con detalles menores.	Interacción inconsistente (detalle o pronóstico falla a veces) o dependiente de recargas.	Sin interacción funcional (no abre detalle ni muestra pronóstico).
Navegación, UX y accesibilidad básica	Navbar/links claros; flujo Home a Detalle sin fricción; contraste y foco visibles.	Navegación clara con 1–2 detalles por mejorar (p. ej., no se marca la página actual).	Navegación confusa en algún punto; accesibilidad mínima ausente.	Enlaces rotos o sin flujo; nula accesibilidad.
Calidad de código (organización y buenas prácticas)	Archivos separados; nombres coherentes; consistencia en estilos; comentarios útiles.	Organización general correcta con leves inconsistencias.	Estructura pobre o nombres poco claros.	Código desordenado; difícil de mantener.
Git/GitHub y README	≥3 commits atómicos con mensajes descriptivos; repo público weather-frontend-m2 ; README claro con enlace al repo y cómo ejecutar.	3 commits con mensajes aceptables; README con enlace pero incompleto.	<3 commits o mensajes genéricos; README escueto.	Sin control de versiones adecuado o sin README/enlace.