

Fase 1: crear app e implementar enrutamiento

Enunciado.

Vamos a desarrollar una aplicación de Angular con **enrutamiento** (*routing*) que se llame **MyWeather**. La aplicación permitirá consultar el tiempo meteorológico de cualquier ciudad del mundo. También permitirá que un usuario especifique sus ciudades favoritas y cuál quiere mostrar en la pantalla de inicio.

Hoja de estilos

De manera general, la aplicación usará **Bootstrap**, por lo que es necesario enlazar los archivos requeridos. Los estilos que sean exclusivos de un componente estarán en el CSS de dicho componente.

Origen de datos

Los datos van a ser obtenidos de las siguientes fuentes por orden de prioridad, es decir, se intentará la fuente (1) y si hay algún obstáculo insalvable pasaremos a la fuente (2) y así sucesivamente:

1. La **API** de **OpenWeatherMap** en <https://openweathermap.org/>.
2. Si fallara el acceso a este último servicio entonces el profesor os proporcionaría un archivo **JSON** para integrar en nuestra app como recurso local.

Las URL base para consultar datos son:

URL	http://api.openweathermap.org/data/2.5/find
Parámetros	q : cadena de búsqueda de ubicaciones por nombre (patrón) appid , la clave API del usuario
URL	http://api.openweathermap.org/data/2.5/weather
Parámetros	id : identificador de la ubicación de la que queremos consultar el tiempo appid , la clave API del usuario units=metric : (opcional) permite indicar que quieres los datos en unidades del Sistema Métrico Internacional

Modelos de *templates* entregados por el profesor

Para ahorrar tiempo, se entregan algunas *templates* con código HTML genérico que ayudan mucho, pero deben ser adaptadas previamente para funcionar bien.

Componente raíz

A continuación, se indica una lista de los 3 componentes que será necesario crear para dar contenido al componente raíz:

- **Componente para el encabezado.** Muestra la imagen logotipo de la aplicación y el nombre de la aplicación.
- **Componente para la barra de navegación.** Ofrecerá la posibilidad de clicar en las secciones listadas en el apartado **Navegación y enrutamiento** más abajo.
- **Componente para el pie.** Muestra el nombre del autor o autora de la aplicación, es decir, tu nombre. También cita la fuente de los datos.

Navegación y enrutamiento

Justo debajo del encabezado aparecerá una barra de navegación horizontal con varias secciones:



The screenshot shows the 'Home' section of the MyWeather application. It includes the logo and navigation bar. Below them, the current location is displayed as 'Murcia, España' with a temperature of '13 °C'. There are two small icons (a heart and a plus sign) below the temperature. At the bottom, there is a footer with the author's name 'Luis Miguel Fernández Costa', a Creative Commons license notice, and a link to 'Visit OpenWeatherMap.org'.	<p>Sección <i>Home</i>: muestra el tiempo de la ciudad elegida como ciudad predeterminada. Si todavía no ha sido elegida ninguna ciudad como ciudad predeterminada, entonces se mostrará la primera ciudad favorita. Y si tampoco hay ciudades favoritas entonces no se mostrará nada.</p>
--	---

<p>Sección <i>Search</i>: muestra un campo de texto para buscar por nombre de ciudad. Listará resultados al presionar la tecla Intro.</p>	The screenshot shows the 'Search' section of the MyWeather application. It includes the logo and navigation bar. Below them, there is a search bar labeled 'Search cities' with the text 'Lepe' entered. Below the search bar, there are two search results for 'Lepe (ES)', each with a heart icon. At the bottom, there is a footer with the author's name, a Creative Commons license notice, and a link to 'Visit OpenWeatherMap.org'.
--	--

The screenshot shows the 'Manage' section of the MyWeather application. It includes the logo and navigation bar. Below them, there is a section titled 'Your favourite cities' with a search bar. Below the search bar, there are four city cards: 'Murcia, España' (13 °C), 'Paris, not Spain' (9 °C), 'Rome, not Spain' (9 °C), and 'Barcelona, España' (13 °C). Each card has a heart icon and a plus icon. At the bottom, there is a footer with the author's name, a Creative Commons license notice, and a link to 'Visit OpenWeatherMap.org'.	<p>Sección <i>Manage</i>: muestra tus ciudades favoritas y un campo de texto para filtrarlas por nombre.</p>
--	---

Fase 2: desarrollar servicios y filtros

Servicios

Tienes libertad para implementar servicios a tu criterio. Sin embargo, si te sirve de guía, aquí tienes una sugerencia:

- **Un servicio para obtener datos del origen de datos.** Es capaz de hacer llamadas *HTTP* para obtener los datos en formato *JSON* necesarios para alimentar a la aplicación.
- **Un servicio para las preferencias del usuario.** Es capaz de almacenar tus ciudades favoritas y tu ciudad predeterminada.

Las preferencias del usuario no deben perderse, por lo que hay que gestionar la **persistencia** en el almacenamiento local del navegador (**local storage**).

Filtros

Debes implementar al menos uno de los siguientes filtros:

- Filtro para que se muestren únicamente las ciudades favoritas cuyo nombre incluye una cadena patrón.
- Filtro para que el nombre del país que se muestra debajo del nombre de la ciudad no sea un simple código internacional. Por ejemplo, gracias a este filtro, en lugar de Murcia (ES) se mostrará Murcia (España).
- Filtro para convertir temperaturas de °K (Kelvin) a °C (Celsius).

Objetivos

Objetivos obligatorios	OK
Los contenidos del componente raíz se ajustan a lo pedido.	
La barra de navegación funciona.	
Un servicio obtiene datos del servidor mediante HTTP y dichos datos son usados y mostrados correctamente en alguna parte de la aplicación.	
El buscador muestra resultados correctos al buscar por nombre sin importar mayúsculas o minúsculas.	
Gestión completa de ciudades favoritas: añadir/listar/borrar.	
Una ciudad puede ser establecida como ciudad predeterminada y se muestra en la página de inicio de la aplicación.	
Al menos un filtro ha sido implementado.	
Las preferencias del usuario se conservan en <i>LocalStorage</i> .	
Objetivos opcionales	OK
No se realizan llamadas HTTP innecesarias y el aspecto está cuidado y los errores en la consola controlados.	
Son implementados dos o tres filtros.	
En cada ciudad se muestra un icono que representa el tiempo que hace (soleado, nublado, lluvia, nieve, viento...).	
Cuando se intenta hacer favorita una ciudad, aparece una notificación/alerta de <i>Bootstrap</i> que nos dice si ha habido éxito o si ya estaba en favoritas.	

Conocimientos examinados

En este examen se **requiere** que el alumno o la alumna sepa:

- Crear aplicaciones web con Angular
- Insertar nuevos componentes en la aplicación
- Capturar eventos producidos por el usuario sobre ciertos elementos
- Definir enrutamiento para navegar dentro de la aplicación
- Establecer vías de comunicación entre componentes
- Añadir servicios e inyectarlos en los componentes que los requieran
- Realizar llamadas HTTP asíncronas para interactuar con una base de datos

En este examen se **valora** que el alumno o la alumna sepa:

- Detectar y manejar los posibles errores en tiempo de ejecución
- Ahorrar ancho de banda
- Implementar aspectos avanzados en aplicaciones web para aumentar la funcionalidad de la misma

Calificación del examen

La correcta resolución de todos los objetivos obligatorios otorga una calificación de 5 sobre 10. Solamente si todos los objetivos obligatorios han sido resueltos adecuadamente se procederá a sumar puntuación adicional por cada uno de los objetivos opcionales solucionados con éxito, hasta un máximo de 5 puntos adicionales si se implementan 5 de los objetivos opcionales.