# Web scraping

# Aplicaciones híbridas

- Web scraping es una técnica utilizada mediante programas de software para extraer información de sitios web.
- Estos programas simulan la navegación de un humano en internet ya sea utilizando el protocolo HTTP manualmente, o incrustando un navegador en una aplicación.

# Aplicaciones híbridas

**Web scraping: ejemplos de uso**

- **En marketing de contenidos:** podemos diseñar un robot que haga un ‘scrapeo’ de datos concretos de una web y los podamos utilizar para generar nuestro propio contenido. Ejemplo: scrapear los datos estadísticos la web oficial de una liga de fútbol para generar nuestra propia base de datos, comparadores, etc.
- **Para ganar visibilidad en redes sociales:** podemos utilizar los datos de un scrapeo para interactuar a través de un robot con usuarios en redes sociales. Ejemplo: crear un bot en instagram que seleccione los links de cada foto y luego programar un comentario en cada entrada.
- **Para controlar la imagen y la visibilidad de nuestra marca en internet:** a través de un scrapeo podemos automatizar la posición por la que varios artículos de nuestra web se posicionan en Google o, por ejemplo, controlar la posición en Google de todas las entradas de nuestro blog.

# Aplicaciones híbridas

**Web scraping: medidas para detener a los scrapers**

El administrador de un sitio web puede utilizar varias técnicas para detener o disminuir los pedidos de los scrapers. Por ejemplo:

- **Añadir entradas al fichero robots.txt.** Google y otros bots pueden ser detenidos de esta forma.
- **Bloquear la dirección IP.** Esto también bloqueará todos los accesos desde esa misma IP, por lo que los usuarios no podrán navegar por el sitio web si acceden desde esta.
- **Monitorear el exceso de tráfico proveniente de cierta IP.**
- **Añadir un captcha u otro sistema de verificación manual al sitio web.** No se garantiza el completo bloqueo de los scrapers, pero mediante esta técnica se dificulta su acceso.
- **Servicios comerciales antibots:** algunas empresas ofrecen servicios antibots y antiscraping.
- **Incrementar el uso de JavaScript y AJAX.** De esta forma es más difícil para los scrapers simular las peticiones como si fueran un navegador común.

# Aplicaciones híbridas

**Ejemplo: obtener predicción meteorológica usando técnicas de web scraping.**

- El objetivo es mostrar en nuestra página un fragmento de texto que indica la predicción del tiempo en una ciudad que se introducirá por teclado, obteniendo dicha información de la página http://es.weather-forecast.com
- Para poder hacerlo, debemos observar cómo se comporta la página fuente de la información cuando nosotros introducimos el nombre de una ciudad sobre la que queremos consultar la predicción del tiempo.

# Aplicaciones híbridas

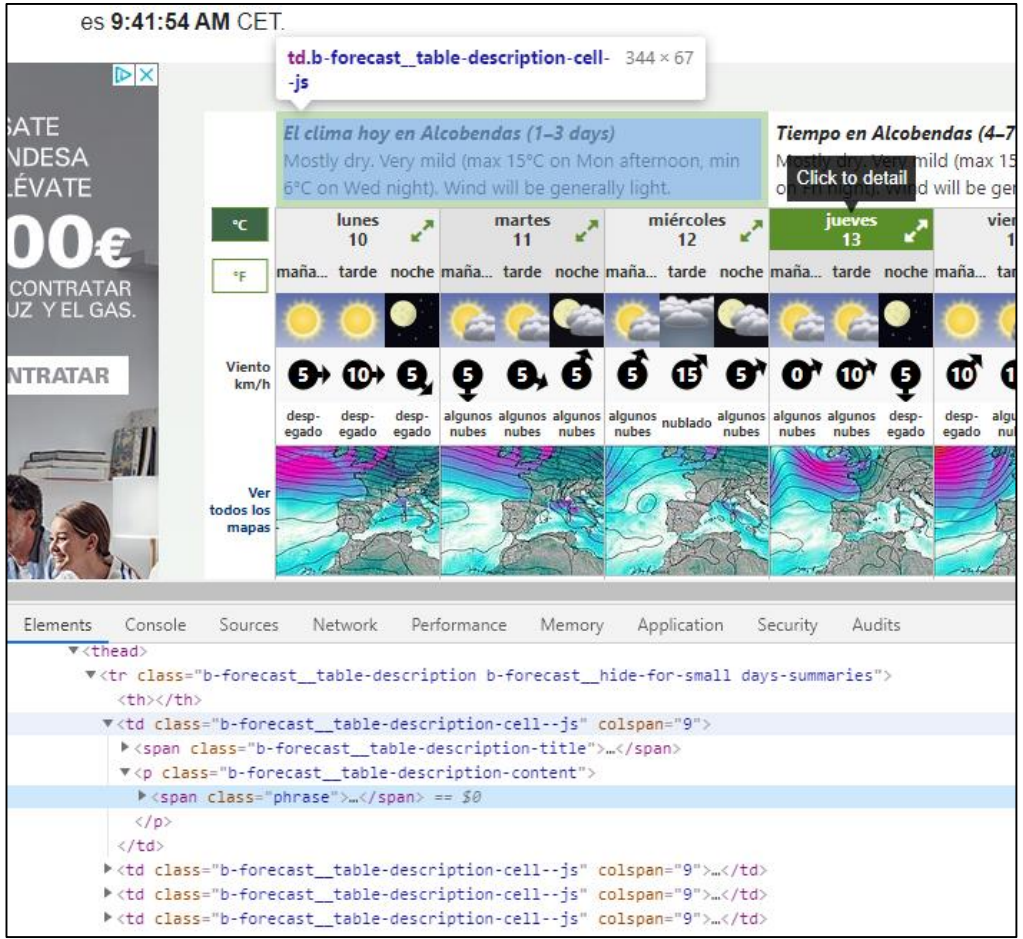
- En el cuadro de búsqueda de la página escribiremos el nombre de una ciudad de la que queremos saber la predicción del tiempo.


- Observando la URL de la página web que nos devuelve podemos comprobar que contiene el nombre de la ciudad que introdujimos. Esto nos indica como debemos realizar una consulta desde nuestro sitio web

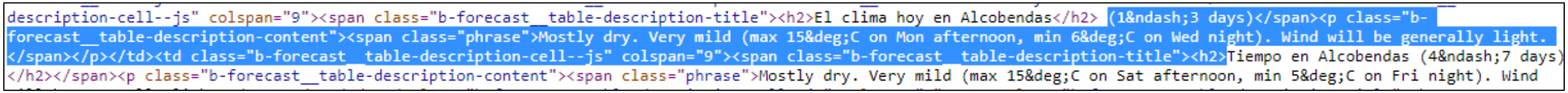
es.weather-forecast.com/locations/Alcobendas/forecasts/latest

# Aplicaciones híbridas

- A continuación, buscaremos en el código fuente de la página origen la parte de código que queremos extraer.
- En concreto, nos interesa obtener la predicción de 1 a3 días.
- Utilizando las herramientas de desarrollador del navegador podemos observar el código y saber cuál es el fragmento que nos interesa recuperar.

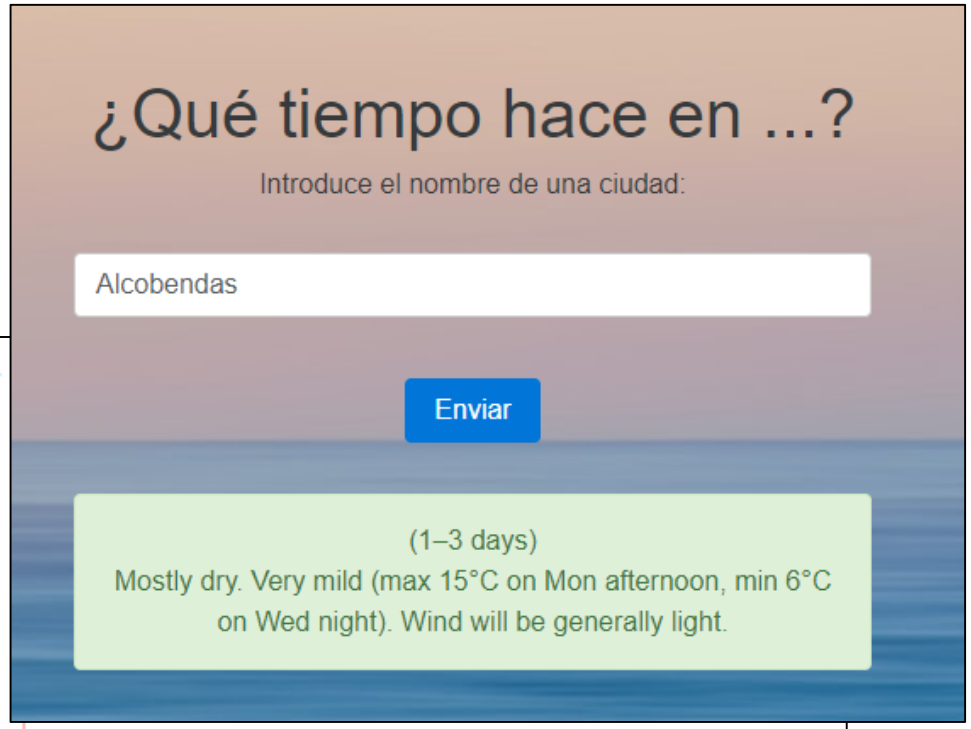



# Aplicaciones híbridas




- Ahora nos toca escribir nuestro código.
- Necesitaremos un formulario en el que introducir el nombre de la ciudad de la que queremos obtener la información meteorológica, y usaremos la función de PHP **file_get_contents(fichero)** que nos devuelve en una cadena de caracteres el contenido de un fichero.
- Una vez tengamos el contenido del fichero en una cadena, extraemos de ella la subcadena que nos interesa utilizando las funciones **strpos()** para calcular las posiciones de inicio  y fin del fragmento de código que queremos recuperar y la función **substr**() para extraer la subcadena que se encuentra entre ambas posiciones.

# Aplicaciones híbridas




```php
//Comprobamos si ha podido leer datos de la URL. Lee toda la página
$pagina = file_get_contents($file);
if(!$pagina){
    $error = "No hemos podido encontrar esa ciudad";
}
else{
/* Establecemos desde donde hasta donde queremos recuperar de la
 * página. Solo queremos la previsión de 1-3 díaas
 * Hay que buscar en el código de la página original ese fragmento-
 */
    $inicio= strpos($pagina,'(1&ndash;3 days)</span><p class="b-forecast__table-description-content">'
            . 'span class="phrase">');
    $fin = strpos($pagina,'</span></p></td><td class="b-forecast__table-description-cell--js" colspan="9">'
            . '<span class="b-forecast__table-description-title"><h2>');
    $long = $fin-$inicio;
    echo $long;
    /* obtiene de la cadena que contiene toda la página la subcadena
     * que nos interesa
     */
    $previsionTiempo=substr($pagina,$inicio,$long);
}
```

# Aplicaciones híbridas

**Incorporar de un sitio web directamente información presenta algunos inconvenientes:**

- Depende del formato HTML de la página para poder extraer la información. Si la página cambia, nuestra página podría dejar de funcionar.
- Al dueño del sitio web no le hace mucha gracia que se use la información de su sitio sin atribuciones o reconocimientos.
- Requiere un trabajo previo de investigación y pruebas. Por ejemplo, en el ejercicio anterior, si pedimos el tiempo de “San Sebastian de los Reyes”, veremos que en la URL va sin tildes y que sustituye los espacios entre palabras por guiones, y que las poblaciones con “ñ” no las reconoce, por lo menos alguna.  Esto se traduce en que en nuestra página deberíamos filtrar el nombre de la población para la página fuente la encuentre.
- Para evitar estos inconvenientes, la mejor forma de incluir código externo en nuestra aplicación web es a través de los servicios web, que nos permiten obtener datos de un propietario con su consentimiento y de forma controlada.