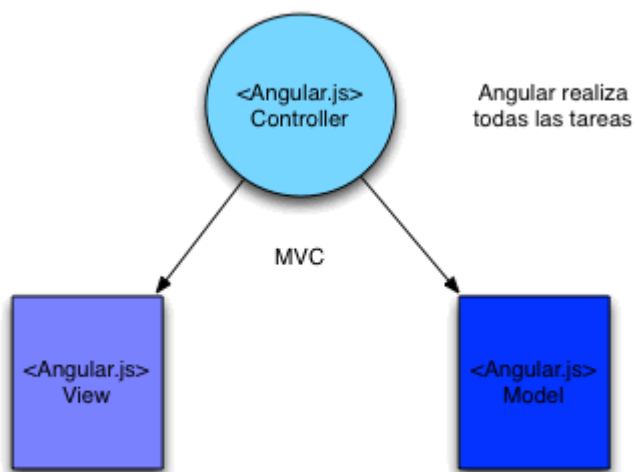
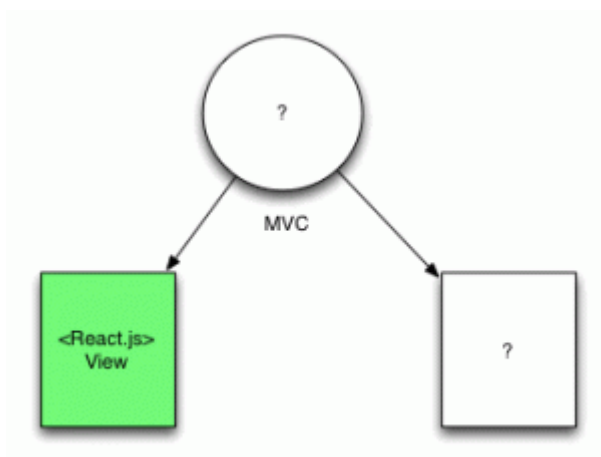


React.js es una librería de JavaScript que se encuentra en un momento de ebullición. El año pasado ocurrió con **Angular.js** y este año le toca a React.js . ¿Qué diferencia React.js de Angular.js?. La diferencia fundamental es que mientras que Angular.js es un framework MVC que se encarga de hacer prácticamente todo.



React.js es una librería encargada de gestionar la vista únicamente.



Esto puede parecer en un principio raro , pero recordemos que frameworks que se han encargado de cosas muy concretas han tenido mucho éxito, caso de jQuery. Existen muchos frameworks que se pueden usar React.js como alternativa al sistema de vistas por defecto , uno de los más conocidos es Backbone.js.

Primeros pasos con React.js

Vamos a comenzar a trabajar con React.js, para ello nos descargaremos la librería de la página web y construiremos el primer ejemplo:

```
<html>
<head>
<script src="https://fb.me/react-with-addons-0.14.0.js"></script>
<script src="https://fb.me/react-dom-0.14.0.js"></script>
</head>
<body>

<div id="zona">

</div>

</body>
<script type="text/javascript">
ReactDOM.render(
  React.createElement("h1", null, "Hola mundo!"),
  document.getElementById("zona")
);
</script>
</html>
```

En este caso se ha construido en elemento utilizando ReactDOM.render para visualizarlo y React.createElement para construirlo. Si cargamos la página en un navegador el resultado será:



El método createElement es el encargado de diseñar un componente , en este caso uno tan sencillo como un H1. React no apuesta por plantillas típicas de un modelo MVC sino que apuesta por un enfoque orientado a componente. Para ello soporta una sintaxis especial denominada JSX (JavaScript Syntax Extension). Vamos a extraer el código JavaScript de la página.

```
<html>
<head>
<script src="https://fb.me/react-with-addons-0.14.0.js"></script>
<script src="https://fb.me/react-dom-0.14.0.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/babel-core/5.8.23/browser.
min.js">
</script>
</head>
<body>

<div id="zona">
```

```
</div>
```

```
</body>
```

```
<script type="text/babel" src="002.js"></script>
```

```
</html>
```

El siguiente paso es ver como queda extraido el código de JavaScript:

```
ReactDOM.render(
```

```
<h1>Hola mundo</h1>
```

```
,
```

```
  document.getElementById('zona')
```

```
);
```

Es curioso que el código no llega comillas en el h1 de holaMundo. Esto es debido a que estamos usando JSX para aumentar la claridad. Demonos cuenta también que aunque hemos puesto de extensión al fichero .js el tipo del script es “text/babel” que es el compilar de ES6 que usa React.js, el resultado será identico.



Hemos construido un ejemplo de Hola Mundo con React.js.

Otros artículos relacionados : [Angular.js vs Node.js](#) , [Angular ng-inspector](#)