

Guía básica de Backbone.js

DevCode.la

Guía básica de Backbone.js

DevCode.la

Este libro está a la venta en <http://leanpub.com/Guia-Basica-de-Backbone-js>

Esta versión se publicó en 2013-12-07



This is a [Leanpub](#) book. Leanpub empowers authors and publishers with the Lean Publishing process. [Lean Publishing](#) is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

©2013 DevCode.la

¡Twitea sobre el libro!

Por favor ayuda a DevCode.la hablando sobre el libro en [Twitter](#)!

El tweet sugerido para este libro es:

Comparte este libro sobre Backbone.js de @devcodela #Devbackbonejs

El hashtag sugerido para este libro es [#DevBackbonejs](#).

Descubre lo que otra gente está diciendo sobre el libro haciendo click en este enlace para buscar el hashtag en Twitter:

<https://twitter.com/search?q=#DevBackbonejs>

Índice general

Introducción	1
<i>¿Que es Backbone.js?</i>	1
<i>¿Que es MV*</i> ?	1
<i>¿Como funciona Backbone.js?</i>	1
Proyecto:	2

Introducción

¿Que es Backbone.js?

Backbone.js es una librería de javascript open source muy liviana (6.4Kb) que fue desarrollada en el año 2010 por Jeremy Ashkenas de DocumentCloud.

La importancia de Backbone.js es que nos ayuda a estructurar nuestro código del lado del cliente, evitando que se vuelva código espagueti y de esta manera poder mantenerlo fácilmente a través del tiempo. Para esto utiliza una variante del patrón de arquitectura MVC, conocida como MV* o MV(estrella).

¿Que es MV*?

Bueno, antes de responder esta pregunta primero voy a explicar que es MVC. El Modelo Vista Controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario. En los framework backend como Rails, Zend, Laravel, Django y otro podemos identificar claramente este patrón, aunque con algunas pequeñas variaciones que no vienen al caso en este curso.

Ahora detallare un poco mas esta definición:

Modelo. Es la representación de los datos que maneja la aplicación. Generalmente, se sostiene por una base de datos. Vista. Representa la interfaz de usuario o lo que el usuario llegar a percibir. Controlador. Un intermediario entre la Vista y el Modelo. Recibe peticiones del browser, las efectúa ya sea interactuando con el modelo o no, y devuelve a la vista el resultado de la operación. Ahora que entendemos que significa el patrón MVC vamos por lo que queríamos saber que era MV*, el cual vendría a ser simplemente una variación del patrón MVC. En el cual el * es una representación que puede significar diferentes cosas según el framework fronted que estemos trabajando.

En el caso particular de Backbone.js el * representaría Collection y Routes. Ahora que entendemos esto vayamos a ver como funciona Backbone.js

¿Como funciona Backbone.js?

Como vemos en el gráfico Backbone.js tiene Modelos Vistas Colecciones y Rutas. Primero voy a explicar cada una de estas:

En el caso de los modelos son los que gestionan los datos con los que trabaja la aplicación, permitiendo su carga, edición y almacenado posterior en el servidor. Las colecciones vienen a

ser conjuntos de instancias de un modelo, que podemos manipular para mantener en el cliente la información necesaria sin tener que estar continuamente cargándola desde el servidor. Las vistas en Backbone tienen la misión de organizar las distintas partes de la interfaz de la aplicación y mantenerlas actualizadas con el modelo. Y por ultimo las Rutas son usadas para manejar las urls de la aplicación. Y como se relacionan todas estas partes? Bueno todo comienza con una petición que realiza un usuario desde el navegador, la ruta envía la petición a una vista para que esta la procese. La vista por su parte pide información a los modelos y colecciones, y una vez devuelta esta información hace cambios en el DOM.

Proyecto:

Una vez explicado toda esta parte les comentamos que todos los cursos de DevCode la tratamos de explicar mediante la realización de aplicaciones. Y este curso no es la excepción así que en esta curso básico de Backbone.js vamos a desarrollar una aplicación para una librería, la cual podrá hacer lo siguiente:

Crear Actualizar Eliminar libros Mostrar detalle de un libro Y por ultimo listar todos los libros y poder buscarlos según autor y titulo. Para el diseño de la aplicación vamos a utilizar Bootstrap y para el lado del servidor utilizaremos node.js, estas partes no explicaremos ya que el curso no los cubre.

Bueno para terminar espero que les haya gustado esta introducción y puedan compartir por Twitter, Facebook y Google+ este video. Les espero en el próximo capítulo.