

jQuery, REST y otros frameworks

Adolfo Sanz De Diego

Octubre 2012

I

Acerca de



Figure: Pronoide

- Pronoide consolida sus servicios de formación superando las **22.000 horas impartidas** en más de 500 cursos (Diciembre 2011)
- En la vorágine de **tecnologías y marcos de trabajo existentes para la plataforma Java**, una empresa dedica demasiado esfuerzo en analizar, comparar y finalmente decidir cuáles son los pilares sobre los que construir sus proyectos.
- Nuestros Servicios de Formación Java permiten ayudarle en esta tarea, transfiriéndoles nuestra **experiencia real de más de 10 años**.

- **Adolfo Sanz De Diego**

- Correo: asanzdiego@gmail.com
- Twitter: [@asanzdiego](https://twitter.com/asanzdiego)
- Linkedin: <http://www.linkedin.com/in/asanzdiego>
- Blog: <http://asanzdiego.blogspot.com.es>

- **Este obra está bajo una licencia:**
 - Creative Commons Reconocimiento-CompartirIgual 3.0
- **El código fuente de los programas están bajo una licencia:**
 - GPL 3.0

II

jQuery

- Es una librería JavaScript que **simplifica el manejo del DOM** del HTML.
- Funciona seleccionando uno o varios elementos y ejecutando una acción sobre ellos.
- **“Write less, do more.”**
- La mejor API que he visto [jQuery Quick API Reference](#)

- Utiliza **los mismos que CSS**, y alguno más propio.

- jQuery está pensado para **recoger y/o lanzar** eventos.
- Estos eventos normalmente son **eventos de ratón, de teclado**.
- También maneja los **eventos de cambio de estado** de algún elemento del DOM.

- Podemos cambiar tanto el **atributo style**, como las **clases** de un elemento.

- **Ocultar, mostrar, desvanecer** elementos.
- También podemos hacer **animaciones** cambiando el CSS.
- Ver también CSS3 transitions jQuery plugin

- Podemos **cambiar el DOM** del HTML:
 - añadiendo texto tanto al principio como al final de un elemento,
 - cambiando el texto de un elemento,
 - añadiendo un elemento tanto antes como después de un elemento,
 - eliminando elementos.

- **Simplifica las peticiones AJAX**, pudiendo manejar su estado.

- Existe un **gran catálogo** de plugins.
- Los plugins **se crean de una forma muy sencilla**.
- Ver Tutorial Oficial

III

jQuery UI

- Son un conjunto de **componentes visuales**, con temas personalizables.
- Las clases CSS se pueden reutilizar. Ver The jQuery UI CSS Framework
- Las demos están muy bien, con muchos y muy buenos ejemplos jQuery UI Demos

- **Interactions:** draggable, droppable, resizable, selectable, sortable
- **Widgets:** accordion, autocomplete, button, datepicker, dialog, menu, progressbar, slider, spinner, tabs, tooltip
- **Effects:** blind, bounce, clip, drop, explode, fade, fold, highlight, pulsate, scale, shake, slide, transfer

IV

jQuery Mobile

- Son componentes pensados para **aplicaciones móviles**.
- Están pensados para el manejo de **eventos táctiles**.

- Toda la documentación del site está hecha con el propio framework.
- Ver jQuery Mobile
- Ver jQuery Mobile Documentation

V

REST

- REST (Representational State Transfer) es una técnica de arquitectura software para sistemas hipermedia distribuidos como la World Wide Web.
- Es decir, **una URL (Uniform Resource Locator) representa un recurso al que se puede acceder o modificar mediante los métodos del protocolo HTTP (POST, GET, PUT, DELETE).**
- Ver Artículos de REST de Enrique Amodeo Rubio (@eamodeorubio)

- Es **más sencillo** (tanto la API como la implementación).
- Es **más rápido** (peticiones más ligeras que se puede cachear).
- Es **multiformato** (HTML, XML, JSON, etc.).
- Se complementa muy bien con **AJAX**.

GET	-> <code>http://myhost.com/person</code>	-> Devuelve todas las personas
POST	-> <code>http://myhost.com/person</code>	-> Crear una nueva persona
GET	-> <code>http://myhost.com/person/123</code>	-> Devuelve la persona con id=123
PUT	-> <code>http://myhost.com/person/123</code>	-> Actualiza la persona con id=123
DELETE	-> <code>http://myhost.com/person/123</code>	-> Borra la persona con id=123

- **Se pueden utilizar los errores del protocolo HTTP:**
 - 200 OK Standard response for successful HTTP requests
 - 201 Created
 - 202 Accepted
 - 301 Moved Permanently
 - 400 Bad Request
 - 401 Unauthorised
 - 402 Payment Required
 - 403 Forbidden
 - 404 Not Found
 - 405 Method Not Allowed
 - 500 Internal Server Error
 - 501 Not Implemented

VI

Otros frameworks

- backbone.js implementa el **paradigma MVC en cliente**.
- Permite crear un **modelo que se actualiza con el servidor (mediante REST y AJAX)**, y que refresca las vistas que tenga asociadas en el **cliente**.
- Ver Tutorials, blog posts and example sites
- Ver Tutplus Backbone Tutorial

- require.js permite la **gestión de dependencias** entre módulos.
- Los **módulos se cargan de forma asíncrona** (según se vayan necesitando).
- Ver Organizing your application using Modules with require.js

- dust.js es un **motor de plantillas** en JavaScript.

- lesscss.js permite crear **archivos CSS de forma programática**.
- Se pueden usar archivos .less y la librería JavaScript (desarrollo) o “compilar” los archivos .less en archivos .css (producción)
- Aquí hay un buen tutorial sobre less

- modernizr.js **detecta las características CSS3 y HTML5 que soporta el navegador.**