Sistemas Web Desconectados

Defossé Nahuel, van Haaster Diego Marcos

8 de marzo de 2009

Índice general

Índice de figuras

Índice de cuadros

Agradecimientos

Resumen

Capítulo 1

Introducción

Capítulo 2

Aplicación del lado del servidor

2.1. El lenguaje de programación Python

2.2. Frameworks web basados en Python

2.3. Django

Django es un framework web escrito en Python[1] el cual sigue vagamente el concepto de Modelo Vista Controlador. Ideado inicialmente como un adminsitrador de contenido para varios sitios de noticias, los desarrolladores encontraron que su CMS era lo sufcientemente genérico como para curbir un ámbito más aplio de aplicaciones. Fue liberado[2] bajo la licencia BSD en Julio del 2005 como Django Web Framework en honor a Django Reinhart. En junio del 2008 fue anuncidada la cereación de la Django Software Fundation, la cual se hace cargo hasta la fecha del desarrollo y mantenimiento.

Django implementa una estructura *modelo*, *vista*, *plantilla*. O MTV por sus siglas en inglés (model, view, template).

Modelo Modelo de datos de la aplicación. Consiste en un mapeador Objeto Relacional.

Vista Funciones a ser ejecutadas cuando un cliente accede a la aplicación.

Template La generación de salida (típicamente código HTML)

[1] Lenguaje de programación orientado a objetos multiparadigma

[2] En el ámbito del software libre, la liberación es la fecha en la cual se pone a disposición de la comunidad del software en cuestión

2.4. Aplicación del lado del cliente

2.5. Modelo de ejecución en el cliente

2.6. Liberías de Javascript

Breve explicación de la existencia de tantas librerías de javascript, necesidad de utilización de estas librerías para consistencia, modularidad y portabilidad del código

2.6.1. Elementos de una librería de javascript

Manejo unificado de DOM

Problemas con las implementaciones de DOM de cada navegador, sacar material de la charla que dio Jhon Reisig (está en Google Techtalks)

Enfoque funcinal (encadenamiento)

Tipos de datos

Eventos

Ajax

Widgets

2.7. Ejemplos de librerías

2.7.1. jQuery

2.7.2. Prototype

2.7.3. YUI (Yahoo User Interface)

2.8. Javascript 1.7

Explicación de Javascript 2.0.

2.8.1. Protopy

Javascript 1.7 + modularidad + ejecución en contexto

Capítulo 3

Aplicación web desconectada

Doff, Django implementado sobre protopy.

3.0.2. Problemas de la traducción del framework a Javascript

3.0.3. Soporte en Django

Protopy

Apéndice A

Referencia Protopy

A.1. Módulos

A.1.1. event

event.connect Concetar

A.2. Plataforma Mozilla

- Porque desarrollaron e implementaron Javascript 1.7
- Porque javascript 1.7 tomo semántica (y sintaxis???) de Python
- Porque es código abierto
- Porque es extensible mediante plugins
  - Tiene firebug
  - Gears y firebug = muy compardor para el desarrollador.
-

Protopy persigue acercar la semántica del Javascript 1.7 a la del lenguaje Python. Las funcionalidades principales son las siguientes:

- Ámbito de nombres
- Semántica de objetos, dentro de la cual se hace una adaptación de
  - Iteradores
  - Generadores
- Iteradores
- Generadores
- Tipos básicos de la librería estándar de python, entre los que se encuentran: *bool*,

type Type sirve para defninir clases.

Bibliografía

[1] Versiones de Javascript, Jhon Reisig