

Creando aplicaciones web con CakePHP 2.x, HTML5 y CSS3



Alex Arriaga

@alex_arriaga_m
<http://fenix.cs.buap.mx>



Temario

- Instalación de CakePHP 2.x
- Comprendiendo MVC, ORM y ActiveRecord
- Personalizando CakePHP
- *Scaffolding*
- Helpers
- Consola
- HTML5 en Layouts
- CSS3 básico

¿Framework?



- ¿Cuándo usar un Framework?
- Frameworks vs CMS vs Programación desde cero
- ¿Por qué usar un framework?
- ¿Existen más frameworks?
- ¿Qué necesito para comenzar?

¿Por qué CakePHP?



- Construcción rápida
- Configuración mínima
- Basados en MVC
- Seguro
- Escalable
- Estricto
- Software libre

Herramientas



Editores de texto, IDE's

Windows

- [Notepad++](#), [Eclipse](#), [Netbeans](#), [Aptana Studio](#)

Linux

- Bluefish, Gedit, Nano, Vim, [Eclipse](#), [Netbeans](#), [Aptana Studio](#)

Mac

- [Eclipse](#), [Netbeans](#), [Aptana Studio](#)

Navegadores web



Requerimientos



Software

- PHP 5.2.8+

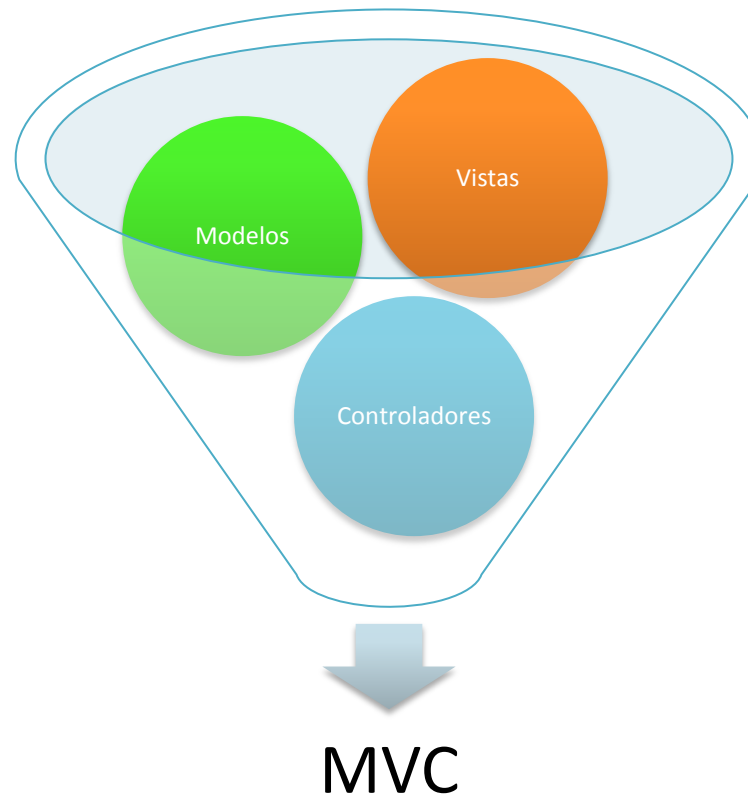
Bases de Datos

- MySQL 4+
- Microsoft SQL Server
- PostgreSQL
- SQLite

Servidores web soportados

- Apache 2.x+

Comprendiendo el MVC



MVC

Layouts, Pages, HTML5 y CSS3...

De la vista nace el amor



- ¿Por qué usar HTML5 y CSS3 en nuestras aplicaciones?
- ¿Qué pasa con IE 7, IE8?
- ¿Vale la pena invertir tiempo, dinero y esfuerzo en aprender nuevas tecnologías?
- ¿Puedo integrar HTML5 a CakePHP2?



HTML5

HTML5 y las funcionalidades del “futuro inmediato”



Semántica



Multimedia



Desempeño e integración



CSS3



Conectividad



Acceso desde móviles



Almacenamiento local y *Offline*



3D, Gráficos y Efectos

CSS



CSS 2 (conocida como CSS 2.1) corregida y publicada por recomendación oficial el 7 de junio de 2011.

CSS 3

- Dividida en documentos separados (módulos)
- Preserva funcionalidades de CSS2
- Primeros borradores en 1999

Desventajas de CSS

- No existen expresiones de cálculo (p.e. padding: 10% - 2em + 2px)
- El centrado y el alineamiento vertical es complejo y no estándar
- Pseudo-clases (p.e. a:hover) no se pueden deshabilitar desde el navegador
- Algunos navegadores (especialmente IE 7+) no respetan los estándares CSS

CSS



Ventajas

- Separación entre la **estructura** del documento y su **diseño**
- Un mismo archivo CSS puede contener reglas para múltiples documentos
- Mejora la **accesibilidad** web, pues se evitan prácticas antiguas (p.e el manejo de tablas)
- Existen **validadores** para este lenguaje
- Se puede combinar con bibliotecas/frameworks javascript (p.e. **jQuery**, **Mootools**, **Modernizr**, etc.) para proveer compatibilidad con navegadores antiguos
- Con pocas reglas se pueden “estilizar” documentos completos
- Si alguna vez se desea realizar una **modificación** sobre diseño, con modificar **unas** líneas automáticamente se modifican **todos** los documentos.

¡Manos a la obra!



Práctica 1. Definiendo la estructura de un documento HTML5 y dando estilos base

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <link rel="stylesheet" href="css/style.css">
    ...
    <script src="js/script.js"></script>
</head>
<body>
    <header></header>
    <section></section>
    ...
    <footer></footer>
</body>
</html>
```

¡Manos a la obra!



Práctica 2. Clases vs Id's en CSS y los estilos en etiquetas HTML

Clases:

Especifica estilos para un grupo de elementos.

Se define mediante “.”

Id:

Especifica estilos para un único elemento en todo el documento.

Se define mediante ‘#’.

También se pueden dar estilos particulares a las etiquetas HTML de un documento. Veamos cómo...

¡Manos a la obra!




Práctica 3. Implementando HTML5 en un layout

- Tomar como base el layout default
- Generar la estructura básica de un documento HTML5
- Generar la estructura solicitada por CakePHP2
- Integrar jQuery, jQuery UI y Modernizr
- Estilizar mediante CSS2 y CSS3 el nuevo layout

Gracias por su atención



- Más material en:
<http://softlite.mx/cakephp/>
<http://softlite.mx/html5/>
- Dudas, comentarios y sugerencias:
alex_arriaga_m
 **alex.arriaga.m@gmail.com**

