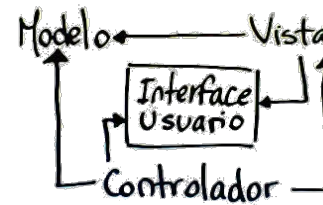
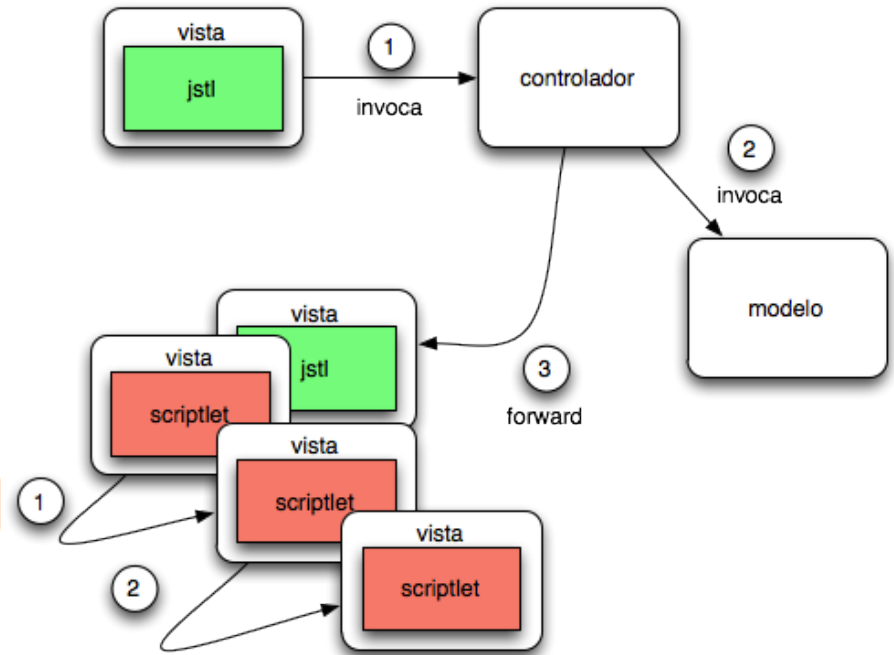
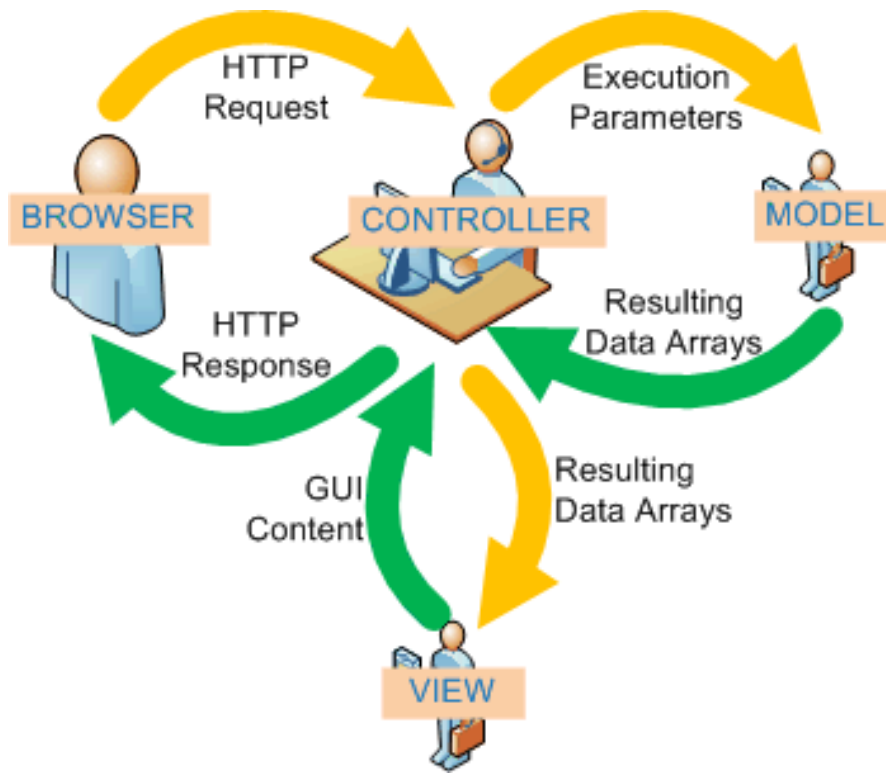


Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos y Patrón MVC en PHP5

Pablo Ramirez A. (Arquitecto de Software)

Presentación:



Pregrado:

Técnico Superior en Informática (**CUAM**) 2000
Ingeniero en Sistemas (**UBA**) 2009
Ingeniería en Informática (**IUTOMS**) (Cursando)

Postgrado:

Especialista en Gerencia de la Ciencia y Tecnología (**URBE**) 2013
Especialista en Gerencia General (**UNESR**) (Cursando)

Diplomados:

- Diplomado en Gerencia de Proyectos (UCAB - PMBOK 2012)
- Diplomado en Administración Tecnológica y Funcional del SIGESP
- Estudios Avanzados en Tecnologías de Información y Comunicación (CIAP UCAB)

Certificaciones:

- SL-110 SE6 Programación Java TM para no programadores.
- SL-275 SE6 Programación Java.
- SL-314-EE5 Desarrollo de Componentes Web con Servlet y JSP.
- Certificado en Genexus 9.0 (Aplicaciones Web con Java)

Pablo Ramirez A.

Programación Orientada a Objetos y Patrón MVC en PHP5

Temario:

1.- Introducción:

- Como funciona una aplicación WEB
- Proyección de la WEB
- Introducción a PHP5
- PHP5 y Programación Orientada a Objetos

2.- Introducción a la Programación Orientada a Objetos (POO).

- Elementos de la POO.
- Características conceptuales de la POO.

3.- Programación Orientada a Objetos en PHP5.

- Clases y Objetos en PHP5.
- Propiedades en PHP5.
- Acceso a las propiedades de un objeto.
- Constantes de clase
- Métodos en PHP5

4.- Introducción al Patrón Arquitectónico MVC

- ¿Qué es la arquitectura de software?
- Tendencias de la Arquitectura de Software
- El Patrón Arquitectónico modelo-vista-controlador (MVC)
 - ¿Qué es el patrón MVC?
 - ¿Cómo funciona el patrón MVC?

5.- Framework

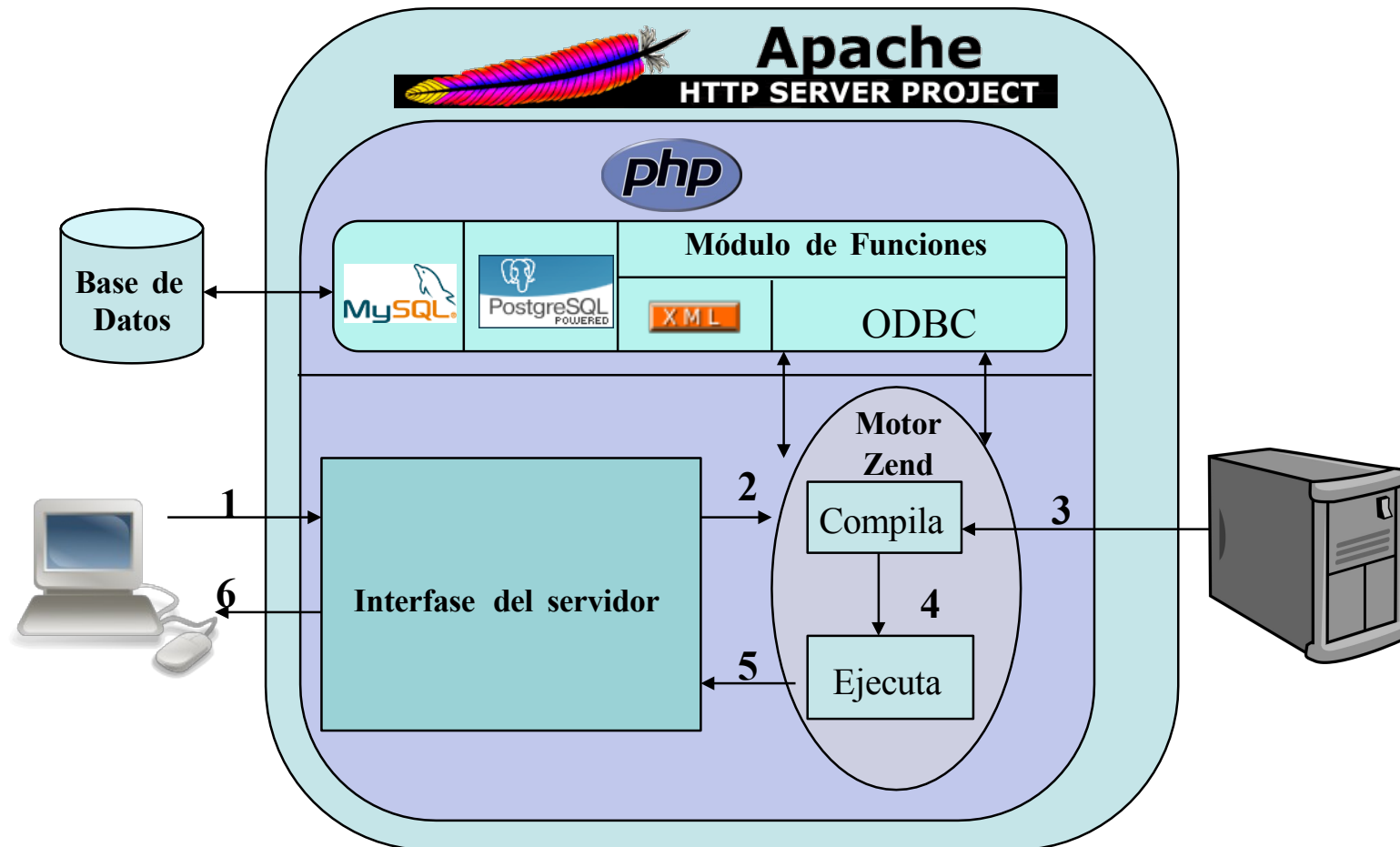
6.- Comparativa de 10 Framework en PHP

7.- Demo.

8.- Conclusión

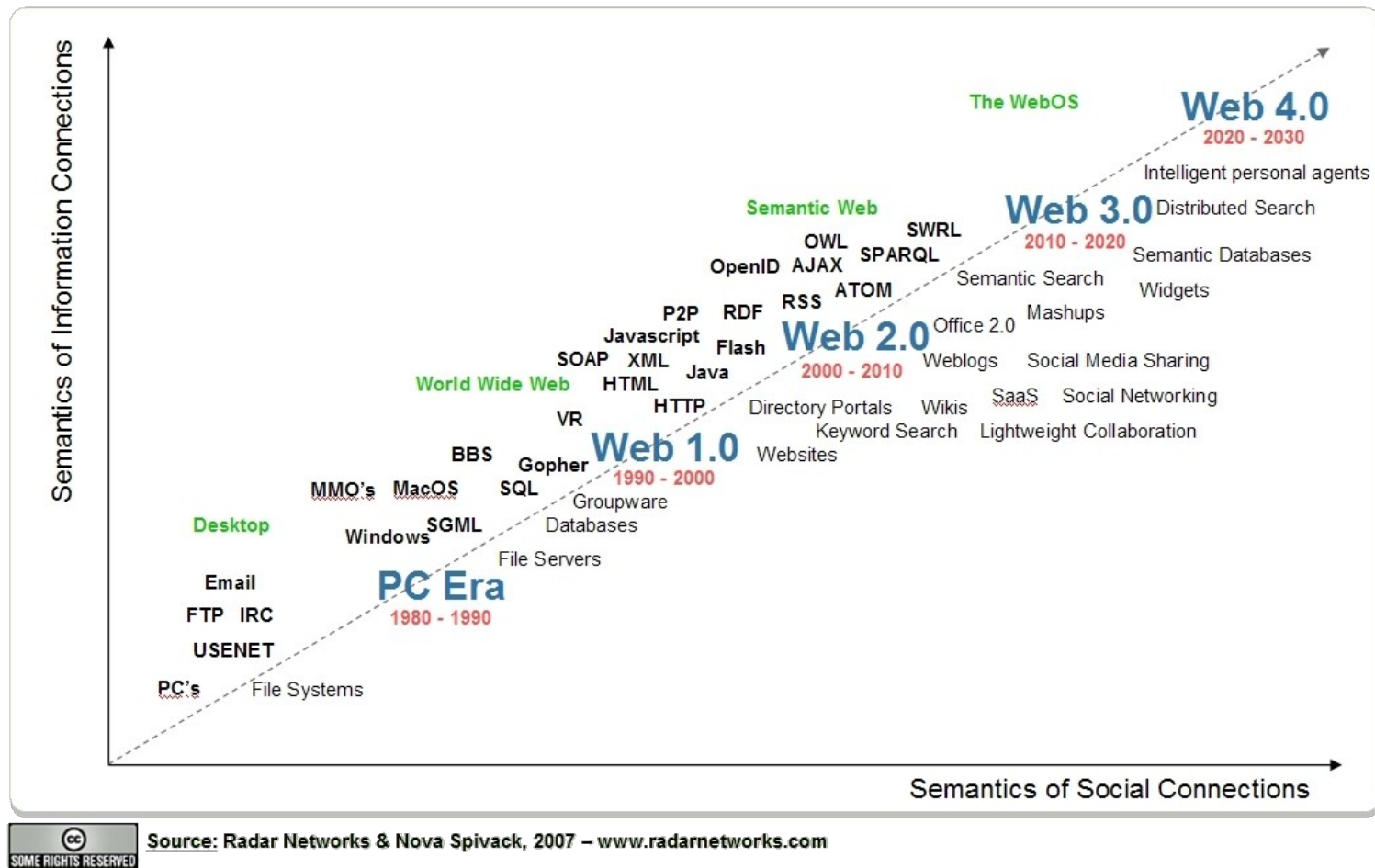
Programación Orientada a Objetos y Patrón MVC en PHP5

Como Funciona una Aplicación Web



Programación Orientada a Objetos y Patrón MVC en PHP5

Proyecciones de la Web.



Programación Orientada a Objetos y Patrón MVC en PHP5



Introducción a PHP5.

- Soporte sólido para Programación Orientada a Objetos (OOP)
- Data Objects.
- Mejoras de rendimiento.
- Mejor soporte para MySQL
- Mejor soporte a XML (XPath, DOM, entre otros).
- Soporte nativo para SQLite.
- Soporte integrado para SOAP.
- Iteradores de datos.
- Excepciones de errores.

Programación Orientada a Objetos y Patrón MVC en PHP5



PHP5 y Programación Orientación a Objetos.

PHP5 soporta una gran cantidad de funcionalidades para OOP.

- Métodos constructores y destructores.
- Métodos setter y getter.
- Métodos mágicos.
- Clases, objetos y variables estáticas, privadas y protegidas.
- Clases abstractas.
- Interfaces.
- Abstracción de datos.
- Standard PHP Library (SPL).
- Clases extendidas, excepciones, iteradores.

Programación Orientada a Objetos y Patrón MVC en PHP5



Introducción a la Programación Orientada a Objetos (POO)

La POO es un paradigma de programación (o técnica de programación) que utiliza objetos e interacciones en el diseño de un sistema.

Elementos de la POO.

Clases

Una clase es un modelo que se utiliza para crear objetos que comparten un mismo comportamiento, estado e identidad.

```
class Persona {  
    # Propiedades  
    # Métodos  
}
```


Programación Orientada a Objetos y Patrón MVC en PHP5



Introducción a la Programación Orientada a Objetos (POO)

Elementos de la POO.

Objetos

Es una entidad provista de métodos o mensajes a los cuales responde (comportamiento); atributos con valores concretos (estado); y propiedades (identidad).

```
$persona = new Persona();  
/*  
    El objeto, ahora, es $persona,  
    que se ha creado siguiendo el modelo de la clase Persona  
*/
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Introducción a la Programación Orientada a Objetos (POO)

Elementos de la POO.

Métodos

Es el algoritmo asociado a un objeto que indica la capacidad de lo que éste puede hacer.

```
function caminar() {  
    #...  
}
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Introducción a la Programación Orientada a Objetos (POO)

Elementos de la POO.

Evento y Mensaje

Un evento es un suceso en el sistema mientras que un mensaje es la comunicación del suceso dirigida al objeto.

Propiedades y atributos

Las propiedades y atributos, son variables que contienen datos asociados a un objeto.

```
$nombre = 'Juan';  
$edad = '25 años';  
$altura = '1,75 mts';
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Características Conceptuales de la POO

Abstracción

Aislación de un elemento de su contexto. Define las características esenciales de un objeto.

Encapsulamiento

Reúne al mismo nivel de abstracción, a todos los elementos que puedan considerarse pertenecientes a una misma entidad.

Modularidad

Característica que permite dividir una aplicación en varias partes más pequeñas (denominadas módulos), independientes unas de otras.

Ocultación (aislamiento)

Los objetos están aislados del exterior, protegiendo a sus propiedades para no ser modificadas por aquellos que no tengan derecho a acceder a las mismas.

Programación Orientada a Objetos y Patrón MVC en PHP5



Características Conceptuales de la POO

Polimorfismo

Es la capacidad que da a diferentes objetos, la posibilidad de contar con métodos, propiedades y atributos de igual nombre, sin que los de un objeto interfieran con el de otro.

Herencia

Es la relación existente entre dos o más clases, donde una es la principal (madre) y otras son secundarias y dependen (heredan) de ellas (clases “hijas”), donde a la vez, los objetos heredan las características de los objetos de los cuales heredan.

Recolección de basura

Es la técnica que consiste en destruir aquellos objetos cuando ya no son necesarios, liberándolos de la memoria.

Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

CLASES Y OBJETOS EN PHP5.

CLASE:

```
class NombreDeMiClase {  
    #...  
}
```

CLASE ABSTRACTA:

```
abstract class NombreDeMiClaseAbstracta {  
    #...  
}
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

CLASES Y OBJETOS EN PHP5.

HERENCIA DE CLASES:

```
class NombreDeMiClaseMadre {  
    #...  
}  
  
class NombreDeMiClaseHija extends NombreDeMiClaseMadre {  
    /* esta clase hereda todos los métodos y propiedades de  
       la clase madre NombreDeMiClaseMadre  
    */  
}
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

CLASES Y OBJETOS EN PHP5.

CLASES FINALES:

```
final class NombreDeMiClaseFinal {  
    #esta clase no podrá ser heredada  
}
```

INSTANCIAR UNA CLASE

```
# declaro la clase  
class Persona {  
    #...  
}  
  
# creo el objeto instanciando la clase  
$persona = new Persona();
```


Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

PROPIEDADES EN PHP5.

```
class Persona {  
    var $nombre;  
    var $edad;  
    var $genero;  
}
```

PROPIEDADES PÚBLICA

```
class Persona {  
    public $nombre;  
    public $genero;  
}
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

PROPIEDADES EN PHP5.

PROPIEDADES PRIVADAS

```
class Persona {  
    public $nombre;  
    public $genero;  
    private $edad;  
}
```

PROPIEDADES PROTEGIDAS

```
class Persona {  
    public $nombre;  
    public $genero;  
    private $edad;  
    protected $pasaporte;  
}
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

PROPIEDADES EN PHP5.

PROPIEDADES ESTÁTICAS

```
class PersonaAPositivo extends Persona {  
    public static $tipo_sangre = 'A+';  
}
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

ACCEDIENDO A LAS PROPIEDADES DE UN OBJETO

ACCEDIENDO A UNA VARIABLE DESDE UNA CLASE

```
return $this->nombre;  
print self::$variable_estatica_de_esta_clase;  
print parent::$variable_estatica_de_clase_madre;
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

ACCEDIENDO A LAS PROPIEDADES DE UN OBJETO

ACCEDIENDO A UNA VARIABLE DESDE EL EXTERIOR DE UNA CLASE

```
# creo el objeto instanciando la clase
$persona_a_positivo = new PersonaAPositivo();

# accedo a la variable NO estática
print $persona_a_positivo->nombre;
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

CONSTANTES DE CLASE

```
const MI_CONSTANTE = 'Este es el valor estático de mi constante';
```

METODOS EN PHP5

```
# declaro la clase  
class Persona {  
    #propiedades  
  
    #métodos  
    function donar_sangre() {  
        #...  
    }  
}
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

MÉTODOS MÁGICOS EN PHP5

```
# constructor de la clase
function __construct() {
    $this->set_estado_producto('en uso');
}

# destructor de la clase
function __destruct() {
    $this->set_estado_producto('liberado');
    print 'El objeto ha sido destruido';
}
```

Programación Orientada a Objetos y Patrón MVC en PHP5



Programación Orientada a Objetos (POO) en PHP5

Según el Manual Oficial de PHP.

MÉTODOS MÁGICOS EN PHP5

Otros métodos mágicos

PHP nos ofrece otros métodos mágicos tales como `__call`, `__callStatic`, `__get`, `__set`, `__isset`, `__unset`, `__sleep`, `__wakeup`, `__toString`, `__invoke`, `__set_state` y `__clone`.

Puede verse una descripción y ejemplo de su uso, en el sitio Web oficial de PHP:

<http://www.php.net/manual/es/language.oop5.magic.php>

Programación Orientada a Objetos y Patrón MVC en PHP5

Introducción al Patrón Arquitectónico MVC

Que es la arquitectura de software

Es la forma en la que se organizan los componentes de un sistema, interactúan y se relacionan entre sí y con el contexto, aplicando normas y principios de diseño y calidad, que fortalezcan y fomenten la usabilidad a la vez que dejan preparado el sistema, para su propia evolución”.

Tendencias de la Arquitectura de Software

La Arquitectura de Software Orientada a Objetos (como “ingeniería” de sistemas).

La Arquitectura Estructurada (como “desarrollo” de una aplicación).

La arquitectura basada en patrones.

La arquitectura basada en procesos y metodologías.

Programación Orientada a Objetos y Patrón MVC en PHP5

El Patrón Arquitectónico MVC

¿Qué es el patrón MVC?

El patrón MVC es un patrón de arquitectura de software encargado de separar la lógica de negocio de la interfaz del usuario y es el más utilizado en aplicaciones Web, ya que facilita la funcionalidad, mantenibilidad y escalabilidad del sistema, de forma simple y sencilla.

MVC divide las aplicaciones en tres niveles de abstracción:

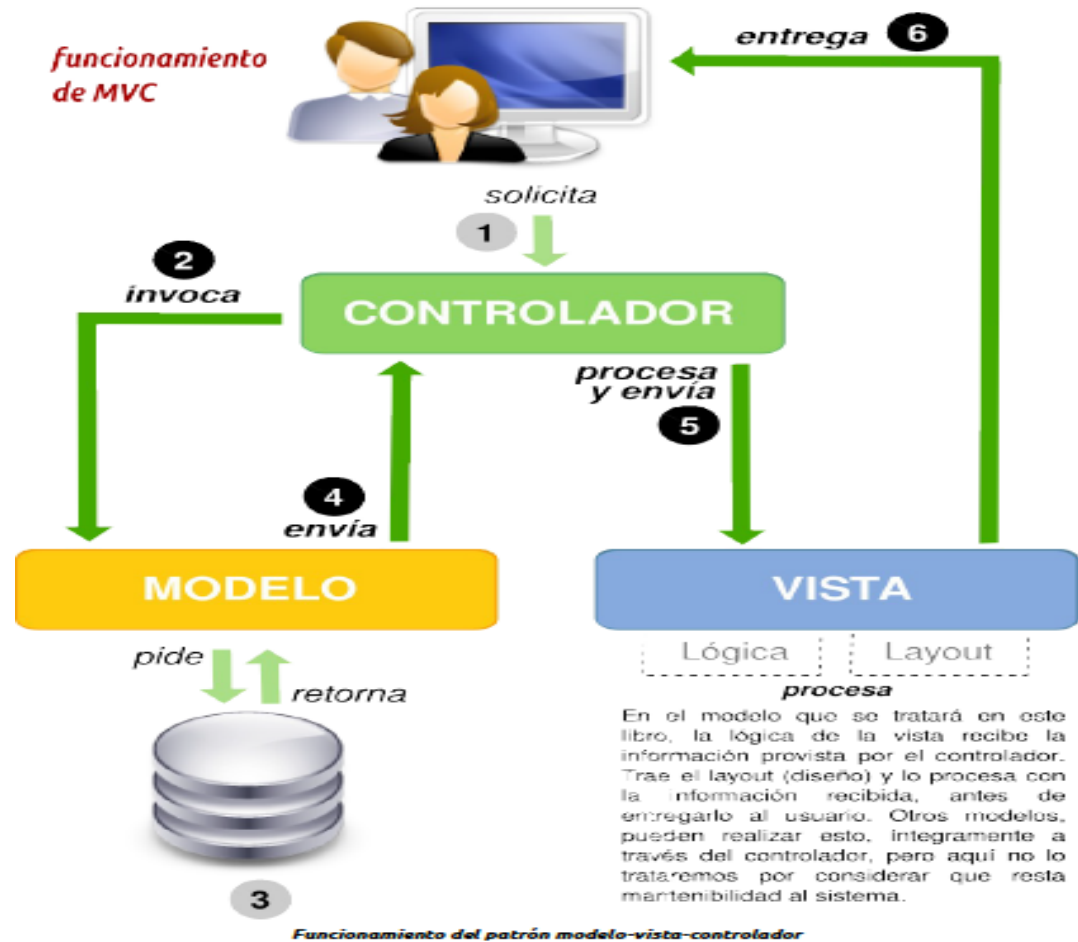
Modelo: representa la lógica de negocios.

Vista: es la encargada de mostrar la información al usuario de forma gráfica y “humanamente legible”.

Controlador: es el intermediario entre la vista y el modelo.

Programación Orientada a Objetos y Patrón MVC en PHP5

Como Funciona el Patrón Arquitectónico MVC



Programación Orientada a Objetos y Patrón MVC en PHP5

Frameworks

Un Framework, “Marco de Trabajo” o WAF (Web Application Framework), es una serie de librerías (toolkits) que se han unido bajo un único esquema de colaboración para que de manera rápida (RAD: Rapid Application Development) logremos desarrollar nuestras aplicaciones.

Entre las ventajas se cuenta:

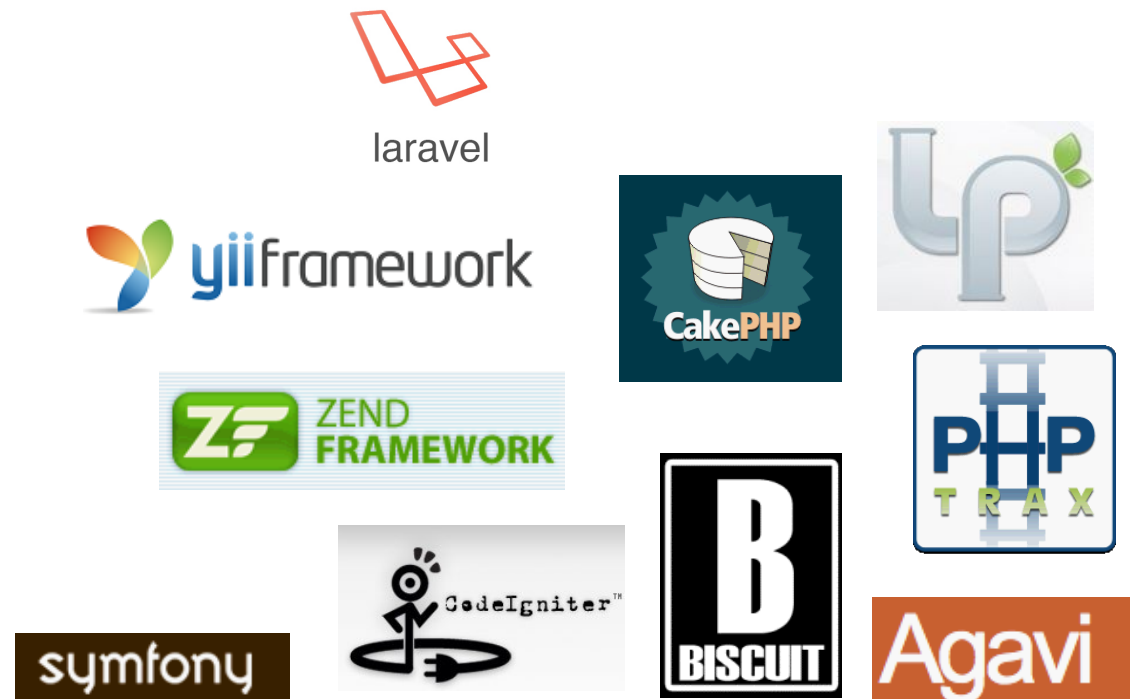
- No hay que definir “marco de desarrollo”, solo “rellenar” los huecos que los frameworks nos indican.
- Trabajamos sobre un Estándar que miles de personas ya conocen.
- Facilidad para encontrar herramientas, librerías o documentación.
- Relación Coste de aprendizaje vs. Aplicabilidad mínima.

Programación Orientada a Objetos y Patrón MVC en PHP5

Frameworks

Algunos Framework MVC:

- Yii Framework.
- CakePHP.
- CodeIgniter.
- Symfony.
- PHP on Trax.
- Zend Framework.
- Laravel Framework, entre otros...



Programación Orientada a Objetos y Patrón MVC en PHP5



Comparativa de 10 Framework en PHP

Framework	PHP4	PHP5	MVC ¹	Multiple DB's ²	ORM ³	DB Objects ⁴	Templates ⁵	Caching ⁶	Validation ⁷	Ajax ⁸	Auth Module ⁹	Modules ¹⁰
Zend Framework	-	✓	✓	✓	-	✓	-	✓	✓	-	-	✓
CakePHP	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	-
Symfony Project	-	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	-
Seagull Framework	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	✓	✓
WACT	✓	✓	✓	✓	-	✓	✓	-	✓	-	-	-
Prado	-	✓	-	✓	-	-	✓	✓	✓	✓	✓	✓
PHP on TRAX	-	✓	✓	✓	✓	✓	-	-	✓	✓	-	-
ZooP Framework	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	✓
eZ Components	-	✓	-	✓	-	✓	✓	✓	✓	-	-	✓
CodeIgniter	✓	✓	✓	✓	-	✓	✓	✓	✓	-	-	✓

#1: Indicates whether the framework comes with inbuilt support for a Model-View-Controller setup.

#2: Indicates whether the framework supports multiple databases without having to change anything.

#3: Indicates whether the framework supports an object-record mapper, usually an implementation of ActiveRecord.

#4: Indicates whether the framework includes other database objects, like a TableGateWay.

#5: Indicates whether the framework has an inbuilt template engine.

#6: Indicates whether the framework includes a caching object or some way other way of caching.

#7: Indicates whether the framework has an inbuilt validation or filtering component.

#8: Indicates whether the framework comes with inbuilt support for Ajax.

#9: Indicates whether the framework has an inbuilt module for handling user authentication.

#10: Indicates whether the framework has other modules, like an RSS feed parser, PDF module or anything else (useful).

Copyright © 2006 Dennis Pallett, [PHPit - Totally PHP](#).

Programación Orientada a Objetos y Patrón MVC en PHP5

Conclusiones

El desarrollo de web aplicaciones es un conjunto de tecnologías, estándares y normativas de accesibilidad, reusabilidad y operatividad de la Web que se deben cumplir para evitar posteriores “dolores de cabeza” por desconocimiento del entorno de despliegue de la aplicación, es decir la Web.

Los patrones de diseño, es una forma simple de resolver problemas cotidianos y que que son muy repetitivos en nuestro entorno. No requiere una implantación con lenguaje de programación por lo que ofrece, un idioma común entre programadores de distintos lenguajes.

Programación Orientada a Objetos y Patrón MVC en PHP5

Conclusiones :: Continuación.

La programación MVC aplicada a la web no es una opción, ¡es una necesidad!

¿Por que?

- Desarrollas para una interfaz, no para una implementación.
- Todo tu equipo se dedica a su especialidad preferida.
- Desarrollo sostenible y mantenible.
- Es posible migrar en diferentes direcciones.
- Evolución rápida de la aplicación.
- ¡Terminas más rápido!.

Programación Orientada a Objetos y Patrón MVC en PHP5

Conclusiones :: Continuación.

Los llamados “Framework” o , “WAF”, es una serie de librerías (toolkits) que se han unido bajo un único esquema de colaboración para que de manera rápida (RAD: Rapid Application Development) logremos desarrollar nuestras aplicaciones.

Programación Orientada a Objetos y Patrón MVC en PHP5

¡Gracias por su Atención!

"Programar sin una arquitectura o diseño en mente es como explorar una gruta sólo con una linterna: no sabes dónde estás, dónde has estado ni hacia dónde vas"

Autor: Danny Thorpe

"En el mundo del software, los activos más importantes de la compañía se van a casa todas las noches. Si no se les trata bien, pueden no volver al día siguiente"

Autor: Peter Chang