



# Introducción al patrón MVC (Modelo-Vista-Controlador)

**2º DAW - Desarrollo Web en Entorno Servidor**

*David Soler Talens*

# Que es MVC?



- **MVC**

*“Patrón de arquitectura de software encargado de separar la lógica de negocio de la interfaz del usuario”*

- **Es el más utilizado en aplicaciones Web**

- **facilita la**

- Funcionalidad
- Mantenibilidad
- escalabilidad del sistema

# Niveles de abstracción



- **Modelo:**

- Representa la lógica de negocios.
- Accede de forma directa a los datos actuando como "intermediario" con la base de datos.

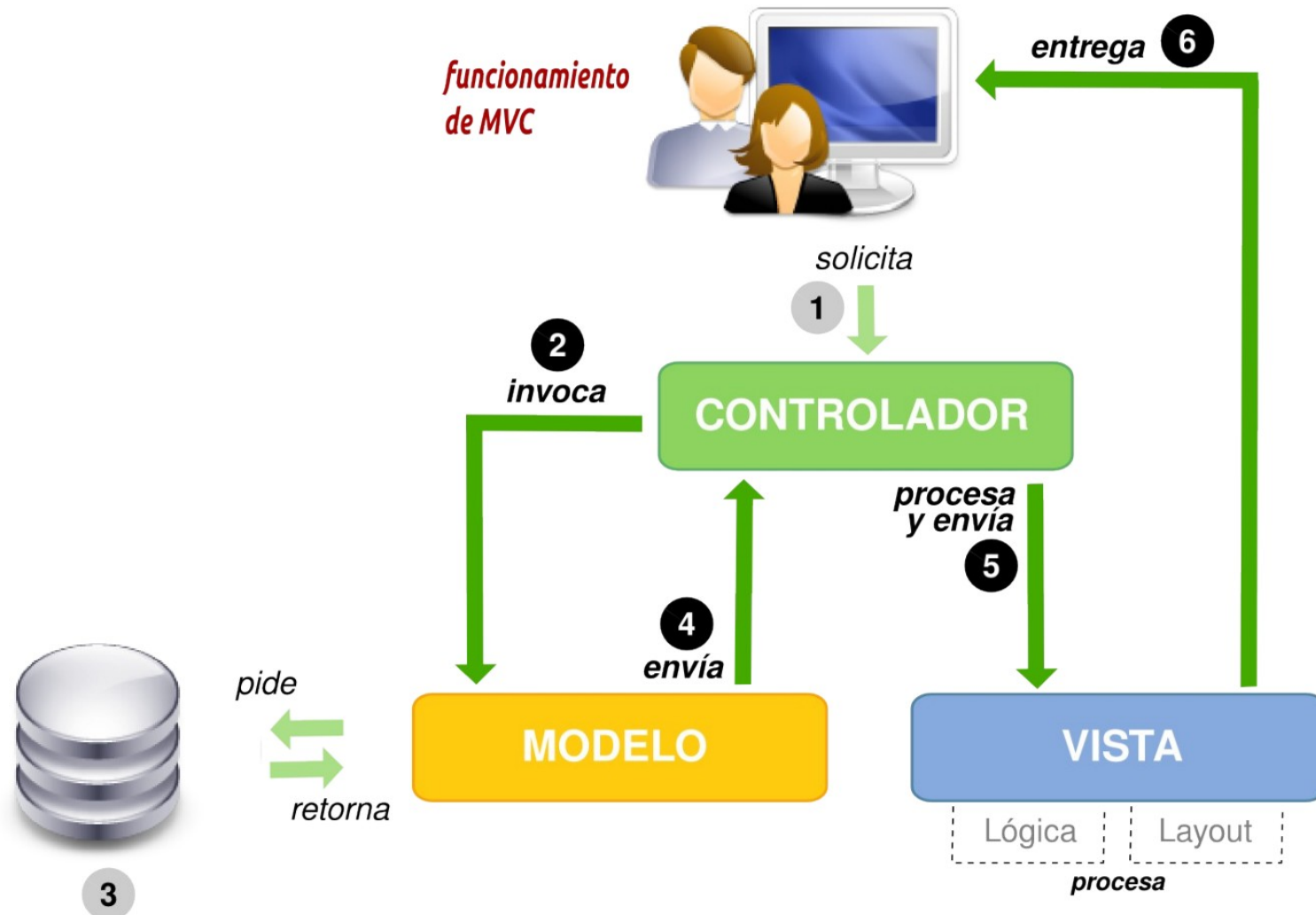
- **Vista:**

- Encargada de mostrar la información al usuario de forma gráfica y "legible".

- **Controlador:**

- Intermediario entre la vista y el modelo.
- Controla las interacciones del usuario solicitando los datos al modelo y entregándolos a la vista.

# Funcionamiento básico de MVC (I)



# Funcionamiento básico de MVC (II)



- 1. El usuario realiza una petición**
- 2. El controlador captura el evento**
- 3. Hace la llamada al modelo/modelos correspondientes.**
- 4. El modelo interactúa con la base de datos y retorna la información al controlador.**
- 5. El controlador recibe la información y la envía a la vista**
- 6. La vista, procesa la información**
  - a) Capa de abstracción para la lógica (procesa datos)
  - b) Capa para el diseño de la interfaz gráfica (los "acomoda" al GUI)

# MVC sencillo (I)



## 1. Crear un diccionario de datos

Qué referencias deben ser reemplazadas por qué datos

## 2. Obtener la plantilla HTML

- Traer la plantilla HTML previamente diseñada, al código PHP
- La plantilla debe definir las referencias p.ej. con `{}`

## 3. Reemplazar el contenido

Hacer la sustitución de referencias por datos en la plantilla

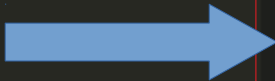
## 4. Mostrar el contenido final al usuario

Salida de la plantilla «*renderizada*»

# MVC sencillo (II)



```
1  <!DOCTYPE HTML>
2  <html>
3  <head>
4      <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
5      <title>{page_title}</title>
6      <meta name="title" content="{page_title}">
7      <meta name="keywords" content="{keywords}">
8      <meta name="description" content="{description}">
9      <link type="text/css" rel="stylesheet" href="/MVC/sample_code/css/template.css">
10 </head>
11
12 <body>
13     <div id="page">
14         <div id="cab">
15             </a>
16         </div>
17         <div id="titulo">
18             <h1>{page_title}</h1>
19         </div>
20         <div id="contenido">
21             {content}
22         </div>
23     </div>
24 </body>
25 </html>
```




- ▼ css
  - # template.css
- ▼ img
  - 🖼 logo.jpeg
- ▼ templates
  - <> template.html
- 🐘 controller.php
- 🐘 model.php
- 🐘 view.php

# MVC sencillo (III)



```
1  <?php
2      // modelo -> proporciona los datos necesarios a la aplicación
3      $diccionario = array(
4          'page_title'=>'MVC en PHP',
5          'keywords'=>'poo, mvc, php, arquitectura de
6          software',
7          'description'=>'ponemos en práctica el MVC
8          en PHP',
9          'content'=>'Lorem ipsum dolor sit amet,
          consectetur adipiscing elit. In ut lorem eu
          justo blandit laoreet. Curabitur sem nisl,
          eleifend fermentum sem nec, condimentum
          sodales mauris. Duis facilisis hendrerit
          fermentum. Suspendisse venenatis ipsum nisl,
          nec scelerisque velit faucibus ac. Aliquam
          erat volutpat. Donec quis enim non risus
          gravida semper eget eget tellus. Sed
          fermentum cursus consectetur.'
8      );
9  ?>
```



- ▼ css
  - # template.css
- ▼ img
  - logo.jpeg
- ▼ templates
  - <> template.html
- controller.php
- model.php
- view.php


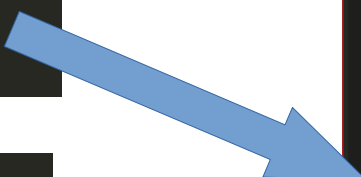


# MVC sencillo (IV)



```
1 <?php
2     require('model.php'); //pide al modelo los
   datos necesarios
3     require('view.php') //envia a la vista los
   datos para su visualización
4 ?>
```

```
1 <?php
2     $template = file_get_contents('templates/
   template.html');
3     foreach ($diccionario as $clave=>$valor) {
4         $template = str_replace('{'.$clave.'}',
   $valor, $template);
5     };
6     echo $template;
7 ?>
```



- ▼ css
  - # template.css
- ▼ img
  - 🖼 logo.jpeg
- ▼ templates
  - <> template.html
- 🐘 controller.php
- 🐘 model.php
- 🐘 view.php

# MVC sencillo (V)



## 1. Modelo

- Deberia contener la interaccion con BBDD
- Deberia implementar métodos como mínimo para
  - Conectar, desconectar
  - CRUD (Create, Read, Update, Delete)

## 2. Vista

- Deberia poder renderizar diferentes situaciones además de las del tipo referencia => valor, como estructuras repetitivas, condicionales, etc...

## 3. Controlador

- Deberia poder renderizar diferentes plantillas, en función de diferentes peticiones del usuario
- Necesitamos un «*Front Controller*»