



Ayudantía de Proyecto IIC2413

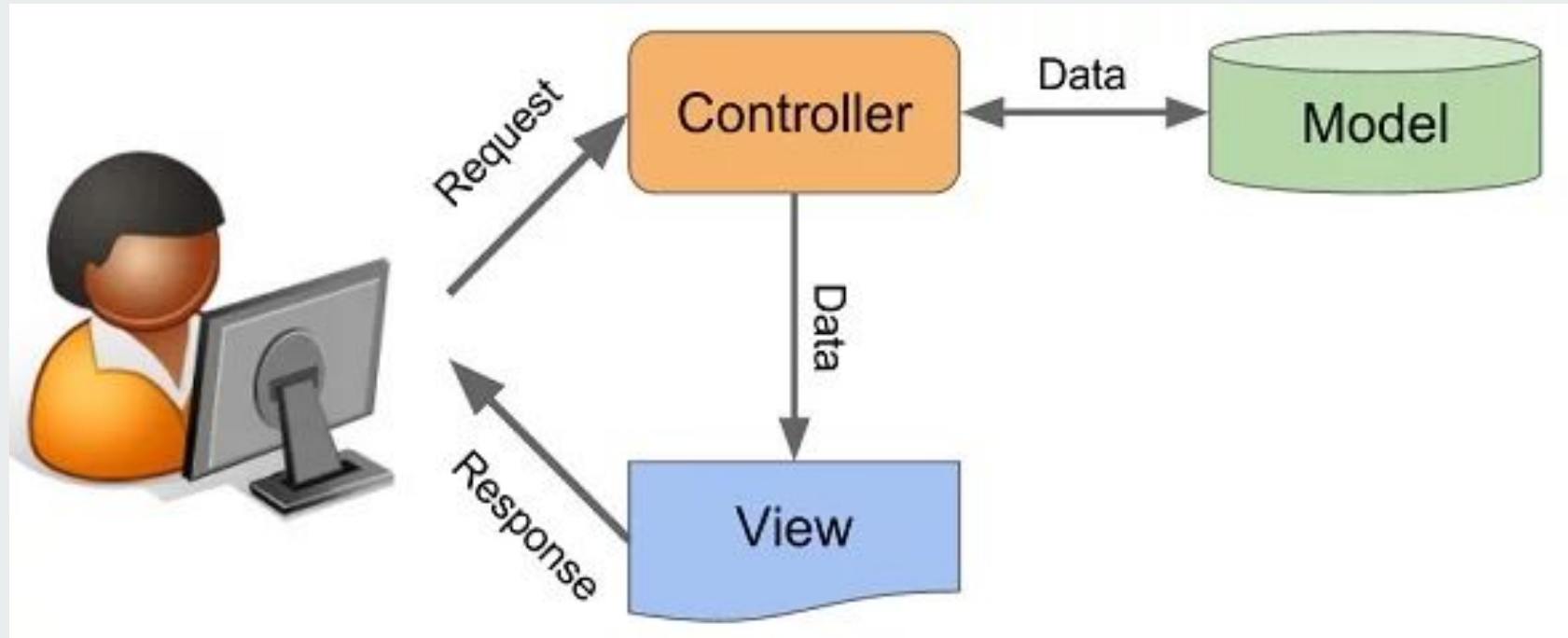
2019 - 2

Patrones de Arquitectura de Software

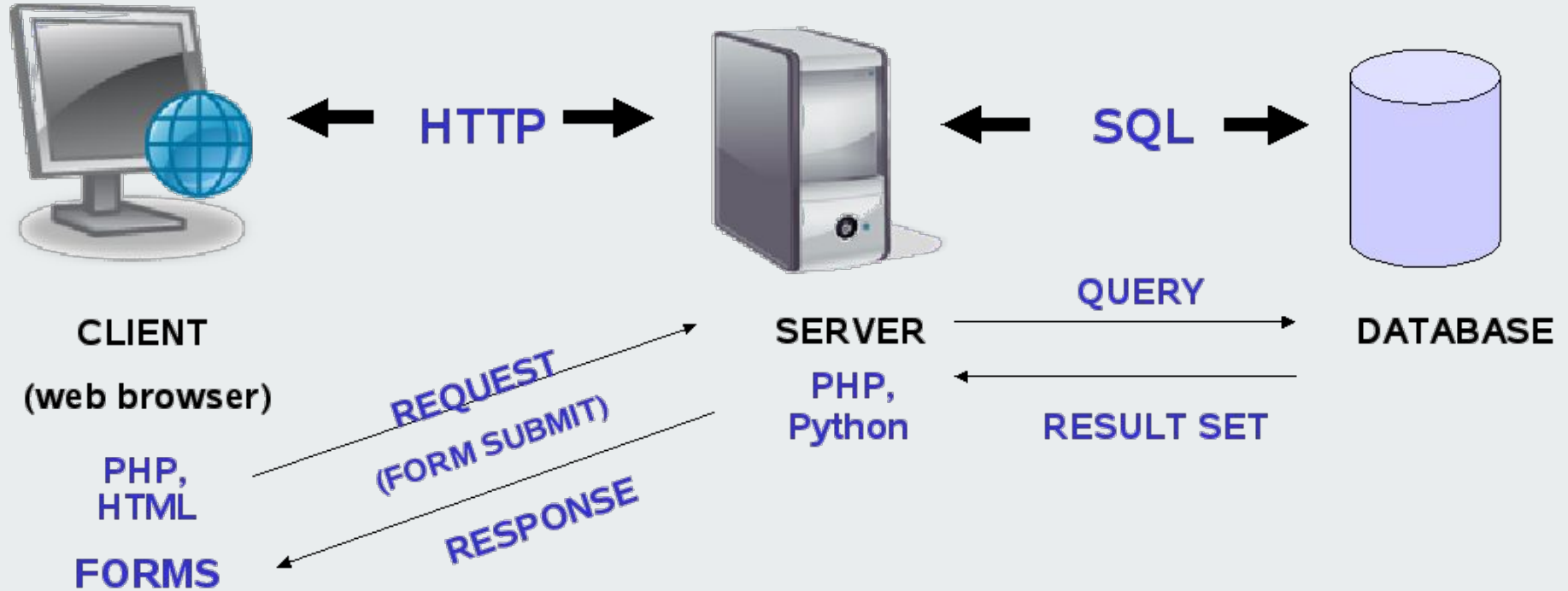


- Arquitectura cliente-servidor
- Arquitectura por capas
- Arquitectura de microservicios
- Arquitectura dirigida por eventos
- Arquitectura punto a punto
- Arquitectura orientada a servicios
- Modelo Vista Controlador

Modelo Vista Controlador



En nuestro caso...



En nuestro caso...



Modelo:

- No haremos archivos que contengan lógica del modelo, porque consultaremos directamente los datos.
- Usaremos un modelo relacional implementado en una base de datos Postgresql.

Vista:

- HTML
- CSS y Javascript

Controlador:

- PHP (incrustado en la Vista)

En nuestro caso...



Modelo:

- No haremos archivos que contengan lógica del modelo, porque consultaremos directamente los datos.
- Usaremos un modelo relacional implementado en una base de datos Postgresql.

Vista:

- HTML
- ~~CSS y Javascript~~ (bonus)

Controlador:

- PHP (incrustado en la Vista)

En nuestro caso...



Modelo:

- Modelo relacional
- Base de datos Postgresql

Vista:

- HTML
- ~~CSS (bonus)~~
- ~~Javascript (bonus)~~

Controlador:

- PHP

HTML



HyperText Markup Language - Lenguaje de marcas de hipertexto

- Es el **estándar** para la creación de páginas web
- La sintaxis es: `<tag> Contenido </tag>`
- Existen *tags* sin etiqueta de cierre: `
`
- Los *tags* pueden tener atributos:
 `<tag atr="valor"> Contenido </tag>`

Ejemplos de *Tags*

- **html:** contiene todo el documento HTML. Todas las marcas van dentro de él.
- **head:** contiene el encabezado (información sobre el documento o metadata)
- **title:** contiene el título de la página (el que se ve en las pestañas del navegador)
- **body:** contiene el cuerpo del documento. Dentro están todos los elementos que serán visualizados.
- **h1:** contiene un título.

Estructura típica de un documento HTML



```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>

<link rel="stylesheet" href="mystyle.css">
<meta charset="UTF-8">
<meta name="description" content="Free Web tutorials">
<meta name="keywords" content="HTML, CSS, XML, JavaScript">
<meta name="author" content="John Doe">
<meta http-equiv="refresh" content="30">
<meta name="viewport" content="width=device-width, initial-scale=1.0">

<style>
  body {background-color: powderblue;}
  h1 {color: red;}
  p {color: blue;}
</style>

</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Ejemplos de *Tags*



- **table:** define una tabla. Todas las otras marcas de la tabla van en su interior.
- **tr:** contiene una fila de la tabla.
- **th:** contiene el encabezado de una columna de la tabla.
- **td:** contiene el dato de una celda.

Tabla en HTML

Table Example

```
<table border="1">
  <tr>
    <td>row 1, cell 1</td>
    <td>row 1, cell 2</td>
  </tr>
  <tr>
    <td>row 2, cell 1</td>
    <td>row 2, cell 2</td>
  </tr>
</table>
```

How the HTML code above looks in a browser:

row 1, cell 1	row 1, cell 2
row 2, cell 1	row 2, cell 2

PHP



PHP Hypertext Preprocessor - Preprocesador de Hipertexto PHP

- Lenguaje de programación de propósito general, diseñado inicialmente para el desarrollo web.
- Permite incrustar etiquetas dentro de un documento HTML mediante lógica.
- Además de código PHP, Los archivos .php pueden contener HTML, CSS y javascript.

PHP



Para poder ejecutarlo de forma local, deberían descargarlo e instalarlo en su computador.

Pero esto no es necesario! Se encuentra instalado en el servidor del proyecto. Solo deben subir los archivos a la carpeta **/Sites/** y verlos desde el navegador en la ruta:

`codd.ing.puc.cl/~grupoXX/archivo.php`

Estructura de PHP



`<?php Código ?>`

El código debe ir dentro de esas marcas. Estas marcas permiten separarlo del resto del documento.

El código es procesado por el servidor y reemplazado por el texto que es impreso por el mismo código (*echo*).

Instrucciones de PHP



- **echo**: impresión de texto (muy importante para imprimir en el documento sus marcas html)
- **\$**: declaración de variables
- **array(var1, ..., varN)**: arreglo de elementos
- **//**: comentarios dentro del código
- **foreach**: para iterar sobre un arreglo
- **include**: para importar módulos (también existe require)

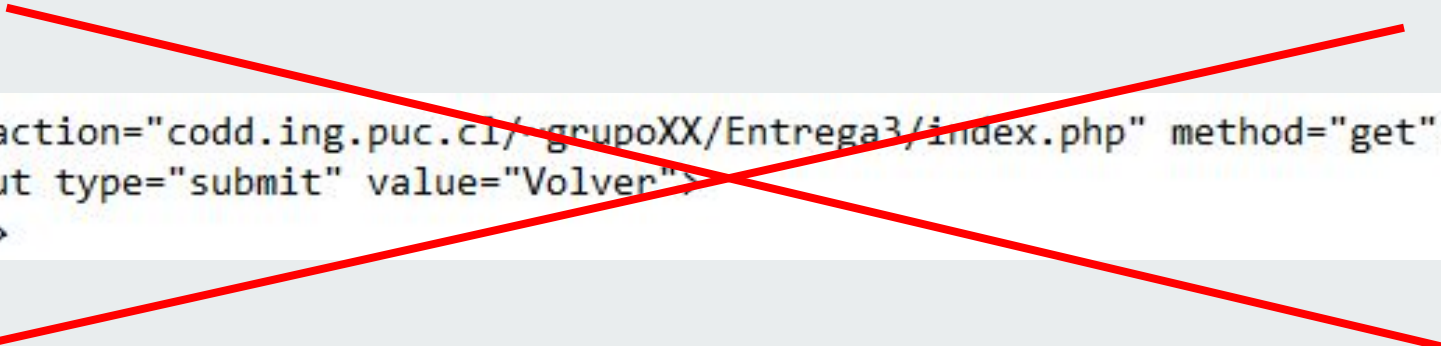

Conexión con Postgresql desde PHP



Para conectarse se debe crear una instancia de PDO y pasarle los siguientes parámetros:

- **Host:** en este caso será *localhost*
- **Port:** en este caso será *5432*
- **User:** para ustedes será *grupoXX*
- **Password:** para ustedes será *grupoXX*
- **Database name:** para ustedes será *grupoXX*

OBLIGACIÓN: Usar rutas relativas



```
<form action="codd.ing.puc.cl/grupoXX/Entrega3/index.php" method="get">  
  <input type="submit" value="Volver">  
</form>
```

```
<form action="../index.php" method="get">  
  <input type="submit" value="Volver">  
</form>
```



Ejercicio Práctico!



Crear un archivo .php que sea capaz de:

- conectarse a su BD de postgresql
- leer la tabla y tupla que crearon para la E1
- mostrar esa información en una tabla HTML

Luego, subirlo a su carpeta Sites para poder verlo desde el navegador.

Conexión con Postgresql



```
$db = new PDO("pgsql:  
    dbname=grupoXX;  
    host=localhost;  
    port=5432;  
    user=grupoXX;  
    password=MyPassWord");
```