

---

# **Desarrollo web en entorno servidor**

***Versión 1.0***

**Ricardo Pérez López**

**febrero de 2017**



<b>I</b>	<b>Introducción</b>	<b>1</b>
1.	Introducción al desarrollo web	3
2.	Protocolo HTTP y lenguaje HTML	7
3.	Sistemas de control de versiones	9
<b>II</b>	<b>PHP</b>	<b>11</b>
4.	Conceptos básicos de PHP	13
5.	Persistencia de datos con PHP	17
6.	Programación avanzada en PHP	19
7.	Interoperabilidad	21
<b>III</b>	<b>Yii2</b>	<b>23</b>
8.	Introducción a Yii2	25
9.	Conceptos fundamentales de Yii2	27
10.	Estructura de una aplicación Yii2	29
11.	Gestión de peticiones en Yii2	31
12.	Acceso a bases de datos en Yii2	33
13.	Recogida de datos y validación de formularios	35
14.	Visualización de datos en Yii2	37
15.	Características adicionales de Yii2	39
16.	Seguridad y cacheado en Yii2	41

<b>17. Pruebas, documentación y mantenimiento</b>	<b>43</b>
<b>18. Computación en la nube</b>	<b>45</b>
<b>19. Contenedores</b>	<b>47</b>
<b>IV Índices y tablas</b>	<b>49</b>

## **Parte I**

# **Introducción**



---

## Introducción al desarrollo web

---

sdfsdfklsdjflkdsf.

### 1.1 Introducción a la tecnología web

#### 1.1.1 Conceptos básicos

Navegadores y servidores web

Web estática vs. dinámica

Estructura vs. contenido

#### 1.1.2 Ejemplos de aplicaciones web

Redes sociales: Facebook, Twitter...

Comercio electrónico: Amazon, eBay...

Administración electrónica...

Portales

ERP, CRM

#### 1.1.3 Tecnologías de desarrollo de aplicaciones web

.NET

Java

**Ruby/Rails**

**Python/Django**

**PHP**

**El Kung-Fu de la programación**

**Odoo**

**PrestaShop**

**Drupal**

**WordPress**

## **1.2 Preparación del entorno de desarrollo**

### **1.2.1 Terminal**

**Zsh**

**Oh My Zsh**

### **1.2.2 Navegador**

**Herramientas de desarrollador**

**Complementos de depuración**

### **1.2.3 Editor**

**Atom**

**Instalación**

**Configuración**

**Paquetes**

**sync-settings**

**Alternativa: PhpStorm**

**Vim**



Cloud9

#### 1.2.4 DokuWiki

Elaboración de documentación

La wiki como sistema de control de versiones

La wiki como herramienta colaborativa



## CAPÍTULO 2

---

### Protocolo HTTP y lenguaje HTML

---



#### 3.1 Git básico

1. config, init, add, commit
2. status, log, diff
3. Alias lg
4. checkout, reset, revert, –amend
5. show
6. rm, mv
7. Atom y Git

#### 3.2 Git avanzado

1. Ramas: branch
2. merge, rebase
3. Resolución de conflictos

#### 3.3 Ramas remotas

1. clone, fetch, push, pull
2. Ramas de seguimiento (tracking branch)

#### 3.4 Repositorios de código (Github.com, GitLab.com, Bitbucket.org...)



## **Parte II**

## **PHP**





## **4.1 PHP como lenguaje embebido**

### **4.1.1 Introducción a PHP**

**Instalación de PHP**

**Página web de PHP**

**Documentación y búsqueda de información**

### **4.1.2 Funcionamiento del intérprete**

**Modo dual de operación**

**Etiquetas `<?php` y `?>`**

### **4.1.3 Sintaxis básica**

### **4.1.4 SAPIs**

**CLI: Uso en línea de comandos**

**Apache**

**Integración de PHP con Apache**

**Configuración básica con `php.ini`**

**Etiqueta `<?='`**

CGI: PHP-FPM (FastCGI Process Manager)

Módulos de extensión

## 4.2 Tipos de datos básicos

4.2.1 Enteros (int, sinónimo de integer)

4.2.2 Números en coma flotante (float, sinónimo de double)

4.2.3 Cadenas (string)

Funciones de manejo de cadenas

Extensión mbstring

4.2.4 Lógicos (bool, sinónimo de boolean)

4.2.5 NULL

`is_null()` vs. `=== null`

4.2.6 Comparaciones

Operadores de comparación

`==` vs. `===`

Ternario (?:)

Fusión de null (??)

Reglas de comparación de tipos

4.2.7 Comprobaciones

`gettype()`

`is_*()` (poco útiles en formularios, ya que sólo se reciben strings)

`is_numeric()`

`ctype_*()`

4.2.8 Conversiones

Coerción, moldeado, forzado o casting

`intval()`, `floatval()`, `strval()`, `boolval()`

`number_format()`, `money_format()`, `setlocale()`

## 4.3 Variables

### 4.3.1 Conceptos básicos

### 4.3.2 Asignación por valor y por referencia

### 4.3.3 Ámbito: globales, superglobales

### 4.3.4 Variables predefinidas

## 4.4 Constantes

## 4.5 Estructuras de control

### 4.5.1 `require`, `require_once`, `include`, `include_once`

## 4.6 Funciones predefinidas

### 4.6.1 `isset()`

### 4.6.2 `empty()`

Para evitar el problema de `empty("0") === true`:

```
function is_blank($value)
{
    return empty($value) && !is_numeric($value);
}
```

## 4.7 Arrays

### 4.7.1 Funciones de manejo de arrays

### 4.7.2 foreach

### 4.7.3 isset() vs. array\_key\_exists()

## 4.8 Funciones

### 4.8.1 Variables locales

### 4.8.2 Uso de global

### 4.8.3 Argumentos por defecto

### 4.8.4 Paso de argumentos por valor y por referencia

### 4.8.5 Declaraciones de tipos

Declaraciones de tipo de argumento

Declaraciones de tipo de devolución

Tipificación estricta: `declare(strict_types=1);`

## 4.9 Manejo de datos de entrada: \$\_GET y \$\_POST

## CAPÍTULO 5

---

### Persistencia de datos con PHP

---



## CAPÍTULO 6

---

### Programación avanzada en PHP

---





## CAPÍTULO 7

---

### Interoperabilidad

---



## **Parte III**

### **Yii2**



## CAPÍTULO 8

---

### Introducción a Yii2

---



## CAPÍTULO 9

---

### Conceptos fundamentales de Yii2

---





## CAPÍTULO 10

---

### Estructura de una aplicación Yii2

---



## CAPÍTULO 11

---

### Gestión de peticiones en Yii2

---



## CAPÍTULO 12

---

### Acceso a bases de datos en Yii2

---



## CAPÍTULO 13

---

### Recogida de datos y validación de formularios

---





## CAPÍTULO 14

---

### Visualización de datos en Yii2

---



## CAPÍTULO 15

---

### Características adicionales de Yii2

---



## CAPÍTULO 16

---

### Seguridad y cacheado en Yii2

---



## CAPÍTULO 17

---

### Pruebas, documentación y mantenimiento

---





## CAPÍTULO 18

---

### Computación en la nube

---



## CAPÍTULO 19

---

Contenedores

---



## **Parte IV**

# **Índices y tablas**



- genindex
- search