

PHP

Avanzado Edición 2023



Santiago Aguirre

[www](http://www.ra-ma.es)



Desde www.ra-ma.es podrá descargar material adicional.



Ra-Ma®

edü®

Ediciones de la U

PHP Avanzado

Edición 2023

Santiago Aguirre



Ra-Ma®



BOGOTÁ - MÉXICO, D.F.

Aguirre, Santiago, *et. al.*

PHP Avanzado. Edición 2023 / Santiago Aguirre --. Bogotá: Ediciones de la U, 2024

340 p. ; 24 cm

ISBN 978-958-792-647-7 e-ISBN 978-958-792-648-4

1. Encriptación 2. Interfaz 3. Plataforma I. Tít.

621,39 ed.

Edición original publicada por © Editorial Ra-ma (España)

Edición autorizada a Ediciones de la U para Colombia

Área: Sistemas e informática

Primera edición: Bogotá, Colombia, enero de 2024

ISBN. 978-958-792-647-7

- © Santiago Aguirre
- © Ra-ma Editorial. Calle Jarama, 3-A (Polígono Industrial Igarsa) 28860 Paracuellos de Jarama
www.ra-ma.es y www.ra-ma.com / E-mail: editorial @ra-ma.com
Madrid, España
- © Ediciones de la U - Carrera 27 #27-43 - Tel. (+57) 601 6455049
www.edicionesdelau.com - E-mail: editor@edicionesdelau.com
Bogotá, Colombia

Ediciones de la U es una empresa editorial que, con una visión moderna y estratégica de las tecnologías, desarrolla, promueve, distribuye y comercializa contenidos, herramientas de formación, libros técnicos y profesionales, e-books, e-learning o aprendizaje en línea, realizados por autores con amplia experiencia en las diferentes áreas profesionales e investigativas, para brindar a nuestros usuarios soluciones útiles y prácticas que contribuyan al dominio de sus campos de trabajo y a su mejor desempeño en un mundo global, cambiante y cada vez más competitivo.

Coordinación editorial: Adriana Gutiérrez M.

Carátula: Ediciones de la U

Impresión: DGP Editores SAS

Calle 63 #70D-34, Pbx (+57) 601 7217756

Impreso y hecho en Colombia

Printed and made in Colombia

No está permitida la reproducción total o parcial de este libro, ni su tratamiento informático, ni la transmisión de ninguna forma o por cualquier medio, ya sea electrónico, mecánico, por fotocopia, por registro y otros medios, sin el permiso previo y por escrito de los titulares del Copyright.

ÍNDICE

ACERCA DEL AUTOR	9
PRÓLOGO	11
SOBRE ESTA OBRA.....	13
PARTE 1. PDO. ENCRYPTACIÓN. SISTEMA DE LOGIN	15
CAPÍTULO 1. INTRODUCCIÓN	17
1.1 EL LENGUAJE	17
1.2 HERRAMIENTAS Y CONOCIMIENTOS PREVIOS	18
1.3 ¿QUÉ DEBO SABER?.....	18
1.4 PHP	18
1.5 LENGUAJE DE MARCADO HTML5	21
1.6 HOJAS DE ESTILO EN CASCADA-CSS	21
1.7 JAVASCRIPT	21
1.8 ¿DÓNDE APRENDER TODO ESTO?	22
1.8.1 HTML5 y CSS	22
1.8.2 JavaScript	24
1.9 ACTIVIDADES.....	24
1.9.1 Test de autoevaluación	24
CAPÍTULO 2. CONCEPTOS INICIALES.....	25
2.1 PHP	25
2.2 CREACIÓN DEL PROYECTO EN PHP.....	32
2.3 ACTIVIDADES.....	44
2.3.1 Test de autoevaluación	44
2.3.2 Ejercicios prácticos	44

CAPÍTULO 3. INTERFAZ	45
3.1 MEJORAR LA INTERFAZ.....	45
3.2 SESIONES EN EL SISTEMA.....	54
3.3 CREAR EL DASHBOARD.....	58
3.4 ACTIVIDADES.....	61
3.4.1 Test de autoevaluación	61
3.4.2 Ejercicios prácticos	62
CAPÍTULO 4. PERFIL	63
4.1 INFORMACIÓN DEL PERFIL	63
4.2 MODIFICAR LA NAVEGACIÓN.....	70
4.3 ACTUALIZAR LOS DATOS DEL USUARIO	73
4.4 ACTIVIDADES.....	77
4.4.1 Test de autoevaluación	77
4.4.2 Ejercicios prácticos	77
CAPÍTULO 5. CUENTAS	78
5.1 VERIFICACIÓN	78
5.2 EDITAR LOS MODELOS USER	86
5.3 ACTIVIDADES.....	94
5.3.1 Test de autoevaluación	94
5.3.2 Ejercicios prácticos	94
GLOSARIO	95
PARTE 2. PASARELA DE COMPRAS. ALGORITMO DE REDES.	
GESTIÓN DE CONTENIDOS	97
CAPÍTULO 6. PLATAFORMA DE STREAMING	99
6.1 NUEVOS MODELOS	99
6.2 FRONT-END DEL PANEL	106
6.3 MOSTRAR LOS VIDEOS	114
6.4 ACTIVIDADES.....	116
6.4.1 Test de autoevaluación	116
6.4.2 Ejercicios prácticos	116
CAPÍTULO 7. APARIENCIA	117
7.1 FRONT-END	117
7.2 AGREGAR CAMPOS DE CLASE.....	126
7.3 ELIMINAR REGISTROS	134
7.4 ACTIVIDADES.....	136
7.4.1 Test de autoevaluación	136
7.4.2 Ejercicios prácticos	136
CAPÍTULO 8. PASARELA DE PAGO	137
8.1 LIBRERÍAS DE PAGO	137
8.2 CREAR PASARELA DE PAGOS	145
8.3 CONTROLADORES PARA LOS PAGOS	149

8.4	ACTIVIDADES.....	154
8.4.1	Test de autoevaluación	155
8.4.2	Ejercicios prácticos	155
CAPÍTULO 9. DETALLES FINALES		156
9.1	ALMACENAR CLIENTES Y COMPRAS.....	156
9.2	CONSULTAS JOIN EN SQL	166
9.3	UNIR TRES TABLAS CON JOIN.....	169
9.4	ACTIVIDADES.....	176
9.4.1	Test de autoevaluación	176
9.4.2	Ejercicios prácticos	176
GLOSARIO		177
API. CONSULTAS AVANZADAS CON PDO. COMPOSER Y VERSIONES		179
CAPÍTULO 10. API.....		181
10.1	¿QUÉ ES UNA API?	181
10.2	PETICIONES GET	187
10.3	PETICIONES POST Y CREACIÓN DE DATOS	191
10.4	BORRADO DE DATOS	196
10.5	ACTIVIDADES.....	199
10.5.1	Test de autoevaluación	199
10.5.2	Ejercicios prácticos	199
CAPÍTULO 11. DASHBOARD		200
11.1	COMENTARIOS	200
11.2	LLAMAR A TU API CON JAVASCRIPT.....	204
11.3	VALIDAR LOS COMENTARIOS	208
11.4	CREACIÓN DE COMENTARIOS	213
11.5	ACTIVIDADES.....	215
11.5.1	Test de autoevaluación	215
11.5.2	Ejercicios prácticos	215
CAPÍTULO 12. MAQUETA.....		216
12.1	CREAR LA MAQUETA.....	216
12.2	CONSUMIR LA API CON JAVASCRIPT	224
12.3	UTILIZAR FETCH EN JAVASCRIPT	228
12.4	ACTIVIDADES.....	231
12.4.1	Test de autoevaluación	231
12.4.2	Ejercicios prácticos	231
CAPÍTULO 13. COMENTARIOS Y RESEÑAS		232
13.1	CREAR TU FORMULARIO AJAX	232
13.2	CREAR LAS PETICIONES AL BACK-END	239
13.3	CONTACTO DESDE EL FRONT-END	243
13.4	ACTIVIDADES.....	250
13.4.1	Test de autoevaluación	250
13.4.2	Ejercicios prácticos	250
GLOSARIO		251

PARTE 3. PHPMAILER, NEWSLETTERS, GESTIÓN DE CORREOS Y MAQUETAS HTML.....	253
CAPÍTULO 14. PHP AVANZADO.....	255
14.1 TRABAJO EN EQUIPO CON PHP.....	255
14.2 DESARROLLO DE LOS MODELOS.....	264
14.3 HERENCIA DE MÉTODOS.....	269
14.4 MAQUETAR LAS VISTAS.....	271
14.5 ACTIVIDADES.....	274
14.5.1 Test de autoevaluación.....	274
14.5.2 Ejercicios prácticos.....	274
CAPÍTULO 15. ROLES.....	275
15.1 CREAR NUEVOS MÉTODOS.....	275
15.2 GETTERS, SETTERS O ACCESORIOS.....	283
15.3 ACTIVIDADES.....	290
15.3.1 Test de autoevaluación.....	291
15.3.2 Ejercicios prácticos.....	291
CAPÍTULO 16. INTERFAZ.....	292
16.1 CREAR LA PLANTILLA WEB.....	292
16.2 DEVOLVER EL OBJETO ROLE.....	300
16.3 CARGA DE LOS DOCUMENTOS.....	306
16.4 ACTIVIDADES.....	309
16.4.1 Test de autoevaluación.....	310
16.4.2 Ejercicios prácticos.....	310
CAPÍTULO 17. CARACTERÍSTICAS ADICIONALES.....	311
17.1 AGREGAR FUNCIONALIDADES.....	311
17.2 VER TODOS LOS USUARIOS.....	320
17.3 MODIFICAR LOS DOCUMENTOS.....	323
17.4 ACTIVIDADES.....	327
17.4.1 Test de autoevaluación.....	327
17.4.2 Ejercicios prácticos.....	327
CAPÍTULO 18. NEWSLETTERS Y CORREOS ELECTRÓNICOS.....	328
18.1 PHPMAILER.....	328
18.2 EL PANEL DE ADMINISTRACIÓN.....	331
18.3 ACTIVIDADES.....	336
18.3.1 Test de autoevaluación.....	336
18.3.2 Ejercicios prácticos.....	336
GLOSARIO.....	337
MATERIAL ADICIONAL.....	339



ACERCA DEL AUTOR

Santiago Aguirre Pérez es programador y desarrollador web, además de entusiasta de la tecnología.

Trabajó durante cinco años en servicio técnico, y estudió las carreras de Comunicación Social y Desarrollo Web en la Universidad de La Matanza. Es desarrollador en Java, PHP, Python, JavaScript, y en tecnologías como Angular y Bootstrap. Actualmente se desempeña como redactor para esta editorial y como desarrollador web en empresas de consultoría tecnológica.



PRÓLOGO

PHP es, hoy en día, uno de los lenguajes más utilizados y demandados en el mundo del desarrollo, tanto en solitario, trabajando bajo patrones de desarrollo; como en conjunto con sus frameworks y librerías; junto con otras tecnologías; como así también bajo el uso de distintos CMS, entornos para la creación de sitios y sistemas, que permiten utilizar menor cantidad de código. Lleva muchos años al frente del desarrollo web, y la aparición de tecnologías nuevas y de diversas características, además de su comunidad creciente, parecen implicar que seguirá así por más tiempo.

Si deseas aprender a desarrollar en PHP de manera completa, con diversas herramientas y en distintos entornos, esta colección es para ti.

SOBRE ESTA OBRA

En esta colección estudiarás todos los aspectos avanzados del desarrollo en PHP, trabajando bajo el patrón de programación Modelo Vista Controlador y utilizando conceptos como relaciones entre tablas, programación orientada a objetos y elementos de las últimas versiones del lenguaje. Crearás un sistema base, sobre el cual irás agregando nuevas e interesantes características en cada capítulo y en cada entrega de la obra, así que no te lo pierdas.

En cada volumen, trabajarás con distintos aspectos de PHP y crearás funcionalidades nuevas, que siempre son demandadas y suelen requerir experiencia en el ámbito laboral.

Este aprendizaje te servirá para manejarte con mayor velocidad, y con conocimiento que te respalde y te permita llevar a cabo cualquier tarea fácilmente.

Aprenderás a utilizar sesiones, sistemas de login y contraseñas encriptadas; trabajarás con librerías de desarrollo en PHP como PHPMailer, la librería por excelencia del lenguaje para el envío de correos electrónicos; verás cómo utilizar pasarelas de pago y sistemas de integración de tarjetas de crédito y cobros en línea; conocerás el manejo seguro de productos y envío de newsletters; crearás sistemas modernos basados en aplicaciones reales, con usuarios y clientes; y podrás gestionar sus compras.

- ▀ **Parte 1:** Trabajarás con PHP para crear un sistema base, el cual utilizarás en los distintos volúmenes de esta colección. Para comenzar, vas a desarrollar el sistema de login, que permitirá acceder al panel de control.
- ▀ **Parte 2:** Avanzarás en la creación del front-end, generando una interfaz interesante que permita vender tus productos al cliente final, desde la cual podrán acceder a distintos pagos y obtener tu contenido.

- ▼ **Parte 3:** Utilizarás PHP para crear un sistema API que te permita aprovechar todo el potencial de AJAX, las ventajas de JavaScript asíncrono, que divide la aplicación en front-end y back-end, y crearás un sistema de calificaciones y opiniones en el sistema.
- ▼ **Parte 4:** Aprenderás a trabajar más en profundidad con PHPMailer y el envío de correos electrónicos, así como también a realizar newsletters, mejorar todo el sistema para dejarlo listo para producción y crear un sistema documental.

Parte 1

PDO. ENCRIPCIÓN. SISTEMA DE LOGIN

Introducción
Conceptos iniciales
Interfaz
Perfil
Cuentas

1

INTRODUCCIÓN

El objetivo de esta colección es mostrarte cómo trabajar con PHP de manera profesional, avanzada, y bajo entornos y patrones de desarrollo modernos, como los principios SOLID y el patrón de diseño de sistemas Modelo Vista Controlador.

1.1 EL LENGUAJE

PHP es uno de los lenguajes más conocidos en el mundo del desarrollo orientado a la Web. Creado como un lenguaje del lado del servidor para programar sitios dinámicos, su popularidad fue creciendo desde el momento en que salió al mercado.

Las distintas mejoras que obtuvo, y los diversos programas que se crearon con él y se popularizaron en el mundo entero le dieron una inmensa difusión y generaron una enorme comunidad que lo utiliza.

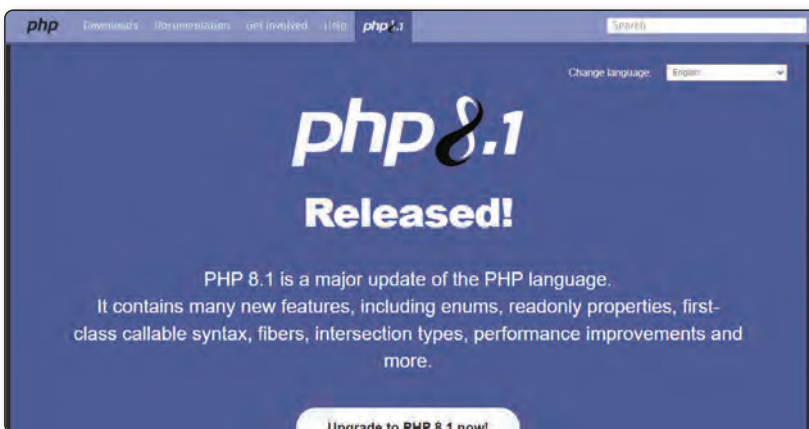


Figura 1.1. PHP es uno de los lenguajes más populares del mercado.

Además de ser una herramienta ampliamente distribuida, PHP se caracteriza por ser muy versátil, y por poseer muchas alternativas entre sus frameworks y librerías. Desde Laravel, uno de los entornos de trabajo más populares; pasando por Symphony, una opción muy interesante y utilizada en este lenguaje; hasta otros recursos como CodeIgniter, conocido como el peso pluma o framework ligero de PHP. También están los CMS, creados por lo general en PHP debido a su orientación a la Web. Se trata de programas que permiten desarrollar sitios web sobre la base de distintos contenidos, como sucede con Wordpress, Drupal o Magento, herramientas destinadas a instalar y gestionar el sitio sin necesidad de trabajar con código en todos los casos, ya que el sistema viene preparado y ready-to-work.

1.2 HERRAMIENTAS Y CONOCIMIENTOS PREVIOS

Esta colección está pensada para aquellas personas que cuentan con conocimientos básicos o intermedios en PHP, es decir, que conocen la sintaxis del lenguaje, saben crear o interpretar condicionales, bucles y variables, modificar o leer funciones, así como también escribirlas desde cero. También se requiere conocer el paradigma de Programación Orientada a Objetos, al menos en forma básica, entendiendo qué es una clase, qué son las interfaces y traits, qué es la herencia y qué son los modificadores de acceso o niveles de acceso en clases. En este capítulo introductorio, verás dónde puedes obtener este conocimiento si aún no lo tienes.

1.3 ¿QUÉ DEBO SABER?

Además de tener cierta noción sobre PHP y entender los conceptos antes mencionados, se recomienda conocer sobre la Web en general; tener una idea de cómo está creada y formada; saber qué es un servidor y un cliente; manejar los conceptos de front-end y back-end; saber qué es HTML5, CSS3 y JavaScript; y poder utilizar, al menos un poco, estos lenguajes.

1.4 PHP

Si no conoces este lenguaje y te gustaría iniciarte como desarrollador back-end o full stack, es aconsejable comenzar por el libro PHP7, de esta misma editorial, donde verás conceptos clave para introducirte en el desarrollo en general y en PHP en particular.



Figura 1.2. Obras de PHP desde cero en RedUSERS.

Para aprender PHP desde cero si no tienes conocimientos sobre este lenguaje de programación, RedUSERS cuenta con libros que te permitirán introducirte a este tema. Por otro lado, si ya conoces un poco sobre PHP y deseas profundizar tus bases con elementos avanzados, también hay disponibles colecciones con esta clase de contenido.

PHP7 es un volumen que te permitirá aprender conceptos muy importantes para cualquier desarrollador, y puedes tomarlo como puntapié inicial del camino para convertirte en uno de ellos.

En caso de que ya tengas conocimientos sobre el lenguaje, sepas cómo es su sintaxis y comprendas los elementos básicos, puedes pasar al siguiente nivel y aprender algunos aspectos más avanzados, como la Programación Orientada a Objetos.



Figura 1.3. Obras de PHP avanzado en RedUSERS.

En esta colección de tres volúmenes, verás conceptos más complejos de PHP, como creación de clases, constructores, campos de clase, métodos accesorios, herencia, abstracción, interfaces, y mucho más. Este roadmap, o camino de tres colecciones, te dará un gran conocimiento sobre el mundo del desarrollo que te permitirá crear cualquier tipo de sitio web.

Una vez que hayas leído las tres colecciones –PHP7, Programación orientada a objetos en PHP y PHP Avanzado–, puedes continuar aprendiendo un framework como Laravel, en RedUSERS Premium, donde encontrarás un libro completo introductorio, con conceptos básicos y avanzados, así como una colección completa del framework que integra otras tecnologías.



Figura 1.4. Laravel en RedUSERS.

1.5 LENGUAJE DE MARCADO HTML5

HTML es un lenguaje de etiquetas, cuya sigla en inglés proviene de HyperText Markup Language, o Lenguaje de Marcado de Hipertexto; fue creado para almacenar información dentro de sus etiquetas definidas. No se trata de un lenguaje de programación, sino de un lenguaje estructurado, cuyo objetivo es guardar información por medio de sus etiquetas, que se escriben con los signos <>. Dentro de ellos, se debe colocar el nombre de las etiquetas que se desea utilizar, y entre la de apertura y la de cierre, incluir la información por guardar.

El lenguaje es una de las bases más importantes para la creación de sitios web en la actualidad, siendo una de las herramientas principales que reemplazó al sistema de Adobe Flash Player. Su estándar es definido por el WWW Consortium, organismo que se dedica a la revisión y el mantenimiento del lenguaje, junto con otras tecnologías ligadas al desarrollo web.

1.6 HOJAS DE ESTILO EN CASCADA—CSS

El lenguaje CSS es un estándar creado para dar estilo visual a la información almacenada en etiquetas HTML. Así como las etiquetas HTML se encargan de almacenar información con su notación, no tienen como objetivo mostrar la información con distintos estilos visuales, sino solo en texto plano, con cierta semántica.

CSS, o Cascade Style Sheets (hojas de estilo en cascada), es un lenguaje creado para dar estilo a los elementos HTML, y así lograr interfaces visuales que pueden renderizarse en un navegador, de forma visualmente agradable.

1.7 JAVASCRIPT

JavaScript es el lenguaje estándar de programación de los navegadores web. Es un dialecto o derivado del estándar ECMAScript, el cual fue desarrollado en 1996, con inspiración en lenguajes como Java y C.

Este lenguaje es un estándar dentro de los navegadores porque todos poseen un motor de intérprete para él, que les permite ejecutar las instrucciones de los programas creados en JavaScript. JavaScript permite trabajar con HTML y CSS en conjunto, para crear páginas web dinámicas que pueden realizar todo tipo

de funcionalidades. Esto hace que sea uno de los lenguajes más solicitados en el mercado actual.

Además de poder utilizarse en navegadores, existen marcos de trabajo como Node que permiten su uso para la creación de servidores y programas fuera del mundo de los navegadores web, algo que será útil para la instalación de elementos de Bootstrap en esta colección.



Figura 1.5. La programación orientada a objetos es un concepto muy importante que todo desarrollador debe saber.

1.8 ¿DÓNDE APRENDER TODO ESTO?

El punto de partida en el cual te basarás para adquirir o reforzar conocimientos en uno o más de los temas mencionados es el sitio web RedUSERS Premium: <https://premium.redusers.com>. Con la experiencia y las décadas de evolución del ecosistema de la información, RedUSERS Premium cuenta con manuales, e-books y guías necesarias para potenciar tu conocimiento básico y así entender a la perfección esta obra.

1.8.1 HTML5 y CSS

RedUSERS Premium tiene varias obras orientadas al aprendizaje a fondo, y desde cero, de los lenguajes de la Web HTML5 y CSS.

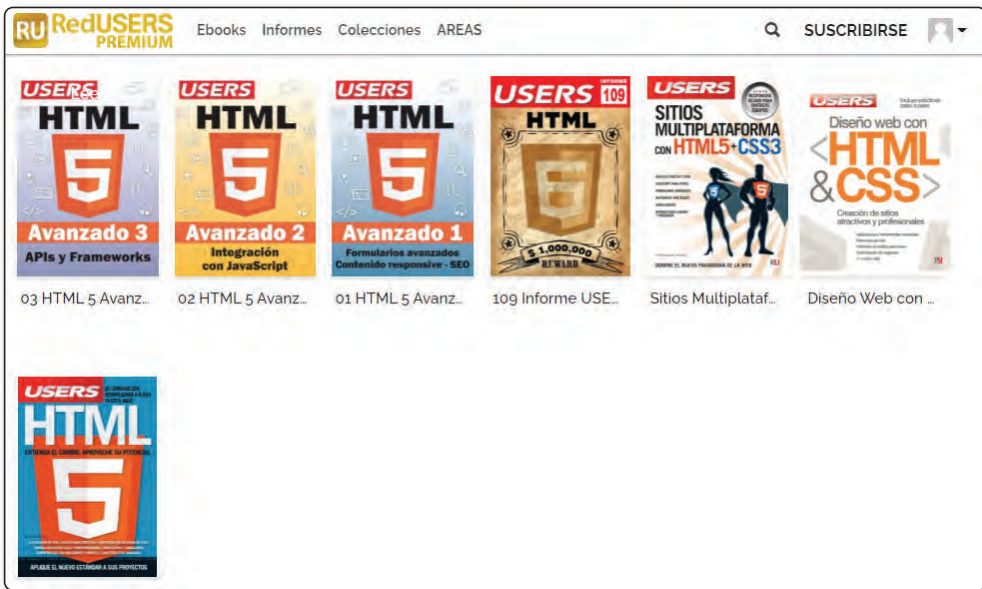


Figura 1.6. HTML5 y CSS en RedUSERS.



Figura 1.7. Sitios Multiplataforma con HTML5 y CSS3 en RedUSERS.

1.8.2 JavaScript

El lenguaje de programación de la Web es uno de los más demandados en la actualidad, y RedUSERS pone a tu disposición varias obras que te serán de utilidad para aprenderlo, tanto si tienes conocimientos como si partes desde cero.

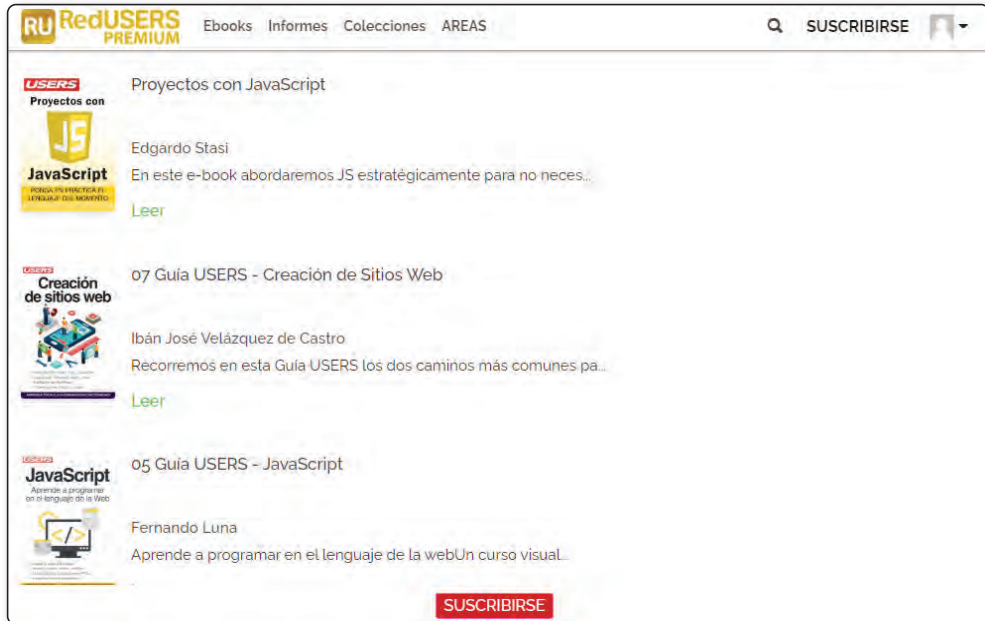


Figura 1.8. JavaScript en RedUSERS.

1.9 ACTIVIDADES

A continuación se presentan las preguntas que deberías saber responder para considerar aprendido el capítulo.

1.9.1 Test de autoevaluación

1. *¿Qué es PHP?*
2. *¿Qué es HTML y qué es CSS?*
3. *¿Qué es JavaScript y en dónde se ejecuta?*
4. *¿Qué conceptos es necesario conocer para avanzar con esta colección?*
5. *¿Qué frameworks de PHP conoces?*
6. *¿Qué se recomienda conocer antes de trabajar con un framework?*

2

CONCEPTOS INICIALES

PHP ha sido, desde sus comienzos, uno de los lenguajes de programación más demandados del mundo, y lo sigue siendo en la actualidad. Es utilizado para generar todo tipo de aplicaciones web, desde sistemas CMS hasta páginas dinámicas como Ecommerce, tiendas electrónicas, y más. Su versatilidad para crear servidores web lo ha llevado a ser el lenguaje más difundido dentro de Internet.

2.1 PHP

PHP es un lenguaje que permite crear toda clase de programas para la Web. Se lo utiliza dentro de sistemas para páginas dinámicas, permite crear APIs que conectan servicios entre sí, y su versatilidad para el uso orientado a objetos o mediante funciones lo ha llevado a ser el elegido para la mayoría de los sistemas en Internet. En entregas anteriores, puedes acceder a una introducción al lenguaje de programación de la Web, donde se te iniciará en la sintaxis de PHP, el uso de bucles, condicionales, funciones, variables, y otras características interesantes. Esto te permitirá aprender lo básico sobre el lenguaje para poder comenzar a trabajar creando tus propias páginas web. Si deseas consultar dicha obra, puedes hacerlo desde el siguiente *enlace*.



Figura 2.1. PHP es un lenguaje de código abierto, con una amplia comunidad de desarrollo.

Una vez que domines las características básicas del lenguaje, será una gran idea que aproveches la orientación a objetos que este posee. El paradigma de la programación orientada a objetos es una característica de los lenguajes modernos, como **Java**, **C#** o **JavaScript**, que permite desarrollar de manera escalable, creando código limpio, fácil de leer e interpretar, y de una forma segura. PHP posee una

fuerte orientación a este paradigma, y puedes aprender sobre clases, objetos, métodos, herencia y otras particularidades interesantes en la colección Programación Orientada a Objetos en PHP, donde verás a fondo las características más relevantes de este lenguaje y del paradigma. Accede a la colección en este [enlace](#).



Luego de aprender sobre estos temas, puedes continuar conociendo las herramientas más utilizadas del lenguaje: las librerías de bases de datos avanzadas, como PDO, el uso de servidores de correo electrónico, las librerías creadas para PHP, los gestores de paquetes de software, el tratamiento de pagos y dinero online, y otras características muy demandadas dentro de la industria del software, que se tratan en esta colección.

Esta obra se centrará en aprovechar todos tus conocimientos sobre PHP, y en explotarlos al máximo para que puedas crear sistemas web avanzados, con características como las mencionadas anteriormente. De este modo, estarás preparado para afrontar cualquier desafío laboral que implique el manejo de este lenguaje.

En este caso, para comprender cómo se trabaja con cada una de estas herramientas, utilizarás un sistema base que podrás manejar a lo largo de los próximos volúmenes de esta obra. En esta primera entrega, crearás este sistema que se encargará de generar una conexión con la base de datos y el sistema de login, para que los usuarios puedan ingresar e iniciar sesión. Para hacerlo, necesitarás instalar en tu computadora una serie de tecnologías y herramientas destinadas a desarrollar en PHP con bases de datos. En primer lugar, tendrás que instalar el lenguaje de programación PHP, junto con el servidor Apache. Luego, necesitarás instalar el motor de bases de datos MySQL, para poder ejecutar tu aplicación. Para esto, si te encuentras en Windows, la forma más sencilla de trabajar con estas tecnologías es mediante un paquete de desarrollo como XAMPP o WAMP, los cuales te permiten adquirirlas en una sola instalación.



Figura 2.2. El paquete de instalación XAMPP ofrece PHP, MySQL, Perl y Apache Server.

Una vez que hayas descargado e instalado el paquete en tu equipo, es recomendable definir en el **PATH** una variable de entorno para PHP, que te permitirá utilizar Composer y PHP en la terminal de tu sistema operativo. También podrás instalar de forma individual cada tecnología, pero esta es una tarea más compleja.

Si trabajas con un sistema operativo Mac, puedes recurrir a un paquete como MAMP, que instala las mismas tecnologías pero en los sistemas de Apple.

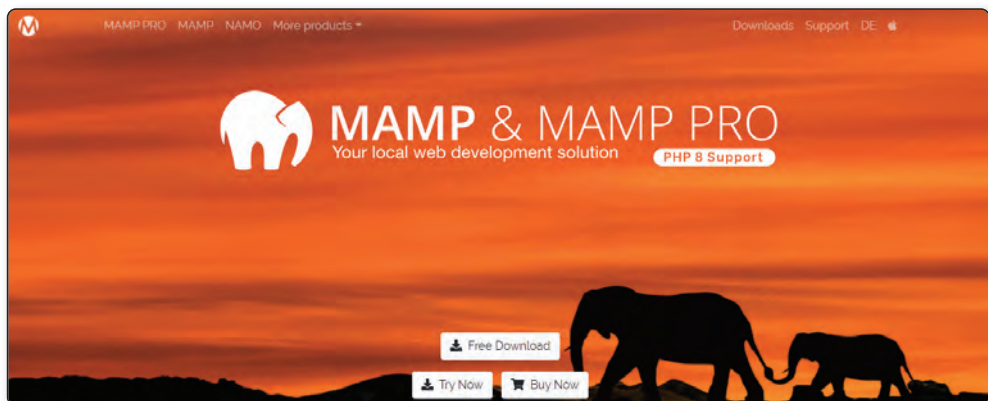


Figura 2.3. MAMP es un paquete de software para desarrollar con PHP y MySQL en MacOS.

En sistemas Linux, tendrás que actualizar la lista de **repositorios disponibles** para tu sistema, y luego instalar el lenguaje mediante el comando **sudo apt install php8.0 libapache2-mod-php8.0**. Además, deberás instalar Apache2 y MySQL, y reiniciar el servidor con el comando correspondiente. También puedes instalar el paquete de desarrollo LAMP en tu computadora, para instalar el lenguaje PHP junto con el motor de bases de datos y el servidor Apache, todo en una misma instalación. Si deseas conocer más acerca de la manera de instalar estas tecnologías, puedes leer la primera entrega de la colección Programación Orientada a Objetos en PHP, en el siguiente *enlace*.



Una vez que tengas todo instalado en tu sistema operativo, deberás instalar Composer, el gestor de paquetes de PHP, que te permite trabajar con el lenguaje, instalar librerías o incluso frameworks como Laravel o **Symfony**, de manera rápida y cómoda.

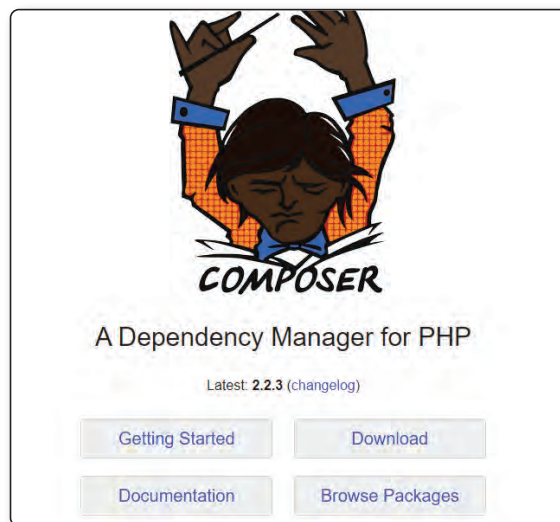


Figura 2.4. Composer es el gestor de paquetes de PHP por excelencia.

Para instalar Composer, debes dirigirte en tu navegador a <https://getcomposer.org>, donde podrás acceder a un instalador o a las instrucciones necesarias para instalar este software. Una vez que tengas el lenguaje de programación junto con el gestor de paquetes, para verificar que todo esté como corresponde, abre una terminal y ejecuta los comandos **php --version** y **composer --version**. Estos deberían indicarte qué versión de cada tecnología tienes instalada.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19043.1415]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Santiago>php --version
PHP 8.0.6 (cli) (built: May 4 2021 23:31:45) ( ZTS Visual C++ 2019 x64 )
Copyright (c) The PHP Group
Zend Engine v4.0.6, Copyright (c) Zend Technologies

C:\Users\Santiago>composer --version
Composer version 2.0.14 2021-05-21 17:03:37

C:\Users\Santiago>
```

Figura 2.5. Ambos comandos deberían devolver las versiones instaladas de cada programa.

A continuación, podrás comenzar a desarrollar y a aprender las características más avanzadas del lenguaje. Para finalizar, tendrás que utilizar un editor de código destinado a desarrollar en PHP. Una alternativa ligera e interesante es Visual Studio Code, que permite desarrollar en cualquier lenguaje, e instalar plugins para ayudarte a programar con sugerencias y corrección de errores en la sintaxis.

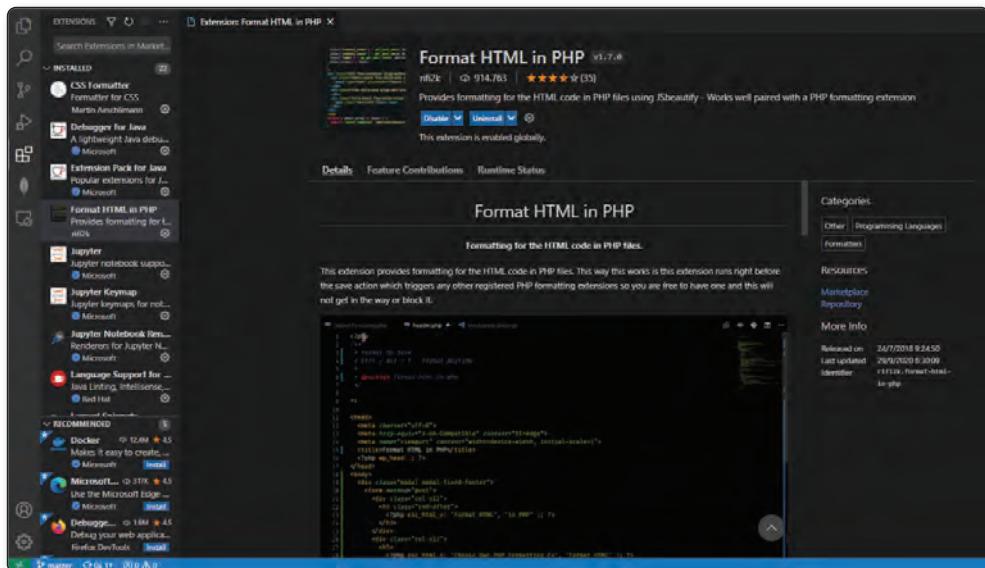


Figura 2.6. Visual Studio Code permite instalar plugins para desarrollar en PHP o en cualquier otro lenguaje.

Dentro de los entornos de desarrollo integrados, se encuentra Eclipse, un software de código abierto que posee distintas herramientas muy útiles, como aquellas destinadas a ejecución de software, integración con **PHPUnit**, trabajo con Git y repositorios remotos, entre otras.

Para finalizar, otra alternativa popular, aunque paga, es PHPStorm, un IDE de la empresa JetBrains que ofrece una gran cantidad de herramientas, como control de código, conexión a servidores remotos y trabajo con bases de datos. Si tienes una cuenta de correo universitaria como alumno o profesor, puedes obtener acceso gratuito a la versión community del software.

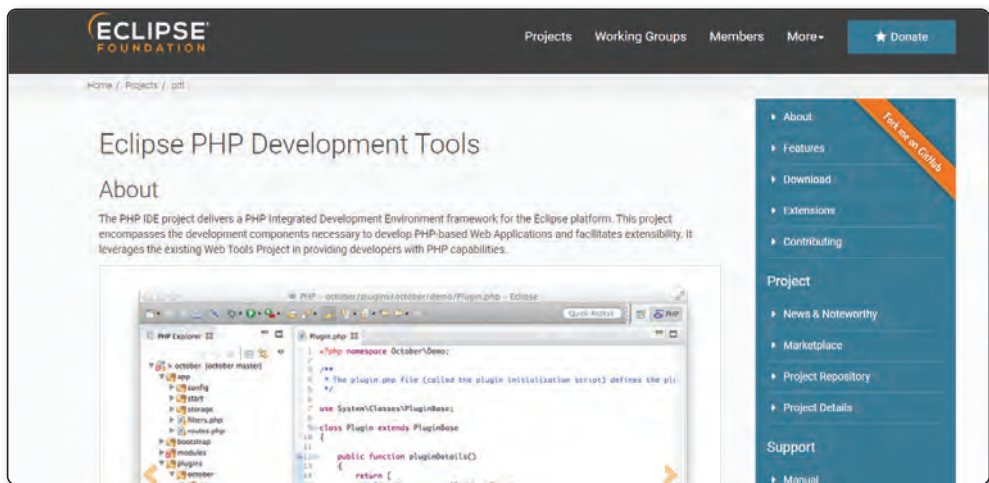


Figura 2.7. Eclipse permite desarrollar en PHP de manera cómoda.

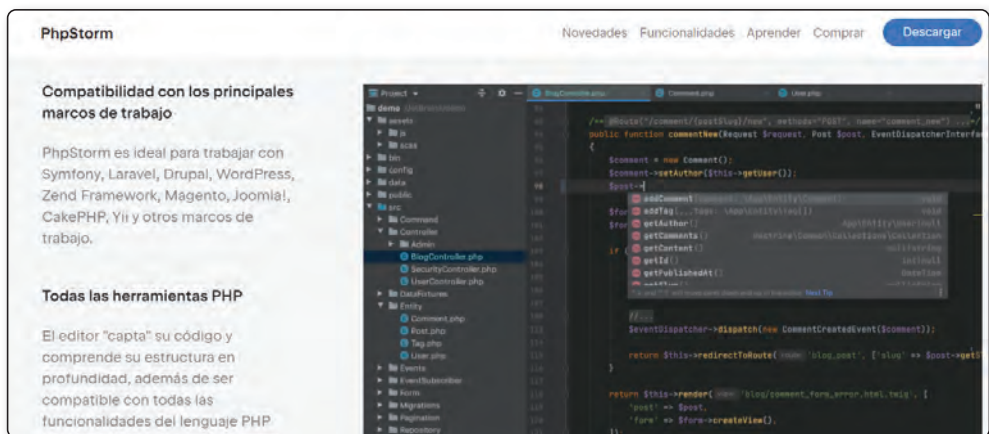


Figura 2.8. PhpStorm es uno de los IDEs más utilizados y completos dentro de la comunidad.