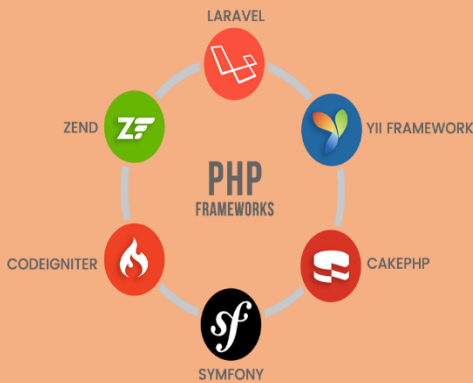


¿Qué es



Framework?

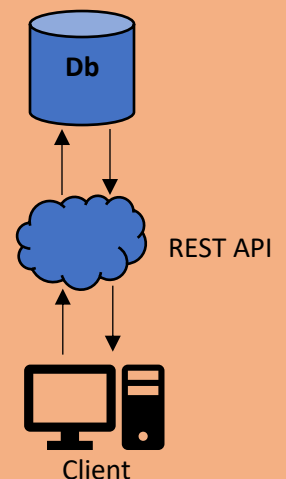
Es una especie de plantilla, un esquema conceptual, que simplifica la elaboración de una tarea, ya que solo es necesario complementarlo de acuerdo a lo que se quiere realizar, los Frameworks permiten entregar un proyecto en menos tiempo y con un código más limpio, cuya eficacia ya ha sido comprobada ya que dentro de estos existen diversas reglas que nos obligan a utilizar buenas prácticas para nuestro código con funciones ya desarrolladas, también nos ofrecen una serie de herramientas con el objetivo de ahorrarnos trabajo, mejorando también el trabajo en equipo ya que todo el sistema estará en la misma lógica y cada programador tendrá en cuenta todo el funcionamiento. (Muenste, 2020)



¿Qué es una

REST API?

Es una aplicación web en el lado del backend, en el cual se tiene una serie de rutas configuradas que hacen ciertas funcionalidades como la iteración con la base de datos, lógica etc. Dentro de estas iteraciones se realizan diferentes peticiones y estas devuelven resultados, estas pueden ser consumidas mediante el protocolo http por cualquier tipo de cliente ya sea web, dispositivo móvil etc. que comprenda el protocolo http. Normalmente un api rest devuelve los datos en xml o json, en la actualidad la mayoría devuelven los datos en json ya que es una información mejor organizada y con menos peso. (desarrolladores, 2016)



CUADRO COMPARATIVO DE ALGUNOS FRAMEWORKS DE PHP:

Frameworks	características	ventajas	desventajas
Laravel	<ul style="list-style-type: none">• Motor de plantillas ligeras e integradas.• Compatible con arquitectura MVC.• incorporación de una herramienta de línea de comandos (llamada Artisan).• ofrece sus propias bibliotecas modulares.	<ul style="list-style-type: none">• fácil de usar y aprender.• su documentación.• es compatible con casi todas las bases de datos existentes.	<ul style="list-style-type: none">• para algunas librerías y demás depende de Symfony.• Fuerte orientación a la programación estática, es decir, uso de manera abusiva de clases y métodos.
CodeIgniter	<ul style="list-style-type: none">• Versatilidad.• Compatibilidad.• Facilidad de instalación.• Documentación tutorializada.	<ul style="list-style-type: none">• Facilidad de edición del código ya creado.• Facilidad para crear nuevos módulos, páginas o funcionalidades.• Estandarización del código.	<ul style="list-style-type: none">• No cuenta con un soporte oficial para ayuda.• No se recomienda para aplicaciones complejas.
Symfony	<ul style="list-style-type: none">• Se puede instalar en cualquier sistema operativo.• instalación de manera muy sencilla.• Hay muy poco código.• Tiene plantillas que puedes personalizar.	<ul style="list-style-type: none">• Compatibilidad.• Organización del trabajo.• Cuenta con el esqueleto de una aplicación o página.	<ul style="list-style-type: none">• su rendimiento• la complejidad.

Bibliografía

desarrolladores. (23 de marzo de 2016). *BBVA API Market*. Obtenido de BBVA API Market: <https://www.bbvaapimarket.com/es/mundo-api/api-rest-que-es-y-cuales-son-sus-ventajas-en-el-desarrollo-de-proyectos/>

Muente, G. (8 de enero de 2020). *rockcontent*. Obtenido de rockcontent: <https://rockcontent.com/es/blog/framework/>