

Para que usar un framework en PHP, pros y contras, ventajas y desventajas, porqué emplearlos

En artículos previos del blog hemos dialogado un poco sobre los frameworks en Php; sin embargo creo que aun no queda del todo claro para qué usarlos, cuales son sus pros y sus contras, sus **ventajas y desventajas**, porqué emplearlos. Digamos en primera instancia que no se trata de una moda pasajera o algo así, se trata de algo útil y pertinente, lógico dentro del **desarrollo de la computación y la programación**, respectivamente.

Para nadie es un secreto que el PHP es uno de los **lenguajes más robustos** y empleados para las páginas web, tanto así que los más conocidos **CMS** trabajan sobre el mismo (por ejemplo WordPress o Joomla). Entonces, al emplear un framework, lo que hacemos es aprovechar toda esa potencia de forma dirigida y simple a la vez.

Se trata de un **Software**, un recurso informático para hacer las cosas más sencillas a la hora de programar en este lenguaje; mas rápido el respectivo desarrollo, facilita el trabajo coordinado en grupo, mantener el código ordenado y coherente, etc.

Al extraer los rasgos comunes y la arquitectura subyacente de la aplicación y agruparlos en una **plantilla reutilizable** y una **colección de bibliotecas**, un framework permite a los desarrolladores centrarse en lo que hace que su proyecto se destaque.

Ventajas



Ventajas

Desarrollo rápido: En las startups de hoy en día, basadas en la experimentación, el prototipado rápido y la obtención de su producto viable mínimo en el mundo lo más rápido posible, la velocidad de desarrollo puede ser lo que construya o destruya su negocio. El uso de un marco PHP es una forma de responder a este requisito. Un marco PHP bien diseñado selecciona las mejores implementaciones para problemas comunes de la comunidad de código abierto y, construyendo encima de ellos, le proporciona una arquitectura de aplicaciones (en la mayoría de los casos, basada en el patrón Model-View-Controller) para guiar tu desarrollo. De esta manera, en lugar de perder tiempo planificando la arquitectura de su aplicación y evaluando las diversas bibliotecas disponibles, puede comenzar desde una plantilla completamente funcional, si es que está vacía, y obtener las características específicas para su aplicación. Muchos frameworks PHP también vienen con una herramienta de línea de comandos que puede usar para generar código y ahorrar más de su valioso tiempo de desarrollo.

Un Framework hace que su aplicación sea más segura: Mientras acelera su desarrollo, un marco también lo puede hacer más seguro: usted está trabajando rápido, pero eso no sucede allanando su camino utilizando soluciones rápidas y sucias, sino mediante el uso de mejores prácticas comprobadas y componentes prefabricados. Las soluciones incluidas en un marco PHP moderno son continuamente evaluadas y validadas por múltiples desarrolladores que utilizan el marco. Como los frameworks son de código abierto, los problemas de seguridad y otros inconvenientes que puede pasar por alto en su código propietario se notan y reparan rápidamente. Dicho esto, es

importante recordar que el uso de un marco no significa que pueda olvidarse de todo sobre la seguridad y las buenas prácticas de desarrollo. Asegúrate de usar el framework como se supone que debe usarse, mantenerlo actualizado y hacer tu parte en cosas como la validación de parámetros: al usar las herramientas de frameworks, no es tan difícil.

Mantenimiento más fácil: Al proporcionar una estructura de programación intuitiva para su aplicación y guiarlo hacia las mejores prácticas y patrones de desarrollo de software, un marco PHP lo guía hacia la escritura de código que se acople débilmente y se repita lo menos posible. En otras palabras, es fácil de leer, probar, mantener y desarrollar más. Prácticamente todos los marcos de PHP populares de hoy en día están orientados a objetos y desarrollados con pruebas automatizadas e incluso desarrollo basado en pruebas en mente. Aún tendrá que hacer su parte. Pero en comparación con comenzar de cero, un marco y su documentación que le recuerden los conceptos y lo empujen hacia su uso seguramente lo ayudarán. Otra forma en la que usar un marco hace que mantener su aplicación sea más fácil es a través de sus componentes compartidos. Como las bibliotecas y los asistentes de Framework se desarrollan independientemente de su aplicación, puede descargar una gran parte del mantenimiento de su aplicación a los desarrolladores de framework. En cierto modo, es como tener un equipo de desarrolladores trabajando para nosotros... y de manera gratuita.

Trabajo en equipo más fuerte: Como desarrollador, usted conoce sus clases y funciones de memoria y puede usarlas fácilmente. Pero a medida que traes a los miembros del equipo, la situación cambia. Sin el conocimiento de sus decisiones de diseño y la documentación completa (seamos sinceros: especialmente en el mundo del desarrollo rápido, la documentación es a menudo una idea de último momento), les resultará difícil entender el código de su aplicación. El uso de un marco PHP ayudará a los desarrolladores nuevos a

comenzar a trabajar en el proyecto. Lo más probable es que ya estén familiarizados con el marco y puedan poner toda su atención en comprender los detalles de la aplicación. E incluso si el marco es nuevo para ellos, la documentación, los tutoriales en línea y los patrones de diseño familiares les ayudarán a ponerse al día con un mínimo esfuerzo. Todo esto mientras puedes seguir trabajando en la aplicación en lugar de enseñar a los desarrolladores tus formas de pensar.

La comunidad está ahí para ti: Finalmente, un beneficio significativo de usar un marco PHP proviene de las comunidades activas de desarrolladores y usuarios que se reúnen a su alrededor. Todos los marcos importantes atraen a contribuyentes de código abierto que, además de trabajar en el marco en sí, desarrollan widgets adicionales, bibliotecas y componentes de ayuda para que otros los incluyan en sus aplicaciones. Otros colaboradores escriben tutoriales y libros o crean videos para explicar los pormenores de cómo usan los marcos en proyectos de la vida real. Como parte de dicha comunidad de desarrolladores, crecerás como desarrollador, aprenderás rápidamente y aplicarás lo que aprendas en el desarrollo de tu aplicación. Y si está tan inclinado, puede devolver y participar en proyectos de código abierto de cualquier manera que se sienta natural y satisfactoria para usted.

Desventajas



Desventajas

No todo puede ser dicha en este tema de los marcos o frameworks.

Ejecución lenta: claro, si se compara con los scripts de PHP;

un framework PHP siempre agrega complejidad y sobrecarga en forma de clases y bibliotecas cargadas antes de que su código sea llamado. La diferencia puede ser significativa en scripts y API rápidos con un conjunto limitado de funciones que deben ejecutarse rápidamente. Sin embargo, en la mayoría de las aplicaciones web, las implicaciones de eficiencia de un marco tienen más que ver con la forma en que lo usa que con el marco mismo.

Soluciones generales para problemas específicos: La idea de un framework PHP es ser una solución general para los requisitos recurrentes que la mayoría de los desarrolladores enfrentarán al desarrollar una aplicación web. Pero, ¿dónde deberíamos trazar la línea sobre qué incluir en el marco y qué dejar para las aplicaciones (o complementos de terceros) a la hora implementar? Según algunos desarrolladores, aquí es donde muchos marcos funcionan mal.

Un marco requiere tiempo para aprender y dominar: Si bien un marco de PHP te empuja hacia un buen código, no es garantía de ello. Si trabajas con un framework sin entenderlo y los patrones de diseño detrás de él, lo más probable es que termines con un lío grande y estructurado. Para alguien que ya está familiarizado con estos principios de diseño, desde otros marcos o proyectos, la curva de aprendizaje será relativamente corta. Para un desarrollador con menos experiencia, el aprendizaje tomará más tiempo. Sin mencionar el dominio del marco.

Visibilidad y control limitados: El uso de un marco de software mueve el desarrollo de su aplicación a un nivel más alto de abstracción. Esto es por diseño, y generalmente es algo bueno. Sin embargo, es bueno darse cuenta de que también puede provocar problemas. Si bien muchas partes de la plantilla del marco están destinadas a ser personalizadas, el núcleo del framework y sus bibliotecas son como un cuadro negro que usted usa pero que no tiene mucho control sobre él. Puede sugerir mejoras pero, en última instancia, depende de los

desarrolladores del marco decidir si incluirán ese cambio o no.

Un Framework presenta una Dependencia Externa: Un marco es siempre una dependencia externa para su código de aplicación controlado por alguien que no sea usted y su equipo. Si bien esto generalmente no es un problema, es algo que vale la pena considerar si el software que está creando es una parte fundamental de su negocio . Pregúntese qué hará si el desarrollo del marco se cae o se mueve en una dirección que no desea seguir.

Bueno, espero que este breve resumen de las bondades de los frameworks en Php sirva para entender un poco mejor sus verdaderos alcances.

Leer también: [6 frameworks PHP para el desarrollo ágil de aplicaciones](#); [5 frameworks para trabajar con angular](#)