

FRAMEWORK

LARAVEL | CODEIGNITER | SYMFONY

WORK

1. ¿QUÉ ES FRAMEWORK?



Un framework, o marco de trabajo, es una plantilla para desarrollar software de manera más rápida y eficiente. En lugar de escribir todo el código desde cero, un framework proporciona una estructura predefinida y herramientas que puedes usar como bloques de construcción.

COMPONENTES COMUNES DE UN FRAMEWORK INCLUYEN:

BIBLIOTECAS Y MÓDULOS:

Estos se encargan de funciones específicas, como manipulación de datos, gestión de sesiones, seguridad y autenticación.

PATRONES DE DISEÑO RECOMENDADOS:

Ayudan a estructurar el código y crear una arquitectura coherente en la aplicación.

HERRAMIENTAS DE DESARROLLO:

Permiten generar y depurar el código, realizar pruebas unitarias y manejar bases de datos.

ESTÁNDARES DE CODIFICACIÓN:

Aseguran que los archivos y directorios se organicen de la misma manera, manteniendo el código consistente y legible.

Un framework es como una herramienta poderosa que te ayuda a construir aplicaciones de manera más eficiente, sin tener que reinventar la rueda en cada proyecto.

2. ¿QUÉ ES UNA REST API?

Transferencia de Estado Representacional (REST): Fue definida por el científico informático Dr. Roy Fielding en su disertación doctoral en el año 2000. REST proporciona un alto nivel de flexibilidad, escalabilidad y eficiencia para los desarrolladores. Por esta razón, las API REST se han convertido en un método común para conectar componentes y aplicaciones en arquitecturas de microservicios.



Principios de diseño

Uniformidad:

Todas las solicitudes de la API para el mismo recurso deben tener el mismo aspecto, independientemente de su origen. Cada dato, como el nombre o la dirección de correo electrónico de un usuario, debe pertenecer a un único identificador uniforme de recursos (URI).

Independencia entre Cliente y Servidor:

Las aplicaciones de cliente y servidor deben ser completamente independientes entre sí. El cliente solo necesita conocer el URI del recurso solicitado, sin interactuar de ninguna otra manera con el servidor.

Sin Estado (Stateless):

Cada solicitud debe incluir toda la información necesaria para procesarla. No se requiere ninguna sesión del lado del servidor, y las aplicaciones de servidor no pueden almacenar datos relacionados con una solicitud de cliente.

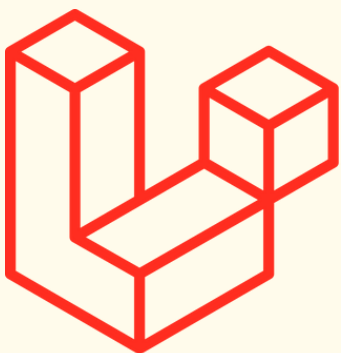
Almacenamiento en Caché:

Los recursos deben almacenarse en caché en el lado del cliente o del servidor para mejorar el rendimiento y la escalabilidad.

Comunicación a Través de Capas:

Las llamadas y respuestas pasan por diferentes capas, y no se debe asumir que las aplicaciones de cliente y servidor se conectan directamente entre sí.

Una API REST es una herramienta poderosa que permite a los desarrolladores integrar aplicaciones y conectar componentes de manera flexible y ligera, siguiendo los principios de REST



3.

LARAVEL

Laravel es un popular framework de código abierto para el desarrollo de aplicaciones web en PHP. Fue creado por Taylor Otwell en 2011 y desde entonces ha ganado una gran popularidad en la comunidad de desarrollo web debido a su elegante sintaxis, su potente conjunto de características y su comunidad activa.

VENTAJAS	DESVENTAJAS
Productividad mejorada: Su sintaxis limpia y clara facilita la escritura de código rápido y de calidad.	Curva de aprendizaje: Simplifica muchas tareas de desarrollo, puede llevar tiempo familiarizarse con su sintaxis.
Bibliotecas y módulos:	
Arquitectura MVC: Proporciona una estructura organizada para el desarrollo de aplicaciones web.	Desempeño: Algunos argumentan que su rendimiento puede ser inferior a opciones más ligeras y optimizadas.
Abundantes bibliotecas y paquetes: Cuenta con una gran cantidad de bibliotecas y paquetes listos para usar.	Dependencia de terceros: Al utilizar numerosas bibliotecas y paquetes de terceros, se crea una dependencia en dichos componentes
Comunidad activa: Cuenta con una gran cantidad de bibliotecas y paquetes listos para usar.	N/D

4.

CODEIGNITER

CodeIgniter es otro framework de desarrollo de aplicaciones web en PHP, aunque es un poco más antiguo que Laravel. Es conocido por su simplicidad y facilidad de uso, así como por su velocidad de ejecución. Aunque no es tan extenso como Laravel en términos de funcionalidades integradas, es una opción popular para proyectos más simples o cuando se necesita un rendimiento muy rápido.



VENTAJAS	DESVENTAJAS
Rendimiento rápido: El núcleo de CodeIgniter es ligero, lo que permite que las páginas se procesen rápidamente.	Curva de aprendizaje: Aunque es más sencillo que otros frameworks, puede llevar tiempo acostumbrarse.
Fácil instalación: Se instala fácilmente subiendo los archivos al servidor y configurando el acceso a la base de datos.	Dependencia de versiones antiguas de PHP: Funciona mejor con versiones antiguas de PHP, lo que puede generar problemas de seguridad.
Edición de código sencilla: Facilita la edición del código existente.	N/D
Separación de la lógica y la arquitectura web (MVC): Utiliza el patrón Modelo-Vista-Controlador para una estructura clara.	N/D



3. SYMFONY

Symfony es otro framework de desarrollo web en PHP que se centra en la modularidad, la flexibilidad y el rendimiento. Es conocido por su estabilidad, su gran comunidad y su enfoque en las mejores prácticas de desarrollo de software. Symfony se utiliza ampliamente en proyectos de todos los tamaños, desde pequeñas aplicaciones hasta grandes sistemas empresariales.

VENTAJAS	DESVENTAJAS
Flexibilidad en la creación de proyectos: Los desarrolladores pueden elegir su propio ORM (Mapeo Objeto-Relacional).	Curva de aprendizaje: Aunque es más sencillo que otros frameworks, puede llevar tiempo acostumbrarse a su sintaxis y características.
Automatización de tareas: Symfony 4 se integra perfectamente con Symfony Flex, que automatiza las tareas más comunes en aplicaciones.	Dependencia de versiones antiguas de PHP: Symfony 4 funciona mejor con versiones antiguas de PHP, lo que puede generar problemas de seguridad.
Estabilidad comprobada: El framework Symfony 4 está construido con la última versión de los Componentes Symfony, las bibliotecas más populares en la historia de PHP.	N/D

FRAGMENTO DE CODÍGO DE CADA FRAMEWORK

LARAVEL

```
app/Http/Controllers/ChirpController.php

<?php
class ChirpController extends Controller
{
    public function store(Request $request)
    {
        $validated = $request->validate([
            'message' => 'required|string|max:255',
        ]);

        $request->user()->chirps()->create($validated);

        return redirect(route('chirps.index'));
    }
}
```

```
var app = new Framework7({
  root: '#app',
  routes: [
    {
      path: '/app',
      async : function(routeTo, routeFrom, resolve, reject){
        app.request.post('/dataserver/server_ajax',{}, function(data){
          console.log('apphelp_v2');
          var data = JSON.parse(data);
          console.log(data);
          resolve(
            {
              componentUrl: '/app/',
            },
            {
              context : {
                feed: data.feed_curation
              }
            }
          )
        });
      },
    },
  ],
});
```

CODEGNITER

SYNFONY

```
~/Sites/symfony-security » symfony console make:user

The name of the security user class (e.g. User) [User]:
> User

Do you want to store user data in the database (via Doctrine)? (yes/no) [yes]:
> yes

Enter a property name that will be the unique "display" name for the user (e.g. email, username,
uuid) [email]:
> email

Will this app need to hash/check user passwords? Choose No if passwords are not needed or will b
e checked/hashed by some other system (e.g. a single sign-on server).

Does this app need to hash/check user passwords? (yes/no) [yes]:
> 
```

BIBLIOGRAFIA

<https://ebac.mx/blog/frameworks>

<https://www.iebschool.com/blog/framework-que-es-agile-scrum/>

<https://www.arimetrics.com/glosario-digital/framework>

<https://unirfp.unir.net/revista/ingenieria-y-tecnologia/framework/>

<https://www.ibm.com/es-es/topics/rest-apis>

<https://www.redhat.com/es/topics/api/what-is-a-rest-api>

<https://oxygenacademy.es/los-5-frameworks-php-mas-solicitados-en-2023/>

<https://einavirtual.com/php/framework-lista>

<https://github.com/laravel/framework>

<https://laravel.com/docs/10.x/installation>

<https://openwebinars.net/blog/que-es-laravel-caracteristicas-y-ventajas/>

<https://kinsta.com/es/base-de-conocimiento/que-es-laravel/>

<https://ventajasydesventajasstop.com/laravel-ventajas-y-desventajas/>

<https://carlos-herrera.com/laravel-vs-codeigniter/>

https://codeigniter.com/user_guide/intro/index.html

<https://aprendeinformaticas.com/guia-codeigniter/>

<https://codeigniter.com/download>

<https://www.desarrollolibre.net/blog/laravel/laravel-vs-codeigniter>

<https://openwebinars.net/blog/que-es-symfony/>

https://programacion.net/articulo/pros_y_contras_de_los_frameworks_php_mas_populares_1122

<https://linube.com/blog/symfony-4-0-novedades/>