



INSTITUTO TECNOLÓGICO DEL VALLE DE OAXACA



CARRERA: ING. INFORMATICA

SEMESTRE: 9º

GRUPO: A

ASIGNATURA: FRAMEWORKS PARA DESARROLLO WEB





ALUMNO: JUAN PABLO AMBROSIO CRUZ





PROFESOR: CARDOSO JIMENEZ AMBROSIO




UNIDAD I: INTRODUCCIÓN





30 DE AGOSTO DE 2022

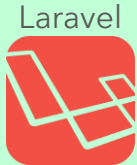


CUADRO COMPARATIVO DE
FRAMEWORKS PARA DESARROLLO WEB

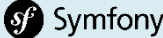



Nombre	Curva de aprendizaje	Lenguaje	Backend	Frontend	Características	Ventajas	Desventajas	Ultima Versión	Licencia
 Reat.JS (Librería)	Media	JavaScript	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<ul style="list-style-type: none"> Modelo de objetos de documento virtual (DOM). Ayuda a desarrollar sitios web y aplicaciones dinámicas rápidamente. Compuesto por múltiples componentes que se pueden reutilizar. Rendimiento multiplataforma. 	<ul style="list-style-type: none"> Muy simple de aprender y usar Viene con un conjunto de herramientas muy prácticas Ayuda con la mejora del rendimiento Tamaño más pequeño Capacidad para probar y depurar rápidamente. 	<ul style="list-style-type: none"> Es un desafío para un principiante obtener todas las funciones de React a medida que se introducen nuevas funciones a diario. 	16.8.6 27	MIT
 Vue.JS	Fácil	JavaScript TypeScript	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<ul style="list-style-type: none"> Opción de vista DOM virtual rentable. Viene con un componente incorporado que ayuda con las transiciones y animaciones CSS. Opciones de enlace de datos bidireccionales. 	<ul style="list-style-type: none"> Fácil de integrar. Ofrece comunicación bidireccional. Fácil de entender y aprender. 	<ul style="list-style-type: none"> No es adecuado para un escalado extenso. Demasiado flexible para el bien. 	3.2.1	MIT
 Angular	Alta	JavaScript TypeScript	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<ul style="list-style-type: none"> Capacidad para ofrecer páginas web progresivas y de alto rendimiento Puede funcionar en varios sistemas operativos Ofrece una API intuitiva y avanzada Capacidad para obtener una finalización de código inteligente Ofrece opciones de detección instantánea de errores y comentarios. 	<ul style="list-style-type: none"> Sincronización perfecta entre el modelo y la vista Los cambios se muestran instantáneamente en la vista Capacidad para separar datos y verlos. 	<ul style="list-style-type: none"> Angular proporciona muchas formas de hacer lo único que puede llevar a confusión. Es más complejo para aprender. 	12.1.4	MIT
 Angular.JS	Media	JavaScript	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<ul style="list-style-type: none"> Se basa en el famoso concepto de MVC (Modelo-View-Controller). No es necesario escribir código especial para enlazar datos a los controles HTML. Esto puede hacerlo Angular simplemente agregando algunos fragmentos de código. 	<ul style="list-style-type: none"> Angular.js mantiene sincronizada la capa de datos y presentación. Admite pruebas, pruebas unitarias y pruebas de integración. 	<ul style="list-style-type: none"> Es grande y complicado debido a las múltiples formas de hacer lo mismo. Las implementaciones no siempre escalan bien. 	1.8.2	MIT

 Express Express.JS	Fácil	JavaScript	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<ul style="list-style-type: none"> ▪ Soporta JavaScriptOffers de solicitudes de enrutamiento de cliente simples. ▪ Funciona como middleware. 	<ul style="list-style-type: none"> ▪ Ayuda a reducir a la mitad el tiempo de creación de aplicaciones web ▪ Puede ser utilizado incluso por principiantes ▪ Muy fácil de aprender e implementar ▪ Peso ligero. 	<ul style="list-style-type: none"> ▪ No se pueden manejar tareas computacionales pesadas ▪ Tiene una API inestable ▪ Crea la necesidad de varios cambios de código. 	4.17.1	MIT
 Node.JS (Entorno)	Media	JavaScript	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<ul style="list-style-type: none"> ▪ Muy rápido de integrar y usar. ▪ Capacidad para escalar ampliamente. ▪ Ejecución de subprocesos asíncronos y sin bloqueo. ▪ Puede manejar grandes volúmenes de conexiones simultáneas. 	<ul style="list-style-type: none"> ▪ Se puede implementar rápidamente. ▪ Muy fácil de aprender. ▪ Un rico ecosistema con múltiples herramientas gratuitas. ▪ Mejora la productividad de los desarrolladores. ▪ Mejora el crecimiento de la conversión. 	<ul style="list-style-type: none"> ▪ Dificultad para procesar tareas basadas en CPU. ▪ La necesidad constante de devolución de llamada afecta a la calidad de los códigos. 	LTS 14.17.6 16.9.1	MIT
 METEOR Meteor.JS	Media	JavaScript	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<ul style="list-style-type: none"> ▪ Latencia: Tiempo de respuesta corto. ▪ Rendimiento: Contiene paquetes listos para hacer que aceleren el proceso de desarrollo. ▪ Idioma: Mismas API en el servidor y el cliente. Utiliza un lenguaje para todos, es decir, JavaScript. 	<ul style="list-style-type: none"> ▪ Proporciona desarrollo de pila completa para aplicaciones web. ▪ Solo se utiliza un lenguaje para todo el desarrollo, front-end, back-end y base de datos. ▪ Los paquetes y bibliotecas están actualizados. 	<ul style="list-style-type: none"> ▪ La única opción de back-end para Meteor es MongoDB, pero contiene problemas de integridad de datos. ▪ Proporciona una integración compleja con otros marcos. ▪ Con una conexión a Internet deficiente, no habrá sincronización entre el front-end y el back-end. 	2.3.6	MIT
 Svelte.JS	Alta	JavaScript	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<ul style="list-style-type: none"> ▪ Sin DOM virtual. ▪ Esto convierte los códigos en JavaScript Vanilla. ▪ Muy reactivo. 	<ul style="list-style-type: none"> ▪ Ligero y fácil de implementar. ▪ Repetición menor ▪ Muy fácil de aprender. ▪ Viene con una tienda reactiva incorporada. 	<ul style="list-style-type: none"> ▪ Necesidad constante de aprender y actualizar la información. ▪ No hay muchas herramientas Svelt en la industria en este momento. ▪ Ecosistema más pequeño. 	3.38.2	MIT

 <p>Next.JS</p>	Fácil	JavaScript TypeScript	☒	☒	<ul style="list-style-type: none"> Enrutamiento automático. Representación del servidor. Reacciona bien con otros ecosistemas, incluidos React y Angular. Precarga automáticamente las páginas. 	<ul style="list-style-type: none"> JS adaptable y receptivo. Ayuda a cumplir los objetivos de ventas y tráfico. Ofrece una experiencia de usuario única. Tiene soporte CSS incorporado. 	<ul style="list-style-type: none"> Se vuelve costoso de personalizar. La conversión de una aplicación del lado del servidor a Next.js es una tarea bastante difícil. 	11.1.0	MIT
 <p>Ember.JS</p>	Alta	JavaScript TypeScript	☐	☒	<ul style="list-style-type: none"> Puede ser utilizado fácilmente por equipos grandes. Viene con propiedades calculadas. Tiene más de 1500 complementos que solo aumentan con el tiempo. Reutilizable y fácil de mantener también. 	<ul style="list-style-type: none"> Compatibilidad perfecta con URL. Ofrece servicios de prestación de clientes. Capa de datos bien integrada. Minimiza el DOM. 	<ul style="list-style-type: none"> No es fácil para los desarrolladores aprenderlo. No es adecuado para proyectos grandes; apoya mejor los proyectos pequeños y medianos. No admite mucha personalización en el proyecto. 	4.0.0	MIT
 <p>Backbone.JS</p>	Alta	JavaScript TypeScript	☐	☒	<ul style="list-style-type: none"> Los eventos proporcionan la capacidad a los objetos backbone para enlazar y crear eventos personalizados. Los modelos contienen las partes significativas de la lógica, y Views tiene una razón para presentar los datos del modelo. Las colecciones se utilizan para definir varias instancias a la vez. Los enrutadores ayudan a conectar el lado del cliente a la acción y los eventos de la aplicación. 	<ul style="list-style-type: none"> Ayuda a explorar las técnicas de código JavaScript. Tiene una biblioteca de utilidades de subrayado. El código se organiza sistemáticamente y se arma después. Backbone no requiere almacenar datos en DOM. 	<ul style="list-style-type: none"> Requiere tiempo para que el desarrollador aprenda a codificar en Backbone. La curva de aprendizaje es alta para Backbone. 	1.4.0	MIT
<p>Polymer.JS</p>	Fácil	JavaScript TypeScript	☐	☒	<ul style="list-style-type: none"> Tiene esquemas de nomenclatura definidos por el usuario. Uno puede nombrar componentes en HTML con cualquier nombre que desee. Cada elemento tiene su plantilla en lo que respecta a los estilos y el comportamiento. 	<ul style="list-style-type: none"> Polymer es un marco de JavaScript muy fácil de usar. Polymer JS admite el enlace de datos en ambos sentidos. La compatibilidad entre navegadores es una característica muy necesaria. En Polymer, el front-end se desarrolla mediante un 	<ul style="list-style-type: none"> Para una acción compleja, el polímero puede ser lento ya que funciona en el navegador. No hay representación del lado del servidor. 	3.0	MIT

						<ul style="list-style-type: none"> enfoque basado en características. Tiene una biblioteca en crecimiento, por lo que ayuda en el desarrollo rápido y fácil de aplicaciones móviles personalizadas. 	<ul style="list-style-type: none"> Polymer JS no tiene buena documentación. 		
Nest.JS 	Alta	JavaScript TypeScript	☒	☐	<ul style="list-style-type: none"> Estructura modular, cuyo uso simplifica la división del proyecto en bloques separados. Simplifica las pruebas al admitir características como contenedores y módulos DI. Permite crear soluciones de software extensibles donde no hay un acoplamiento fuerte entre los componentes. Se puede emplear Nest JS en todas las plataformas. 	<ul style="list-style-type: none"> Con Nest no hay limitaciones, puedes diseñar una aplicación web simple o una idea posible compleja que una aplicación pueda hacer. Nest Framework admite middleware, filtros de excepción, tuberías, protectores, interceptores, GraphQL, Websockets y muchos otros componentes. 	<ul style="list-style-type: none"> Requiere tiempo para que el desarrollador aprenda a codificar en Nest. 	8.0.6	MIT
Ruby On Rails 	Media	Ruby	☒	☒	<ul style="list-style-type: none"> Basado en la arquitectura MVC (modelo-vista-controlador) RoR enfatiza los principios DRY (don't repeat yourself, o "no se repita") y CoC (convención sobre configuración) La gran comunidad de Rails trabaja de manera proactiva para parchear nuevas vulnerabilidades Modificar un código existente es sencillo Biblioteca potente y robusta, llamada Active Record 	<ul style="list-style-type: none"> Preferido para la creación de prototipos Procesos sencillos de automatización de pruebas Desarrollo rápido de aplicaciones Marco MVC y acceso variado a bibliotecas y herramientas 	<ul style="list-style-type: none"> Ausencia de citas apropiadas Formas lentas de tiempo de arranque No se prefiere para aplicaciones enormes 	7.0.1	MIT
Gin 	Media	Go	☒	☐	<ul style="list-style-type: none"> Permite crear un servidor HTTP sin esfuerzo. Encabeza la lista en términos de popularidad debido a su marco minimalista y rendimiento. Marco de Golang más adecuado para 	<ul style="list-style-type: none"> Útil para crear API rest Amplia documentación Bien probado y numerosos middlewares Alto rendimiento 	<ul style="list-style-type: none"> Es posible que no sea adecuado para aplicaciones backend grandes 	1.7.7	MIT

					desarrollar API REST de alto rendimiento.				
	Media	php	☒	☐	<ul style="list-style-type: none"> Plantillas ligeras integradas y múltiples widgets que incorporan código JS y CSS Soporte de arquitectura MVC Fuerte seguridad de las aplicaciones web ORM elocuente (mapeo relacional de objetos) Herramientas de comando integradas Artisan, que se encargan de automatizar las tareas de programación tediosas y repetitivas Bibliotecas modulares y orientadas a objetos preinstaladas 	<ul style="list-style-type: none"> Herramientas prediseñadas Motor de plantillas-Blade Enrutamiento Automatización con Artisan CLI Pruebas automatizadas 	<ul style="list-style-type: none"> Congestión de backend porque Laravel es liviano Bibliotecas de terceros poco fiables El vasto tamaño de la comunidad es un inconveniente aquí porque se vuelve difícil encontrar una biblioteca confiable. 	8	MIT
	Media	php	☐	☒	<ul style="list-style-type: none"> Trabaja con MVC (Model View Controller) Cuenta con módulos para manejar archivos PDF, canales RSS, Web Services (Amazon, Flickr, Yahoo), etc Incluye objetos de las diferentes bases de datos, por lo que es extremadamente simple realizar consultas a la base de datos, sin tener que escribir ninguna consulta SQL. Una solución para el acceso a base de datos que balancea el ORM con eficiencia y simplicidad. 	<ul style="list-style-type: none"> Zend es el framework PHP más utilizado por las grandes empresas. Asociado con Google, IBM, Adobe, Microsoft Zend tiene una gran biblioteca de clases predefinidas con las que los desarrolladores pueden crear aplicaciones web mantenibles y estables. 	<ul style="list-style-type: none"> Dado que Zend tiene una arquitectura poco acoplada, significa que la documentación también será bastante suelta. 	7.3	BSD
	Media	php	☐	☒	<ul style="list-style-type: none"> Licencia flexible Compatible con las versiones de PHP 5.2.6 y superiores. Contiene CRUD para la interacción de la base de datos. Andamiaje de código. 	<ul style="list-style-type: none"> Cake tiene un ORM incorporado que es bastante fácil de usar y aprender. El desarrollo en CakePHP es muy rápido y flexible. 	<ul style="list-style-type: none"> Debido al código heredado, los conceptos antiguos y la priorización de la velocidad de desarrollo sobre todo lo demás, CakePHP es pesado y lento. 	3.10	MIT

					<ul style="list-style-type: none"> ▪ Generación automática de código. ▪ Arquitectura MVC ▪ URLs personalizadas ▪ Función de Validación. 				
	Alta	php	☒	☐	<ul style="list-style-type: none"> ▪ La base estándar sobre la que se construyen las mejores aplicaciones PHP. ▪ Utiliza el motor de plantillas Twig. Twig es rápido, seguro y flexible. 	<ul style="list-style-type: none"> ▪ La comunidad más grande Actual ▪ Las plataformas populares que usan Symfony, como Drupal, Magento, eZ Publish, recomiendan Symfony. 	<ul style="list-style-type: none"> ▪ Algunas aplicaciones que usan Symfony funcionan lentamente porque dependen de otras tecnologías. ▪ El tiempo de desarrollo es lento porque el desarrollo reutilizable es para múltiples usos. 	6.0.2	MIT
	Media	Python	☒	☐	<ul style="list-style-type: none"> ▪ Ofrece funciones de depurador y servidores de desarrollo ▪ Cumplimiento de WSGI ▪ Backend basado en Unicode ▪ Compatible con Google App Engine ▪ Utiliza plantillas Jinja2 ▪ Ayuda integrada para pruebas unitarias ▪ Extensiones para una mejor funcionalidad ▪ Se pueden crear cookies seguras 	<ul style="list-style-type: none"> ▪ Escalable ▪ Flexible ▪ Seguridad 	<ul style="list-style-type: none"> ▪ No hay muchas herramientas ▪ Costes de mantenimiento 	2.0.2	BSD
	Alta	Python	☒	☒	<ul style="list-style-type: none"> ▪ Django incluyen sintaxis simple, arquitectura central MVC, ORM (Object Relational Mapper), soporte de Middleware y bibliotecas HTTP. ▪ Django cuenta con su propio sistema de fabricación para todas las herramientas y características. 	<ul style="list-style-type: none"> ▪ Capacidades Inteligentes De Desarrollo Aplicaciones Y Escalabilidad Más ▪ Desarrollo Rápido ▪ Flexibilidad ▪ Seguridad 	<ul style="list-style-type: none"> ▪ Falta De Convenciones ▪ No Apto Para Proyectos Más Pequeños ▪ Difícil De Aprender 	4.0.1	BSD
	Media	JAVA	☒	☐	<ul style="list-style-type: none"> ▪ Tecnologías: como la inyección de dependencias, eventos, recursos, i18n, validación, enlace de datos, 	<ul style="list-style-type: none"> ▪ Los desarrolladores no tienen que usar un servidor o cualquier otro contenedor empresarial. 	<ul style="list-style-type: none"> ▪ El marco de Spring tiene muchas variables y complicaciones. ▪ Los desarrolladores tienen que saber 	5.3.8	Apache

					<div>conversión de tipo, SpEL.</div> <ul style="list-style-type: none">▪ Acceso a datos: soporte DAO, JDBC, ORM, Marshalling XML.▪ Gestión de transacciones.▪ Programación orientada a aspectos (AOP): permite la implementación de rutinas transversales.▪ MVC (Modelo Vista Controlador).	<ul style="list-style-type: none">▪ Los desarrolladores tienen la opción de elegir anotaciones basadas en XML o Java para fines de configuración.▪ La inyección de dependencias de Spring ayuda a los desarrolladores a insertar datos de prueba.	<div>qué características serán útiles, y tomar las decisiones equivocadas podría llevar a retrasos significativos.</div> <ul style="list-style-type: none">▪ Dentro de la documentación de Spring, no dice nada sobre cómo lidiar con amenazas como XSS o la falsificación de solicitudes entre sitios.		
--	--	--	--	--	--	--	---	--	--

REFERENCIAS BIBLIOGRÁFICAS

Frameworks Javascript para el 2021. (2021). Recuperado el 20 de septiembre de 2021, de <https://medium.com/zurvin/frameworks-javascript-para-el-2021-28e8f8ad181a>

Anwar, M. (2021). 13 BEST JAVASCRIPT FRAMEWORKS IN 2021 - Code Leaks. Recuperado el 20 de septiembre de 2021, de <https://www.codeleaks.io/13-best-javascript-frameworks-in-2021/#Cons-9>

Ermigiotti, L. (2021). 17 JavaScript Frameworks that You Should Know in 2021 - A Comprehensive Guide - Codemotion Magazine. Recuperado el 20 de septiembre de 2021, de <https://www.codemotion.com/magazine/dev-hub/javascript/javascript-frameworks-guide/#5-ember-js>

Documentación oficial

Polymer library - Polymer Project. (2021). Recuperado el 20 de septiembre de 2021, de <https://polymer-library.polymer-project.org/3.0/docs/devguide/feature-overview>

@nestjs/core. (2021). Recuperado el 20 de septiembre de 2021, de <https://www.npmjs.com/package/@nestjs/core>

Releases - Ember.js. (2021). Recuperado el 20 de septiembre de 2021, de <https://emberjs.com/releases/>

Using TypeScript With Ember Effectively. (2021). Recuperado el 20 de septiembre de 2021, de <https://docs.ember-cli-typescript.com/ts/using-ts-effectively>

Angular Vs AngularJs: Difference Between Angular & AngularJS | upGrad blog. (2021). Recuperado el 20 de septiembre de 2021, de <https://www.upgrad.com/blog/angular-vs-angularjs/>

The Good and the Bad of Angular Development. (2021). Recuperado el 20 de septiembre de 2021, de <https://www.altexsoft.com/blog/engineering/the-good-and-the-bad-of-angular-development/>

¿Qué es AngularJS? Arquitectura y características - Guru99. (2021). Recuperado el 20 de septiembre de 2021, de <https://guru99.es/angularjs-introduction/>

NestJS: qué es y por qué empezar a usarlo» GFourmis. (2021). Recuperado el 20 de septiembre de 2021, de <https://gfourmis.co/nestjs-que-es-y-por-que-empezar-a-usarlo/>