



Introducción a los Stacks Tecnológicos

TECNOLOGÍAS WEB I

Carlos Rojas Sánchez

Licenciatura en Informática

Universidad del Mar

1. Stacks Tecnológicos
2. Tipos de Aplicaciones Web

Stacks Tecnológicos

¿Qué es un Stack Tecnológico?

- Conjunto de tecnologías usadas para desarrollar y ejecutar una aplicación.
- Incluye:
 - Lenguajes de programación
 - Frameworks
 - Bases de datos
 - Servidores
- Hay stacks especializados para diferentes tipos de proyectos.

LAMP = Linux, Apache, MySQL, PHP

- Muy usado en WordPress y apps PHP
- Arquitectura clásica de servidor web

MEAN = MongoDB, Express, Angular, Node.js

- 100% JavaScript (frontend y backend)
- Ideal para aplicaciones SPA

MERN = MongoDB, Express, React, Node.js

- Alternativa moderna a MEAN
- React ofrece más flexibilidad y comunidad activa

MEVN = MongoDB, Express, Vue, Node.js

- Vue.js es simple y accesible para nuevos desarrolladores
- Muy popular en Latinoamérica y Asia

Python + Django

- Framework robusto, seguro y rápido
- Compatible con PostgreSQL, MySQL, SQLite
- Servidor: Gunicorn/Nginx

JavaScript, APIs y Markup

- Sitios estáticos + funciones backend como servicio
- Frameworks: Next.js, Nuxt, Astro
- Hosting: Netlify, Vercel

Java + Spring Boot

- Para aplicaciones empresariales
- Base de datos relacional (MySQL, PostgreSQL)
- Servidor embebido (Tomcat)

¿Cuál elegir?

Proyecto	Stack recomendado
Blog/CMS	LAMP
App JS moderna	MERN / MEVN
App Python	Django Stack
Sitio estático rápido	JAMstack
App empresarial compleja	Spring Boot

Conclusión

- Hay un stack para cada necesidad.
- La elección depende del tipo de proyecto y equipo.
- Es útil conocer varios stacks para elegir el más adecuado.

Tipos de Aplicaciones Web

MPA (Multi Page Application)

Aplicaciones de múltiples páginas. Cada interacción del usuario (como hacer clic en un enlace) provoca la carga completa de una nueva página desde el servidor.

- Ejemplos: sitios de noticias, e-commerce tradicionales, Wikipedia.
- Tecnologías típicas: HTML, CSS, JS, junto con backend en PHP, Python (Django), Ruby (Rails), Java, etc.

SSR (Server-Side Rendering)

El contenido de la página se renderiza en el servidor y se envía completamente al cliente.

- Ejemplos: aplicaciones que priorizan el SEO y el tiempo de carga inicial.
- Frameworks: Next.js (React), Nuxt.js (Vue), Ruby on Rails, Django con templates.

SSG (Static Site Generation)

Las páginas se generan de forma estática en tiempo de construcción (build) y luego se sirven como archivos HTML planos.

- Ejemplos: blogs, documentaciones, portafolios.
- Herramientas: Gatsby, Next.js (modo estático), Hugo, Jekyll.

PWA (Progressive Web Application)

Aplicaciones web que usan tecnologías modernas para comportarse como apps nativas: pueden funcionar sin conexión, enviar notificaciones, instalarse en el escritorio/móvil, etc.

- Ejemplos: Twitter Lite, Starbucks PWA, Spotify Web.
- Tecnologías: Service Workers, Web App Manifest.

Aunque no son aplicaciones web puras, vale la pena mencionarlas por su relación:

- Híbridas: usan web technologies dentro de un contenedor nativo (ej: Ionic, Cordova).
- Nativas: escritas específicamente para Android (Kotlin/Java) o iOS (Swift/Obj-C).
- Multiplataforma: frameworks como Flutter o React Native.

SPA (Single Page Application)

Una Aplicación de Página Única es un tipo de aplicación web que carga una sola página HTML y actualiza dinámicamente su contenido sin recargar toda la página desde el servidor. Esto proporciona una experiencia más fluida y rápida para el usuario, similar a la de una aplicación de escritorio.

SPA (Single Page Application)

- Al cargarse, se descarga una página HTML inicial junto con los scripts de JavaScript necesarios.
- Las interacciones del usuario (como hacer clic en un enlace o enviar un formulario) no provocan recargas completas de página.
- En su lugar, el contenido se actualiza dinámicamente utilizando AJAX o tecnologías similares (como fetch o Axios).
- Las rutas suelen ser manejadas del lado del cliente mediante librerías de enrutamiento.