



**TECNOLÓGICO
NACIONAL DE MÉXICO**

Instituto Tecnológico Nacional de México Campus Ensenada

Trabajo de Investigación

Materia: Programación Web

Maestro: Roberto Luna Martinez

Tema: Evolución y Tecnologías de las Aplicaciones Web

Alumna: Ariam Anette Zurita Torres

Grupo: 5SS

Lugar y fecha: Ensenada, B.C. 4 de Septiembre del 2025

INTRODUCCIÓN

Las aplicaciones web han cambiado la manera en la que las personas se comunican, trabajan, estudian y realizan actividades cotidianas. Desde sus inicios, en los años 90 's, empezaron como solo páginas estáticas hasta la actualidad, que son plataformas interactivas en las que la mayoría se basan en inteligencia artificial. Su progreso ha estado ligado al desarrollo del internet y a la necesidad de poder contar con herramientas accesibles en cualquier parte del mundo.

Una ventaja de las aplicaciones web es que no debemos instalarlas, ya que se pueden ejecutar desde cualquier navegador, como Chrome, Edge, Firefox, entre otros; sólo necesitamos conexión a internet. Le permite a empresas, gobiernos, instituciones educativas y usuarios en general utilizar las plataformas para diferentes actividades, como, por ejemplo, comercio electrónico, redes sociales, entretenimiento, trabajo colaborativo, servicios de mensajería y educación a distancia. Un ejemplo es que, en mi caso, no soy de educación a distancia, pero podemos hacer entrega de tareas a través de nuestra plataforma Moodle, que nos facilita a los estudiantes que estamos en modalidad para trabajadores el poder trabajar y estudiar al mismo tiempo. Es una gran herramienta para nosotros.

Mi trabajo se centra en tres aspectos fundamentales: la evolución histórica de las aplicaciones web, la arquitectura que permite su funcionamiento y las tecnologías más relevantes utilizadas en su desarrollo. A través de este análisis, busco comprender cómo estas herramientas no solo han modificado la interacción con la información, sino que también han impactado en la economía, la educación y la sociedad en general.

1.1 Evolución de las aplicaciones WEB

La evolución en las aplicaciones web ha sido marcada por distintas generaciones que reflejan El avance tecnológico y la necesidad de los usuarios

Etapas iniciales web 1.0 (1990 a 2000)

- Las primeras páginas web eran estáticas, ya que fueron diseñadas con un HTML básico.
- Los usuarios sólo podían consultar información ya que no existía la integración ni la retroalimentación.
- Las estructuras en ese tiempo eran simples solo texto imágenes y enlaces.
- Las utilizan principalmente universidades y el gobierno para difundir información.
- Ejemplos de ellos son las páginas como GeoCities o los primeros directorios como Yahoo

Web 2.0 (2000 a 2010)

- En esta surge la interactividad gracias al lenguaje JavaScript, PHP AJAX y CSS.
- Los usuarios se convierten en prosumidores o sea que consumen y generan contenido.
- Empezaron las redes sociales como los wikis, plataformas de video y los blogs.
- Las características principales eran que tenía dinamismo, participación social y personalización.
- Ejemplo de ellos eran Facebook Wikipedia YouTube blogs entre otros

Web 3.0 (2010 a la actualidad)

- Es conocida como la web semántica
- Muchas se basan en inteligencia artificial, Big data y análisis predictivos.
- La información se conecta a una base de datos lo que permite mayor personalización de experiencia de los usuarios.
- Las aplicaciones funcionan de forma multiplataforma que podemos usar en PC, laptops, celulares y tablets.

- Ejemplos pueden ser Netflix, que recomienda contenido personalizado ya que depende de qué tipo de series o películas veas, si es de comedia, terror, drama etcétera. Amazon ofrece productos basados en compras que hemos hecho.

1.2 ARQUITECTURA DE LAS APLICACIONES WEB

La arquitectura de una aplicación web se podría definir en cómo se organizan los componentes para lograr seguridad, escalabilidad, rendimiento y eficiencia. Como por ejemplos son:

Arquitectura Cliente-Servidor

- Es la más básica de todas las arquitecturas
- El cliente (navegador) envía solicitudes a un servidor que responde con la información solicitada. Ejemplos de ello es cuando un usuario llena un formulario de Google o un formulario para una beca, procesa los datos y manda la información de dicho usuario.

Arquitectura en Capas

Divide la aplicación en tres partes:

1. **Capa de presentación:** Interfaz gráfica (HTML, CSS, JS).
2. **Capa de lógica de negocio:** Son las reglas que definen cómo funciona el sistema (por ejemplo, calcular precios, validar usuarios, validar un método de pago, calcular el envío de un paquete).
3. **Capa de datos:** base de datos donde se guarda la información (MySQL, MongoDB).

Ventaja: Facilita el mantenimiento y la seguridad.

Arquitectura de Microservicios

- Cada módulo de aplicación funciona como servicio independiente
- Un ejemplo puede ser Amazon que es un microservicio gestionado por pagos, otro por pedidos y otro por los envíos.
- La ventaja de ella es su independencia, escalabilidad y la facilidad de las actualizaciones
- La desventaja es mayor complejidad en la comunicación entre servicios, un ejemplo puede ser que haces un pago de unos audífonos que compraste y a veces tarda en llegar al pago por lo tanto no te pueden enviar nada si no se refleja el pago

Arquitectura basada en la nube

- Los servidores físicos se sustituyen por servicios en la nube como (AWS, Google cloud, Azure, Onedrive)

- Una ventaja es que tenemos acceso global, son de bajos costos, tenemos un respaldo automático y es muy seguro guardar nuestra información, tanto puede ser de tu trabajo o escuela
- Un ejemplo pues son las aplicaciones como Dropbox, Google drive o Onedrive

1.3 TECNOLOGÍAS PARA EL DESARROLLO DE APLICACIONES WEB

Requieren múltiples tecnologías, las cuales están agrupadas en distintas categorías aquí nuestro cuáles son:

1.- Lenguajes de programación

Frontend (interfaz de usuario)

- **HTML5:** Es una estructura básica de las páginas web, como por ejemplo formularios en línea o etiquetas multimedia
- **CSS3:** Son estilos de visuales, como los colores, animaciones y diseños responsivos, un ejemplo es como un diseño adaptable en páginas como Wikipedia o Google
- **JavaScript:** Nos permite interactuar ya que hay botones dinámicos, menús desplegables, validación de formularios. Ejemplos más comunes son Gmail o Trello ya que tienen efectos en tiempo real

Backend (servidor):

- **PHP:** Ya que es ampliamente usado en páginas dinámicas y administradores de contenido. Ejemplo: Wordpress.
- **Python:** Es muy popular en las aplicaciones de datos y aprendizaje automático. Ejemplos: Instagram y Spotify.
- **Java:** Es muy usado en aplicaciones bancarias y empresariales Ejemplo: Sistemas de banca en línea de cualquier banco, HSBC, BBVA, etcétera.
- **Node.js:** Se ejecuta con servidores como JavaScript que son ideales para el chat en tiempo real como por ejemplo WhatsApp web.
- **Ruby (Con Ruby on Rails):** Es fácil de usar y es muy ágil para startups Ejemplo: Airbnb o Github

2.- Frameworks y librerías:

Frontend:

- **React:** Basado en componentes reutilizables, ofrece un excelente rendimiento , solo las partes necesarias de la interfaz. Desarrollado por Facebook es usado en Instagram, Netflix y Uber.
- **Angular:** Un framework completo con una estructura robusta y herramientas integradas para construir aplicaciones complejas y escalables. Creado por Google, es ideal para grandes proyectos. Un ejemplo de su uso son Gmail y Google docs.
- **Vue.js:** Se caracteriza por su ligereza, flexibilidad y una curva de aprendizaje suave. Es usado en Alibaba y Xiaomi.

Backend:

- **Django (Python):** Es una base sólida, rápida y segura Ejemplo: es Instagram.
- **Laravel (PHP):** Es muy popular por su sencillo sistema, ejemplo de ello son los sistemas de e-commerce personalizados.
- **Spring boot (Java):** Usado normalmente en aplicaciones bancarias y corporativas. Un ejemplo son plataformas de seguros o bancas en línea.
- **Express.js (Node.js)** Es ligero y muy rápido, utilizado en APIs modernas. Uno de sus ejemplos es Uber.

3. Bases de datos

- **Relacionales (SQL):**
- **MySQL:** Es una base de datos de código abierto muy popular y utilizada en aplicaciones web. Muy usado en WordPress y Wikipedia.
- **PostgreSQL:** Son de código abierto y conocida por ser potente, extensible y cumplir con los estándares SQL. Un ejemplo es la app Spotify
- **Oracle Database:** Ampliamente usada en grandes empresas financieras, ya que tienen una gran base de datos

No relacionales (no sql):

- **MongoDB:** Tiene un tipo de base de datos de documentos, ya que es muy flexible y usado en link y Ebay.
- **Firebase:** Es la base de datos principal de Google, diseñado en aplicaciones móviles y chats
- **Cassandra:** Ideal para cantidades grandes de datos de forma distribuida, como por ejemplo Netflix o redes sociales

4. Protocolos y servicios web

- **HTTP/HTTPS:** Mantiene una comunicación entre el cliente y el servidor. Un ejemplo es cualquier sitio que tenga el “https://”
- **REST :** Un tipo de arquitectura que define principios para la creación de APIs, para la comunicación entre las aplicaciones. Ejemplo: API de Twitter o Google Maps.
- **GraphQL:** Es un lenguaje de tipo consulta para APIs que ofrece flexibilidad y eficiencia. Es muy utilizado en Facebook y Shopify
- **WebSockets:** Tiene un protocolo que permite comunicación bidireccional (full-duplex) entre cliente y un servidor a través de una única conexión TCP

5. Herramientas de despliegue y gestión:

Contenedores:

- **Docker:** Es una plataforma de software que empaqueta aplicaciones y sus dependencias (código, bibliotecas, etc) en unidades estandarizadas. Garantiza que funciona de manera consistente en cualquier entorno
- **Kubernetes:** Es un contenedor de código abierto, creado por Google, que nos ayuda a automatizar la implementación, el escalado y la administración de aplicaciones contenerizadas a gran escala. Ejemplo: Google Cloud.

Plataformas en la nube:

- **Amazon Web Services (AWS):** Es un conjunto de servicios de computación en la nube que ofrece infraestructura bajo demanda, con servicios como Amazon ECS
- **Microsoft Azure:** La plataforma de la nube de Microsoft, que ofrece servicios de orquestación de contenedores como Azure
- **Google Cloud:** Servicios de Google para computación en la nube, que incluye Google Kubernetes para ejecutar y administrar aplicaciones en los contenedores

Control de versiones y colaboración:

- **GitHub:** Es una plataforma basada en la web que ofrece un sistema para control de versiones para el código fuente, lo que permite a los equipos colaborar en proyectos y realizar un seguimiento en los cambios del código.
- **GitLab y Bitbucket:** Alternativas a GitHub que proporcionan funcionalidades similares para la administración de repositorios de código abierto.

Conclusión

El análisis de la evolución de la arquitectura tecnológica de las aplicaciones web nos permite ver cómo estas herramientas pasaron de ser simples páginas estáticas a convertirse en plataformas muy complejas que forman parte esencial de la vida moderna.

La evolución es evidencia de un proceso de transformación constante e impulsando por la necesidad de mayor interactividad, personalización y accesibilidad la transformación de la web 1.0 a la web 3.0 muestra cómo las aplicaciones dejaron de ser tan sencillas para hacerse más dinámicas colaborativas y capaces de aprender sobre los usuarios

En cuanto a la arquitectura es un cambio enorme hacia modelos más flexibles y escalables, destacamos la opción de los microservicios y servicios en la nube esté en arquitecturas modernas nos garantizan una mayor seguridad, mayor disponibilidad y mayor capacidad de crecimiento que es muy indispensable en un entorno donde somos millones de usuarios que interactúan día con día de forma simultánea.

Las tecnologías actuales nos permitieron que el desarrollo sea más eficiente y accesible utilizando frameworks, librerías, lenguajes y bases de datos adaptadas a necesidades diferentes gracias a eso las aplicaciones han podido responder a las demandas de diferentes sectores como la educación el comercio o el entretenimiento

En resumen las apps webs representan un pilar muy importante en nuestra era y sabemos que va a seguir evolucionando con la incorporación de la inteligencia artificial y realidad aumentada. Estas innovaciones nos están garantizando experiencias increíbles e inmersivas, seguras y que se adapten a cada usuario para que la experiencia sea mejor y personalizada con eso las aplicaciones web son y serán las herramientas tecnológicas más influyentes en el sector

Referencias Bibliográficas

1. García Gutiérrez, F., & Ruiz Corbella, M. (2020). *Educación y tecnología digital*. Ciencia Latina. Recuperado el 4 de Septiembre del 2025 de <https://biblioteca.ciencialatina.org/wp-content/uploads/2024/04/Educacion-y-tecnologia-digital.pdf> Biblioteca Ciencia Latina
2. Megías Jiménez, D., & Mateu, C. (2002). *Desarrollo de aplicaciones web*. FUOC-UAB. Disponible en Recuperado el 4 de Septiembre del 2025 <https://libros.metabiblioteca.org/bitstreams/a37985ce-f55b-49a6-9ac4-bff7d082cdbf/download> MetaBiblioteca
3. RCIAssociats. (2024). *El desarrollo de aplicaciones con tecnologías web: un enfoque integral*. Recuperado el 4 de Septiembre del 2025 de <https://rcassociats.com/el-desenvolupament-daplicacions-amb-tecnologies-web-un-enfocament-integral/> RC Associats
4. Alfaomega México. (s. f.). *Aplicaciones Web – un Enfoque Práctico*. Recuperado el 4 de Septiembre del 2025 de <https://alfaomega.com.mx/producto/aplicaciones-web-un-enfoque-practico>
5. Press (Editorial). (2023). *Desarrollo de aplicaciones web con tecnologías modernas*. Recuperado el 4 de Septiembre del 2025 de <https://press.religacion.com/index.php/press/catalog/book/108> Religación Press
6. Wikipedia. (2025, agosto). *Brecha digital*. En *Wikipedia, la enciclopedia libre*. Recuperado el 4 de Septiembre del 2025 de https://es.wikipedia.org/wiki/Brecha_digital
7. Moreno, A. (2019). *Desarrollo de aplicaciones web con Python y Django*. Marcombo. Recuperado el 4 de Septiembre del 2025 de: <https://www.marcombo.com/libro/desarrollo-de-aplicaciones-web-con-python-y-django/>

