



Taller de Programación Web

Arquitectura Web





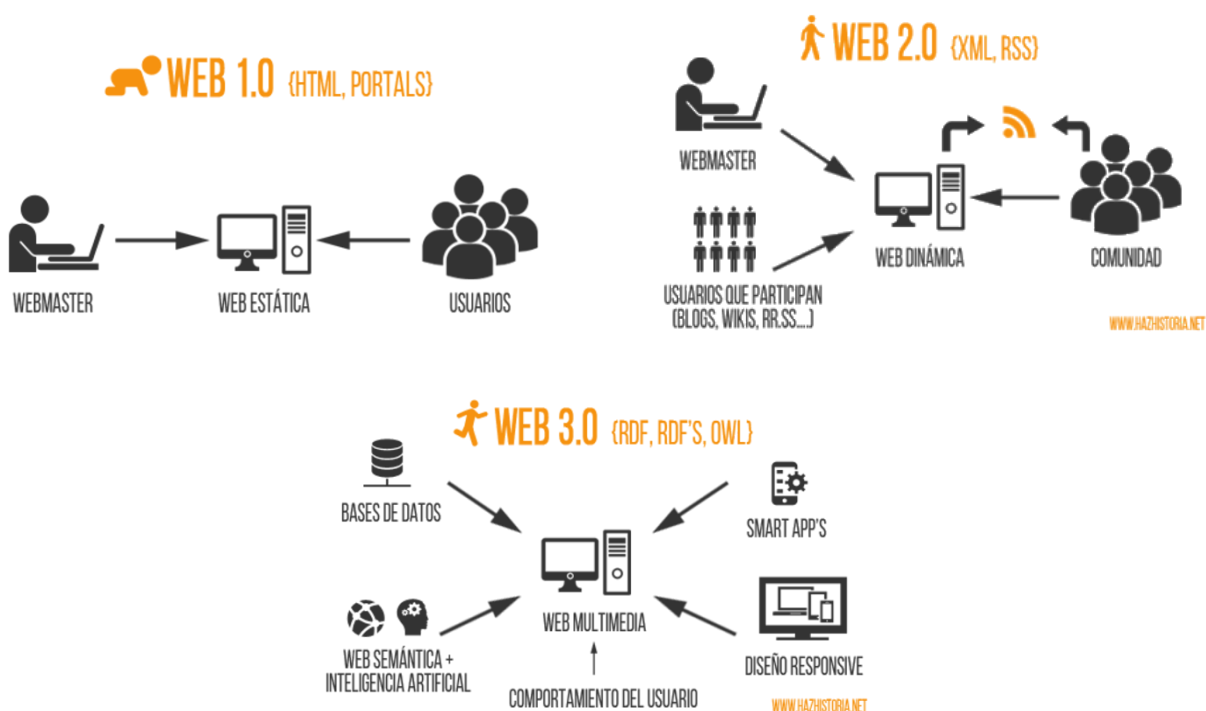
TEMA 3: Desarrollo Web

La Arquitectura Web



La **arquitectura web** se puede definir como la estructura conceptual de internet. Los tipos de arquitectura web incluyen el **Modelo Cliente-Servidor** y el **Modelo de tres niveles** que son los más utilizados actualmente.

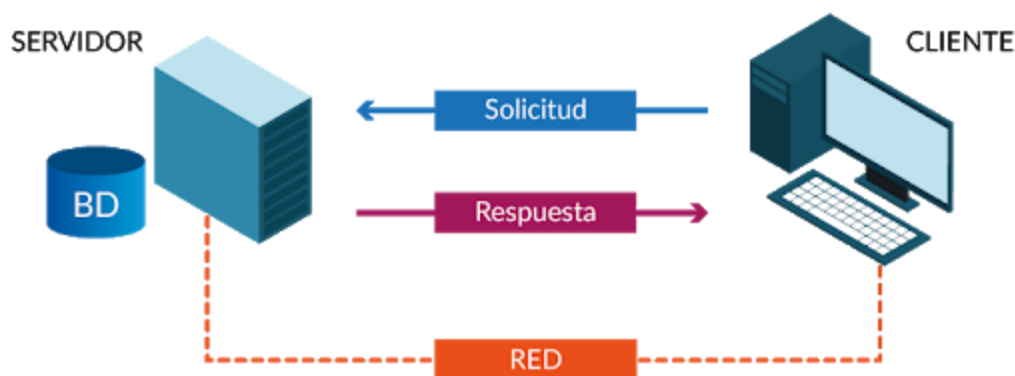
WWW (World Wide Web) es un concepto que se inició en la década de 1990 para que las personas y las máquinas puedan comunicarse entre sí dentro de un determinado espacio. Se utiliza para **intercambiar, distribuir y compartir información en una red**. En ese momento, la web consistía predominantemente en sitios web estáticos basados en HTML, en otras palabras, hipertextos que un navegador puede recuperar. Más adelante se agregaron sitios web dinámicos y servicios web distribuidos.





Modelo Cliente - Servidor

El concepto de **cliente-servidor** hace referencia a un modelo de comunicación en el que se vinculan varios dispositivos informáticos dentro de una red. El **cliente** realiza peticiones de servicios al **servidor** que es el encargado de satisfacer dichos requerimientos del cliente.



En este modelo, las tareas se distribuyen entre los **servidores** (*que son proveedores de servicios*) y los **clientes** (*que demandan servicios*), de esta manera puede repartirse la capacidad de procesamiento, ya que el servidor puede ejecutarse sobre más de un equipo y ser más de un programa.

Gran parte de los servicios de Internet obedecen a la arquitectura cliente servidor.

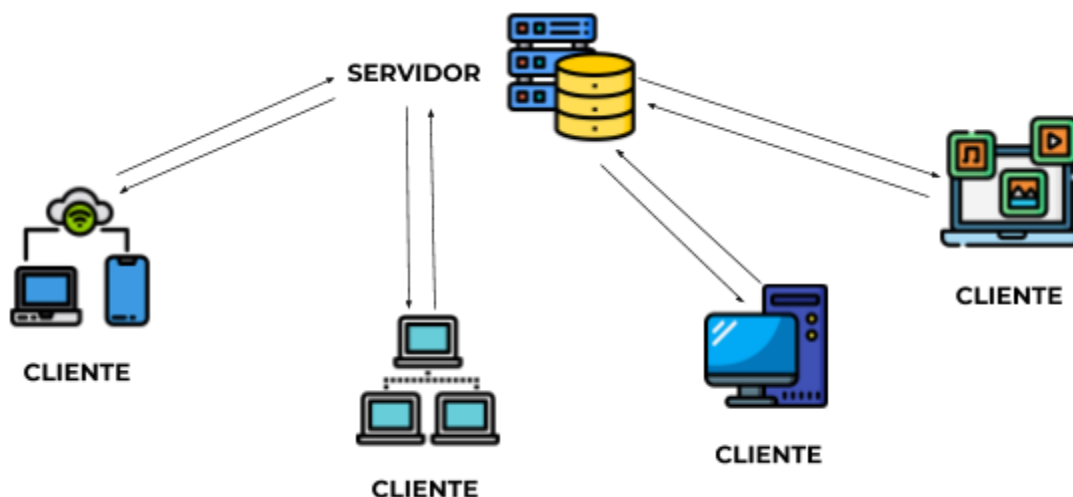
El servidor web pone a disposición del cliente los sitios web, a los cuales el cliente accede a través de su navegador. El servidor, de esta manera, aloja los datos que el cliente solicita mediante el navegador instalado en su computadora.





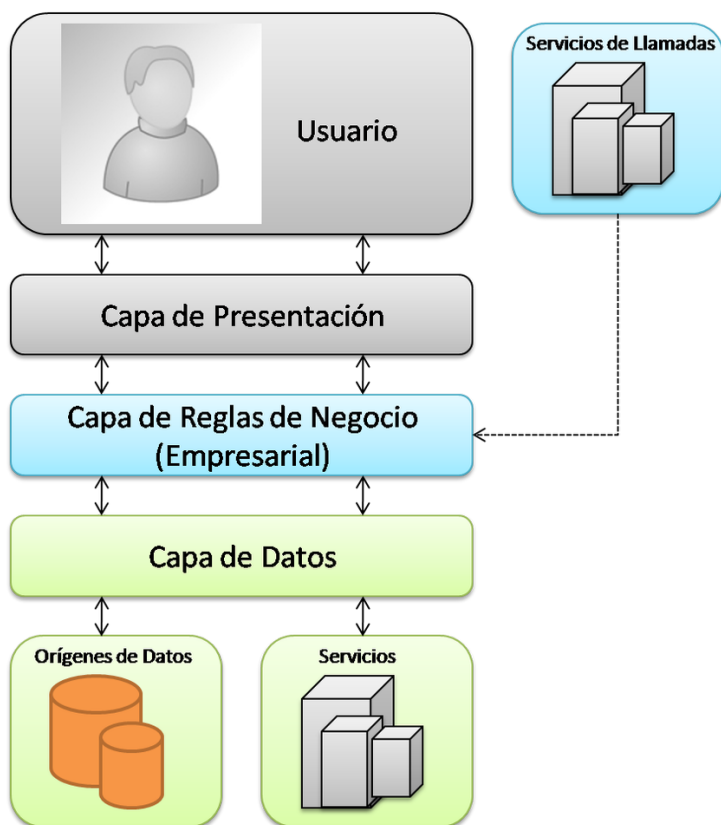
Tal como se ve en la imagen siguiente, el modelo cliente-servidor posee los elementos básicos que nombraremos a continuación:

- **Red:** Una red es un conjunto de clientes, servidores y base de datos unidos de una manera física o no física en el que existen protocolos de transmisión de información establecidos.
- **Cliente:** Este concepto hace referencia a un demandante de servicios, este cliente puede ser un ordenador como también una aplicación de informática, la cual requiere información proveniente de la red para funcionar.
- **Servidor:** Se refiere a un proveedor de servicios, este servidor a su vez puede ser un ordenador o una aplicación informática la cual envía información a los demás agentes de la red.
- **Protocolo:** Un protocolo es un conjunto de normas o reglas y pasos establecidos de manera clara y concreta sobre el flujo de información en una red estructurada.
- **Servicios:** Un servicio es un conjunto de información que busca responder las necesidades de un cliente, donde esta información pueden ser mail, música, mensajes simples entre software, videos, etc.
- **Base de datos:** Son bancos de información ordenada, categorizada y clasificada que forman parte de la red, que son sitios de almacenaje para la utilización de los servidores y también directamente de los clientes.





Modelo de 3 capas



El **modelo de 3 capas** es una arquitectura basada en el Modelo Cliente-Servidor, en el cual la **interfaz de usuario** (Capa de Presentación), la **lógica de negocio** (o reglas de negocio) y el **almacenamiento de la información y su acceso**, son desarrolladas y mantenidas de forma independiente, como módulos o plataformas.

Este modelo permite la *escalabilidad de cada capa*, logrando así la **actualización o reemplazo de manera independiente**, en respuesta a los cambios de requerimiento o tecnología.

Por ejemplo, la actualización de los estilos de la interfaz de usuario, solo afectará a la capa de presentación.





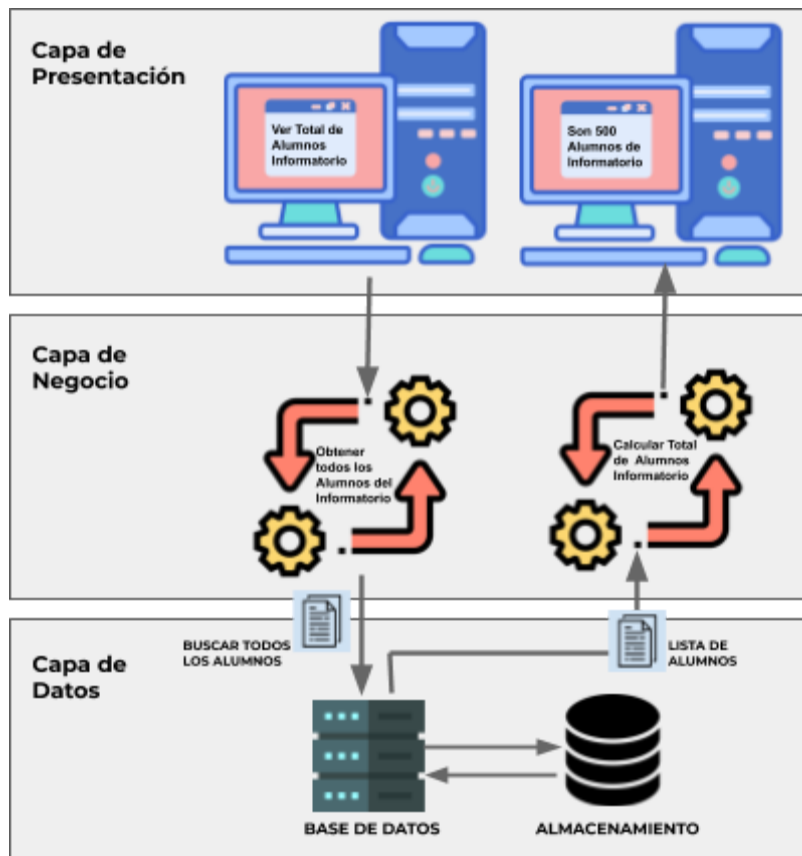
Capa de Presentación	Capa de Negocios	Capa de Datos
<p>Este es el nivel más alto de la aplicación.</p> <p>Muestra información relacionada con los productos y servicios que ofrece una aplicación.</p> <p>En términos simples, es una capa a la que los usuarios pueden acceder directamente (como una página web o la interfaz gráfica de un sistema operativo).</p>	<p>O capa de lógica de negocio, se extrae del nivel de presentación y, controla la funcionalidad de una aplicación realizando un procesamiento detallado.</p> <p>Se encarga de recibir las peticiones del usuario y enviar la respuesta una vez ejecutado el proceso.</p> <p>Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos almacenar o recuperar datos de él.</p>	<p>Incluye los mecanismos de persistencia de datos (servidores de bases de datos, archivos compartidos, etc.) y la capa de acceso a datos que encapsula los mecanismos de persistencia y expone los datos. Evitar las dependencias en los mecanismos de almacenamiento permite actualizaciones o cambios sin que los clientes de nivel de aplicación se vean afectados o incluso conscientes del cambio.</p> <p>Al igual que con la separación de cualquier nivel, existen costos de implementación y, a menudo, costos de rendimiento a cambio de una mejor escalabilidad y facilidad de mantenimiento.</p>





¿Cómo lo vemos en nuestro ámbito?

En el campo del desarrollo web, los tres niveles a menudo se usan para referirse a cualquier aplicación o sitio web, que se crean utilizando tres niveles:



- Un **servidor web front-end** que sirve contenido estático y, potencialmente, contenido dinámico en caché. *En una aplicación basada en la web, el front end es el contenido representado por el navegador. El contenido puede ser estático o generado dinámicamente.*
- Un **servidor de aplicaciones** de nivel de generación y procesamiento de contenido dinámico medio (por ejemplo, Django, Symfony, Spring, ASP.NET, Rails, Node.js).
- Una **base de datos back-end o almacén de datos**, que comprende tanto conjuntos de datos como el software del sistema de gestión de bases de datos que gestiona y proporciona acceso a los datos.



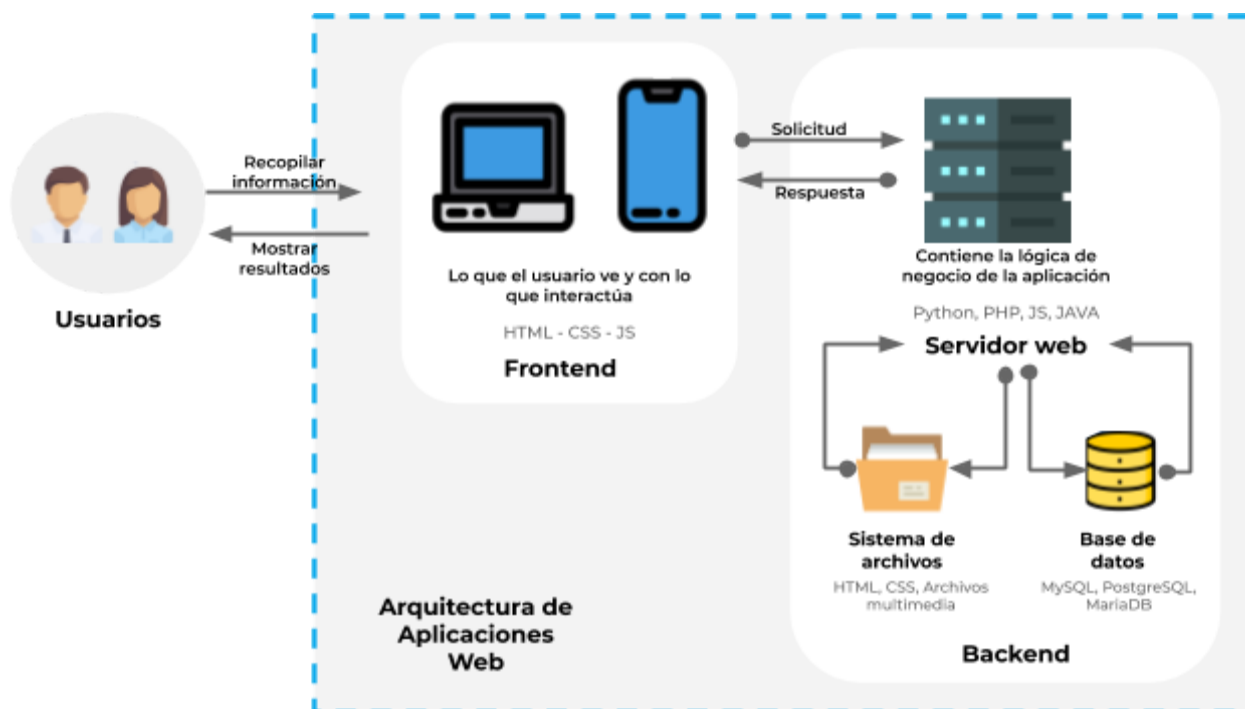


Arquitectura de Aplicaciones Web

La arquitectura de aplicaciones web hace referencia a las **relaciones e interacciones entre los componentes de la aplicación**, como los **sistemas de middleware**, las **interfaces de usuario** y las **bases de datos**.

En términos simples, este es un patrón de interacción entre los componentes de la aplicación web, como el servidor web, el servidor de bases de datos, etc.

El proceso inicia cuando *se hace clic en una URL o cuando el usuario escribe cualquier URL y toca "Ir"*, entonces el navegador encuentra la URL a través de la red, el servidor enviará los datos al navegador y el navegador ejecutará esos datos para mostrar la página solicitada. Ahora, el usuario puede interactuar con el sitio web. Por supuesto, todas estas acciones se ejecutan en cuestión de segundos. De lo contrario, los usuarios no se molestarán en utilizar sitios web.





Para elegir que tipo de arquitectura vamos a utilizar debemos tener en cuenta los **requisitos del usuario**, del **equipo de desarrollo** y de tu **cliente o product owner** del producto de software.

Componentes de la Arquitectura de Aplicaciones Web

→ Interfaz de usuario de la aplicación

Incluyen paneles, datos estadísticos, elementos de notificación, diseños, seguimiento de actividad y otros elementos. Estos componentes crean las imágenes de una página web y sientan las bases para la experiencia del usuario.

→ Componentes estructurales

Los componentes estructurales de una aplicación web se refieren básicamente a la funcionalidad de la aplicación web con la que interactúa un usuario, el control y el almacenamiento de la base de datos.

Cuando se trata de construir los componentes, hay varios modelos para elegir:

→ 1 servidor web y 1 base de datos:

Es el modelo más simple. Con esta arquitectura de servidor web, el funcionamiento exitoso de una aplicación depende de la estabilidad del servidor. En otras palabras, si hay un problema con el servidor, la aplicación no funcionará. Aún así, el modelo es suficiente para pruebas y sesiones privadas.

→ 2 servidores web y 2 bases de datos:

Es un modelo más confiable, ya que hay un servidor de respaldo. Por otro lado, es importante garantizar que la base de datos sea segura y se ejecute siempre.

→ Más de 2 servidores web y bases de datos:

Es la opción más confiable. Debido a su capacidad para administrar y procesar grandes cantidades de datos, este modelo es una base sólida para la arquitectura de una aplicación web empresarial.

