

# Tarea 1

## Unidad 1

**Docente:** Efrén Emmanuel Prado  
López

**Alumno:** Sara del Carmen  
Sanches de la Rosa

**Fecha:** 30/01/2023

**Carrera:** Tic's





**Instrucciones:** Investigar los temas y conceptos claves sobre aplicaciones web, como su evolución, arquitectura y planificación y realiza un resumen.

Primero que todo hay que ver **que son las aplicaciones web** según san Google dice que las aplicaciones web son " Son programas que funcionan en internet, que los datos o los archivos en los que trabajas son procesados y almacenados dentro de la web. Estas aplicaciones, por lo general, no necesitan ser instaladas en tu computador".

Ósea no entendí, sin embargo, según yo las aplicaciones web se crean desde la computadora y después se suben mediante la internet y ya se hace aplicación web :)

Hablando de aplicaciones web su **evolución** fue gracias al Internet y las conexiones de alta velocidad que lo acompañan, se ha podido mejorar de manera significativa la manera de trabajar de algunas personas al poder hacerlo desde sus hogares, Internet ha permitido a estas personas mayor flexibilidad en términos de horarios y de localización.

La Web 1.0 (1991-2003) es la forma más básica que existe, con navegadores de sólo texto bastante rápidos ya que es de sólo lectura. El usuario no puede interactuar con el contenido de la página (nada de comentarios, respuestas, citas, etc), estando totalmente limitado a lo que el Webmaster sube a ésta.

Esto también fue lo que encontré en internet sobre la evolución de aplicaciones web

Algunos elementos de diseño típicos de un sitio Web 1.0 incluyen:

Páginas estáticas en vez de dinámicas por el usuario que la visita.

El uso de framesets o Marcos.

Extensiones propias del HTML como <blink> y <marquee>, etiquetas introducidas durante la guerra de navegadores web.

Libros de visitas online o guestbooks

botones GIF, casi siempre a una resolución típica de 88x31 pixels en tamaño promocionando navegadores web u otros productos.

formularios HTML enviados vía email. Un usuario llenaba un formulario y después de hacer clic se enviaba a través de un cliente de correo electrónico, con el problema que en el código se podía observar los detalles del envío del correo electrónico.

No se podían adherir comentarios ni nada parecido

Todas sus páginas se creaban de forma fija y muy pocas veces se actualizaban.

No se trata de una nueva versión, sino de una nueva forma de ver las cosas.

La **arquitectura** de un Sitio Web tiene tres componentes principales:

- Un servidor Web
- Una conexión de red
- Uno o más clientes

El servidor Web distribuye páginas de información formateada a los clientes que las solicitan. Los requerimientos son hechos a través de una conexión de red, y para ello se usa el protocolo HTTP. Una vez que se solicita esta petición mediante el protocolo HTTP y la recibe el servidor Web, éste localiza la página Web en su sistema de archivos y la envía de vuelta al navegador que la solicitó.

Los componentes de una aplicación Web son:

1. Lógica de negocio.

- Parte más importante de la aplicación.
- Define los procesos que involucran a la aplicación.
- Conjunto de operaciones requeridas para proveer el servicio.

2. Administración de los datos.

- Manipulación de BD y archivos.

3. Interfaz

- Los usuarios acceden a través de navegadores, móviles, PDAs, etc.
- Funcionalidad accesible a través del navegador.
- Limitada y dirigida por la aplicación.

Las aplicaciones web se modelan mediante lo que se conoce como modelo de capas, Una capa representa un elemento que procesa o trata información. Los tipos son:

- Modelo de dos capas: La información atraviesa dos capas entre la interfaz y la administración de los datos.
- Modelo de n-capas: La información atraviesa varias capas, el más habitual es el modelo de tres capas.
- Modelo de dos Capas.

Gran parte de la aplicación corre en el lado del cliente (fat client).

Las capas son:

- Cliente (fat client): La lógica de negocio está inmersa dentro de la aplicación que realiza el interfaz de usuario, en el lado del cliente.
- Servidor: Administra los datos.

Las limitaciones de este modelo son.

- Es difícilmente escalable
- Número de conexiones reducida



- Alta carga de la red.
- La flexibilidad es restringida
- La funcionalidad es limitada.

La **planificación** de aplicaciones web nos sirve para llevar a cabo un proyecto sus pasos son:

#### 1. Planificar el futuro

Puede sonar un poco duro que nos tengamos planear el futuro de algo que aún no existe, pero es así, que pasa si en un futuro tuviéramos problemas de escalabilidad, y sí tuviéramos que migrar las bases de datos de MySQL a Oracle, de una buena planificación depende la dificultad en un futuro de realizar cambios en la plataforma que sostiene nuestra aplicación.

#### 2. Documentación

Documenta todo lo que puedas tu aplicación en especial aquellas partes de la misma que hayan requerido un mayor tiempo de desarrollo, en un futuro cercano te librarás de más de un quebradero de cabeza.

#### 3. No empieces por los pequeños detalles

A menudo es más sencillo plantearse el desarrollo de pequeños servicios cómo pueden ser los RSS o una galería de imágenes, pero es fundamental comenzar por lo que vendrá a ser la base de nuestra aplicación, de este modo podremos ir completando nuestra aplicación en un orden lógico, primero los cimientos.

#### 4. Diseño de la interfaz

Esta entrada puede que te ayude en esta parte, es importante conseguir un buen diseño para nuestra interfaz, es importante conseguir un interfaz que se atractivo y útil para la gran mayoría de los usuarios, lo más importante es llegar a conectar con el usuario y la mejor manera es ofrecerle la mayor cantidad de información posible para que se llegue a sentir parte del proyecto pero todo esto muchas veces no lo llegamos a conseguir por una mala elección en la combinación de los colores que se integran en nuestra interfaz.

#### 5. Evita construirte librerías o frameworks innecesarios

Muchas personas siempre sostienen que es mejor que cada desarrollador utilice sus propias librerías, pero la realidad es que en muchas ocasiones esto supone una gran pérdida de tiempo, es mejor buscar una librería o framework que se parezca lo más posible a lo deseado que desarrollarlo por ti mismo.

#### 6. Ten un plan de proyecto

Asegúrate de que cada desarrollador que trabaje en el proyecto sepa qué es lo que debe de hacer y cuándo tiene que hacerlo, teniendo todo esto redactado en un documento te ayudará a identificar problemas.

## 7. Elige el lenguaje adecuado

Ya sabes que hay numerosas discusiones sobre ¿Cuál es el mejor lenguaje de programación?, la respuesta a menudo es depende de para que quieras utilizarlo, por eso asegúrate de escoger el lenguaje adecuado.

## 8. Evita las distracciones

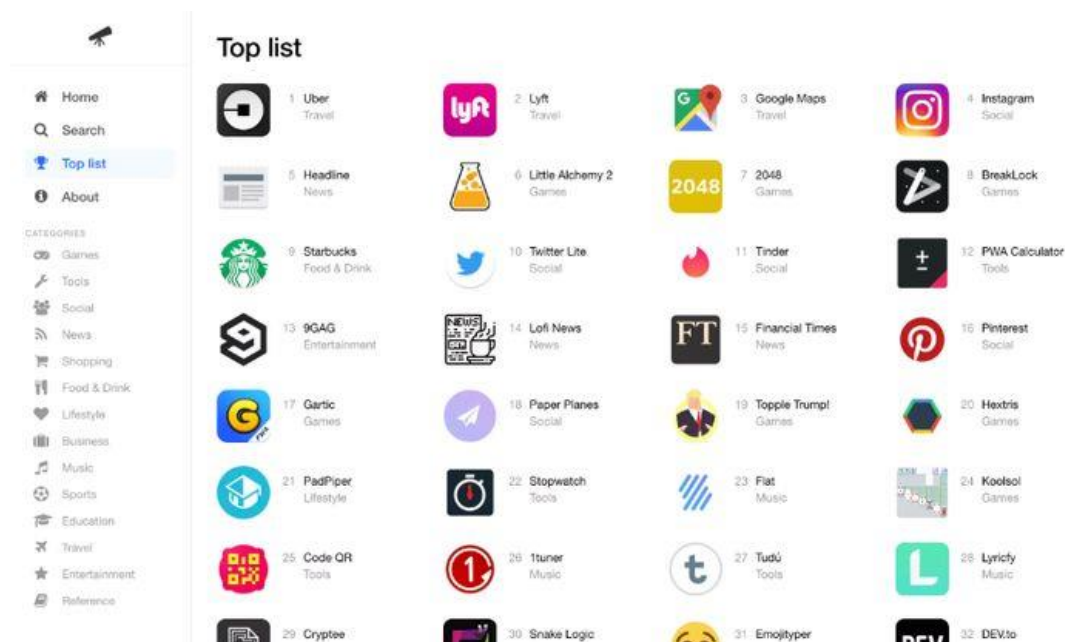
Optimiza tu tiempo, cuando estés trabajando que nada te distraiga, ni llamadas telefónicas, ni clientes de mensajerías instantáneas, proponte un buen número de horas al día de trabajo aislado dedicado al proyecto.

## 9. Sigue un buen control de calidad

Muchas veces nos equivocamos al planificar los tiempos de proyecto y esto lleva a menudo a los desarrolladores a tener que escribir “código sucio”, esta es una buena opción si lo que quieres es una demo de la aplicación funcionando, pero esa actitud te traerá problemas en el futuro.

## 10. Plántate los pasos a seguir después del desarrollo

Ten en cuenta que está fase de desarrollo algún día deberá acabar, estudia con qué frecuencia se deberá actualizar y que se necesita para que este funcione.



Esto es solo unos ejemplos de aplicaciones web