



Facultad de Ingeniería
Escuela de Ingeniería de Sistemas y Computación

Mauricio Gaona
mauricio.gaona@correounivalle.edu.co

Profesor

2023-I

Desarrollo de Software I



01

RESUMEN

02

ESTIMACIÓN DE TIEMPO Y
COSTOS DE UN PROYECTO DE
SOFTWARE.
CONCEPTOS PARTICULARES
DE SCRUM.

03

EL WORD WIDE WEB

04

EXPOSICIONES, TALLER Y
PROYECTO DEL CURSO

Conceptos

Resumen





Conceptos básicos en el desarrollo de APPs





Interfaces de usuario

- Story board o modelo de pantallas o Front-end UI
- Reportes
- Arquitecturas de software





Estimación de Tiempos Y costos de un desarrollo de software usando metodologías ágiles

Estimación de Tiempos en la metodologías ágiles

Release Plan

Sprint	HU	Puntos de HU	Desarrollador
1	HU1	3	Pedro/María Juan/Ricardo
	HU2	4	
2	HU3	2	Maria Pedro Juan
	HU4	4	
	HU6	2	
3	HU5	4	Maria Juan
	HU7	3	
4	HU9	3	Pedro Juan
	HU8	3	
5	HU10	4	Maria Pedro
	HU11	3	

Velocidad

Road Map

Release 1
Octubre 5

Release 2
Noviembre 5

Release 3
Diciembre 8

Velocidad: Cantidad de puntos que un equipo es capaz de hacer en un sprint

Desarrollo de Software I

Estimación de Tiempos en la metodologías ágiles

Sprint	HU	Puntos de HU	Desarrollador
1	HU1	4	Pedro/María Juan/Ricardo
	HU2	4	
2	HU3	2	Maria Pedro Juan
	HU4	4	
	HU6	3	
3	HU5	4	Maria Juan
	HU7	4	
4	HU9	5	Pedro Juan
	HU8	3	
5	HU10	4	Maria Pedro
	HU11	5	
		Total puntos del proyecto: 35	
		Puntos promedio por Sprint: 7	

Estimación de esfuerzo (tamaño del software) Total Puntos del proyecto	(Product Backlog)	35 puntos
Velocidad en puntos del equipo por mes	Puntos por mes (2 Sprints)*	14 puntos
Duración estimada del proyecto	Puntos del proyecto / Velocidad en puntos del equipo por mes	2,5 meses

* Asumiendo sprints de 2 semanas

3 meses



Preguntas ?



Costo de una App en las metodologías ágiles

1. Costo

Es lo que cuesta elaborar o realizar un producto o servicio.

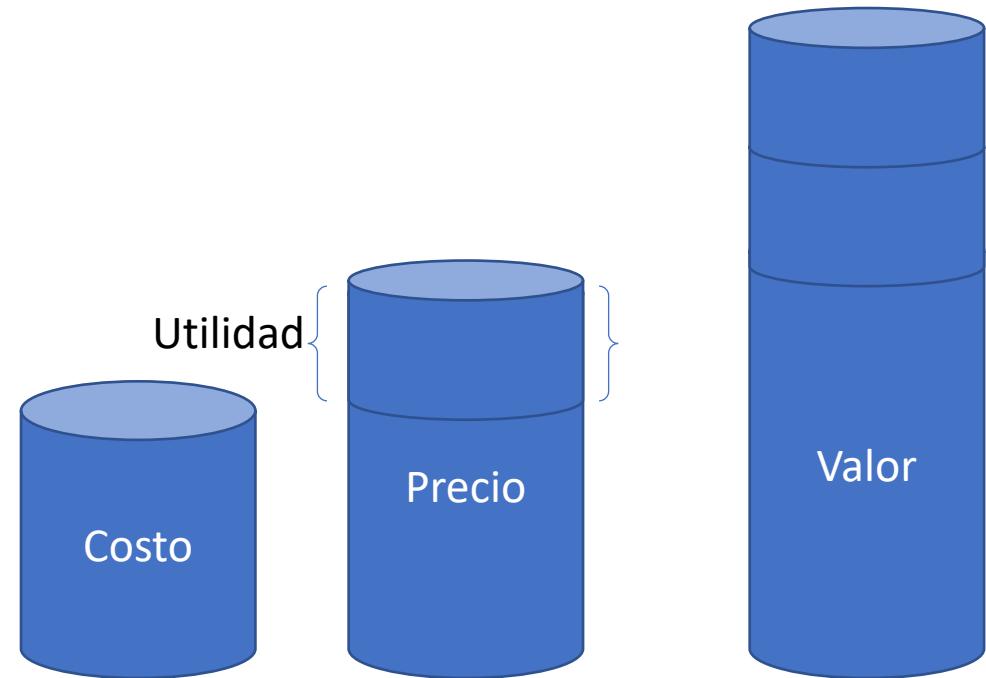
2. Precio

Es la cantidad de dinero que una empresa espera que se le pague por su producto. Todo lo que se cobra por encima del costo será la utilidad (ganancia).

3. Valor

Es el monto que se le da a un producto o servicio dentro del mercado según las necesidades de los clientes.

Mientras el precio es el monto fijado para un producto o servicio, el valor es en esencia el monto que un cliente está dispuesto a pagar.





Costo de una App en las metodologías ágiles

Estimación de costos de un desarrollo de software usando metodologías ágiles

Precio fijo por HU

- Se cobra por cada HU
- Fácil de entender y de aplicar
- Se cobra por el esfuerzo requerido para cada HU
- Enfoca al equipo en trabajar para la entrega
- El cliente puede predecir el impacto de agregar o quitar HU

Hay que **estimar bien** o podría tener impactos en los costos del proyecto

Los costos se hacen para la línea base de HU y si se agregan HU durante el desarrollo la estimación se puede ajustar

Costo de una App en las metodologías ágiles

	Costos del desarrollo		
PB	Estimación de esfuerzo (tamaño del software) Puntos del proyecto	(PB: Product Backlog)	80 puntos
PM	Velocidad en puntos del equipo por mes	(PM: Puntos por mes)	20 puntos
	Duración estimada del proyecto en meses	Puntos del proyecto / Velocidad en puntos del equipo por mes	4 meses
C1	Costo del equipo de trabajo y otros (en millones mensuales)	Salarios del equipo y otros costos como servidores, plantillas, impuestos, etc	56.600.000
CP	Costo aproximado por punto	Costo aproximado por punto = (C1) / PM. Costo aproximado por punto = 56.600.000/20	2.830.000
	Costo estimado del proyecto de software	Costo estimado = Puntos del proyecto x Costo de un punto Costo estimado = 80 x 2.830.000	226.400.000

Importante adicionar utilidad estimada. Ejemplo: 30%, 40%, 100%, 300% ...

Precio = costos + utilidad estimada



Costo de una App en las metodologías ágiles

Precio estimado de un proyecto de software							
	Valor hora	#horas mes	Valor mes		Valor	Meses	Total
Desarrollador 1	30.000,00	160	4.800.000,0	Costo mensuales	56.600.000,0	1	56.600.000,00
Desarrollador 2	30.000,00	80	2.400.000,0				
Desarrollador 3	30.000,00	160	4.800.000,0	Subtotal			56.600.000,00
Product Owner	30.000,00	160	4.800.000,0	Puntos por mes			20
Scrum Master	40.000,00	160	6.400.000,0	Total puntos			80
Tester	30.000,00	160	4.800.000,0	Costo estimado de un punto			2.830.000,00
Infraestructura	30.000,00	80	2.400.000,0	Costo estimado del proyecto			226.400.000,00
Diseñador	30.000,00	80	2.400.000,0	Utilidad (40%)			90.560.000,00
	Subtotal		32.800.000,0	Precio estimado del proyecto (sin impuestos)			316.960.000,00
				IVA 19%			60.222.400,00
Asesor 1	40.000,00	40	1.600.000,0	Otros impuestos (7%)			22.187.200,00
				Precio estimado del proyecto			399.369.600,00
Gerente	40.000,00	160	6.400.000,0				
Secretaria	15.000,00	160	2.400.000,0				
Auxiliar	15.000,00	160	2.400.000,0				
Servicios públicos, arriendo			9.000.000,0				
Hosting, librerías, templates, etc			1.000.000,0				
Otros costos fijos			1.000.000,0				
Costos mensuales			56.600.000,0				

Estimación de costos de un desarrollo de software usando metodologías ágiles

- Para la estimación de costos los equipos requieren experiencia en la estimación de los puntos de las HU
- Estimar **cuanto cuesta un punto de HU** es el factor clave para la mayoría de cálculos
- El tamaño del producto (Product Backlog) suele ser el factor determinante más importante del costo del desarrollo de software, independientemente de la metodología específica que se utilice.
- Planning poker ayuda a estimar los puntos de una HU
- Transformar puntos a horas suele ser una actividad compleja y depende de muchos factores
- Factores que pueden afectar los costos: Habilidades del equipo de desarrollo, Tecnología usada, Costos administrativo e impuestos



Preguntas ?



Pregunta en conferencia.

En la metodología Scrum se recomienda tener equipos de 5 a 10 personas máximo.

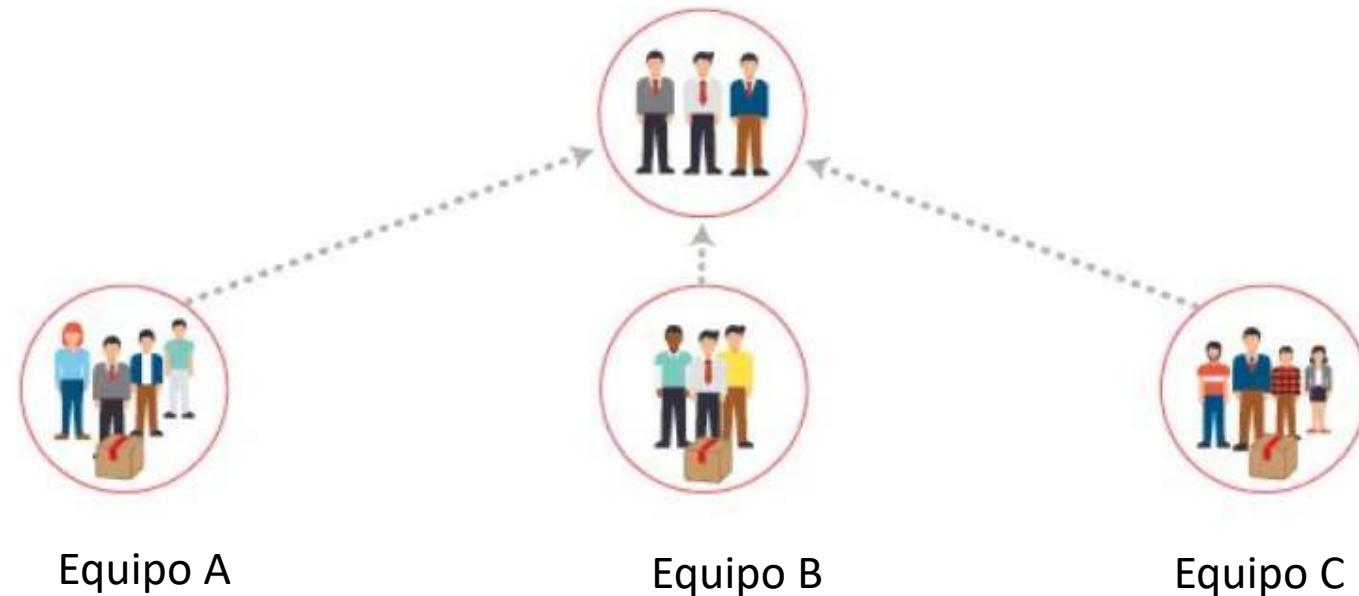
¿Qué pasa si es un proyecto es muy grande y se tienen muchos desarrolladores. Ejemplo 40 desarrolladores ?

Respuesta: **Scrum de Scrums**.

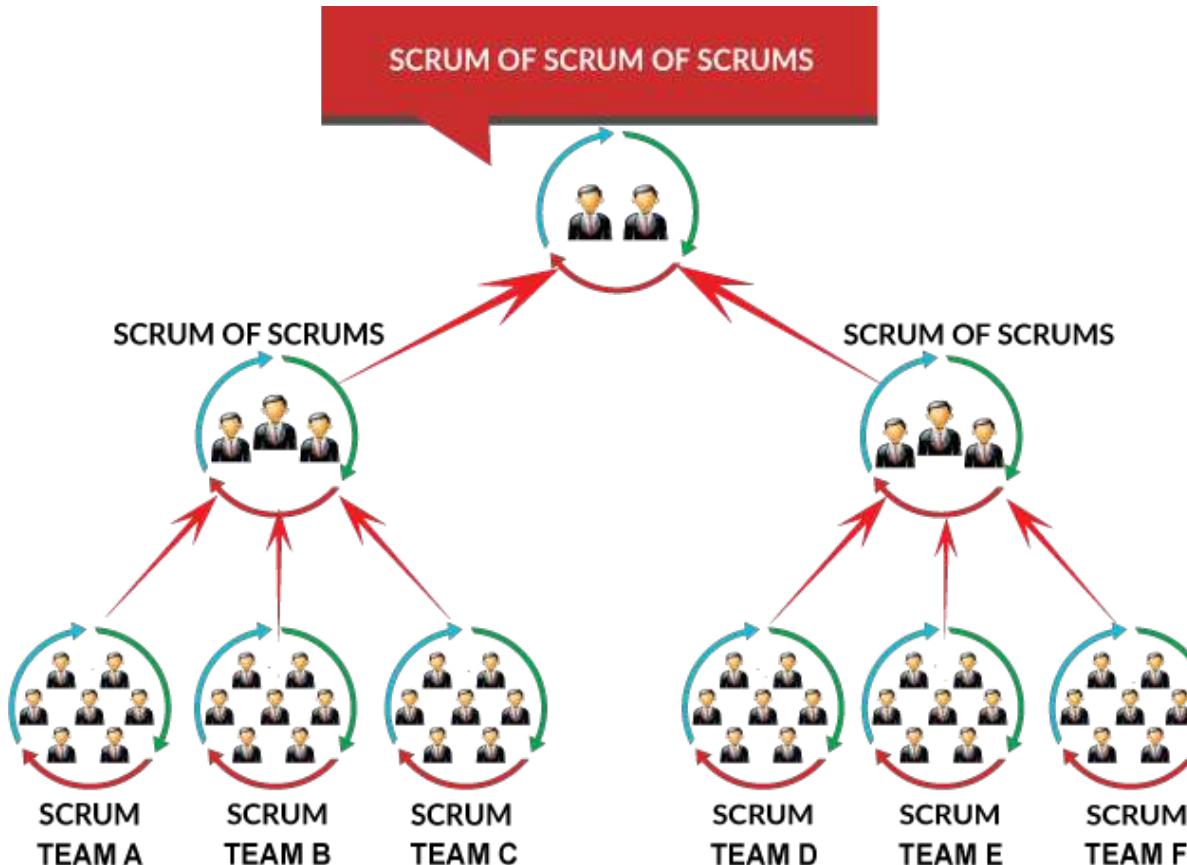
Reunión diaria de representantes de cada equipo. Duración 15 minutos

Scaled Agile Framework® (SAFe®) es un conjunto de patrones de organización y flujo de trabajo que sirve para implementar prácticas ágiles a escala empresarial.

Scrum of Scrums (SoS)



Pregunta en conferencia.



Preguntas

1. ¿Qué ha hecho mi equipo hasta ahora desde la última vez que nos reunimos que podría afectar a otros equipos?
2. ¿Qué hará mi equipo antes de que nos volvamos a encontrar que pueda afectar a otros equipos?
3. ¿A qué problema se enfrenta mi equipo que podría obtener la ayuda de otro equipo para resolverlo?

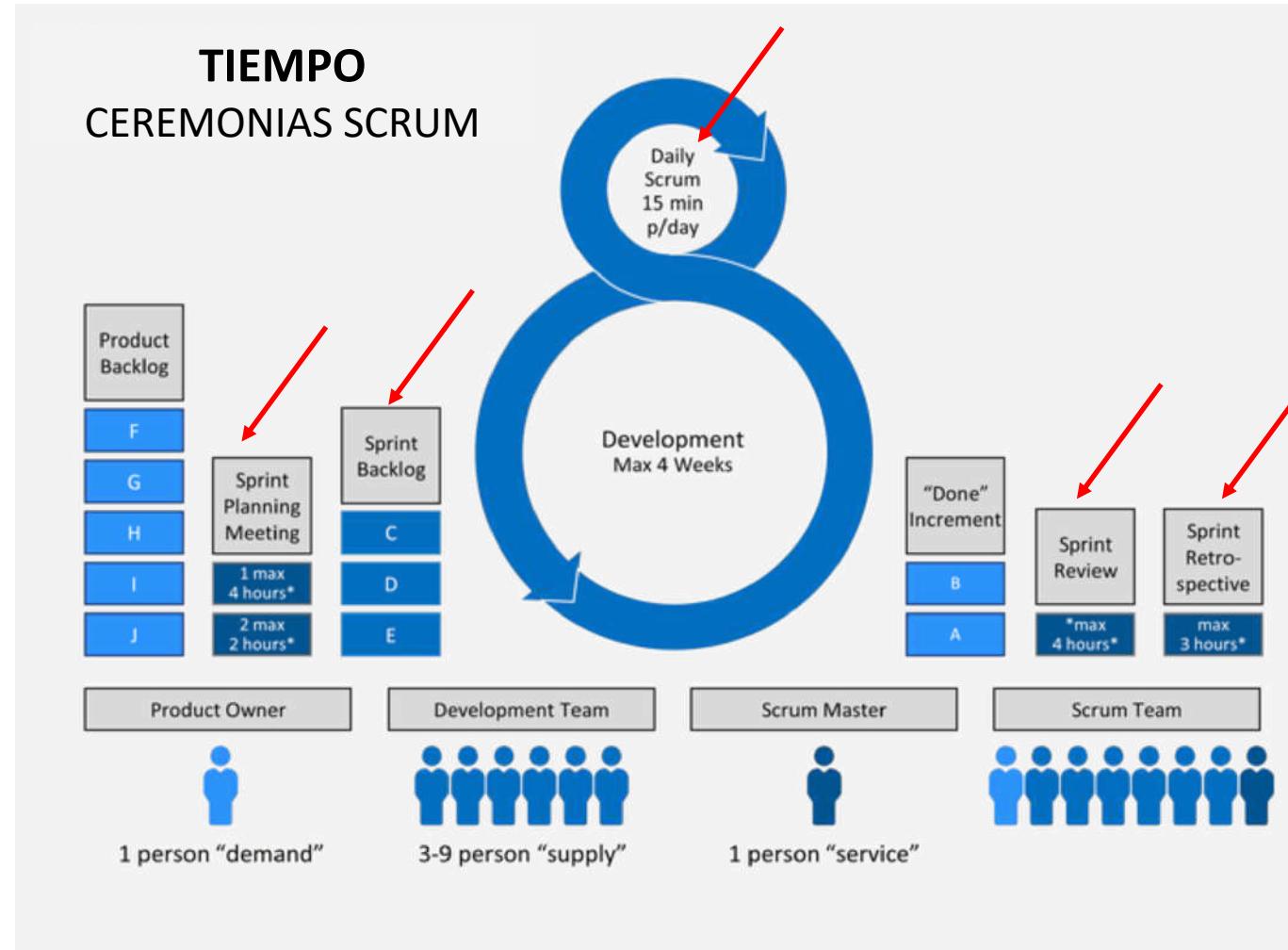


Preguntas ?



Pregunta en conferencia.

En las empresas es común que los desarrolladores trabajen en más de un proyecto. ¿Cómo se maneja esto en Scrum?



Prácticas ágiles

En las empresas es común que los desarrolladores trabajen en más de un proyecto. ¿Cómo se maneja esto en Scrum?

Disponibilidad de los desarrolladores (Sprint de dos semanas 80 horas)

	Desarrollador 1	Desarrollador 2	Desarrollador 3



Prácticas ágiles

En las empresas es común que los desarrolladores trabajen en más de un proyecto. ¿Cómo se maneja esto en Scrum?

Disponibilidad de los desarrolladores (Sprint de dos semanas 80 horas)

	Desarrollador 1	Desarrollador 2	Desarrollador 3
Ceremonias Scrum	12	12	12



Prácticas ágiles

En las empresas es común que los desarrolladores trabajen en más de un proyecto. ¿Cómo se maneja esto en Scrum?

Disponibilidad de los desarrolladores (Sprint de dos semanas 80 horas)

	Desarrollador 1	Desarrollador 2	Desarrollador 3
Ceremonias Scrum	12	12	12
Perdidas por productividad	10	10	10



Prácticas ágiles

En las empresas es común que los desarrolladores trabajen en más de un proyecto. ¿Cómo se maneja esto en Scrum?

Disponibilidad de los desarrolladores (Sprint de dos semanas 80 horas)

	Desarrollador 1	Desarrollador 2	Desarrollador 3
Ceremonias Scrum	12	12	12
Perdidas por productividad	10	10	10
Otros proyectos	0	16	32



Prácticas ágiles

En las empresas es común que los desarrolladores trabajen en más de un proyecto. ¿Cómo se maneja esto en Scrum?

Disponibilidad de los desarrolladores (Sprint de dos semanas 80 horas)

	Desarrollador 1	Desarrollador 2	Desarrollador 3
Ceremonias Scrum	12	12	12
Perdidas por productividad	10	10	10
Otros proyectos	0	16	32
Subtotal	22	38	54
Horas disponibles	(80-22) 58	42	26

Puntos planeados en el sprint 14P 11P 8P

Si un punto equivale a 4 horas 56 44 32





Preguntas ?





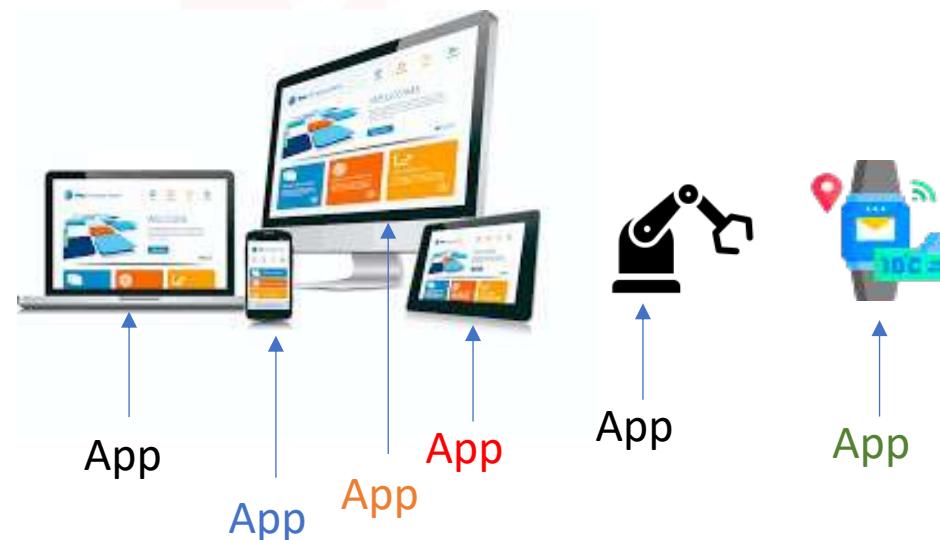
Aplicaciones Web

¿Qué es un App?

App = Application.



Una aplicación es un programa de software diseñado para realizar tareas específicas.



Aplicaciones Web

Tipos de App

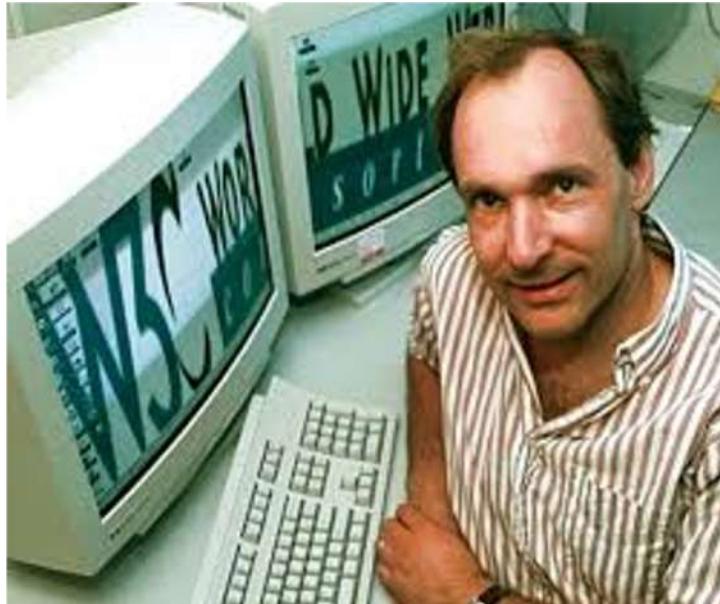




Aplicaciones WEB

El World Wide Web (WWW)

¿Cómo inició? ¿Por qué se creó?



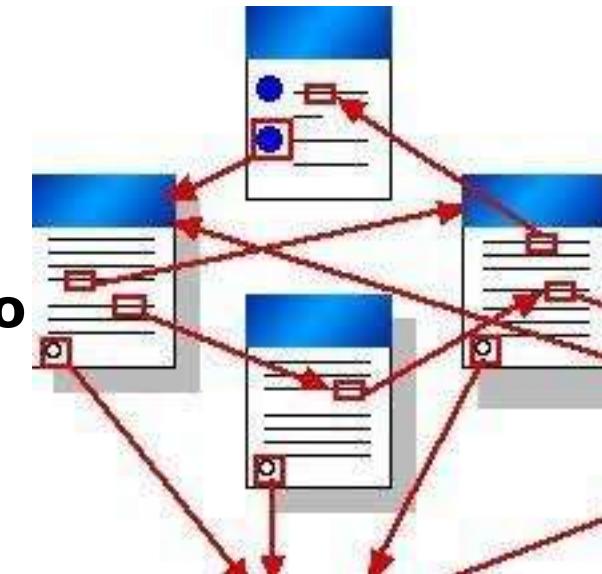
Muchos programas + muchos formatos + muchos protocolos = **caos**

Iniciativa World Wide Web (WWW) (web)
Tim Bernes-Lee 1.989 y asociados del CERN
Nuevo sistema de información

Propósito orden

Programa llamado Browser

Idea central: Metáfora de **Hipertexto**



Concepto

Tecnologías web

Primer Browser diciembre de 1.990.

Enero de 1.991 resultado del proyecto WWW en el CERN.

Febrero de 1.991 liberado para uso público.

Febrero de 1.993 (NCSA) primera versión de Mosaic.

Noviembre de 1.993 liberado para otras plataformas:

Apple Macintosh, Microsoft Windows y Xwindows.



Cello, ViolaWWW, Mosaic
Netscape
Explorer, Amaya, Firefox, Opera
Chrome, Edge, Safari, etc.
Microbrowsers.



Concepto

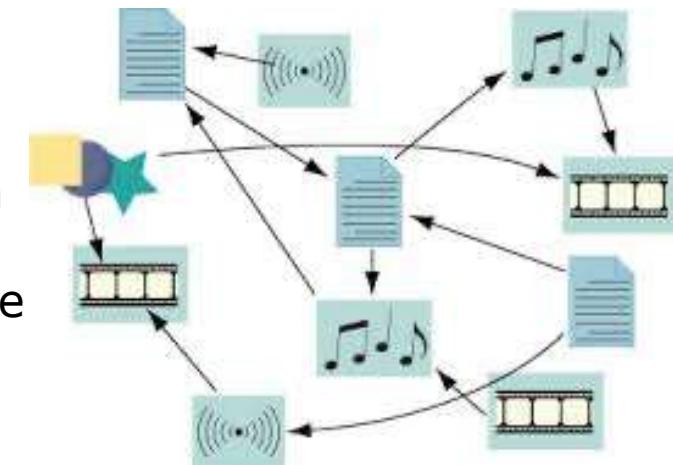
¿Qué es el WWW?

Es un sistema de distribución de documentos de hipertexto o hipermedia interconectados y accesibles a través de Internet



Sus principales características son:

- Disponible para múltiples plataformas.
- Es distribuido (puede accesar a información geográficamente dispersa)
- Es dinámico (la información se puede actualizar)
- Es de carácter multimedial, puede accesar muchas tipos de información en Internet (gráficos, texto, sonido, video, etc)
- Es interactivo, es decir, permite la interactividad con los usuarios o entre usuarios
- Es usado hacer páginas web y aplicaciones web



¿Quién es el dueño del Web?

No existe un individuo o entidad sola que sea dueño del web, es un consorcio internacional de carácter abierto y libre, pero existen ciertas entidades a nivel mundial, que lideran e impulsan su organización y desarrollo.

En los Estados Unidos, existe el WWW (W3); Consortium W3 en el MIT
(<https://www.w3c.org/>)

Universidad
del Valle



En el consorcio W3C trabajan conjuntamente para desarrollar estándares para la Web.

¿Qué puede usted hacer en el Web?

"Casi todo lo que a usted le guste o lo que los medios electrónicos de transferencia y despliegue de información permitan".

Típicamente el Web se ha utilizado en:

- Distribución de información de empresas o entidades
- Uso comercial: - Catálogos - Ventas - Distribuir manuales técnico. - Distribuir software
- En la academia (universidades, colegios).
- Información pública. (Gobiernos)
- Guías, Manuales, Periódicos, Blogs, Redes sociales, juegos, etc.
- En investigación para el intercambio de información, revistas, etc.
- Sistemas de información web (WIS)
- Gerenciar y administrar todo una empresa o institución

Visión Actual y Futura del Web:

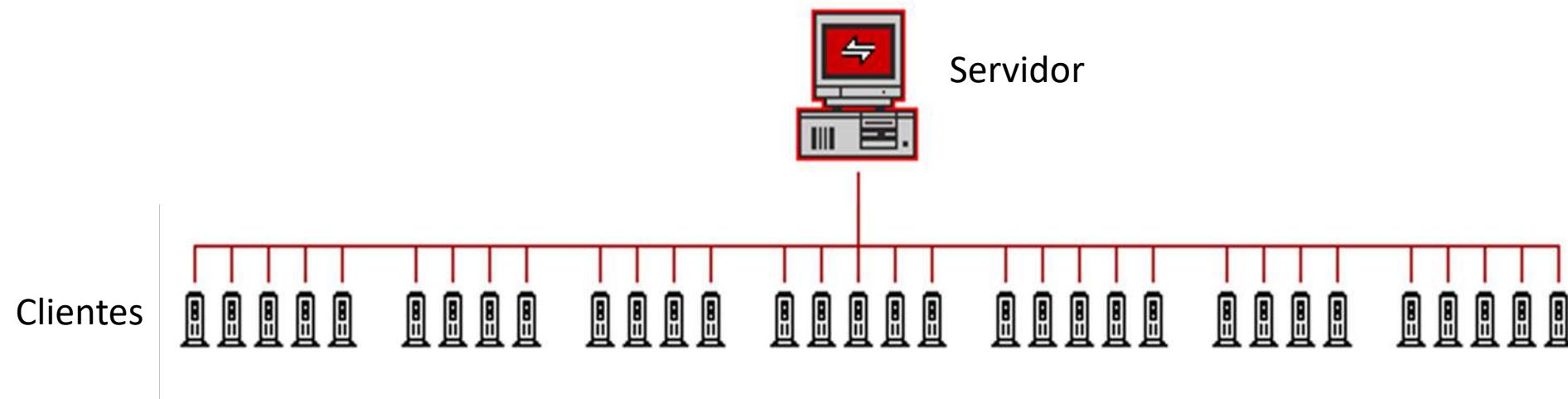
- RIA (Rich Internet Applications), Responsive
- SaaS
- PWA , Motion UI, Smart Apps ...



Arquitectura Cliente Servidor

Es un modelo de diseño de software, basado en la prestación y uso de servicios; para lo cual requiere de un servidor (provee los servicios) y un cliente (utiliza los servicios)

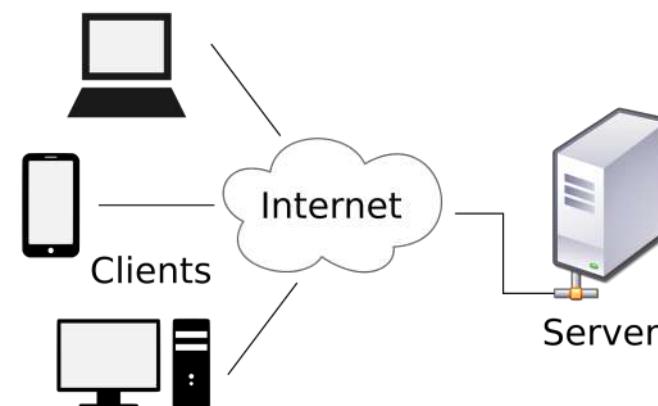
- Poseen conexiones son **con-estado** (State Connections)





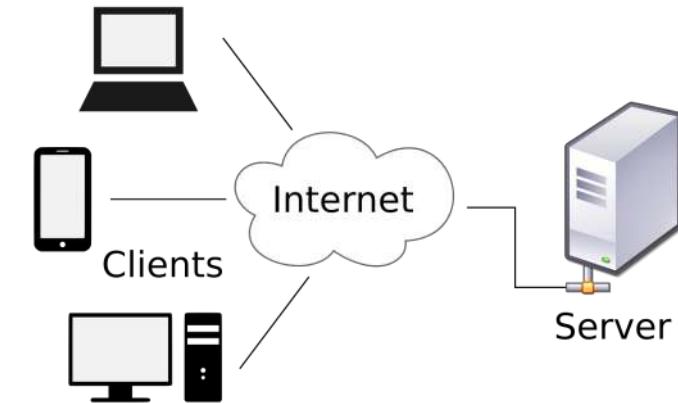
Arquitectura Cliente-Servidor

- Conexiones son **con-estado** (State Connections)
- Los servidores de aplicaciones mantienen dos tipos de conexiones con estado:
 - Estado de intercliente: estado cuyas actualizaciones deben sincronizarse con las transacciones realizadas por otros clientes.
 - Estado de intracliente: estado que debe mantenerse para un cliente dado durante una sesión (que puede abarcar varias conexiones).



Arquitectura Cliente-Servidor

Recursos conexiones en **con-estado** (State Connections)

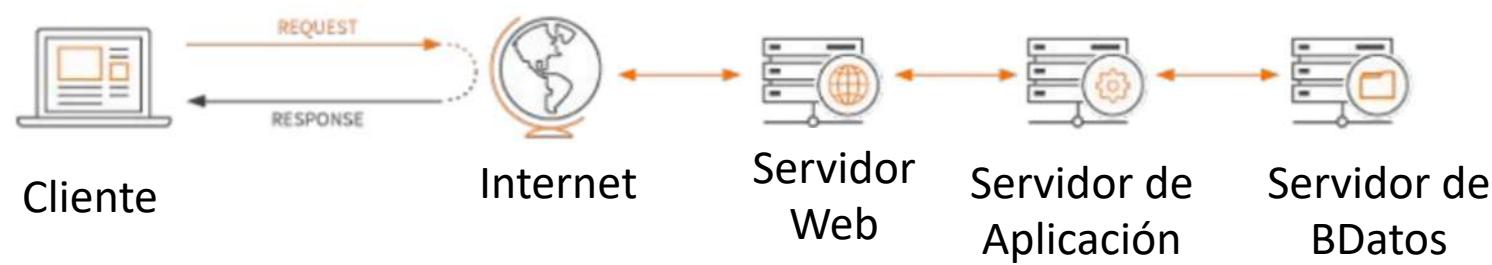




Arquitectura (Cliente Web - Servidor Web)

Es un extensión de la arquitectura cliente/servidor tradicional, donde las conexiones son **sin-estado** (Stateless Connections)

- Abrir una conexión
- Obtener un recurso (ejecutar un archivo y obtener la respuesta)
- Cerrar la conexión





Servidores y Clientes Web

HTTP es un protocolo sin estado

- No guarda ninguna información sobre conexiones anteriores
- Al finalizar la transacción todos los datos se pierden.
- Por esto se popularizaron las cookies, que son mensajes que los servidores web pasan al navegador cuando se visita sitios en Internet. El navegador almacena cada mensaje en un archivo pequeño, llamado “cookie.txt”. Cuando se solicita otra página desde el servidor, su navegador envía el cookie de vuelta al servidor; éstas son usadas comúnmente para guardar información relevante del usuario de una página o aplicación.

Tarea: Investigar que son variables de sesión en una aplicación web

Cliente Web (Web Browser)

Ventana para los usuarios finales hacia el WWW.

Servidor Web (Web Server)

Responsable por atender las peticiones de los clientes web.

Arquitectura (Cliente Web - Servidor Web)

Consumo de recurso de las conexiones son **sin-estado** (Stateless Connections)



Web server



1

Permite compartir recursos soportando procesos simultáneos y controlando el acceso a los servicios

2

Atiende muchos clientes con un solo servidor

3

Ubicación distribuida es decir puede controlar procesos en equipos geográficamente dispersos.

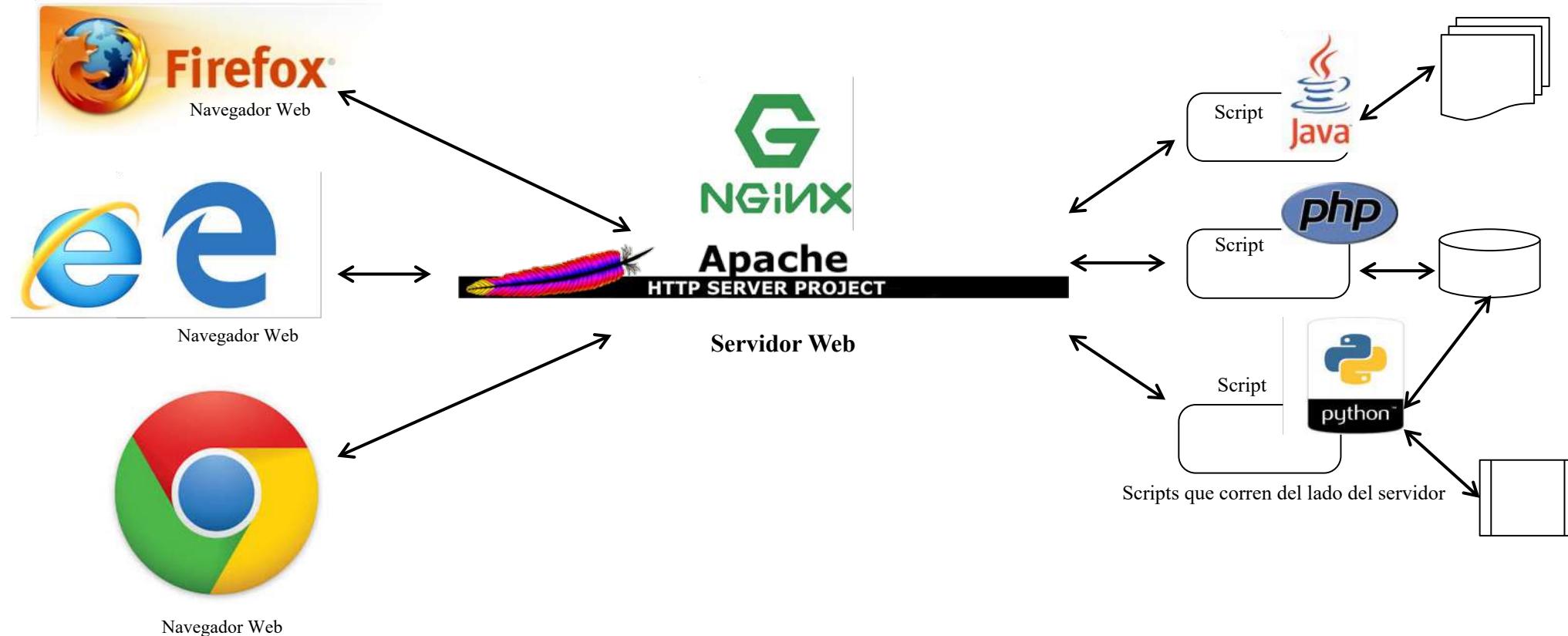
4

Aplicaciones de plataforma independiente dado que funciona en diferentes plataformas como Unix, Windows, IOs y Android.

5

Garantiza la integridad de los servicios que están centralizados en el servidor, lo cual facilita el mantenimiento y reduce los costos; además, garantiza la independencia de cada cliente.

Clients y Servidores Web



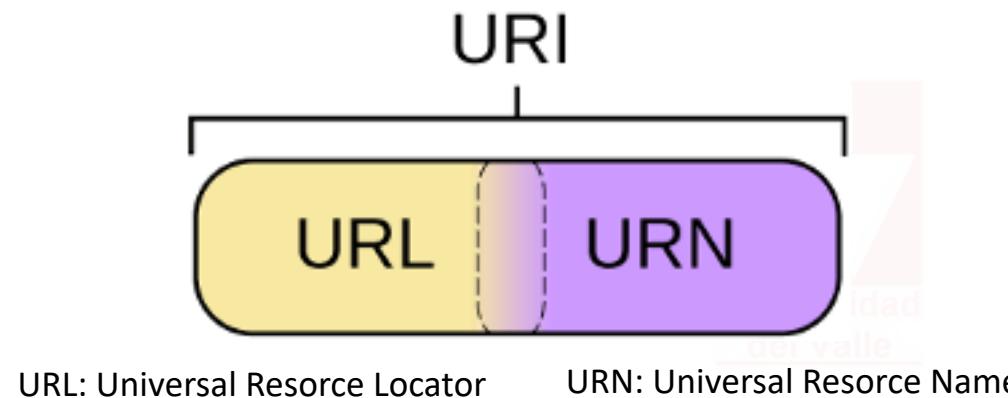
Tarea: Cuáles son las principales diferencias entre Apache y NGinx

Soporte para protocolos de transferencia de datos

HTTP://WWW.GOOGLE.COM

-  **HTTP:** (Hypertext Transfer Protocol) Protocolo nativo del web.
-  **HTTPS:** (Hypertext Transfer Protocol Secure)
-  **FTPS:** (File Transfer Protocol Secure) Protocolo de transferencia de archivos.
-  **GOPHER:** Universidad de Minnesota, Campus Informative Service Protocol.
-  **NNTP:** Protocolo usado para distribuir artículos colocados en Usenet.
-  **FILE:** Protocolo para archivos locales.
-  **TELNET:** Protocolo de comunicaciones basado en comandos de texto.

¿Cómo localizar un recurso en la Web?

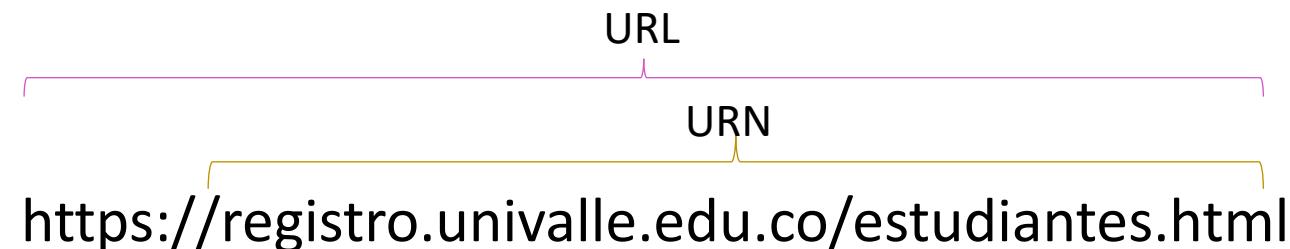


URI

Universal Resource Identifier

Es una forma precisa de nombrar y localizar los recursos web de manera única.

La URI contiene el nombre, el localizador o los dos



Anatomía de una URL

`https://www.univalle.edu.co/info/estudiantes.html`



Direcciones web: (ICANN)

(no patrocinados) (Dominios de alto nivel)

(**.edu, .com, .gov, .net, .mil, .org, .int**)

(.biz [www.neulevel.biz]) para negocios

(.info [www.afiliadas.info]) para todos los usuarios

(patrocinados)

(.aereo, .mobi, .coop, .jobs, .museum, .pro, .travel

Infraestructuras: .arpa, .root

En Fase de inicio: .post, .tel

Propuestos: .asia, .cym, .geo, .kid, .kids, .mail,
.sco, .web, .xxx

Borrados/retirados: .nato

Reservados: .example, .invalid, .localhost, .test

Pseudominios: .bitnet, .csnet, .local, .onion, .uucp, ws

gTLD, nuevos a partir de 2014 <http://newgtlds.icann.org/en/program-status/delegated-strings>

.futbol .click .help .cards. lltda .vision .productions .berlin etc.

Paises (<http://www.iana.org/root-whois/index.html>)

Casos .co , .tv, io, ai



Caracteres legales en una URL

Letras mayúsculas y minúsculas del alfabeto inglés

Números y Caracteres como \$ - @ . _

Otros caracteres como : = ; / # ? % & [espacio en blanco] se usan con un significado especial

Ejemplo

<http://www.tag.org/cftp/?lan=en>

Símbolos ilegales

Tabuladores retorno newline acentos ~

(para ser usados se trasladan a símbolos de escape)

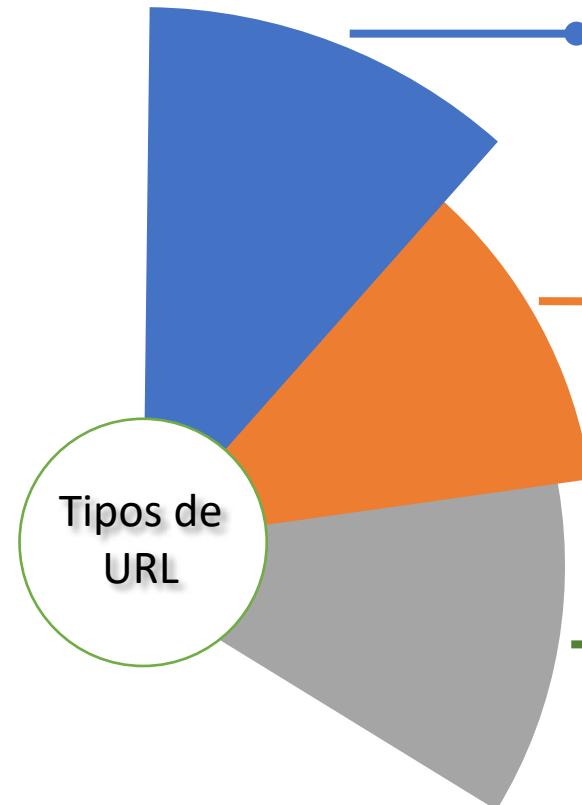
Ejemplo %20 espacio

%0D retorno

%25 %

<http://www.estrellateyarde.org/manual-de-html/manual-de-html-url-encoding>

Tipos de URLs



URL Completa

Cuando tiene todas las partes de una URL
`http://www.osi.org/cgi-bin/datos.py`

URL Parcial o relativa

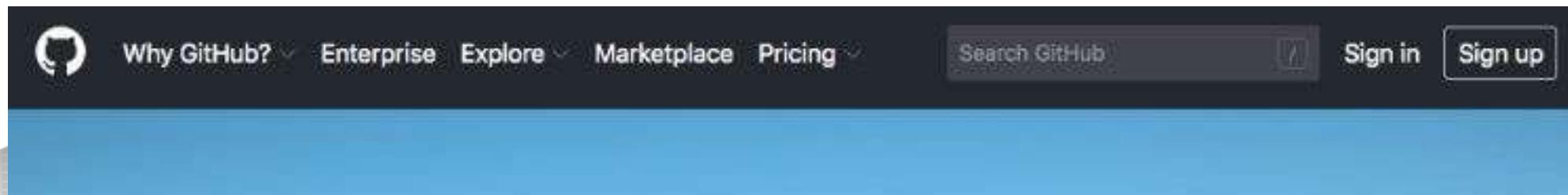
Cuando el protocolo y las partes del servidor(subdominio) no son colocadas, el browser interpreta la URL como relativa a la página actual.

`cgi-bin/datos.py`

Casos especiales

- No existe un archivo predefinido de inicio (`index.html`) muestra el contenido del directorio.
- Página de error.

Códigos de error



http://es.wikipedia.org/wiki/Anexo:C%C3%B3digos_de_estado_HTTP

<https://www.cheatography.com/kstep/cheat-sheets/http-status-codes/pdf/>

Find code, projects, and people on GitHub:

Peticiones HTTP o HTTPS

Una petición de envío o solicitud de información entre el navegador y el servidor web se hace usando GET o POST

PETICION

- Métodos de petición ([GET](#), [POST](#))
- Uniform Resource Identifier (URI) específicamente una (URL)
- Versión de HTTP que entiende el cliente
- Headers (información adicional respecto a la transferencia)



RESPUESTA

- Versión del protocolo y el código de estado
- Respuesta a los headers
- Respuesta al body

Tarea: Investigar la diferencia entre GET y POST

Aplicaciones Web

90-95 “Sitios Web”

- Páginas que tenían texto y gráficos con enlaces

Hoy en día..... “Aplicaciones Web” WebApps

- Capacidad de realizar lógica de la aplicación.
- Funciones para interacción con el usuario final
- Integradas con bases de datos
- Integradas con aplicaciones de negocios
- Múltiples tecnologías trabajando juntas
- Énfasis en las interfaces de usuario
- Arte y tecnología
- Tipos de App web : Estáticas, dinámicas, SPA, MPA, Animadas, CMS, Ecommerce y Portales
- Nuevas características de las Apps PWA, SaaS, Smart APPs, ...

Aplicaciones Web

Características Frecuentes

- Intensidad de red
- Carga impredecible
- Concurrencia
- Alto desempeño
- Disponibilidad (24/7/365)
- Gobernada por los datos
- Evolución continua (tecnologías, dispositivos, etc)
- Inmediatez
- Seguridad
- Facilidad de interacción con el usuario





Preguntas ?





Fecha de Exposiciones, Taller y Proyecto

Grupo	Tema	Fecha exposición
	Pruebas unitarias en Python y Django	Mayo 3 de 2023
	Estándares de codificación en Python y Django	Mayo 3 de 2023
	Taller Django	Mayo 10 de 2023
	Tablero Kanban	Mayo 17 de 2023
	Chrome DevTools	Mayo 17 de 2023
	Git y git flows (incluir git tags)	Mayo 24 de 2023
	*Web services con Django	Mayo 24 de 2023
	Primera entrega del proyecto	Mayo 31 de 2023
	Motion UI	Junio 7 de 2023
	*WAF (Web Applications Firewall)	Junio 7 de 2023
	Desarrollo de aplicaciones web usando ChatGPT	Junio 14 de 2023
	Entrega final del proyecto	Julio 28 de 2023

Equipos para el proyecto del curso y exposiciones:

Trabajo en equipos del proyecto del curso





Ítems que deberían tener en cuenta para la primera presentación de avance del proyecto

Fecha de entrega mayo 24 de 2023

1. Título del Sistema
2. Objetivos (principales funcionalidades solicitadas)
3. Empresas a las que le sirve el sistema
4. Idea del proyecto: Presenta un resumen de no más de 100 palabras que resumen de la idea del proyecto.
5. Fechas críticas: Define las fechas críticas de la aplicación tales como inicio y finalización del proyecto y fechas estimadas de los releases. (Entrega final del proyecto junio 28)
6. Miembros del equipo de desarrollo: El equipo se organiza a través de asignación de los 3 roles básicos de la metodología distribuidos entre los integrantes del grupo
7. Product backlog (o enlace a donde este): Backlog priorizado y estimado.
8. Prácticas ágiles a usar durante el desarrollo: Presenta un listado de las prácticas ágiles que acordaron usar durante el desarrollo.
9. Tecnologías a usar : Define mediante un listado todas las tecnologías de desarrollo de la aplicación
10. Diagrama de bloques(módulos de la aplicación): Presenta un diseño inicial de los módulos de la aplicación que representan el estilo arquitectural de la aplicación.
11. Modelo de datos (modelo relacional): Presenta un modelo de datos inicial.
12. Diseño de las interfaces de la aplicación (Interfaces del primer sprint)
13. Release plan: Presenta un release plan inicial donde presenta los sprints a realizar, HU de cada sprint, responsable de cada HU y fecha estimada de entrega de los sprints y releases del proyecto.
14. Detalles de la Historias de usuario : Presenta el detalle de las HU de al menos el primer sprint en un formato definido.
15. Acrónimos: Presenta una lista de los acrónimos usados con su respectiva definición (si los hay)
16. Primer Sprint del proyecto.

Desarrollo I

Economy of the
European Union

Gracias

