

INTRO-CS-1 - Introducción a la Informática (3 horas)

Descripción: En esta sesión, los participantes aprenderán cómo los protocolos HTTP y TCP/IP hacen que Internet funcione, y cómo podemos usar esa información para aprender conceptos futuros relacionados con el desarrollo de aplicaciones web.

Objetivos de aprendizaje:




Al final de la sesión, seré capaz de:

- Explicar los conceptos básicos sobre cómo funciona Internet.
- Explicar la diferencia entre un cliente y un servidor.
- Explicar el protocolo TCP/IP en un nivel básico.
- Explicar el protocolo HTTP.
- Explicar y utilizar los métodos HTTP GET / POST / PUT / DELETE.
- Usar herramientas de desarrollo.
- Describir la diferencia entre un sitio web y una aplicación web.
- Dar ejemplos de aplicaciones web y sitios web.

Puntos clave:

- Los clientes hacen solicitudes. Los servidores esperan para recibir solicitudes. Los clientes se conectan a servidores para realizar acciones tales como buscar una imagen.
- TCP/IP:
 - Las diferentes capas TCP/IP son:
 - Capa de acceso a la red (enlace de datos): permite el acceso a los recursos de Red
 - Capa de Internet: Proporciona transmisión fiable de datos
 - Capa de transporte: mueve el paquete del punto A al B
 - Capa de aplicación: Responsable de la conexión física
- Protocolo de Transferencia de HiperTexto (HTTP):
 - Cliente/Navegador envía una solicitud HTTP a la web
 - Un servidor web recibe la solicitud
 - El servidor procesa la solicitud
 - El servidor devuelve una respuesta HTTP al explorador
 - El cliente recibe y muestra la respuesta

<ul style="list-style-type: none"> • Métodos HTTP: <ul style="list-style-type: none"> ○ GET: Para recuperar información de un servidor mediante la URL ○ POST: para enviar datos a un servidor; es decir, cargas de formularios o archivos ○ PUT: reemplaza todas las representaciones de un recurso especificado por el contenido cargado ○ DELETE: Elimina la representación actual del recurso de destino • Herramientas de desarrollo: <ul style="list-style-type: none"> ○ Se utilizan para ver los archivos de origen e información sobre una página web ○ Pueden visualizar HTML, CSS, JavaScript ○ Pueden visualizar solicitudes/respuestas HTTP • Los sitios web se basan en contenido. Las aplicaciones web se centran en la interacción del usuario; es decir, Wikipedia vs Google Earth. 			
Habilidad de comportamiento	<ul style="list-style-type: none"> • Comunicación 	Mentalidad	<ul style="list-style-type: none"> • Mentalidad de crecimiento
Palabras clave	<ul style="list-style-type: none"> • Cliente • Servidor • TCP/IP • HTTP • PUT • POST • GET • DELETE • Método de solicitud • Aplicación Web • Inspeccionar Elemento 	Materiales	<ul style="list-style-type: none"> • Computadora • Proyector de video • Pizarrón • Google Chrome • Papel • Pluma <p>Lecturas y Recursos:</p> <ul style="list-style-type: none"> • Documentación del módulo de Introducción a la Informática
Secuencia de la sesión	<ul style="list-style-type: none"> • Primera sesión en el módulo INTRO-CS - Introducción a la Informática. 		
Objetivos de desempeño	<ul style="list-style-type: none"> • d. Aprender nuevas habilidades técnicas y no técnicas de forma proactiva para mejorar su desempeño e incrementar la probabilidad de una promoción. 		

Evaluación de la sesión	<ul style="list-style-type: none"> • Durante todas las Reflexiones, evalúa el conocimiento de los participantes haciendo preguntas y validando sus respuestas.
Íconos	 Actividad  Reflexión  Evaluación

Tiempo	Actividad
10 minutos	Apertura <ul style="list-style-type: none"> • Mini ejercicio • Reflexión
85 minutos	Contenido <ul style="list-style-type: none"> • Explicando a detalle el internet <ul style="list-style-type: none"> ◦ Práctica ◦ Reflexión • TCP/IP <ul style="list-style-type: none"> ◦ Reflexión • Protocolo de Transferencia de HiperTexto (HTTP) <ul style="list-style-type: none"> ◦ Práctica ◦ Reflexión • Métodos de solicitud HTTP <ul style="list-style-type: none"> ◦ Práctica ◦ Reflexión • Descubriendo las herramientas de desarrollo <ul style="list-style-type: none"> ◦ Práctica ◦ Práctica ◦ Reflexión
70 minutos	Práctica <ul style="list-style-type: none"> • Usando las herramientas como un desarrollador • Explorando el protocolo HTTP

	<p>* Nota al instructor: Si los ejercicios de práctica se realizan en otro momento y/o con otro instructor, continúa con la sección de cierre para finalizar la experiencia de aprendizaje.</p>
15 minutos	<p>Cierre</p> <ul style="list-style-type: none"> • Conexión con habilidades de comportamiento y mentalidades

APERTURA (10 minutos)

Nota al instructor: En esta sección se revisan los objetivos de aprendizaje de la sesión y se lleva a cabo una rápida actividad, seguida de una reflexión grupal.

- **Objetivos:** Revisa los objetivos y presenta un argumento sobre el valor de la sesión.

👉 Mini ejercicio

- **Da instrucciones:**

- ☐ **Pide** a los participantes que formen **grupos de 2 o 3**.
- ☐ **Da 5 minutos** para realizar el ejercicio.
- ☐ **Lee** el siguiente escenario:
 - ☐ *Se les pide que expliquen a sus abuelos cómo funciona Internet. Sus abuelos no conocen los conceptos básicos de la tecnología. ¿Cómo lo explicarían utilizando la mayoría (o todas) las palabras clave de esta sesión?*
 - ☐ **Muestra** las palabras clave en el pizarrón.
- ☐ **Recuerda** a los participantes que no pueden usar Internet para completar esta tarea.


🕒 Reflexión

- **Pide** a algunos grupos que presenten su definición de cómo funciona Internet al grupo.

CONTENIDO (85 minutos)

Nota al instructor: En esta sección los participantes aprenderán sobre qué es el Internet, TCP/IP, Protocolo de Transferencia de HiperTexto (HTTP), métodos de solicitud HTTP, y herramientas de desarrollo.

Explicando a detalle el Internet

Contenido	Reflexión
<ul style="list-style-type: none"> ● Explica el modelo cliente-servidor. Concéntrate en los siguientes conceptos: <ul style="list-style-type: none"> ○ Definiciones de cliente y servidor y las diferencias entre ellos. ○ La base de cómo funciona Internet. ○ Cómo Internet desglosa los datos en paquetes. <p> Práctica</p> <p>Nota al instructor: Deja abierta la sección sobre solicitudes HTTP para utilizar el mismo ejemplo.</p> <ul style="list-style-type: none"> ● Abre Google Chrome. ● Navega a 'Ver' > 'Desarrollador' > 'Inspeccionar Elemento'. ● Abre la pestaña Red. ● En la barra de direcciones, escribe la dirección URL de un sitio web, por ejemplo. Netflix.com. ● Haz clic en la primera solicitud con la etiqueta 'Netflix.com'. ● Lee los encabezados de solicitud. 	<p>Haz las siguientes preguntas. Comparte ^{TF}:</p> <ul style="list-style-type: none"> ● ¿Qué es Internet? <ul style="list-style-type: none"> ○ Respuesta: Internet es una red global de computadoras. Imaginen un sistema postal que entregue y reciba paquetes a velocidades extremadamente rápidas. Utiliza cosas como TCP/IP y HTTP para comunicarse. Cualquier persona con el permiso correcto puede obtener información de otra computadora. ● ¿Qué es un cliente? <ul style="list-style-type: none"> ○ Respuesta: Una pieza de hardware o software que accede a un servicio puesto a disposición por un servidor. ● ¿Qué es un servidor? <ul style="list-style-type: none"> ○ Respuesta: Proporciona datos a otras computadoras. Menciona LAN/WAN. Existen muchos tipos de servidores. ● ¿Qué es un paquete? <ul style="list-style-type: none"> ○ Respuesta: Los datos enviados a través de Internet se denominan mensajes. Antes de que se envíen los mensajes, se dividen en fragmentos llamados paquetes. Los paquetes se envían de forma independiente. ● ¿Cómo hacen los paquetes a Internet tal como la conocemos hoy en día? <ul style="list-style-type: none"> ○ Respuesta: <ul style="list-style-type: none"> ■ Cuando se desea acceder a un sitio web, se escribe la dirección del sitio web en la barra de direcciones. ■ La solicitud se divide en paquetes. ■ Cada paquete toma su propia ruta al servidor de destino.


	<ul style="list-style-type: none"> ■ El servidor lee la solicitud y proporciona la información, que también se divide en paquetes.
--	---

TCP/IP

Contenido	Reflexión
<ul style="list-style-type: none"> ● Explica el modelo TCP/IP. <ul style="list-style-type: none"> ○ Vea la imagen de abajo ● Explica las cuatro capas principales del protocolo: <ul style="list-style-type: none"> ○ Aplicación: <ul style="list-style-type: none"> ■ Permite el acceso a los recursos de red. ■ Interactúa con aplicaciones de software. ■ Por ejemplo, Iniciar la transferencia de archivos o enviar un correo electrónico. ○ Transporte: <ul style="list-style-type: none"> ■ Proporciona entrega confiable de mensajes y errores. ■ Determina cuántos datos se deben enviar a dónde y a qué velocidad. ■ Utiliza Handshakes para retransmitir una correcta transmisión de datos. ○ Internet: <ul style="list-style-type: none"> ■ Para mover paquetes de A a B. ■ Puede utilizar varias redes y nodos para llegar del punto A al punto B. ○ Acceso a la Red: <ul style="list-style-type: none"> ■ Responsable de la transmisión física entre 2 dispositivos en la misma red. 	<p>Haz las siguientes preguntas. Comparte ^{TF}:</p> <ul style="list-style-type: none"> ○ ¿Qué es un protocolo? <ul style="list-style-type: none"> ■ Respuesta: Un procedimiento/conjunto de reglas para la comunicación entre dispositivos. ○ ¿Qué significa TCP? <ul style="list-style-type: none"> ■ Respuesta: Protocolo de Control de Transferencia. ○ ¿Qué significa IP? <ul style="list-style-type: none"> ■ Respuesta: Protocolo de Internet. ○ ¿Qué es TCP/IP? <ul style="list-style-type: none"> ■ Respuesta: Determina cómo una computadora debe conectarse a Internet y transmitir datos.

Protocolo de Transferencia de HiperTexto (HTTP)

Contenido	🕒 Reflexión
<ul style="list-style-type: none"> ● Explica la descripción general del Protocolo HTTP. Concéntrate en las siguientes secciones: <ul style="list-style-type: none"> ○ Diagrama Fetch ○ Diagrama HTTP & Capas ○ Flujo HTTP ● Discute la diferencia entre un cliente y un servidor. ● Describe el aspecto de una solicitud HTTP: <ul style="list-style-type: none"> ○ Ejemplo: <code>GET/path/to/file/index.html HTTP/1.0</code> <div data-bbox="226 638 1150 1015" data-label="Diagram"> <pre> graph TD subgraph Request_Line [Request Line] direction LR M[GET] --- P[/] --- V[HTTP/1.1] end M --> Method P --> Path V --> Version[Version of the protocol] subgraph Headers_Box [Headers] direction TB H1[Host: developer.mozilla.org] H2[Accept-Language: fr] end Headers_Box --> Headers </pre> </div> <ul style="list-style-type: none"> ● Describe las partes de una respuesta HTTP. Concéntrate en: <ul style="list-style-type: none"> ○ Ejemplo: <code>HTTP/1.0 200 OK</code> ● Explica los códigos de respuesta HTTP. Concéntrate en: <ul style="list-style-type: none"> ○ 404 No encontrado - El recurso solicitado no existe. ○ 200 OK - (Normalmente no se ve por el usuario, porque se llega al sitio web) ○ 301 - Movido Permanentemente 	<p>Haz las siguientes preguntas.</p> <p>Llamada con aviso ^{TF}:</p> <ul style="list-style-type: none"> ● <i>¿Qué es HTTP?</i> <ul style="list-style-type: none"> ○ Respuesta: HTTP define cómo se formatean y transmiten los mensajes. ● <i>¿Cómo funciona el modelo cliente-servidor?</i> <ul style="list-style-type: none"> ○ Respuesta: Un cliente HTTP abre una conexión y envía un mensaje de solicitud a un servidor HTTP. Entonces el servidor devuelve un mensaje de respuesta, que generalmente contiene el recurso que se solicitó. Una vez que se abre la conexión, ésta permanece abierta para procesar más solicitudes. ● <i>¿En qué consiste la línea de solicitud inicial?</i> <ul style="list-style-type: none"> ○ Respuesta: Un nombre de método, ruta y versión http.

<ul style="list-style-type: none"> ○ 302 - Movido temporalmente ○ 500 - Error del servidor ● Explica esta información adicional si es necesario: <ul style="list-style-type: none"> ○ HTTP Encabezado de Solicitud <ul style="list-style-type: none"> ■ Se utiliza para pasar información adicional sobre la solicitud al servidor. ○ HTTP Cuerpo de Solicitud <ul style="list-style-type: none"> ■ Se utiliza para procesar información adicional requerida por el servidor. ■ Puede estar vacío <p> Práctica</p> <p>Notas al instructor: Utiliza esta práctica para responder también preguntas en la reflexión.</p> <ul style="list-style-type: none"> ● Vuelve a abrir las herramientas de desarrollo y realiza una solicitud para Netflix.com. ● Lee la información del encabezado. ● Compara con el contenido. ● Intenta otros sitios web como Amazon. 	
---	--

Métodos de solicitud HTTP

Contenido	🕒 Reflexión
<ul style="list-style-type: none"> ● Explicar los 4 tipos principales de solicitudes del protocolo: <ul style="list-style-type: none"> ○ GET 	<p>Haz las siguientes preguntas. Llamada con aviso ^{TF}:</p>

- **POST**
- **PUT**
- **DELETE**

👉 Práctica

Nota al instructor: Si los ejercicios de práctica se realizan en otro momento y/o con otro instructor, continúa con la sección de cierre para finalizar la experiencia de aprendizaje.

- **Pide** a los participantes que levanten la mano si creen que los siguientes escenarios son una solicitud HTTP GET/PUT/POST/DELETE. Con cada respuesta, **pregunta por qué**.

Escenarios:

- *Escenario 1: Alex está usando YouTube, viendo tutoriales sobre cómo crear un sitio web. Hace clic en "Me gusta" en su video favorito. ¿Qué tipo de solicitud HTTP recibiría al hacer clic en "Me gusta"?*
 - **Respuesta:** POST
- *Escenario 2: Tu colega te envía material de lectura para ayudarte a programar. Es un enlace para apilar el desborde. ¿Qué tipo de solicitud HTTP recibiría al hacer clic en el enlace enviado?*
 - **Respuesta:** GET
- *Escenario 3: Estás escribiendo una nueva publicación de Facebook. ¿Qué tipo de solicitud HTTP recibiría al presionar enviar?*
 - **Respuesta:** POST
- *Escenario 4: Cometiste un error en tu publicación original y decidiste editarlo. ¿Qué tipo de solicitud HTTP recibiría?*
 - **Respuesta:** PUT
- *Escenario 5: Estás en medio del desarrollo de un proyecto java que consume mucho tiempo. Te das cuenta que has cargado accidentalmente el fragmento*

- *Define GET en tus propias palabras*
 - **Respuesta:** Tipo de solicitud más común. Se utiliza para recuperar datos de un servidor web.
- *¿Cuál es la diferencia entre POST y PUT?*
 - **Respuesta:** POST se utiliza principalmente cuando no se conoce la URL específica para colocar la información (por ejemplo, crear un nuevo hilo en un foro; diferente de PUT donde conocería el destino, por ejemplo, editar un comentario publicado anteriormente).
- *Define DELETE con tus propias palabras.*
 - **Posible respuesta:** quita el recurso seleccionado actual de la dirección URL de destino.

de código incorrecto en el repositorio, por lo que lo eliminas. ¿Qué tipo de solicitud HTTP recibiría?

- **Respuesta:** DELETE

- *Escenario 6: "¡GANADOR, GANADOR HAGA CLIC AQUÍ!" Tu correo electrónico ha sido congestionado con correo no deseado. Tu cliente de correo electrónico tiene un botón de "limpiar". Este botón está diseñado para eliminar el correo no deseado. ¿Qué tipo de solicitud HTTP recibiría al hacer clic en "limpiar"?*

- **Respuesta:** DELETE

- *Escenario 7: Recibes un nuevo mensaje en Facebook de tu primo lejano sobre su visita a tu país de residencia. ¿Qué tipo de solicitud HTTP recibiría si abre el mensaje?*

- **Respuesta:** GET

- *Escenario 8: Has tenido un error de programación difícil. Has visitado muchos foros en busca de una respuesta, pero te cuesta encontrar algo útil. En última instancia, decides pedir ayuda en varios foros mediante la creación de temas con tu pregunta y un fragmento del código. Al crear un tema, ¿qué tipo de solicitud HTTP sería esa acción?*

- **Respuesta:** PUT

Descubriendo las herramientas de desarrollo

Contenido	🕒 Reflexión
<p>Nota al instructor: En este plan de la sesión, el navegador que se está utilizando es Google Chrome. Si prefieres otro navegador, no dudes en adaptar la sesión.</p> <ul style="list-style-type: none"> ● Explica los conceptos principales de Herramientas de Desarrollo. Concéntrate en lo siguiente: <ul style="list-style-type: none"> ○ ¿Qué es una herramienta de desarrollo? <ul style="list-style-type: none"> ■ En cuanto a los navegadores web: Safari, Chrome, Firefox ○ Para acceder al modo de desarrollador: <ul style="list-style-type: none"> ■ Configuración > Más herramientas > Herramientas de desarrollo ● Disecciona las partes de la vista a continuación. Concéntrate en lo siguiente: <ul style="list-style-type: none"> ○ Barra de herramientas <ul style="list-style-type: none"> ■ Elementos ■ Consola ■ Fuentes ○ Visualizador <ul style="list-style-type: none"> ■ Visualizador HTML ■ Wrappers ■ Visualizador CSS ■ Vista de posicionamiento de elementos ● Explica las diferencias entre un sitio web y una aplicación web. ● Muestra algunos ejemplos de cada uno de ellos a medida que los comentas: <ul style="list-style-type: none"> ○ Sitio web. <ul style="list-style-type: none"> ■ Wikipedia ■ CNN 	<p>Haz las siguientes preguntas. Comparte ^{TF}:</p> <ul style="list-style-type: none"> ○ <i>¿Qué hace el Origen de Página en las herramientas para desarrollo?</i> <ul style="list-style-type: none"> ■ Respuesta: Donde puedes ver todos los archivos de origen de la página web que estás viendo. ○ <i>¿Qué hace el Inspector Web?</i> <ul style="list-style-type: none"> ■ Posible respuesta: Inspector Web es tu centro de comando. Te ayuda a inspeccionar todos los recursos y la actividad en una página web, lo que hace que el desarrollo sea más eficiente. ○ <i>¿Qué es lo que es más probable que encuentres en la consola?</i> <ul style="list-style-type: none"> ■ Respuesta: Cualquier salida basada en JavaScript, incluidos los códigos de error y las advertencias. ○ <i>¿Qué es un sitio web?</i> <ul style="list-style-type: none"> ■ Respuesta: Generalmente informativo, y no hay mucha interacción con el usuario. ○ <i>¿Qué es una aplicación web?</i>

- [Reddit](#)
- Aplicación Web
 - [Facebook](#)
 - [Pinterest](#)
 - [Gmail](#)
- **Ejemplo:** En comparación con un sitio web, ustedes tienen un teléfono inteligente que puede tener diferentes aplicaciones: la reproducción de música, vídeos, imágenes, juegos, etc.

👉 Práctica

- **Elige** 2 o 3 sitios web y aplicaciones web.
- **Nombra individualmente** cada sitio web o aplicación web seleccionado.
- **Pide** a los participantes que pongan los pulgares arriba si nombras una aplicación web y los pulgares abajo si nombras un sitio web. Ejemplos:
 - [Wikipedia](#)
 - [Google Maps](#)
- **Discute** cualquier discrepancia que tengan los participantes.

👉 Práctica

Nota al instructor: Anima a los participantes a hacer esto en sus computadoras para que tengan la libertad de explorar la página.

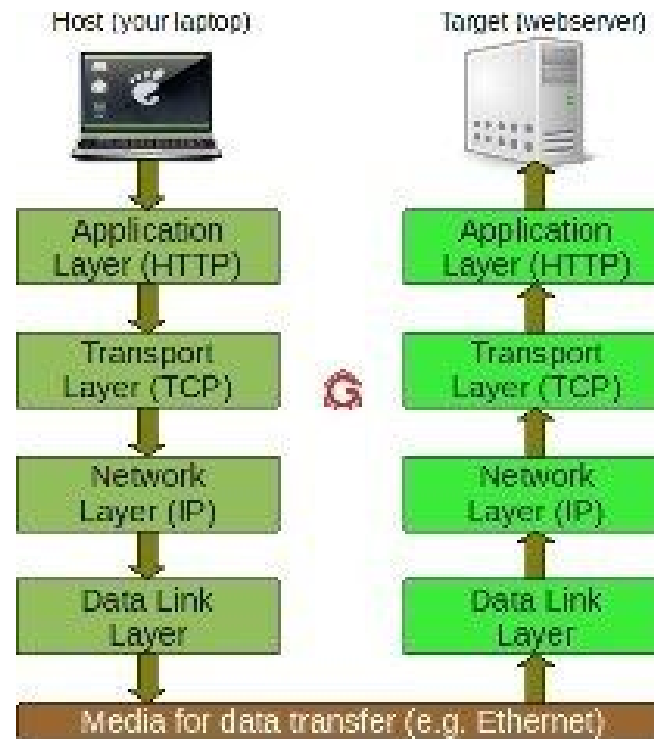
- **Utiliza** las herramientas de desarrollo para examinar los archivos de origen de las páginas web mencionadas en el último ejercicio.
- **Pregunta:** *¿Cuáles son las principales diferencias entre la fuente de las páginas web?*
 - **Respuesta:** Los sitios web tienden a tener más similitudes en los archivos de origen en comparación con las páginas web.

- **Respuesta:** Incluye algún tipo de interacción con el usuario. Por lo general, permite a los usuarios interactuar con esa información.
- Ejemplo: Es posible que tu biblioteca local tenga un sitio web que te permita iniciar sesión para ver qué libros has pedido prestados.
- **Ve** esta discusión sobre [Sitio web vs Aplicación web.](#)

Recursos de imagen

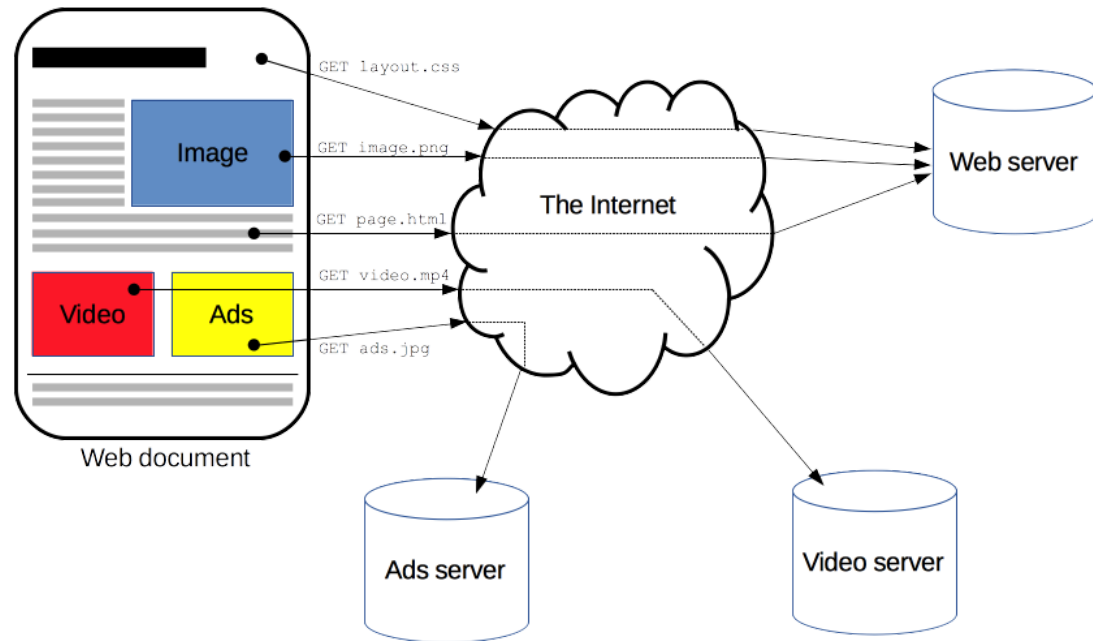
1. Capas TCP/IP

[Enlace a imagen](#)



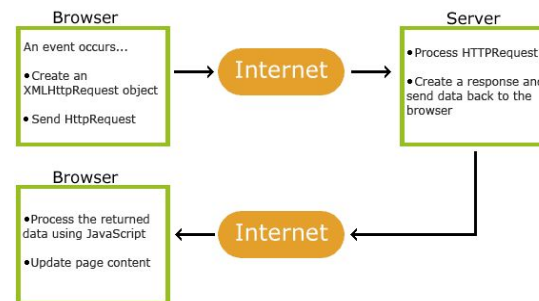
2. Obtener un diagrama de la página

[Enlace a imagen](#)



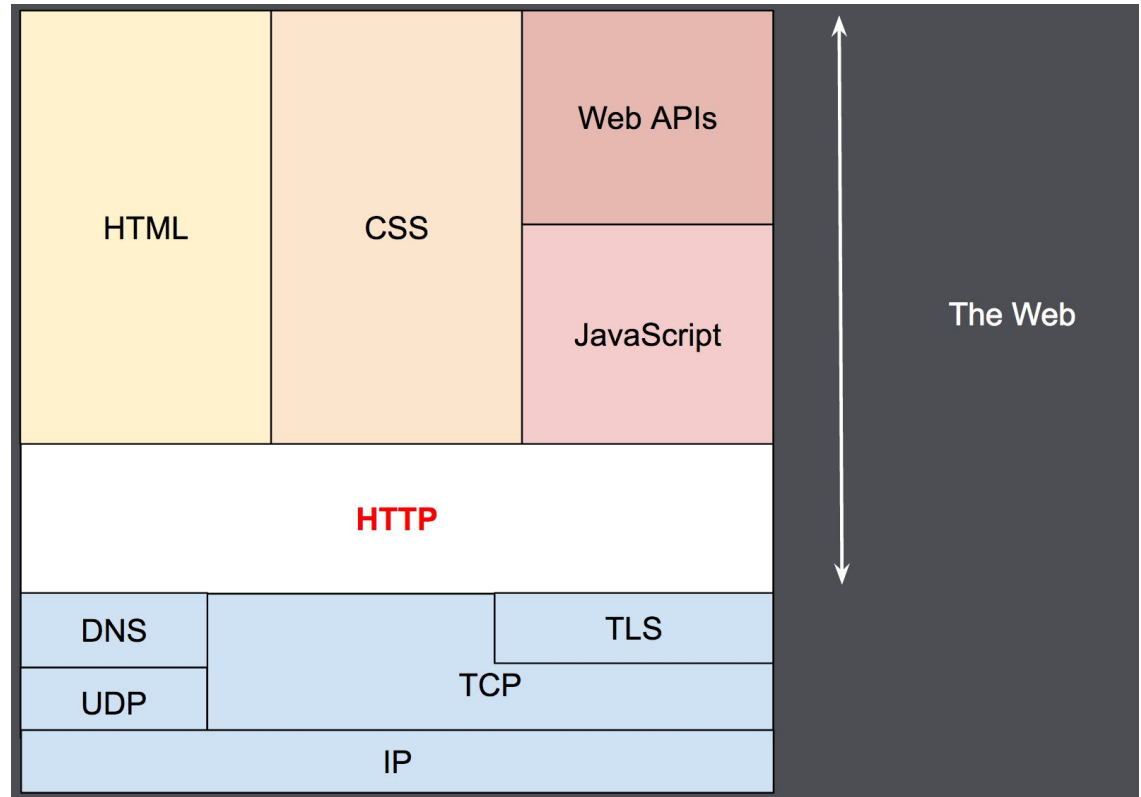
3. HTTP & capas

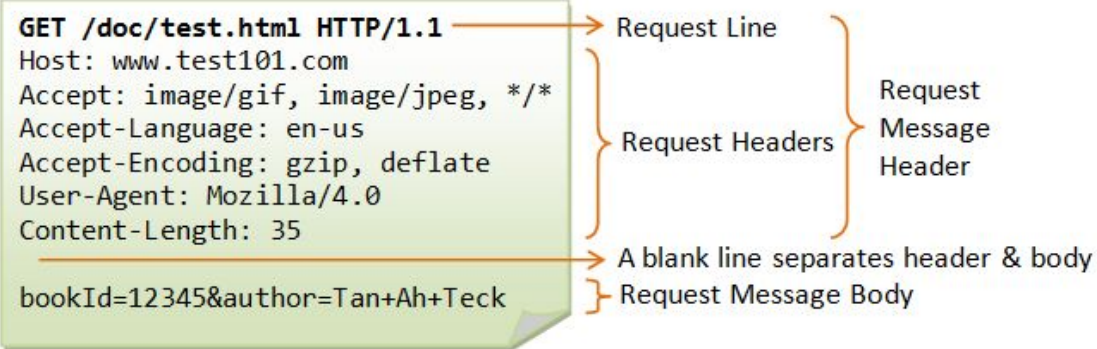
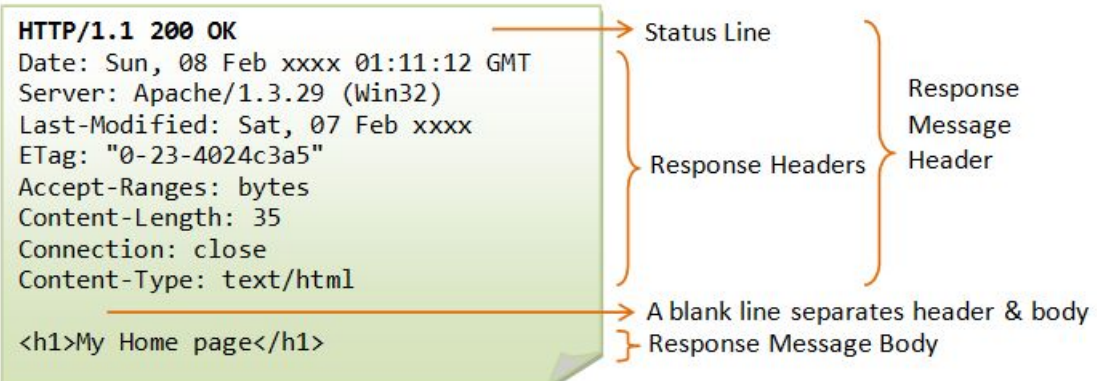
[Enlace a imagen](#)



4. Flujo HTTP

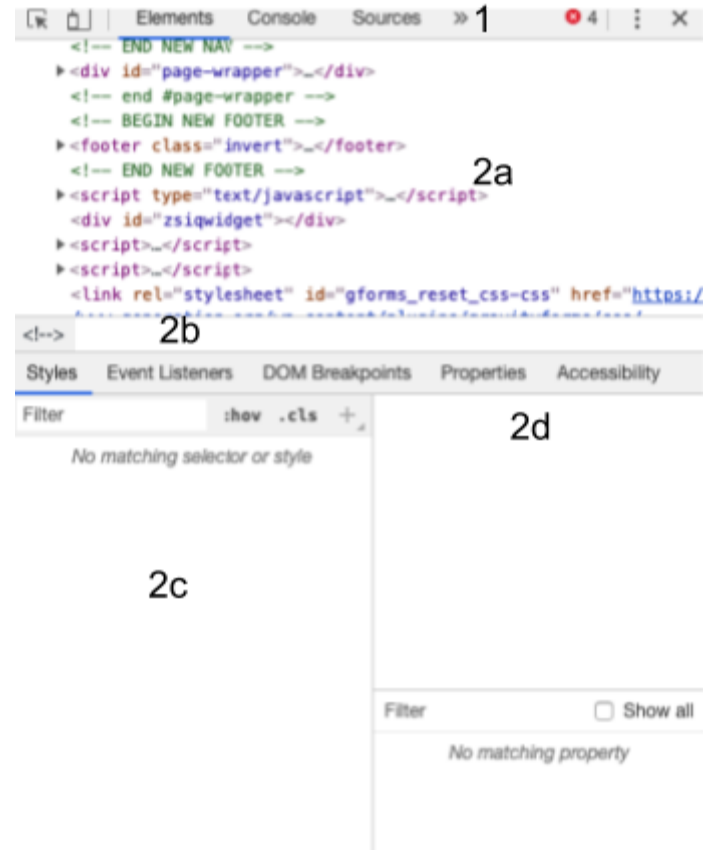
[Enlace a imagen](#)



<p>5. HTTP Mensaje - Solicitud</p> <p>Enlace a imagen</p>	 <pre> GET /doc/test.html HTTP/1.1 Host: www.test101.com Accept: image/gif, image/jpeg, */* Accept-Language: en-us Accept-Encoding: gzip, deflate User-Agent: Mozilla/4.0 Content-Length: 35 bookId=12345&author=Tan+Ah+Teck </pre> <p>Request Line</p> <p>Request Headers</p> <p>Request Message Header</p> <p>A blank line separates header & body</p> <p>Request Message Body</p>
<p>6. HTTP Mensaje - Respuesta</p> <p>Enlace a imagen</p>	 <pre> HTTP/1.1 200 OK Date: Sun, 08 Feb xxxx 01:11:12 GMT Server: Apache/1.3.29 (Win32) Last-Modified: Sat, 07 Feb xxxx ETag: "0-23-4024c3a5" Accept-Ranges: bytes Content-Length: 35 Connection: close Content-Type: text/html <h1>My Home page</h1> </pre> <p>Status Line</p> <p>Response Headers</p> <p>Response Message Header</p> <p>A blank line separates header & body</p> <p>Response Message Body</p>

7. Herramientas de desarrollo

[Enlace a imagen](#)



8. HTTP Encabezado/Cuerpo

[Enlace a imagen](#)

```
-----Request-----
Request Line: GET /utilities/weatherfull/city/hyderabad HTTP/1.1
Request Method: GET
Request Time: 2017-08-27 13:30:30
Accept-Encoding: gzip, deflate
Host address: restapi.demoqa.com
Client IP: 84.245.10.101
Client Port: 42092
HTTP Protocol Version: HTTP/1.1
Connection: close
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.101 Safari/537.36
Accept-Language: en-GB,en-US;q=0.8,en;q=0.6
Request body:
```

9. Ejemplo imagen 1

[Enlace a imagen](#)

Welcome to the BBC

Wednesday, 26 February



LIVE Generation purchase world class
Real Madrid for a "Small" Fee"

EUROPEAN FOOTBALL



London firms send
staff home amid
coronavirus fear

BUSINESS



Snow and ice to bring
UK-wide travel
disruption

UK

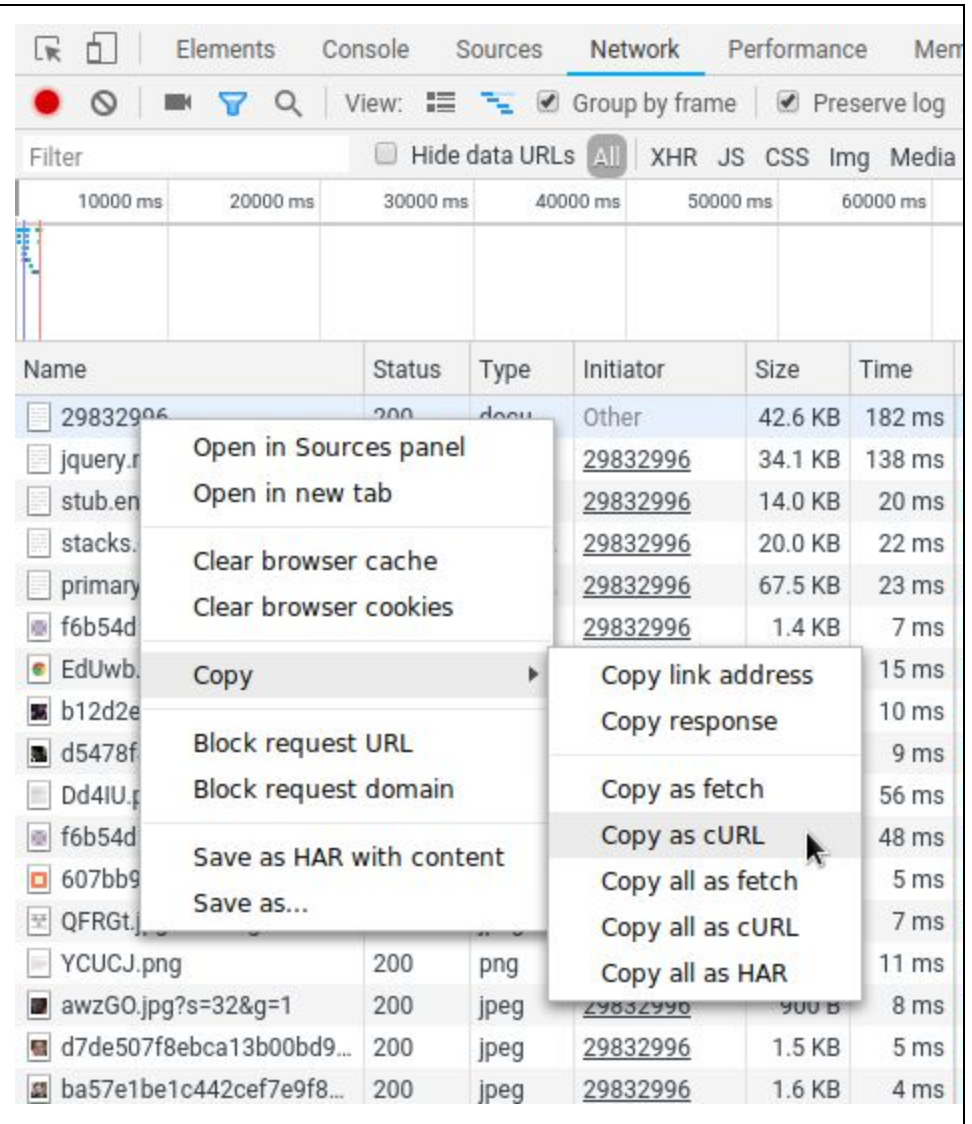
10. Ejemplo imagen 2

[Enlace a imagen](#)

```
▶ <div class="top-story__image-wrapper" data-reactid=".1h9zfrfycsg.0.1.0.$hp-ts-0.0">_</div>
▼ <div class="top-story__content" data-reactid=".1h9zfrfycsg.0.1.0.$hp-ts-0.1">
  ▼ <h3 class="gel-double-pica-bold top-story__title-container" data-reactid=".1h9zfrfycsg.0.1.0.$hp-ts-0.1.0">
    ▶ <span class="hp-live-badge top-story__live-badge hp-badge--blue" data-reactid=".1h9zfrfycsg.0.1.0.$hp-ts-0.1.0.0">_</span>
      ▶ <span class="top-story__title" data-reactid=".1h9zfrfycsg.0.1.0.$hp-ts-0.1.0.1">Generation
        purchase world class Real Madrid for a "Small" Fee"</span> = $0
      ▶ <span class="hp-accessibility-label" data-reactid=".1h9zfrfycsg.0.1.0.$hp-ts-0.1.0.2">. Live
        now.</span>
      </h3>
    </div>
    ▶ <span class="gel-minion top-story__attr-container" aria-labelledby="hp-ts-0-collection-uk-hero-promos-with-ck-and-nations-overrides" data-reactid=".1h9zfrfycsg.0.1.0.$hp-ts-0.2">_</span>
```

11. Ejemplo imagen 3

[Enlace a imagen](#)



PRÁCTICA (70 minutos)

Nota al instructor: Es posible que estas opciones no estén disponibles en otras herramientas para desarrollo del navegador. Consulta esta documentación sobre las [herramientas de desarrollo de Google Chrome](#).

☞ Usando las herramientas como un desarrollador

- **Di** a los participantes que ahora explorarán las herramientas para desarrollo de sus sitios web favoritos.
- **Da instrucciones:**

- ☐ **Abre** Google Chrome.
- ☐ **Navega** a tu sitio web favorito, por ejemplo, Generation.org.
- ☐ **Abre** las herramientas de desarrollo.
- ☐ **Explora** el sitio web con el botón derecho > 'Inspeccionar Elemento'.
 - ☐ **Visita** Fuentes y Consola también.
 - ☐ **Ve** la consola al hacer clic en los botones.
- ☐ **Encuentra** un ejemplo de cada método de solicitud en el panel de red (GET/PUT/DELETE/POST).
 - ☐ **Ve** a las comillas angulares (>>) en la barra de herramientas.
 - ☐ **Haz clic** en Red.
- ☐ **Utiliza** el panel Aplicación para ver todos los recursos que se cargan actualmente en la página.
 - ☐ **Ve** a las comillas angulares (>>) en la barra de herramientas.
 - ☐ **Haz clic** en Aplicación.
- ☐ **Dirige** a los participantes a pasar a las dos extensiones que se enumeran a continuación si tienen tiempo adicional:
 - ☐ **Extensión 1:** **Intenta** editar el texto en la página web para mostrar algo genial.
 - ☐ Ve la [imagen de ejemplo 1](#) y la [imagen de ejemplo 2](#)
 - ☐ **Extensión 2:**
 - ☐ **Realiza** otra solicitud HTTP mediante cURL.
 - ☐ **Ve** al panel de red.
 - ☐ **Haz clic** en el nombre de la solicitud.
 - ☐ **Selecciona** Copiar > Copiar como cURL.
 - ☐ **Pega** en la consola.
 - ☐ **Ve** la [imagen de ejemplo 3](#).

🕒 Reflexión

- Haz las siguientes preguntas. **Llamada con aviso** ^{TF}:
 - ¿Cuál es la forma más fácil de acceder a las herramientas para desarrollo?
 - **Respuesta:**
 - Hacer clic con el botón derecho en > Inspeccionar elemento.
 - Herramientas > Herramientas de desarrollo.
 - ¿Dónde puedes acceder a los métodos HTTP que usan los sitios?
 - **Respuesta:** El panel de red/aplicación.

👁 Explorando el protocolo HTTP

- Da instrucciones:

- ☐ **Abre** la herramienta de inspector del navegador.
- ☐ **Revisa** todos los elementos presentados en esta sesión con los participantes.
- ☐ **Explica** [cómo usar Postman](#).
- ☐ **Explora** el protocolo usando Postman.
 - ☐ **Instala** Postman (si aún no está instalado) en getpostman.com.
 - ☐ **Envía** algunos comandos a los sitios web y examina los encabezados.
 - ☐ **Haz** que los participantes noten el contenido de los encabezados de solicitud (páginas HTML).
 - ☐ **Realiza una** solicitud HTTP POST utilizando Postman a la siguiente API:
 - ☐ <http://dummy.restapiexample.com/create>
 - ☐ **Realiza una** solicitud HTTP GET con Postman y comprueba que se muestran los datos que enviaste mediante POST:
 - ☐ <http://dummy.restapiexample.com/employees>.

🕒 Reflexión

- Haz las siguientes preguntas. **Llamada con aviso** ^{TF}:
 - ¿Cuál es el método HTTP más utilizado y cuál es su función?
 - **Respuesta:** El método GET, que recupera información de un sitio web/terminal en particular.
 - ¿Cuál es la diferencia entre los métodos GET y POST?
 - **Respuesta:** GET se utiliza para recuperar datos y POST se utiliza para enviar datos.
 - ¿Cuál es el código de estatus HTTP cuando la solicitud se realiza correctamente?

- **Respuesta:** 200
- ¿Qué códigos de estatus HTTP devolverá una aplicación web cuando se produce un error en una solicitud?
 - **Respuesta:** Una solicitud fallida devolverá el código de estatus 404 No encontrado o 500 Error interno del servidor.

CIERRE (15 minutos)

Nota al instructor: En esta sección, los participantes hacen un repaso de lo aprendido en la sesión y una conexión con las habilidades de comportamiento y mentalidades.

Revisa puntos clave	
<ul style="list-style-type: none"> • Enlace a puntos clave 	
Preguntas de repaso	
¿Cuál es la diferencia entre GET y POST?	Ver la tabla de comparación
¿Por qué es relevante aprender sobre HTTP e Internet para un desarrollador de Java Junior?	Como desarrollador de software Junior, trabajarán con muchas arquitecturas diferentes. Algunas arquitecturas pueden incluir alojamiento de servidores, administración de entrega o mantenimiento de sitios web.
Conexión con habilidades de comportamiento y mentalidades	
<p><i>La compañía de Sonia está retrasada con una aplicación web que están construyendo y realmente necesitan a alguien que tenga experiencia con los servicios web. Sonia conoce y entiende completamente HTTP, pero nunca ha hecho ningún trabajo en la industria que involucra HTTP.</i></p>	
¿Por qué una mentalidad de crecimiento es útil en este escenario?	Una Mentalidad de Crecimiento ayudará a Sonia a ver la utilidad en el aprendizaje de los servicios web, no sólo por el bien de la empresa, sino también para su propia progresión como programadora. Ayudará a mantener a Sonia aprendiendo y hacer feliz a su jefe.

<p><i>¿Cómo puede ayudar la comunicación?</i></p>	<p>La comunicación ayudará a los gerentes de Sonia a saber que tienen opciones disponibles. Sonia será capaz de mostrar sus capacidades y la voluntad de ayudar que al final acelerará el aprendizaje y la progresión de Sonia en la compañía. Esto impide que Sonia se sume al proyecto a ciegas, sin la ayuda de sus colegas.</p>
---	---

-- Fin de la sesión --