Escribimos el mail XD

Apretamos enter para enviar el mail.

2.1. Al apretar enter estamos enviando una señal eléctrica desde el teclado

2.2. Esta señal es enviada a la motherboard, la motherboard la envia al CPU

2.3. La CPU recibe la señal del teclado y la envia al sistema operativo

2.4. El sistema operativo (a través de drivers, que son softwares que interpretan las señales eléctricas de los periféricos) interpreta la señal eléctrica y la envía al navegador (pues sabe que nosotros queriamos enviarla ahí por donde apretamos el enter (mientras estabamos en el navegador)

2.5. Nosotros cuando estamos en una pestaña de chrome en realidad estamos en un html, un html posee uno o varios script, y ahí un script que esta preparado para hacer algo al momento de pulsar la tecla enter (al momento que ocurra el evento “Pulsar tecla enter”), el cual va a hacer pues hemos apretado a tecla enter.

2.6. Javascript entiende que al ocurrir el evento enter tiene que hacer algo con el mail que hemos escrito.

2.6.1. AJAX permite enviar datos al servidor (el mail en este caso) y que la pagina no se recargue. AJAX es un API (interfaz de programación), que, entre sus multiples funciones, encapsula los datos que deseamos enviar de una manera que luego el servidor comprendera. La forma de encapsular los datos se denomina formato. AJAX encapsula el mail en un formato llamado JSON

2.6.2. AJAX envía el mail en formato JSON a través del protocolo REST (es un conjunto de reglas de como se envían los datos a un servidor) a través de HTTP (esta incluido en REST).

2.6.3. HTTP y HTTPS son protocolos de transferencia de texto. HTTPS es más seguro porque permite que solo al servidor al cual estamos enviando los datos pueda interpretar la información, solo el receptor final podrá ver los datos que enviamos. De esta manera se evita que alguien se pueda meter en el medio de al transferencia y ver los datos. Los datos se envían a una dominio DNS, que es la dirección de un servidor

2.7. El servidor recibe los datos en forma de señales eléctricas, el sistema operativo del servidor (en general linux(muerte al capitalismo 😛 )) a través de ethernet, transforma los datos que están encapsulados según el protocolo TCP/IP. Este protocolo luego crea los datos hacia el protocolo HTTP, HTTP es recibido por un servidor de HTTP de linux.

2.8. El servidor HTTP ahora permite que los programas en el backend procesen la información recibida según corresponda. Los datos procesados son guardados en una base de datos para luego poder acceder a estos datos.

2.9. El mail ahora si es enviado al dominio que sigue al @.

2.10. El mail ahora va al servidor de mail del dominio que se envía, este lo envía a la base de datos del mail que corresponde (el que esta previo al @), es decir va a la bandeja del receptor.

2.11. La bandeja del receptor recibe una notificación

2.12. La bandeja envía una notificación a un servidor de notificaciones, el servidor de notificación avisa a nuestro celular que se ha recibido un mail.

El mail fue envía y el receptor fue notificado a su celular.

Cuando los datos viajan por la red para identificar el proceso existe lo que es llamado el modelo OSI que consta de 7 capas:

1.-Fisica

2.-Enlace de Datos

3.-Red

4.-Transporte

5.-Sesión

6.-Presentación

7.-Aplicación

Al realizar la comunicación los datos van subiendo o bajando las capas que los van encapsulando en el protocolo según sea el caso.

La Capa 4 de Transporte utiliza los protocolos TCP y UDP en el que la diferencia es que TCP va orientado a la conexión confiable y la integridad de los datos.

Cuando mandas un correo o una imagen viaja por TCP porque tienen que llegar los bits completos, si algo se pierde en el camino, se vuelve a reenviar el datagrama.

Cuando realizas una videollamada o ves un video en Youtube los datos viajan por UDP ya que si se pierden datos en el camino son casi imperceptibles y estar reenviando los datos por cada error seria mas notable al verse lento un video.

AJAX: Asynchronous Javascript And XML

JSON: Javascript Object Notation

REST: Representational State Transfer

HTTP: Hypertext Transfer Protocol

FTP: File Transfer Protocol

SSH: Secure Shell

DNS: Domain Name System

TCP: Transmission Control Protocol

IP: Internet Protocol

SMTP: Simple Mail Transfer Protocol

POP3: Post Office Protocol

UDP: User Datagram Protocol

SoC: System on a Chip

ASCII: American Standard Code for Information Interchange