1. Describa e ilustre la arquitectura cliente-servidor de una de las principales aplicaciones de Internet (Web, email o netnews).

* A continuación se detalla como funciona el sistema de commits de git, y su posterior despliegue a un repositorio remoto, es decir, con arquitectura cliente-servidor.



1. Dé dos ejemplos de aplicaciones donde sea beneficioso emplear código móvil.

* 1º Ejemplo: aplicaciones JavaScript ejecutadas en cliente para establecer carga dinámica de datos provenientes de un servidor, como por ejemplo, hace cualquier pagina de deportes para retransmitir en directo vía web un partido de fútbol importante ([www.marca.es](http://www.marca.es/), [www.as.es](http://www.as.es/) y un largo etc).
* 2º Ejemplo: Algoritmos de minado de bitcoins, donde al usar un servicio online, el usuario acepta que en su maquina, mientras usa ese servicio, compute esos datos para beneficio del dueño del servidor que ofrece el servicio que usa el cliente. Generalmente, suelen estar implementados en Applets de Java o bien, en lenguajes interpretados de cliente como JavaScript o TypeScript.

1. Dé dos ejemplos de modelos de fallos que encuentre en Internet.

* 1º Ejemplo: El código QR que usa la compañía Facebook en su aplicación móvil WhatsApp para usar el servicio de forma web. Este código, pasado un tiempo, si no es usado, se invalida, obligando al usuario a obtener otro nuevo. En este caso, sería un fallo por omisión.
* 2º Ejemplo: Los fallos que daba al principio la nube Azure de Microsoft. Esto era porque sus servidores era distribuidos, pero de forma de jerárquica. Es decir, si se pedía información que no estaba en los servidores mas bajos en la jerarquía, se accedía a un servidor central principal. Como al principio el sistema de caché de la información no estaba muy bien pensada, el servidor principal acababa saturándose, cayendo todo el sistema en picado. Esto es claramente un fallo de proceso.