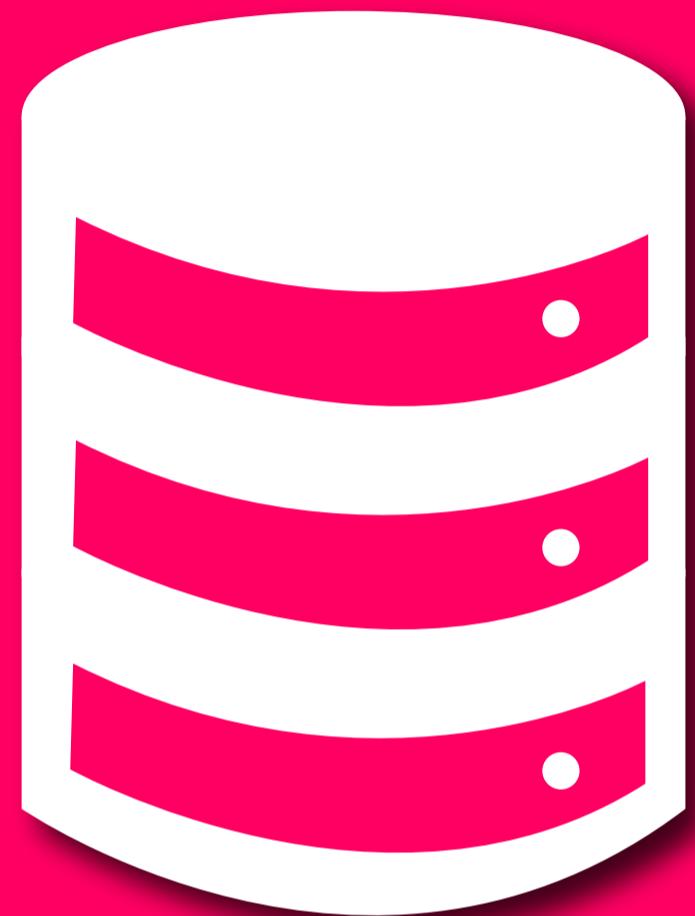
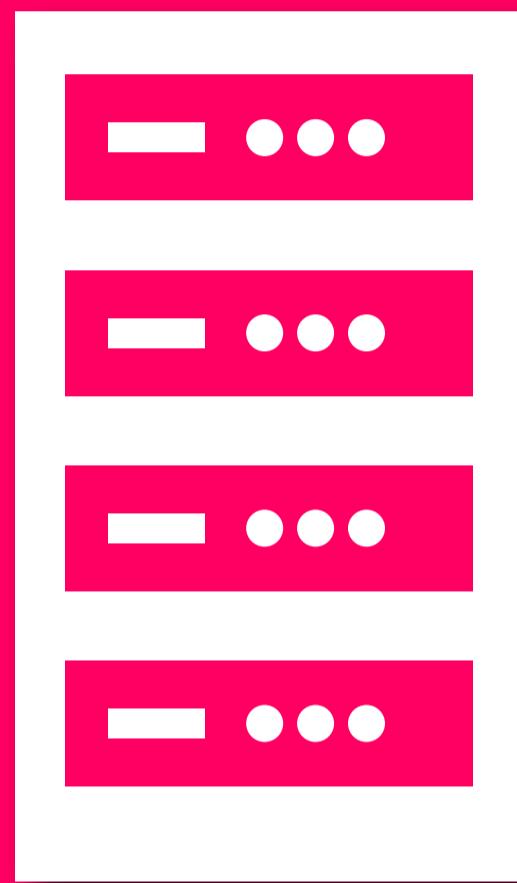
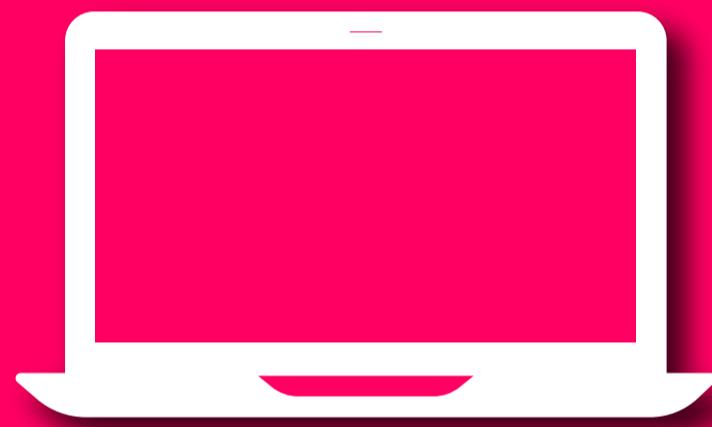




El futuro digital
es de todos

MinTIC



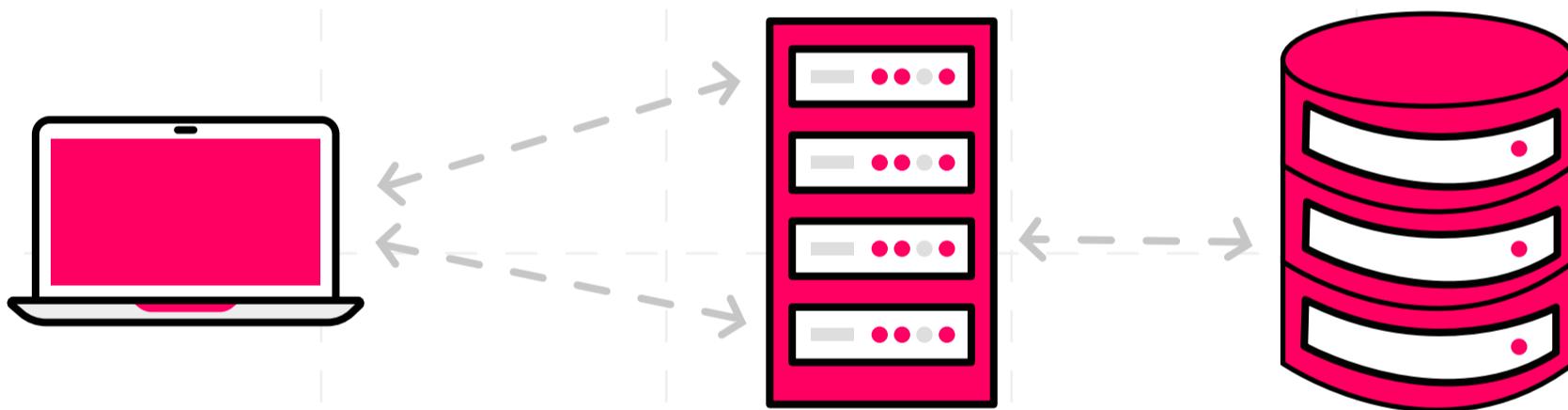
Aplicaciones web: Estructura general



Universidad de Caldas

Hola:

Para entender adecuadamente la estructura o arquitectura general de una aplicación web es necesario recordar como es el protocolo de comunicación con el cliente, que para este propósito, será un navegador web en un dispositivo computacional y el servidor que aloja los recursos que se quiere acceder, es decir, el que contiene la aplicación web. Este proceso se da cuando un usuario hace una petición al servidor a través del dispositivo computacional. Es muy probable que el servidor necesite acceder a algunos recursos (documentos, videos, imágenes, datos persistentes) que se encuentran almacenados en una base de datos para luego retornar la respuesta.



En el desarrollo de aplicaciones web la estructura general está dividida en 3 partes: la primera, se llama *frontend*, que hace referencia a la interacción directa con el usuario final, es decir, se encarga de presentar los formularios web para capturar y mostrar diferente tipo de información al usuario. La segunda, es el *backend*, que es el lugar donde se lleva a cabo todo el procesamiento relacionado con la lógica del negocio y también crea el acceso a los recursos que están en el servidor. Por último, cualquier aplicación con un mínimo de utilidad necesita un acceso a un sistema manejador de bases de datos para poder registrar la información persistente de la aplicación. A esta estructura se le conoce como *fullstack*.

Frontend

Interacción directa con el usuario



Bases de datos

Información persistente



01

02

03

Backend

Lógica del sistema, acceso a recursos



En el siguiente ejemplo hay una persona que atiende en un mostrador de un restaurante. Este sería el *frontend* del servicio del restaurante. Allí llega una persona que hace un pedido, este cliente sería el equivalente al usuario de la aplicación web. Este cliente está haciendo un pedido de capuccino con baileys y una hamburguesa doble carne con queso extra. La persona que atiende (*frontend*) y pregunta si desea el café fuerte o suave, es decir, existe una interacción entre el usuario y el frontend. Por último, luego de definir el pedido, el empleado hace una solicitud a la cocina (*backend*) para que准备 el plato que hace las veces del recurso solicitado. Es importante resaltar que el usuario no visualiza el proceso de preparación.

En la cocina se dispone del *backend*, que es donde se reciben las solicitudes realizadas por parte del empleado. El *backend* es el lugar en donde se tienen los recursos necesarios y se realiza el proceso de elaboración de los platos para satisfacer los diferentes pedidos. Entre los recursos se tiene: las recetas, los ingredientes y el chef. Una vez se termina la preparación del plato solicitado se le entrega al empleado que está en la barra (respuesta) para que este se lo entregue al usuario (cliente del restaurante) y así con esta comunicación *backend-frontend*, termina el proceso.

Estas capas de la estructura de aplicaciones web tienen un grupo de lenguajes y *frameworks* bastante específicos, lo que ha permitido que incluso los desarrolladores de software se especialicen en cada una de ellas, por ejemplo, el *frontend developer*, que utiliza los lenguajes estándar HTML 5 para el manejo del contenido de la página web, las hojas de estilo CSS para la parte visual y para los procesos se utiliza Javascript. Toda esta capas del *frontend* se conocen normalmente como desarrollo del lado del cliente.



En el *backend* el menú es inmenso, se puede encontrar una gran variedad de lenguajes, entre los que están: Node js, php, java, go, C++ Swift y Asp.Net. Y para el manejo de la persistencia están los sistemas que manejan bases de datos: Mongo DB, Mysql, postgressSQL y SQL Server. Además, es común utilizar *frameworks* que facilitan el proceso de desarrollo, por ejemplo: React Angular y Bootstrap para la parte del frontend. En el curso se empleará Asp.Net y Blazors, que es un *framework* para el *frontend* y el *backend* y, para persistencia se usará SQL Server.

En conclusión, se ha partido de la comunicación entre el cliente (navegador) y el servidor, mediante el protocolo sencillo de solicitud y respuesta. Además, del análisis a la estructura general de una aplicación web que implementa este protocolo con el que se hizo énfasis en las capas del *frontend* y del *backend* y, por último, el concepto de *fullstack*.



Universidad de Caldas