

Web Architecture

Client

- Machine which is having the browser.
 - The user who is using the browser for browsing the website
- e.g
- desktop
 - mobile

Browser

It is a special application which is used to browse websites
uses URL to browse a website

- URL
 - Uniform Resource Locator
 - Resource: any file that is hosted (copied) on the server machine
- e.g
 - https://www.google.com
 - http://192.168.0.1
 - http://localhost:8080
 - http://mysite.com/myfile.php?username=user&address=ngp
- every url follows a pattern

Component

- Scheme
 - The protocol that will be used while communication
 - e.g http, ftp, file, https
 - Default it is http or https
- Domain name or IP address
 - The way to find the machine uniquely over the network
 - e.g
 - google.com , apple.com
 - 142.250.67.68
- Port number
 - Port number of the web server running on the server machine
 - If you missed to write port number
 - 80 for http
 - 443 for https

- Query string
 - Used to pass your user data
 - It always starts with ?
 - e.g
 - ?username=user&address=ngp

componentes

- network
 - it is responsible for sending the request and reciving the response
- layout engine
 - it is responsible for converting the html and css to JS objects
- javascript engine
 - it is responsible for executing the JS code
- user interface
 - display output to the user
- web storgae
 - is is aslo know as data store
 - e.g history , session , cookies

server

- machine which serves a request
- server is not a hardware rather server is a software
- types
 - web server
 - server which serves the web (https or http) request
 - e.g : apache2, IIS , express
 - database server
 - server which persists the data
 - e.g: RDBMS: MYSQL, Oracle
 - NoSQL: MongoDB, Firebase

stack

- web stack - web server - database - language - platform

e.g : LAMP : Linux + Apache + MySQL + PHP/python/perl
 MEAN : MongoDB/Mysql + Express + Angular + Nodejs
 MERN

creates HTTP response

- object created to return the content when request
- contains
 - header : collection of key-value pair
 - body: content

status code

- used to represent the status of server execution
- types

- 1xx
 - for debugging or informational messages
 - e.g: 100: continue, 101 : switching protocol
- 2xx
 - success
 - e.g: 200 OK
 - 201: created
 - 202: Accepted
- 3xx
 - redirection
 - e.g 300: multiple choice
 - 301: moved permanently
- 4xx
 - client error
 - e.g : 400: bad request
 - 401: unauthorized
 - 402: payment required
 - 403: forbidden
 - 404: Not Found
- 5xx
 - server side error
 - e.g : 500: internal server error
 - 501: not implemented
 - 502: bad gateway