1.1 Web Basics: Web Browsers, Web Servers, Tier Technology, Static and Dynamic Web Page

Web

- World wide is commonly called web.
- System of interlinking hypertext documents that are accessed via the internet.
- One can view web page that contains text,images,videos and other multimedia and navigate between them via hyperlink.
- Tim Berners lee a British Computer scientist & Belgian Computer scientist Robert Cailliar invented the web.

Web Technology

- Web technologies are infrastructural building blocks of any effective computer network that allows users and devices on the computer network to communicate and share resources.
- Communication between the sender and receiver can only possible with the mechanism called web technology.
- E.g HTML,CSS,CGI,XML

Web Browsers

- Web Browsers are software installed on your PC to access the Web.
- such as Chrome, Safari, Microsoft Internet Explorer or Mozilla Firefox.

Web Servers

- Called service providers to the client.
- Every website sits on a computer known as web server.
- Every Web server that is connected to the Internet is given a unique address made up of a series of four numbers between 0 and 255 separated by periods.
- for example, 68.178.157.132 or 68.122.35.127. When you register a Web address, also known as a domain name, such as cite.com you have to specify the IP address of the Web server that will host the site.

Tier Technology

Software Architecture consists of One Tier, Two Tier, Three Tier and N-Tier architectures.

A "tier" can also be referred to as a "layer".

Three layers involved in the application namely Presentation Layer, Business Layer and Data Layer. Let's see each layer in detail:

Presentation Layer: It is also known as Client layer. Top most layer of an application. This is the layer we see when we use a software. By using this layer we can access the webpages. The main functionality of this layer is to communicate with Application layer. This layer passes the information which is given by the user in terms of keyboard actions, mouse clicks to the Application Layer.

For example, login page of Gmail where an end user could see text boxes and buttons to enter user id, password and to click on sign-in.

In a simple words, it is to view the application.

Application Layer: It is also known as Business Logic Layer which is also known as logical layer. As per the gmail login page example, once user clicks on the login button, Application layer interacts with Database layer and sends required information to the Presentation layer. It controls an application's functionality by performing detailed processing. This layer acts as a mediator between the Presentation and the Database layer. Complete business logic will be written in this layer.

In a simple words, it is to perform operations on the application.

Data Layer: The data is stored in this layer. Application layer communicates with Database layer to retrieve the data. It contains methods that connects the database and performs required action e.g.: insert, update, delete etc.

In a simple words, it is to share and retrieve the data.

Types of Software Architecture:
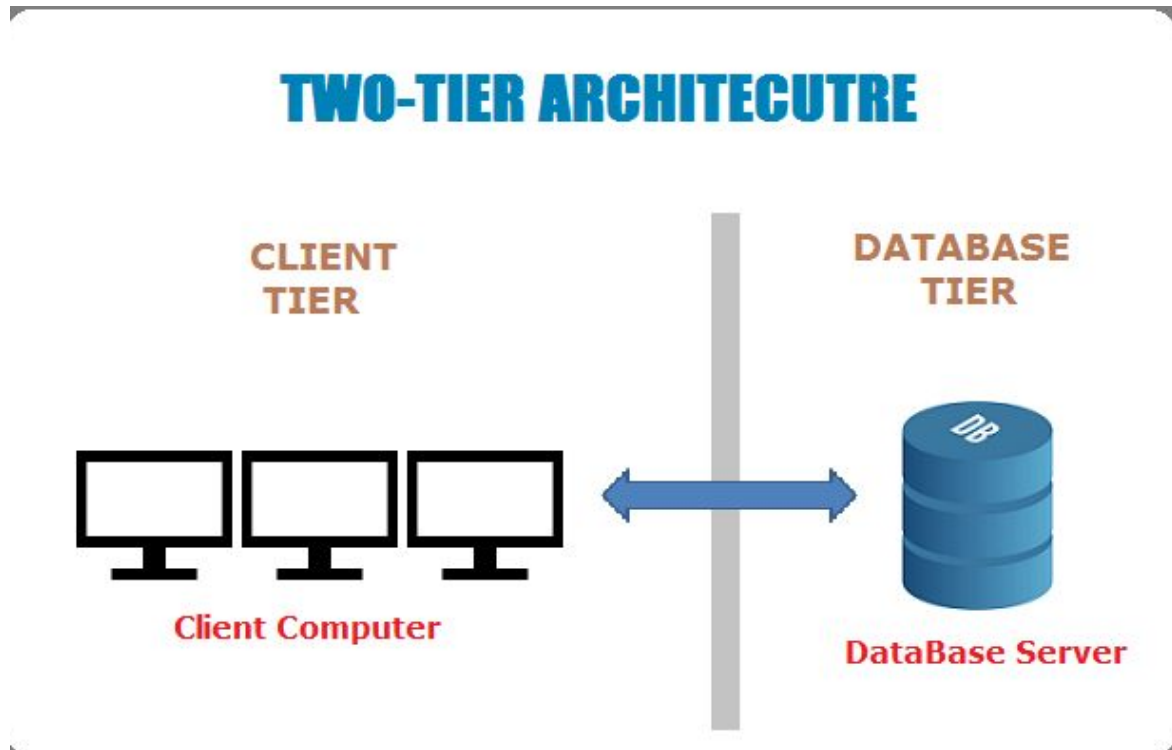
One Tier Architecture:

One Tier application AKA Standalone application
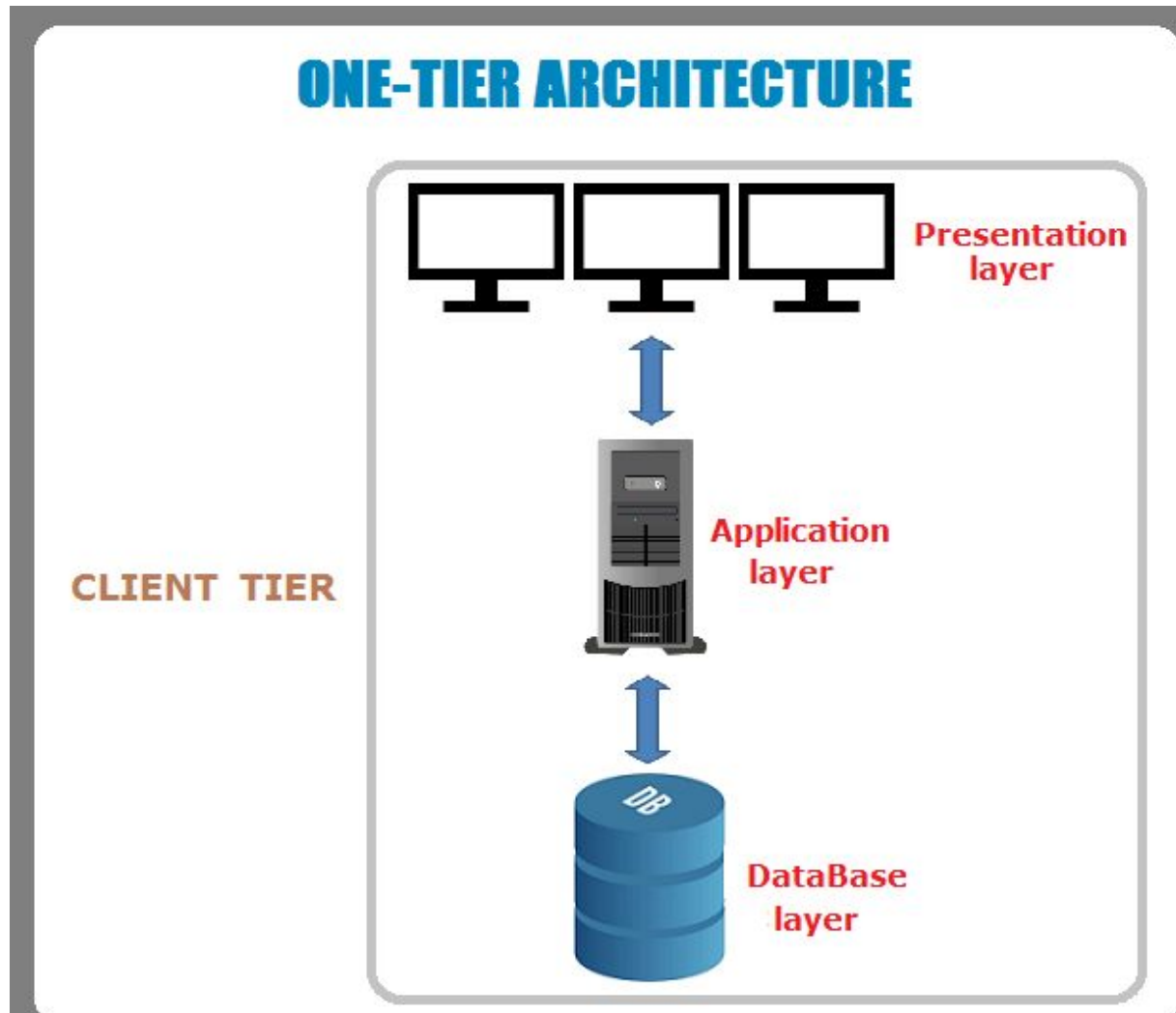
One tier architecture has all the layers such as Presentation, Business, Data Access layers in a single software package. Applications which handles all the three tiers such as MP3 player, MS Office are come under one tier application. The data is stored in the local system or a shared drive.

Two-Tier Architecture:

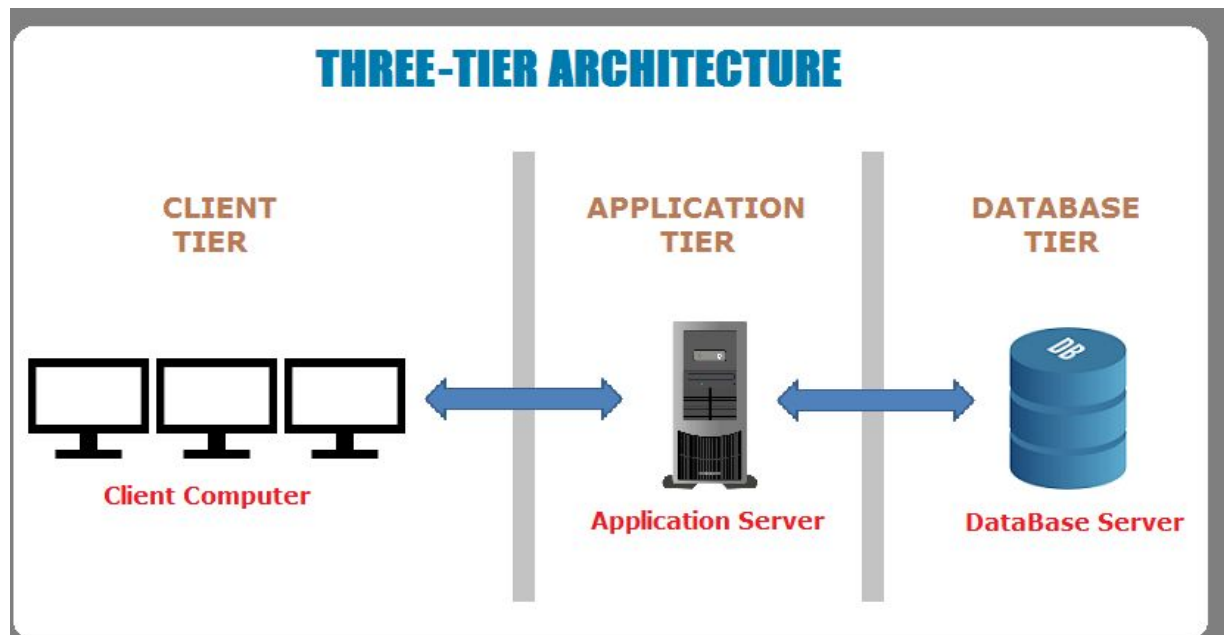Two Tier application AKA Client-Server application

The Two-tier architecture is divided into two parts:

1. Client Application (Client Tier)

2. Database (Data Tier)

Client system handles both Presentation and Application layers and Server system handles Database layer. It is also known as client server application. The communication takes place between the Client and the Server. Client system sends the request to the Server system and the Server system processes the request and sends back the data to the Client System

Three-Tier Architecture:

Three Tier application AKA Web Based application

The Three-tier architecture is divided into three parts:

1. Presentation layer (Client Tier)

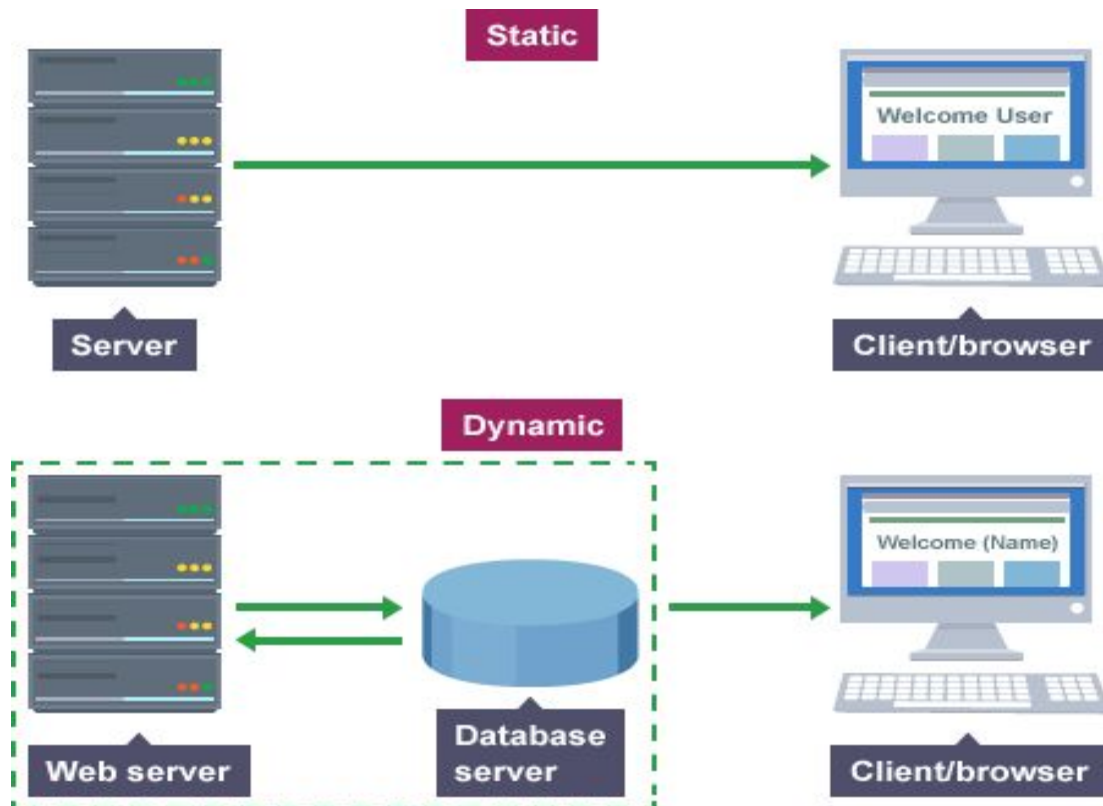2. Application layer (Business Tier)

2. Database layer (Data Tier)

Client system handles Presentation layer, Application server handles Application layer and Server system handles Database layer.

Note: Another layer is N-Tier application. N-Tier application AKA Distributed application. It is similar to three tier architecture but number of application servers are increased and represented in individual tiers in order to distributed the business logic so that the logic will be distributed.

(3-Tier and N-tier)

- Physical, Business and Data layer are physically separated.
- In software engineering, multi-tier architecture (often referred to as n tier architecture) is a client-server architecture in which presentation, application processing, and data management functions are physically separated.
- Most widespread use of multi-tier architecture is three-tier architecture.
- N tier application architecture provides a model by which developers create a flexible and reusable application.

Static and Dynamic Web Page

Static

Welcome User

Server   Client/browser

Dynamic

Web server   Database server   Client/browser

Welcome (Name)

Static Web Page

- Unchanged or constant.
- Contains the prebuilt content each time the web page is loaded.
- Standard HTML web pages are static web page
- HTML defines the structure and content of the web page.
- The content of HMTL will change only if the developer updates and publishes the file.
- Extension for static web pages is .htm or .html

Dynamic Web Page
- Changing content
- These web pages contain server code which allows to change the content dynamically each time if the web page loads.
- Web pages developed using the scripting language such as PHP,ASP,JSP,ROR are dynamic web pages.
- Also called data base driven web pages.
- Extension for dynamic web page is .php,.apsx,.jsp

1.2 HTTP, HTTPs, FTP
HTTP
- Hypertext transfer protocol, the underlying protocol used by the world wide web.
- HTTP defines how the message is transmitted and formatted and what actions the web servers & web browsers should take to respond to various commands.
- For eg. When we enter URL in our web browser, this actually sends an HTTP command to the web server directing to fetch & transmit the requested web page.
- HTTP is called stateless protocol because each command is executed independently, without knowledge of the commands that came before HTTP

HTTPS
- Similar to HTTP.
- Secure HTTP used by world wide web.
- The message is transmitted in encrypted form.

FTP
- File transfer protocol.
- For exchanging files over the internet.
- FTP works same ways as HTTP for transferring web pages from a server to users browsers and SMTP for transferring emails across the internet.

- Uses TCP/IP protocols to enable data transfer.
- Used for downloading files from the server and uploading files to the server.

Free and Open Source Software

Characteristics of free software

For a program to be free software, it should respect these 4 essential freedoms:
- Freedom 0: Freedom to run the program as you wish.
- Freedom 1: the Freedom to study the source code of the program and freedom to change it.
- Freedom 2: the Freedom to help your neighbor, i.e. freedom to make and distribute exact copies of the program whenever you want.
- Freedom 3: Freedom to contribute to the community, i.e. freedom to make or distribute copies of your modified versions of the program.

Open source software
- source code is openly shared so that people are encouraged to voluntarily improve the design of the software.
- Open Source Software Open Source Software is a software that is free to use and which provides the original source code used to create it so that advanced users can modify it to make it work better for them.
- Examples: Linux GIMP Blender Inkscape Mozilla Firefox 3.0 OpenOffice.org KOffice

Advantage of FOSS
- Generally free.
- Easily modified and adopted for the business requirements.
- Decreasing software costs, increasing security and stability (especially in regard to malware), protecting privacy, and giving users more control over their own hardware.
- Free, open-source operating systems such as Linux and descendants of BSD are widely utilized today, powering millions of servers, desktops, smartphones (e.g. Android), and other devices. Free software licenses and open-source licenses are used by many software packages.

- Long term support.

Disadvantages of FOSS
- FOSS can't be sold and will not generate money as that of a commercial product.
- Less user-friendly and easy to use because less attention is paid to developing the UI.
- Less support available when things go wrong.
- It rely on its community of users to respond and fix problems.
- Although open source software is free but there might be some indirect cost evolved as paying for support
- There may be certain bugs with in the software which will exploit security threat

Proprietary Software
- Proprietary Software Proprietary Software (Closed Source Software) means the company that developed the software owns the software and no one may duplicate it or distribute it without that company's permission.
- Users have to pay to the software company if they want to use the proprietary software.
- Examples: Microsoft Office, Adobe Photoshop CS3, Adobe Flash, Corel Office X3 Windows OS (7, 8, 10)

Advantages of Proprietary software
- Good customer service for troubleshooting and setup purpose.
- Developed to meet market demand.
- Fewer bugs compared to FOSS. Disadvantages of Proprietary Software

Disadvantages of Proprietary Software
- Expensive.
- Charge licensing fee.
- Dependents on the developer.
- Not easily adaptable.

Difference between Free Software, Open Source Software, and Proprietary Software

Free VS Open Source Softwares

Free Software	Open Source Software
• Software is given free.	• Software + source code (program is freely distributed)
• Developed by one or a few individuals or an organization.	• Collaborative development.
• May or may not be updated	• Updated
• Individual efforts	• Collaborative effort
• May put price tag	• Always free
• New features may not be available	• New versions with improved features

DIFFERENCES BETWEEN PROPRIETARY & OPEN SOURCE SOFTWARE

Open Software	Proprietary
• Linux, Ubuntu, OpenOffice.org	• Windows Vista, Microsoft Word
• Purchased **with** its source code	• Purchased **without** its source code
• Users **can modify** the software	• Users **cannot modify** the software
• Can get open software **for free** of charge	• **Must pay** to get the proprietary software
• Can install software **freely** into any computer	• Must have a **license** from vendor before install into computer
• No one is responsible to the software	• Full support from vendor if anything happened to the software

COMPARISON OF OPEN SOURCE AND PROPRIETARY SOFTWARE

	proprietary software	open source software
license fee	$100	~~$100~~
maintenance and support	commercial support	community or commercial support, if available
source code access	no access	access, modification possible

1.3) Licensing and its types: Commercial License and Open Source License

The legal agreement governing the use or redistribution of software.

It grants an end user permission to use one or more copies of software.

Software licensing typically contains provisions which allocate liability and responsibility between the parties entering into the license agreement.

Software licenses can generally be fit into the proprietary software and free and open source software

Commercial license

- Publisher of software grants the use of one or more copies of software under the end user license agreement (EULA).
- Ownership of those copies of remains to software publisher.
- Certain rights regarding to the software are reserved by the software publisher.
- Without the acceptance of the license, the end user may not use the software at all
- E.g license for Microsoft Windows
- Proprietary Software license contains the extensive list of activities which are restricted
- Such as reverse engineering, simultaneous use of the software by multiple users and publication of benchmarks or performance tests
- The most common license models is per single user or per user in the appropriate volume discount level, while some manufacturer accumulate existing licenses.

Free and open-source software license

- Aim to preserve the freedoms that are given to the users by ensuring that all subsequent users receive those rights(copy left license).
- E.g copy left free software license is the GNU general public license(GPL).
- This license is aimed at giving all user unlimited freedom to use , study, and privately modify the software,
- Any modifications made and redistributed by the end user must include the source code.

Permissive free software licenses are the BSD license and the MIT license.

-
  - These give unlimited permission to use, study and privately modify the software and include only minimal requirements on redistribution.
  - Apple public source license, Apache license, BSD license, Boost software license.