Client Server Overview

Table of Contents

[Introduction 2](#_Toc78759301)

[Website – What is? 2](#_Toc78759302)

[Types of Websites 2](#_Toc78759303)

[Static Website 2](#_Toc78759304)

[Dynamic Website / Web Application 2](#_Toc78759305)

[What is the Web? 2](#_Toc78759306)

[What is a Client? 3](#_Toc78759307)

[Client Machine 3](#_Toc78759308)

[Client Softwares 3](#_Toc78759309)

[What is a Server? 4](#_Toc78759310)

[Server Machine 4](#_Toc78759311)

[Server Software 4](#_Toc78759312)

[Hosting 5](#_Toc78759313)

[HTTP Protocol 5](#_Toc78759314)

[HTTP Methods 6](#_Toc78759315)

[How do all these things work together? 6](#_Toc78759316)

# Introduction

In this modern digital age, **billions of bytes of data** get generated on a day-to-day basis. The data such as videos, audios, text, images are uploaded every single second onto the millions of servers’ worldwide.

To store and access this huge amount of data, every person sends **request** on the **network** and awaits a **response**.

In order to understand the mechanism of request and response, let us first get an understanding of the **fundamentals of the web**.

# Website – What is?

A Website is a **collection of web pages**. A web page is **designed** by front end web technologies and developed using **back-end web technologies**

## Types of Websites

There are 2 types of websites:

### Static Website

* A static website is a website which contains Web pages with fixed content.
* Each page is coded in HTML and displays the same contents to every visitor at any given time.
* Static sites are the most basic types of websites and are the easiest to create.
* Static websites **do not** require any web programming or database designs.

### Dynamic Website / Web Application

* A dynamic website contains information that changes, depending upon the viewer interaction.
* It contains client-side and server-side scripting to generate dynamic content.
* A Web application requests services over the web to generate dynamic content
* These are created using server side technologies such as Java, PHP, Python, and ASP.Net etc.

Example: Any website that presents dynamic content to the user such as Instagram, Yahoo, Google etc.

# What is the Web?

Simply put, a **web or internet** is an interconnected system of computers. These computers can be categorized as:

* Clients, and
* Servers

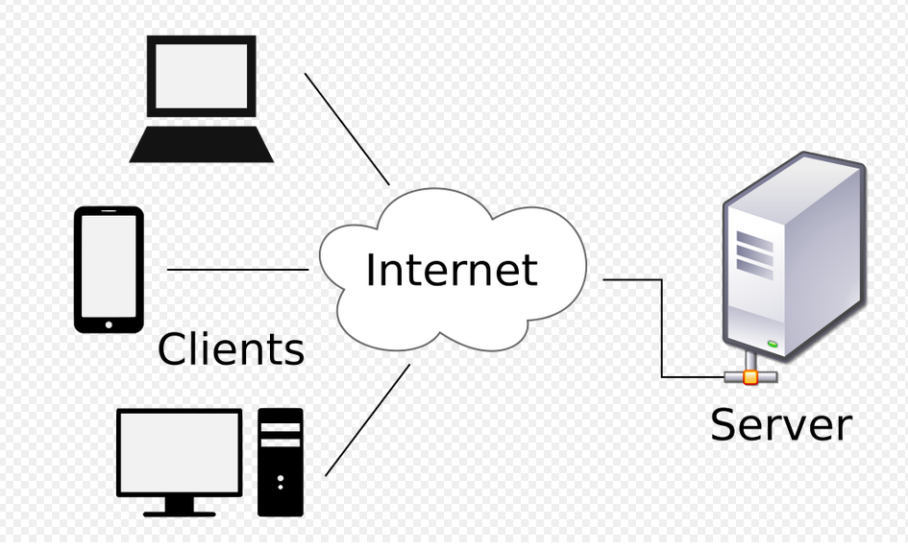


Figure 1. Client-Server Architecture

The mechanism is as simple as: **A client requests, a server fulfills the requests.**

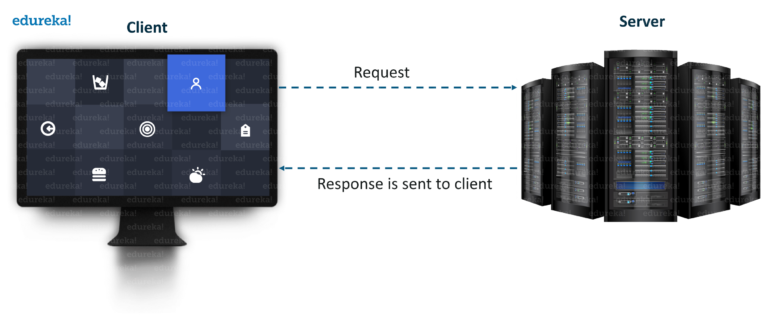


Figure 2. Request Response Flow

## What is a Client?

A client can be anything that is used to request data from a server. There are 2 types of clients:

### Client Machine

A client machine is a piece of computer **hardware** that accesses a service made available by a server. Some examples of client machine are:

1. Desktop/Laptop Computers
2. Mobiles
3. Smart TVs
4. Smart Watches
5. Etc.

### Client Softwares

Client *software* is a piece of computer software that makes the request to the server and parses the response for the user. Some examples of client softwares are:

1. **Web browsers** (**Firefox**, **Chrome**, **Edge**) are clients that connect to web servers and retrieve web pages for display.
2. **Email clients (Thunderbird, Chipmunk)** retrieve email from mail servers.
3. **FTP client:** FTP stands for File Transfer Protocol. Using an FTP client we can upload, download, and manage files on our server.

## What is a Server?

A Server can be anything that responds to the requests sent over the network by the client.

### Server Machine

A server machine is a piece of computer **hardware** that stores in data and provides service to the client requests over the network 24X7. There are following types of server machines:

1. File Servers
2. Web Servers
3. Mail Servers
4. Virtual Servers
5. Application Servers
6. Domain Name Servers

|  |  |
| --- | --- |
| server.png  **Single Dedicated Server** | servers.jpg  **Server Room (Multiple Servers)** |

Figure 3. Physical Servers

### Server Software

All server machines must have server softwares installed on them to **handle requests and send prepared response** to the client. Some examples of server softwares are:

1. **Apache Tomcat**
2. XAMPP
3. Microsoft IIS
4. Lighttpd
5. Etc.

So, it could be said that: In a data center, the physical computer that a server program runs on is also frequently referred to as a server.

### Hosting

When a **server** machine **stores** a file or **group of files** on it for **any specific time**, it is known as Hosting.

If we want our web application to be called from any client machine or any client software, we must **host the web application on a web server.**

There can be different types of hosting:

#### Private Hosting

The big companies (such as Facebook, Google) have their **own servers** to host their web applications.

#### Rented Hosting

However, small companies can take a **server on rent** from **server space provider companies.**

There are many private companies that offer different plans to host websites. Some such companies are:

* Godaddy
* Amazon
* Bigrock
* Hostgator, etc.

#### Local Hosting

A server installed on the local machine for development purposes is called local hosting.

When a software program is developed, the programmers need to test it frequently for correctness. In case the programmer is working directly on a remote server, it could be time and bandwidth consuming because every change will require transaction with the online server, and the server could be anywhere in the world. So, the solution is to use local-host (server on personal machine) for development. When finally the software is completely ready it can be transferred to the main server at once.

Some local servers are:

1. **Apache Tomcat**
2. XAMPP
3. IIS etc.

# HTTP Protocol

* HTTP stands for Hyper Text Transfer Protocol
* It is a network **protocol (set of rules)** that clients and servers on the web use to communicate.
* It is similar to other internet protocols such as **SMTP** (Simple Mail Transfer Protocol), **FTP** (File Transfer Protocol and **TCP/IP** (Transmission Control Protocol/Internet Protocol).
* HTTP is a **stateless protocol** i.e. it supports only **one request per connection**. This means that the protocol treats every request as a new request; it doesn’t remember the client over multiple requests.
* The above mechanism allows more users to connect to a given server over a period of time.
* The client sends an HTTP request and the server responds with an HTTP response.

**For ex**: Web browser and Web Servers communicate using HTTP

## HTTP Methods

The primary and the most commonly used HTTP methods are:

1. POST
2. GET
3. PUT
4. PATCH
5. DELETE etc.

The HTTP request can be made using any of the above of methods, but the ones which are most used are:  **Get** and **Post**.

# How do all these things work together?

1. Some user is browsing the internet on a client machine using a client software, such as Chrome.
2. The client software and client machine together are called **Client**.
3. The **Client** wants to access a **web application**
4. This **website** (web application) is **hosted** on a **remote** **server machine**.
5. The request for the website & response is managed by a **server software** installed on the server machine.
6. Both, server software and server machine are collectively called as **Server**
7. The website is identified by an **address** also known as a **domain name**.
8. The website can be accessed by its **domain name**.
9. The **Client** makes a request to the **Server** to access the website
10. This request is sent through the **network** using the **HTTP protocol.**
11. The **Server** gets the request, **prepares the data** and sends it back to the **Client**
12. The server also uses the **HTTP protocol** to send the response
13. The client software **parses the response** and displays the output to the end user
14. The **Server** again comes into a waiting state to handle more requests