**MVC.Net - Getting started with MVC .Net**

- What is role of client?

The role of a client can vary depending on the context, but generally, a client refers to an individual or organization that seeks or uses the services, products, or expertise of another party, typically a service provider, vendor, or a professional.

-What is role of server?

The role of a server in various contexts, particularly in the context of client-server architecture, is to provide services, resources, or information to clients. Servers play a crucial role in managing and delivering data or functionality to multiple clients or users. Servers serve as the backbone of client-server architecture, handling data storage, processing, communication, and many other functions essential for the proper functioning of applications and services that clients depend on. The specific role and function of a server depend on the type of server and its intended purpose within a given system.

-What is client server architecture?

Client-Server architecture, often referred to as a client-server model, is a computing and network architecture that defines the roles and responsibilities of two distinct types of entities: clients and servers. This architecture is commonly used in distributed computing systems and networked environments. Here's an overview of the client-server architecture:

Client:

A client is a software application or device that initiates requests for services, resources, or information from a server.

Clients can be end-user applications (e.g., web browsers, email clients), devices (e.g., computers, smartphones, IoT devices), or even other servers.

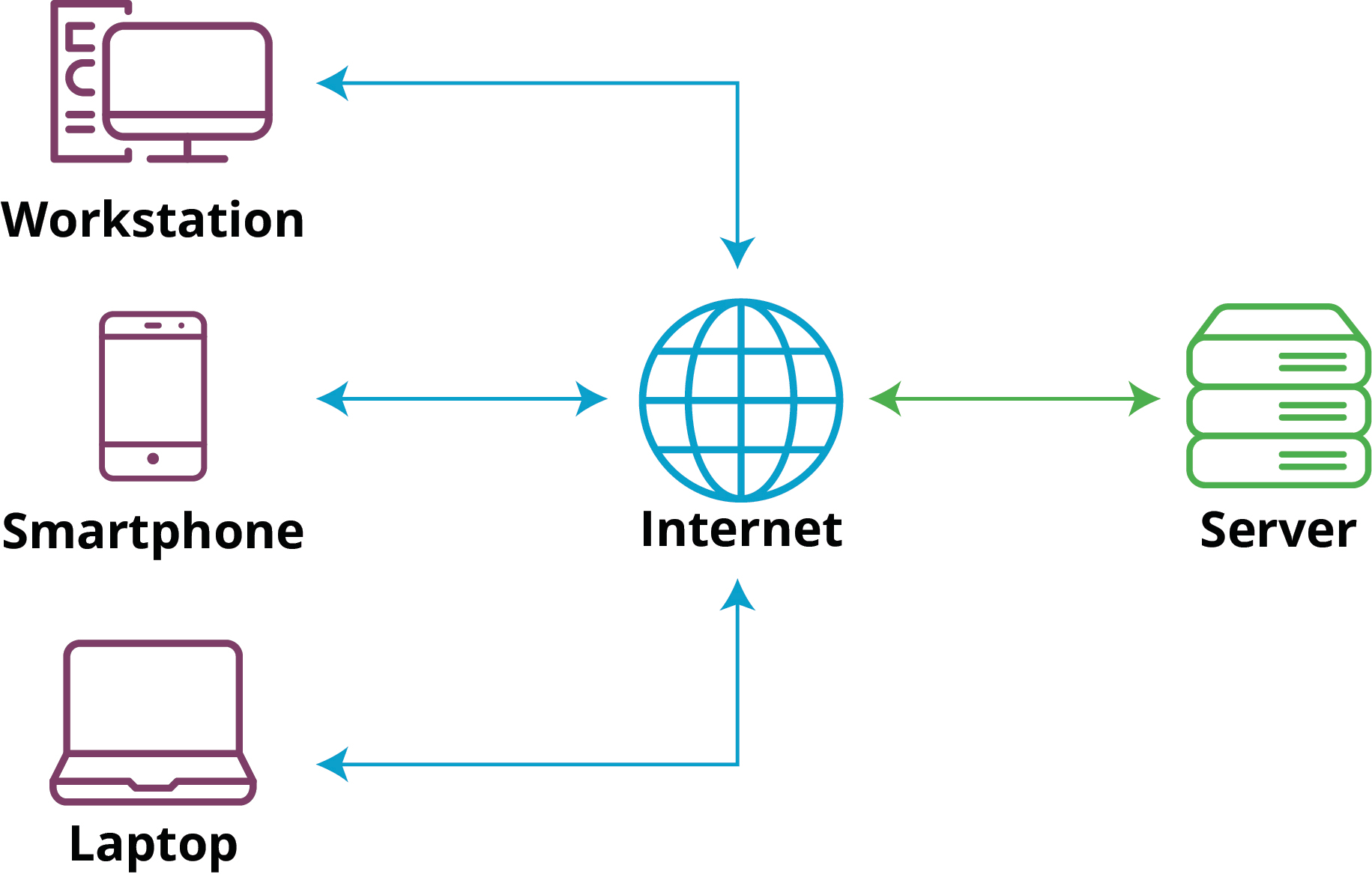
Clients request specific actions or data from the server, such as accessing a web page, fetching email, or retrieving database records.

Server:

A server is a software application or hardware system that provides services, resources, or data to clients.

Servers are designed to listen for incoming requests from clients, process those requests, and respond accordingly.

Servers are typically more powerful and specialized in their functions, capable of handling multiple client requests simultaneously.



-What is role of compiler?

A compiler is a crucial software tool in the field of computer science and programming. Its primary role is to translate high-level programming code written by a human programmer into machine code or a lower-level code that can be executed directly by a computer's central processing unit (CPU). Here are the key roles and functions of a compiler:

-Translation

-Syntax and Semantic Analysis

-Optimization

-Error Detection and Reporting

-Code Generation

-Linking

-Portability

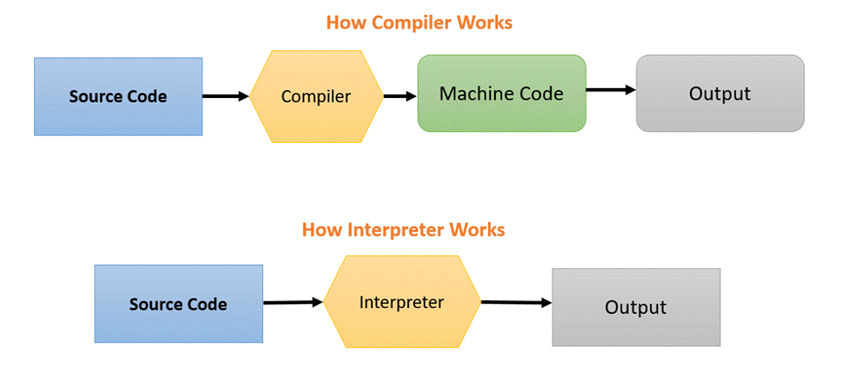
-Performance Enhancement

-Security Enhancement

Compiler is a software tool that takes source code written by a programmer and converts it into a format that can be executed by a computer. It plays a crucial role in ensuring that code is error-free, efficient, and compatible with the target platform, ultimately making software development more accessible and manageable.

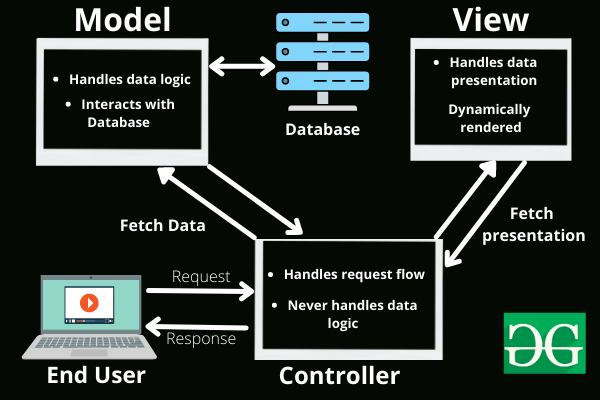
-What is difference between Compiler and Interpreter?

Compiler: A compiler translates code from a high-level programming language (like Python, JavaScript or Go) into machine code before the program runs. Interpreter: An interpreter translates code written in a high-level programming language into machine code line-by-line as the code runs.



-What Is MVC?

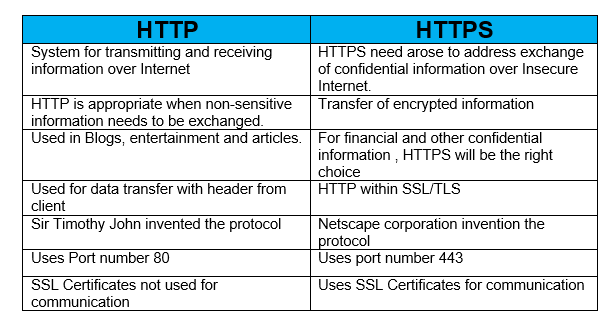
MVC stands for Model-View-Controller, which is a widely used architectural pattern in software development for designing and organizing applications. MVC separates an application into three interconnected components to promote modularization, maintainability, and scalability.



-What is communication protocol and what is difference between HTTP and HTTPS?

A communications protocol is a set of formal rules describing how to transmit or exchange data, especially across a network. A standardized communications protocol is one that has been codified as a standard. Examples of these include WiFi, the Internet Protocol, and the Hypertext Transfer Protocol (HTTP).

HTTPS is HTTP with encryption and verification. The only difference between the two protocols is that HTTPS uses TLS (SSL) to encrypt normal HTTP requests and responses, and to digitally sign those requests and responses. As a result, HTTPS is far more secure than HTTP.



-What is .net?

.NET, pronounced as "dot net," is a software framework and platform developed by Microsoft. It provides a comprehensive and consistent programming model for building a wide range of applications, from web and desktop applications to mobile apps and cloud-based services. .NET encompasses a set of technologies, tools, and libraries that are widely used in the software development industry.

.NET is widely used in enterprise software development, web development, game development, and various other application domains. It provides a robust and versatile platform for creating modern, scalable, and secure applications.

-What is CLR?

At the core of .NET is the **Common Language Runtime** (CLR), a virtual machine that manages the execution of .NET programs. The CLR provides features like memory management, security, and exception handling.

-What is difference between CLS and CTS?

CLS and CTS are both components of the .NET framework designed to facilitate interoperability among different .NET languages. CLS focuses on language-level features and syntax to ensure that code written in one language can be consumed by other languages. CTS, on the other hand, deals with data type definitions, memory representation, and type system rules to ensure that data can be shared and used consistently across languages. Together, CLS and CTS help create a cohesive and interoperable .NET ecosystem.

-What do you mean by design pattern?

A design pattern is a general and reusable solution to a common problem in software design and development. It represents a best practice or a template that can be applied to solve a particular type of problem or achieve a specific objective in a software system. Design patterns are not complete designs or ready-to-use code; instead, they offer guidance and principles for structuring and organizing software.

Design patterns help developers build software that is more maintainable, scalable, and robust by following established practices and solutions to common software design problems. They are a valuable resource for software architects and developers looking to improve the quality and efficiency of their code.

-What is client server architecture?

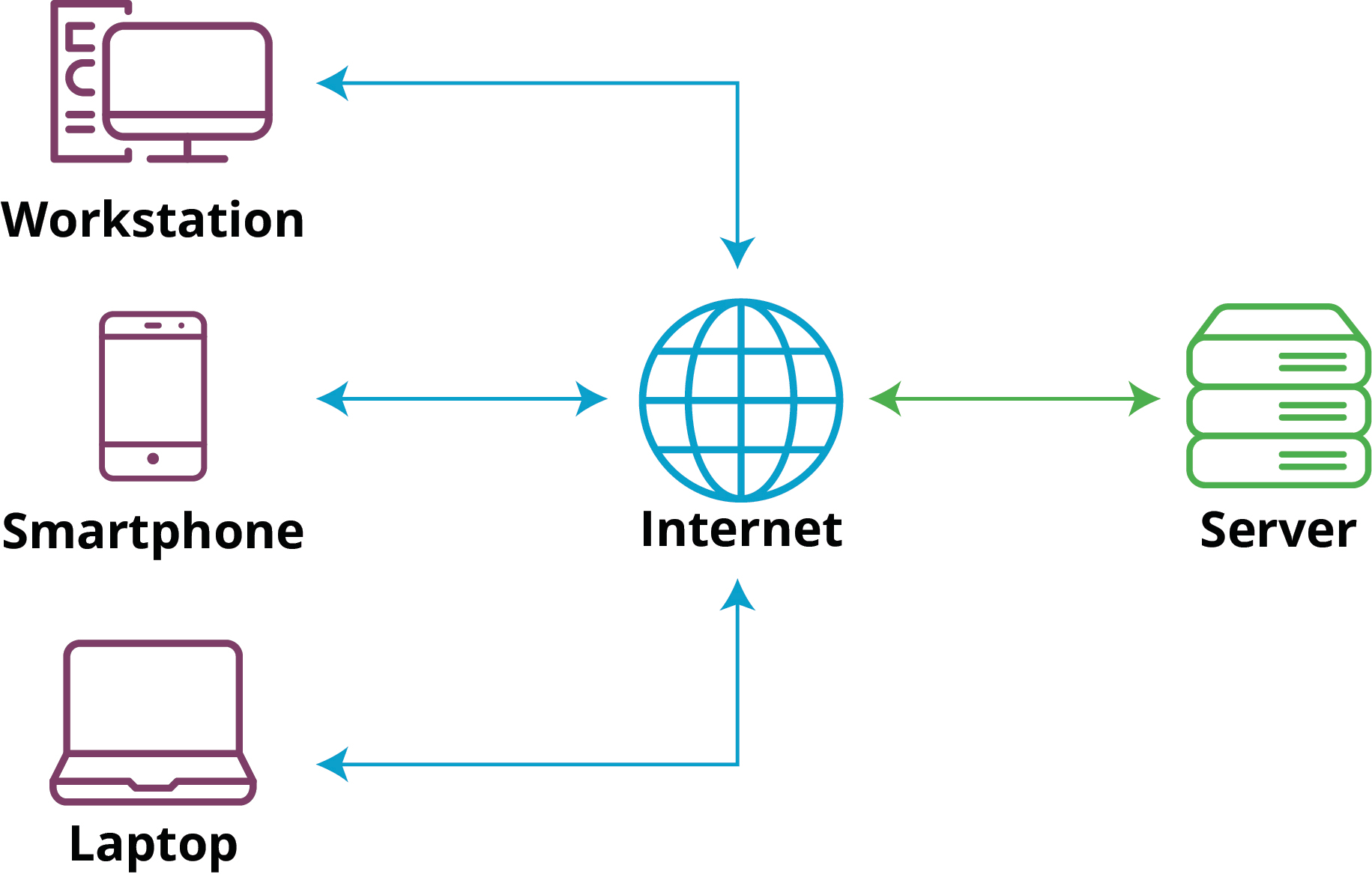
Client-server architecture is a type of computer network where multiple clients request and receive services from a centralized server. The server is responsible for hosting, managing, and delivering services to clients. Clients are connected to the server and communicate with it over the internet using a computer network.

In this model, all requests and services are delivered over a network. Examples of computer applications that use the client–server model include:

-Email

-Network printing

-The World Wide Web



-What is difference between Asp.Net and MVC.Net?

ASP.NET MVC includes built-in support for URL routing, while in ASP.NET, developers have to implement their own URL routing. This makes it easier to define custom URL patterns in ASP.NET MVC, which can be useful for creating more user-friendly and SEO-friendly URLs.

