

- **Git<sup>1</sup>**: A version control system that lets you collaborate on code.
- **HTTP<sup>2</sup>**: A protocol that defines how web browsers and servers communicate.
- **API<sup>3</sup>**: An interface that allows applications to interact with each other.
- **RESTful API<sup>4</sup>**: A type of API that follows the principles of REST (Representational State Transfer), such as using HTTP methods, URIs, and JSON.
- **SSL<sup>5</sup>**: A security protocol that encrypts the data exchanged between web browsers and servers.
- **Cookie<sup>6</sup>**: A small piece of data that a web server sends to a web browser and is stored on the user's device.
- **Session**: A way of maintaining state between multiple requests from the same user or client.
- **Token**: A piece of data that proves the identity or authorization of a user or client.
- **OAuth**: A standard protocol that allows users to grant access to their online resources to third-party applications without sharing their credentials.
- **Cloud**: A model of computing that delivers services over the internet, such as storage, networking, databases, analytics, and artificial intelligence.
- **CDN**: A network of servers that distributes web content to users based on their geographic location and performance.
- **Load Balancing**: A technique that distributes workloads across multiple servers or resources to optimize performance and availability.
- **Cache**: A temporary storage of data that can be accessed faster than the original source.
- **Microservices**: A style of software architecture that breaks down an application into small, independent, and loosely coupled components that communicate through APIs.

- **Webservices:** Applications that provide functionality over the internet using standard protocols such as HTTP and XML.
- **Cloud Computing:** A model of computing that delivers services over the internet, such as storage, networking, databases, analytics, and artificial intelligence.
- **Lazy Loading:** A technique that delays loading of data or resources until they are needed. This can improve performance and reduce memory usage.
- **Eager Loading:** A technique that loads all data or resources at once. This can avoid multiple requests and ensure data consistency.
- **Database Transactions:** A unit of work that ensures data integrity by following the ACID properties (Atomicity, Consistency, Isolation, Durability).
- **Design Patterns:** Reusable solutions to common software design problems. They describe how classes and objects can interact to achieve a specific goal.
- **FTP:** A protocol that allows transferring files between computers over a network.
- **HTTPS:** A protocol that encrypts and authenticates web traffic using SSL/TLS certificates.
- **SSH:** A protocol that allows secure remote access to servers using encryption and authentication.
- **Asynchronous and Synchronous:** Two modes of communication or execution. Asynchronous means that the sender and receiver do not have to wait for each other, while synchronous means that they do.
- **How does internet work?:** A question that can be answered at different levels of abstraction. In general, the internet works by sending packets of data across a network of routers, switches, cables, satellites, and wireless devices using various protocols such as IP, TCP, UDP, DNS, HTTP, etc.

- **DNS:** A system that translates domain names into IP addresses.
- **TLS:** A security protocol that encrypts and authenticates data exchanged over a network. It is also known as SSL.
- **CORS:** A mechanism that allows web browsers to request resources from different origins than the current web page.

- **Asynchronous and Synchronous**

- asynchronous and synchronous are two modes of communication or execution. Asynchronous means that the sender and receiver do not have to wait for each other, while synchronous means that they do.
- For example, when you send an email to someone, you don't have to wait for them to reply before you can do something else. This is asynchronous communication. But when you make a phone call to someone, you have to wait for them to answer and talk to them in real time. This is synchronous communication.
- Similarly, when you write a program that performs some tasks, you can choose whether to make them asynchronous or synchronous. Asynchronous tasks can run in the background without blocking the main thread of execution, while synchronous tasks have to finish before the next task can start.
- Asynchronous programming can improve performance and responsiveness of your application, but it can also introduce complexity and challenges such as handling errors, callbacks, promises, async/await syntax etc.
- Synchronous programming can be simpler and easier to understand and debug, but it can also cause delays and bottlenecks if some tasks take too long or depend on external factors such as network latency etc.

- **How does internet work?**

- The internet is a global network of computers that can communicate with each other by sending and receiving data<sup>1</sup>. The data is divided into small segments called packets, which contain both information and instructions for routing them to the correct destination<sup>23</sup>. The internet uses a set of rules called protocols to ensure that packets are delivered correctly and efficiently. One of the most important protocols is Internet Protocol (IP), which assigns a unique address to each computer on the internet<sup>45</sup>.
- There are many other protocols besides IP that enable different types of communication and functionality on the internet. Some of them are:
  - TCP: Transmission Control Protocol ensures reliable and ordered delivery of data across IP connections<sup>1</sup>.
  - UDP: User Datagram Protocol enables low-latency and non-reliable data transmission for time-sensitive applications like voice or video<sup>12</sup>.
  - FTP: File Transfer Protocol allows users to transfer files between computers on a network.
  - HTTP: Hypertext Transfer Protocol defines how web browsers and servers exchange data over the web.
- **Encrypting** is the process of converting plain text into cipher text by using an algorithm. The encrypted message can only be read by knowing the encryption key<sup>12</sup>.
- **Hashing** is the process of converting plain text into a hash or digest by using an algorithm. The hash cannot be reversed to get the original plain text<sup>13</sup>.
- **SHA** and **MD5** are examples of hashing algorithms that produce a fixed-length string of characters from any input data<sup>4</sup>. They are used for verifying the integrity of data and preventing tampering.
- **Integration Testing** and **Unit Testing** are types of software testing that check if different components or

units of a system work correctly together or individually<sup>56</sup>.

- **TDD**, or Test-Driven Development, is a software development approach that involves writing tests before writing code and then refactoring the code until it passes the tests<sup>6</sup>.
- **Functional Testing**: A type of testing that verifies that the software is performing as expected, when used by any user or system that interfaces with it<sup>1</sup>.
- **CI/CD**: Continuous Integration and Continuous Delivery, a set of practices that enable faster and more reliable software delivery by automating code integration, testing, deployment, and feedback<sup>2</sup>.
- **Serverless**: A cloud computing model that enables developers to build and run applications without managing servers or backend infrastructure<sup>345</sup>.
- **DRY**: Don't Repeat Yourself, a principle that states that every piece of knowledge must have a single, unambiguous, authoritative representation within a system<sup>6</sup>.
- **Reusability**: The ability to use a piece of code for different purposes or contexts without duplication or modification<sup>7</sup>.
- **Pluggability**: The ability to add or remove components from a system without affecting its functionality or performance<sup>8</sup>.
- **Agile**: A set of values and principles that promote adaptive planning, collaborative working, early delivery, and continuous improvement in software development<sup>9</sup>.
- **UML** stands for **Unified Modeling Language**, a standardized modeling language for software engineering<sup>12</sup>.
- **Scrum** is an empirical process for agile project management that involves delivering value incrementally in a collaborative manner<sup>34</sup>.

- **GPG** stands for **GNU Privacy Guard**, a free software implementation of the OpenPGP standard for encryption and signing<sup>56</sup>.
- **CMS** stands for **Content Management System**, a software application that allows users to create, manage, and modify digital content on a website<sup>12</sup>.
- **SOAP** stands for **Simple Object Access Protocol**, an XML-based messaging protocol for exchanging information among computers<sup>34</sup>. It can also refer to a salt of a fatty acid used for cleansing and lubricating<sup>5</sup>.
- **JWT** stands for **JSON Web Token**, an open standard that defines a compact and self-contained way for securely transmitting information between parties as a JSON object<sup>67</sup>.
- **TCP/IP** stands for **Transmission Control Protocol/Internet Protocol**, a suite of communication protocols used to interconnect network devices on the internet or a private network<sup>89</sup>.
- **PWA** can stand for **Progressive Web App**<sup>1</sup>, a type of web application that works on any device and has features such as offline access, push notifications, and fast loading.
- **SPA** can stand for **Single Page Application**<sup>2</sup>, a type of web application that loads a single HTML page and dynamically updates it as the user interacts with it.
- **WSGI** can stand for **Web Server Gateway Interface**<sup>3</sup>, a specification that describes how a web server communicates with web applications or frameworks written in Python.
- **ASGI** can stand for **Asynchronous Server Gateway Interface**<sup>4</sup>, a successor to WSGI that allows multiple, asynchronous events per application and supports both sync and async apps.