

# SDLC: Software Development Life cycle

## 16/05/2025

Task: 1

### What is SDLC?

SDLC is Software Development Life Cycle, it is structured process used to design, develop, test the software and provide the better software to users

It is systematic approach that divides software development process into diff phases according to planning through maintenance the Application

It includes SDLC models like waterfall, agile, spiral and V-model.

Task 2:

### Why is SDLC ?

Here are some point:

1. **Structure & Organization:** SDLC provide a systematic framework that helps in Software development by braking down complex process.
2. **risk management:** early identification of problems through structured phases  
Minimise the risk
3. **Quality Assurance:** it helps to better quality and provide specific deliveries

Task 3:

### What are the stages of SDLC ? write 2 lines about each.

There are 6 stages of Software Development Life cycle:

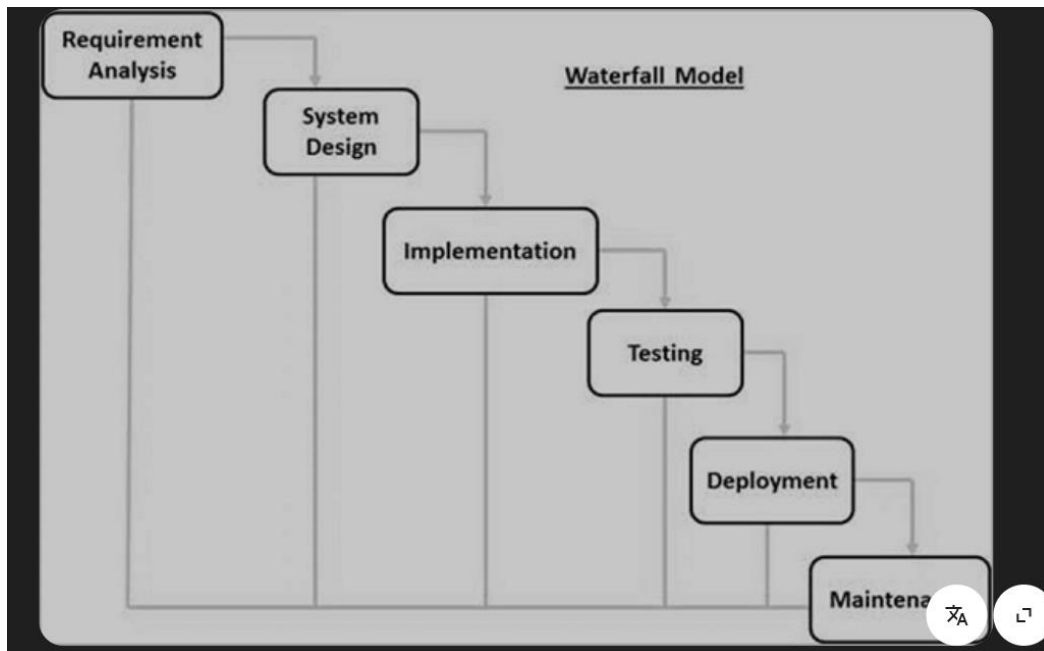
1. **Planning and Requirement** - Gather all the System requirements from the Client/user what are the expectation from the software.
2. **Design** - Creating the architecture and software design according to the requirements. which includes database design.
3. **Development/Implementation** - building the software by writing the Code on design with structured/Systematic code.
4. **Testing** - perform testing the software(unit, integration, system, acceptance and etc).
5. **Deployment** - Deploy the Software on the Servers.
6. **Maintenance**- maintenance the software and fixing the bugs, features enhancements, checking wether it is working smoothly or not and security updates .

Task 4:

## SDLC Models:

### Waterfall Mode:

Waterfall Model is very simple and structured model and It is linear and sequential model process . which includes requirements, design, coding, testing and deployment. which is easy to understand and manage. well documented and structured it is used for small scale projects



#### Applications:

library management.

construction companies used it for building management software

#### Advantages:

Simple and easy understand.

Well-defined requirements.

#### Disadvantages:

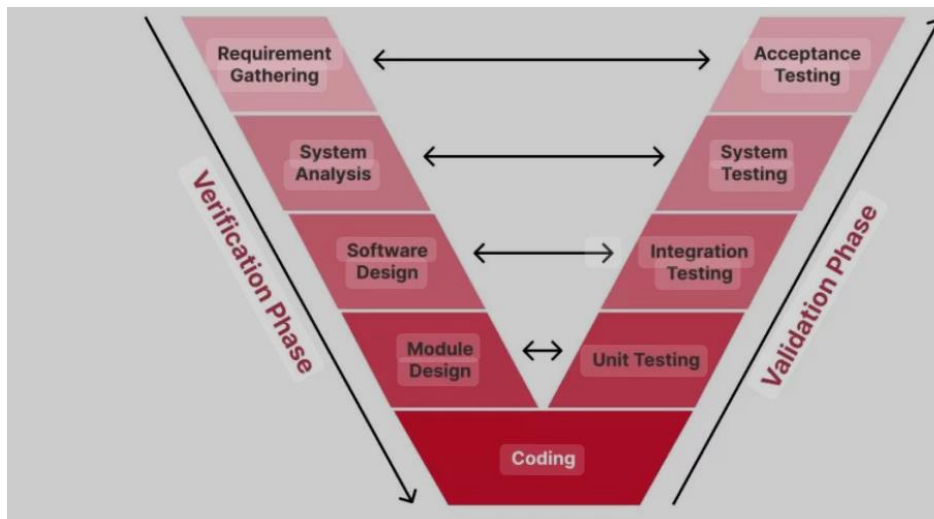
Difficult to handle changes.

## V-Model:

It is extension of the waterfall model with a testing phase for each development. it is also called verification and validation model.

in this model also includes requirement gathering, analysis, design , coding and each development phase has a corresponding testing phase, with testing planned in parallel to development which are unit testing, integration testing, system testing, Acceptance testing.

in here bugs are found earlier while developing the software.



### Applications:

traffic control software.

Medical device software.

### Advantages:

Easily found bug early in development

testing at every phase

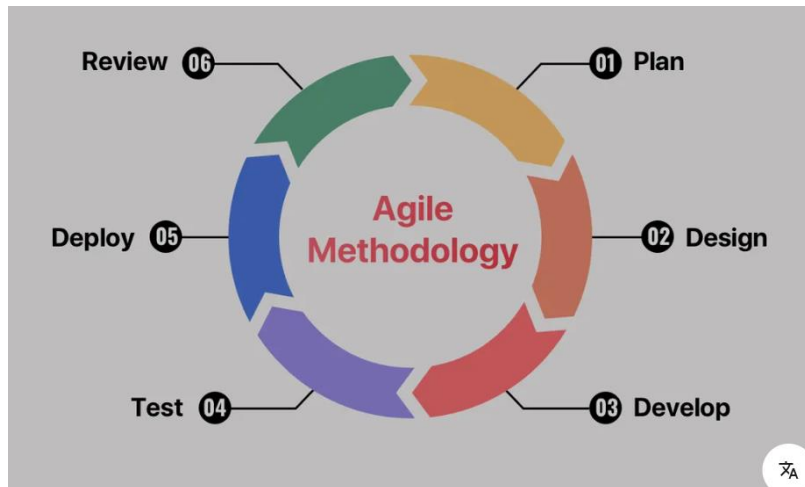
Aviation Software for safety - critical system

### Disadvantages:

difficult for complex project

## Agile Model:

Agile is an iterative and flexible development model that breaks the project into small cycle “sprints”, where teams develop, test, and gather feedback continuously. It embraces change and promotes constant collaboration between developers and customers, making it highly adaptable to evolving project requirements.



### Application:

Mobile apps like Netflix , ecommerce website like amazon, flipkart.

### Advantages:

faster delivery through the sprints.

highly flexible

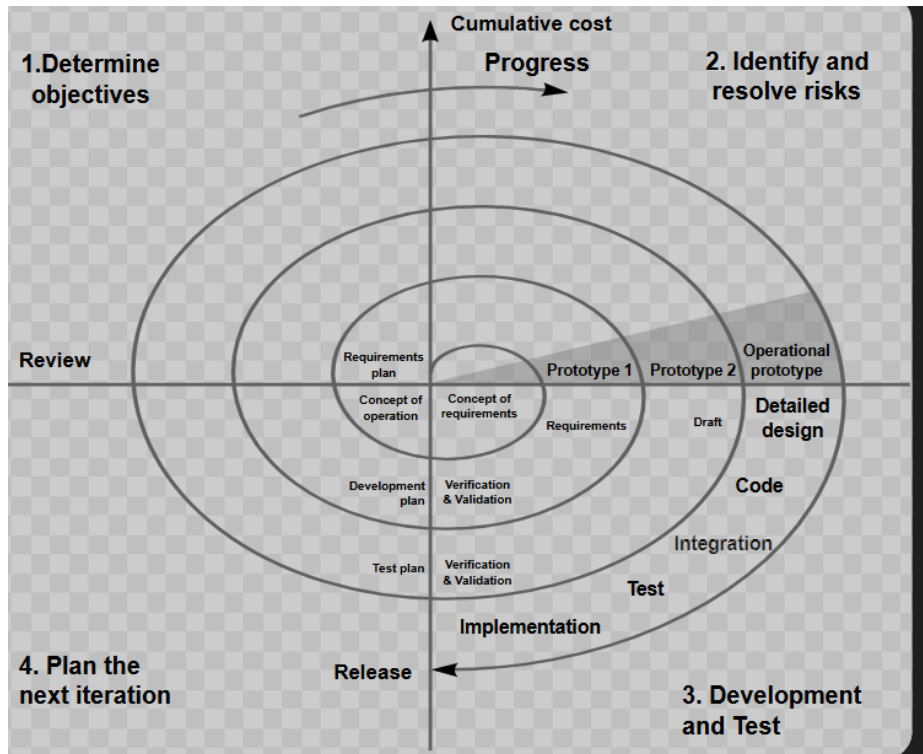
### Disadvantages:

difficult to estimate effort and budget

## Spiral Model:

The Spiral model combines elements of both design and prototyping in stages like waterfall and Iterative model, where the project repeatedly passes through four phases: planning, risk analysis, engineering, and evaluation, like a spiral getting larger.

This model is particularly effective for large, high-risk projects



### Application:

Banking systems for security purpose.  
Department of defense large scale project.

### Advantages:

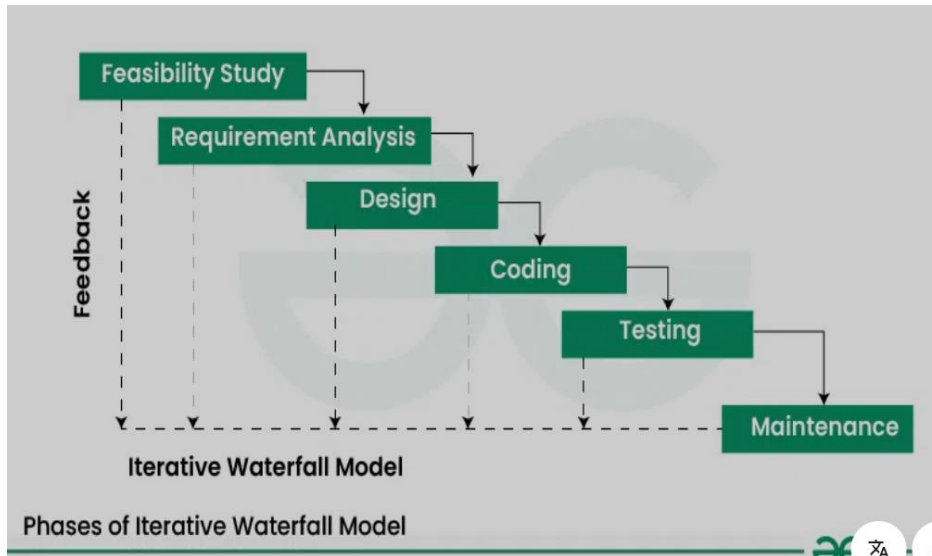
mainly focus on risk management  
large projects

### Disadvantages:

complex and manage of time  
cost is very high

## Iterative Model:

This model breaks down development into smaller, repeating cycles, allowing for flexibility and feedback throughout the process. Each iteration goes through requirements, design, implementation, and testing phases, delivering a working but enhanced version of the product



### Application:

streaming platform like youtube , Instagram

### Advantages:

- better progress visibility
- early working product

### Disadvantages:

- requires customer Involvement
- repeated testing may cause for high costly

---

**Scrum:**

it is an agile framework that breaks down work into fixed-length sprints (2-4 weeks), where cross-functional teams work from a prioritized product backlog.

It employs defined roles, artifacts, and ceremonies to manage development work effectively while promoting transparency and continuous improvement through regular feedback loops.

product Backlog --- Sprint plan --- Sprint Backlog --- Scrum Team --- Sprint review --- Increment

**Scrum Master –**

The Scrum Master serves as a servant-leader who facilitates Scrum practices, removes obstacles, and guides the entire Scrum Team to ensure effective implementation of the Scrum framework.

**Sprint:**

A Sprint is a fixed time-boxed period during which the Scrum Team works to complete a set amount of work selected from the Product Backlog. During each Sprint, the team commits to delivering a potentially shippable product increment while following Scrum ceremonies including Daily Standups, Sprint Review, and Sprint Retrospective.

**what we have to do and what we have avoid while working with sprints:**

Do's :

1. Sprint Goals
2. Daily Communication
3. Quality Standards

DON'T DO :

1. don't add new requirements mid sprint.
2. doing multi task with other project.
3. Process Violation like skipping documentation

**Backlogs in Sprints:**

In Scrum, the Product Backlog is a prioritized list of all desired features, requirements, and fixes maintained by the Product Owner, from which the Development Team selects items during Sprint Planning to create the Sprint Backlog, **Sprint backlogs contain user stories**

The sprint backlog is **a list of tasks identified by the Scrum team to be completed during the Scrum sprint.**

**Stories in Sprints:**

Stories are bite-sized, user-focused requirements that describe desired functionality, which the Scrum Team selects for completion during a Sprint, with each story following the 'As a, I want, So that' format and including acceptance criteria to define when it's considered done.

On an agile team, stories are **something the team can commit to finish within a one- or two-week sprint.**

Oftentimes, developers would work on dozens of stories a month. Epics, in contrast, are few in number and take longer to complete.

## Scrum Artifacts:

**Product Backlog :** The Product Backlog is a prioritized list of all the work that the team needs to do to deliver the product. It's a living document, constantly being updated with new ideas, features, and bug fixes.

**Sprint Backlog:** The Sprint Backlog is a subset of the Product Backlog, containing the items that the development team plans to work on during a specific Sprint (a fixed time period for completing a defined amount of work).

**Increment:** The Increment is the result of a Sprint, containing all the completed Product Backlog items that meet the Definition of Done.

---

## Networking:

**ports:** Ports are logical endpoints for communication. They help a device know which service or application should handle incoming or outgoing data. Which service?  
works at Transport Layer , it is numerical.  
Ex: Port 80, Port 443, Port 25

**protocols:** Protocols are rules and standards that define how data is transmitted over a network. Each protocol serves a specific purpose. How to communicate .  
it operates at multiple OSI layers  
Ex: HTTP, HTTPS, FTP, TCP, UDP

## Types of Networks:

- Personal area network, or PAN.
- Local area network, or LAN.
- Metropolitan area network, or MAN.
- Wide area network, or WAN.

## Types of Server

### Web Server

A web server hosts websites and delivers web pages to users through HTTP or HTTPS protocols. For example, Apache or Nginx are popular web servers.

### Application Server

An application server runs the business logic for apps. It works as a middle layer between the web interface and the database, ensuring smooth processing of client requests.

### Database Server

his server stores and manages data. Applications connect to it to retrieve or update information. MySQL, Oracle, and SQL Server are widely used.

### File Server

A file server provides centralized storage so users on a network can upload, download, and share files easily.



**Mail Server**

Mail servers handle the sending and receiving of emails using protocols like SMTP, POP3, and IMAP. Microsoft Exchange is a common example.

**DNS Server**

A DNS server translates human-readable domain names like google.com into IP addresses, helping us access websites without remembering numbers.

**DHCP Server**

This server automatically assigns IP addresses and network configurations to devices when they connect to the network, reducing manual setup.

**Proxy Server**

A proxy acts as a gateway between the user and the internet. It helps with content filtering, caching, and improving security.

**FTP Server**

An FTP server allows file transfers between devices using the File Transfer Protocol. It's commonly used for uploading and downloading large files remotely.

**Virtual Server**

These are software-based servers running on a physical machine using virtualization. They allow multiple servers to run independently on the same hardware.

What do you know about DNS? Domain Name System

DNS stands for Domain Name System, and it's like the phonebook of the internet. It translates human-readable domain names like www.google.com into IP addresses like 142.250.195.36, which computers use to communicate.

It plays a key role in accessing any online service smoothly, and without DNS, users would have to remember complex IP addresses for each website

**Main functions of DNS:**

Converts domain names to IP addresses

Supports email delivery by resolving mail servers , Helps in load balancing and content distribution

Types of DNS servers involved:

DNS Resolver (Client-side) ,

Root DNS Server ,TLD Server (.com, .org, etc.) ,

Authoritative DNS Server (gives the final IP)

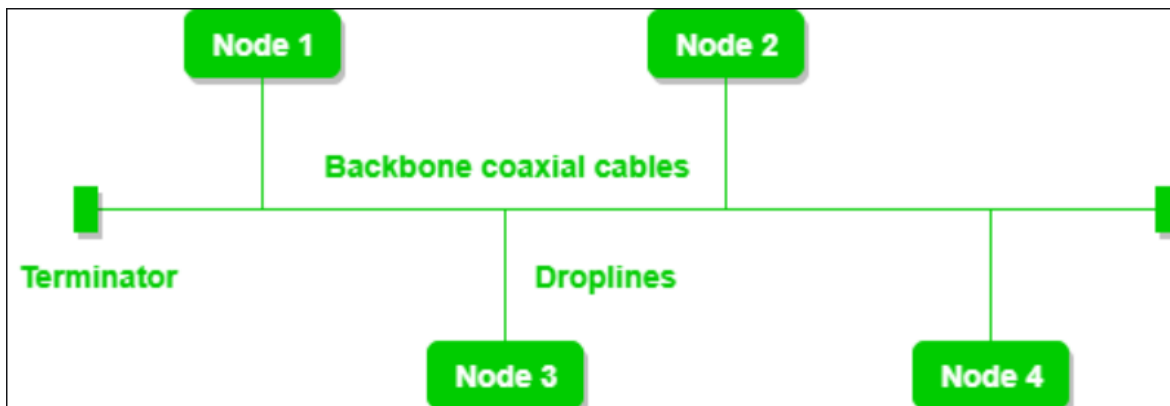
## Topology:

Network topology is the way devices are connected in a network. It defines how these components are connected and how data transfer between the network.

## Types of Topology:

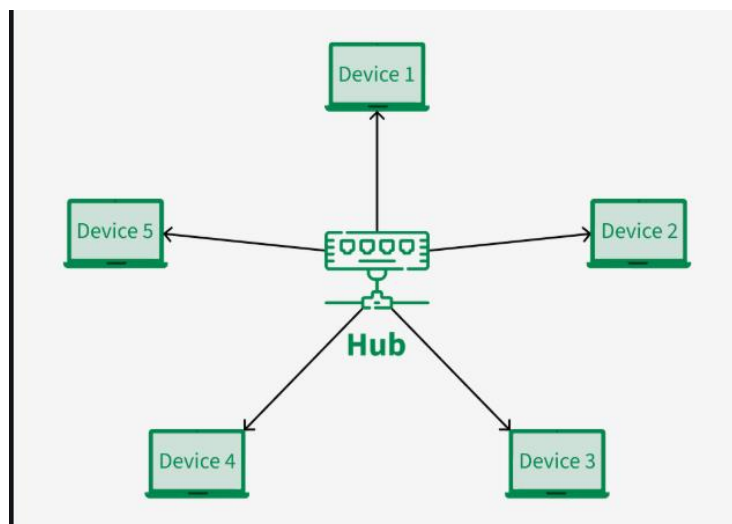
### Bus Topology

All devices are connected to a single central cable called the bus. Data travels in both directions, but if the main cable fails, the whole network goes down.



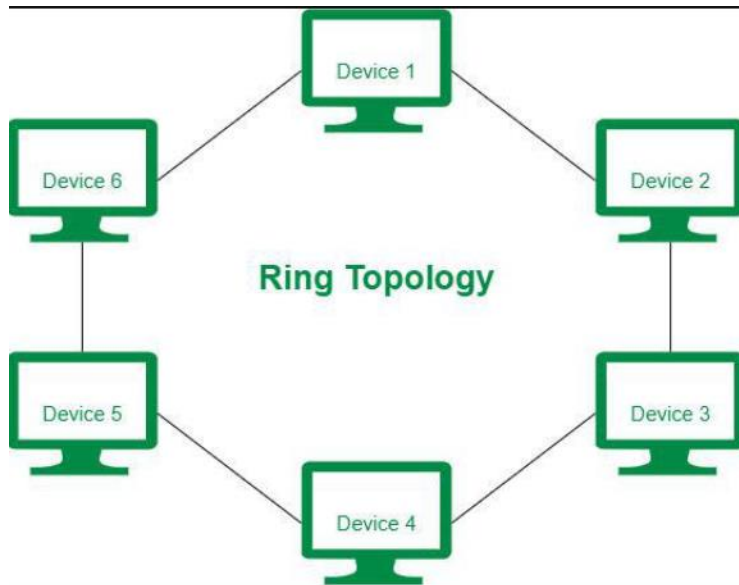
### Star Topology

All devices connect to a central device, usually a switch or hub. If one device fails, it doesn't affect the rest, but if the central device fails, the whole network is impacted.



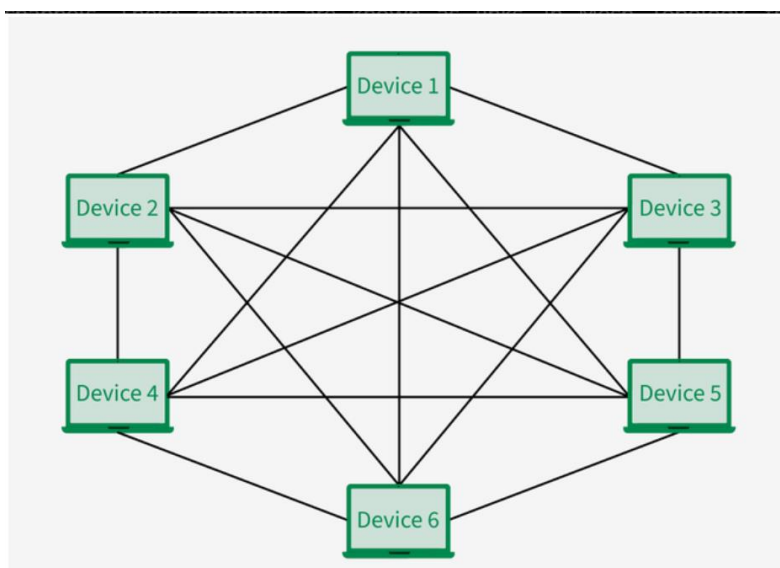
## Ring Topology:

Devices are connected in a circular fashion, and data travels in one direction around the ring. It's simple but if one device or connection breaks.



## Mesh Topology:

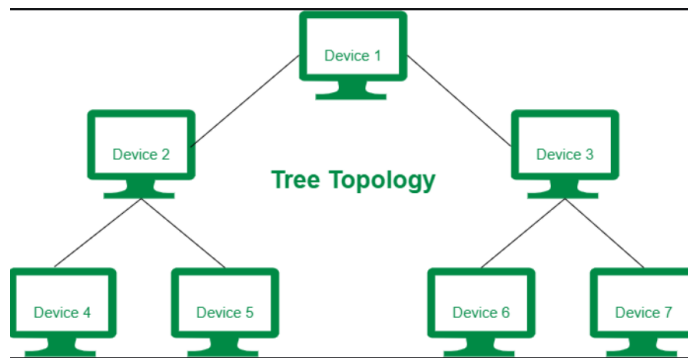
Every device is connected to every other device. It provides high redundancy and reliability, but it's expensive and complex to implement. These channels are known as links.



## Tree Topology

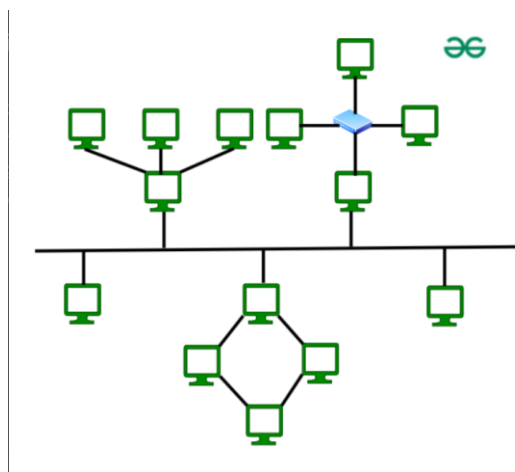
A combination of star and bus topologies. Groups of star-configured networks are connected to a linear bus backbone.

It is a multi-point connection and a non-robust topology because if the backbone fails the topology crashes.



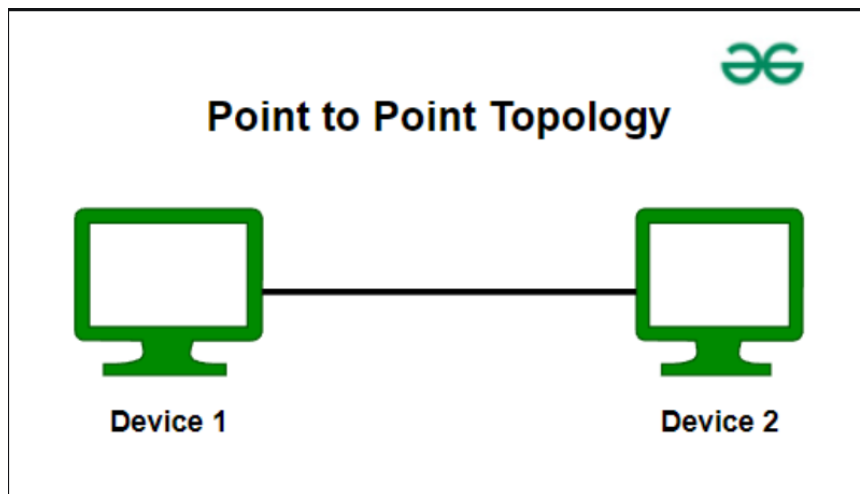
## Hybrid Topology:

A mix of two or more different topologies to suit specific network needs. This is used when the nodes are free to take any form. These can be individuals or a combination of various types of topologies.



## Point to Point Topology :

Point-to-point topology is a type of topology that works on the functionality of the sender and receiver. It is the simplest communication between two nodes, in which one is the sender and the other one is the receiver.



---

## OSI Model (Open System Interconnection):

The OSI (Open Systems Interconnection) Model is responsible to explain how different computer systems communicate over a network.

There are 7 layers:

### 1. Physical Layer:

The lowest layer of the OSI reference model is the Physical Layer. It is responsible for the actual physical connection between the devices.

### 2. Data Link Layer:

The data link layer is responsible for the node-to-node delivery of the message. The main function of this layer is to make sure data transfer is one node to another using the MAC addresses.

### 3. Network Layer:

This layer works for the transmission of data from one host to the other located in different networks. The sender and receiver's IP address are placed in the header by the network layer.

### 4. Transport Layer:

It is responsible for the end-to-end delivery of the complete message. It will provide acknowledgment of the successful data transmission and re-transmits the data if an error is found.

**5.Session Layer:**

in the OSI Model is responsible for the establishment of connections, management of connections, terminations of sessions between two devices. It also provides authentication and security.

**6.Presentation Layer:**

this layer is also called the Translation layer. It translate, encrypts or compress data so application can understand it.

**7.Application Layer:**

These applications produce the data to be transferred over the network. Using applications like browsers,email.

# Thank You