UNIT NO. 01
COMPUTER SYSTEMS

Short Questions

Q.No.01: What is the basic unit of data representation in a digital computer?

Ans: In a computer, all data is stored in binary form. A binary digit has two possible states, 1 and 0.A binary digit is known as a bit. A bit is the smallest unit of data a computer can use. The binary unit system is used to describe bigger numbers too. Eight bits are known as a byte.

Q.No.02: How are analog signals different from digital signals?

Here are the important difference between Analog and Digital transmission:

Analog	Digital
An analog signal is a continuous signal that represents physical measurements.	Digital signals are time separated signals which are generated using digital modulation.
It is denoted by sine waves	It is denoted by square waves
It uses a continuous range of values that help you to represent information.	Digital signal uses discrete 0 and 1 to represent information.
Temperature sensors, FM radio signals, Photocells, Light sensor, Resistive touch screen are examples of Analog signals.	Computers, CDs, DVDs are some examples of Digital signal.
The analog signal bandwidth is low	The digital signal bandwidth is high.
Analog signals are deteriorated by noise throughout transmission as well as write/read cycle.	Relatively a noise-immune system without deterioration during the transmission process and write/read cycle.
Analog hardware never offers flexible implementation.	Digital hardware offers flexibility in implementation.
It is suited for audio and video transmission.	It is suited for Computing and digital electronics.
Processing can be done in real-time and consumes lesser bandwidth compared to a digital signal.	It never gives a guarantee that digital signal processing can be performed in real time.

Analog	**Digital**
Analog instruments usually have s scale which is cramped at lower end and gives considerable observational errors.	Digital instruments never cause any kind of observational errors.
Analog signal doesn't offer any fixed range.	Digital signal has a finite number, i.e., 0 and 1.

Q.No.03: Why have analog signals declined in usage with the advent of digital signals?

Ans: Due to following reasons:

- Digital signals can convey information with less noise, distortion, and interference.
- Digital signals can be reproduced easily in mass quantities at comparatively low costs.
- Digital signal processing is safer because digital information are often easily encrypted and compressed.
- Digital systems are more accurate, and therefore the probability of error occurrence are often reduced by employing error detection and correction codes.
- Digital signals are often easily stored on any magnetic media or optical media using semiconductor chips.
- Digital signals can be transmitted over long distances.

Q.No.04: What is the primary function of logic gates in digital electronics?

Ans: A logic gate is an electronic circuit that performs Boolean operations (logical functions) on one or more inputs to produce a single binary output (boolean expression). They are the building blocks of digital electronics and are used to process and manipulate digital signals. Logic gates are typically made up of transistors, which are tiny electronic switches that can be used to turn the flow of electricity on or off. By combining these switches in different ways, logic gates can be created to perform a variety of logical operations.

Q.No.05: Explain the functions of AND, NOT and NAND gate.

Ans: AND Gate: The AND gate is a digital logic gate that performs the logical operation on two or more binary inputs. The output of an AND gate is **only HIGH (1) when all of its inputs are HIGH**. If any of the inputs are LOW (0), then the output of the AND gate is also LOW (0).

* you may see the following terms used interchangeably

1 | HIGH | ON

O | LOW | OFF

The Boolean expression given for an AND gate is that for Logical Multiplication, which is denoted by a single dot or full stop symbol (**.**) giving us the Boolean expression of: Z = A**.**B

NOT Gate he NOT Gate is a digital logic gate that **performs the logical negation operation** on a single binary input to produce a single binary output. As the NOT Gate inverts the input signal, it is also known as *Inverter*.

The output of a NOT Gate is **HIGH (1) only when its input is LOW (0)**. If the input signal is HIGH (1), then the output of the NOT Gate is LOW (0).

The Boolean expression given for a NOT operation is (–) bar above the output, giving us the Boolean expression of $\bar{A}$. So if the input signal is A, the output signal will be $\bar{A}$. If A is 1, then $\bar{A}$ = 0; if A is 0, then $\bar{A}$ = 1.

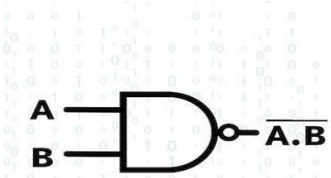
The following terms are used interchangeably

1 | HIGH | ON

O | LOW | OFF

In the truth table ON = 1 and OFF = 0. Let's show you what that looks like.

NAND Gate: The NAND Gate has a normal HIGH logic level and only goes LOW when all inputs are HIGH. The truth table for an NAND Gate is:



2 input NAND gate

A	B	$\overline{A.B}$
0	0	1
0	1	1
1	0	1
1	1	0

Q.No.06: What is the purpose of the Analysis phase in SDLC?

Ans: During this software development lifecycle phase, the specialists meticulously collect precise requirements from the customer to present a solution fine-tuned to their needs. Any unclarities must be elucidated in this stage only.

The analysis phase also gathers business requirements and identifies any potential risks. This step in SDLC also includes a feasibility study, which defines all fortes and weak points of the project to assess the overall project viability.

The goals you achieve at this stage are identified as the system of functions your business needs or wants to develop and implement. For that, software developers complete three primary activities:

- Listing Business Needs Or Requirements
- Developing Process Diagrams Or A Development Pipeline
- Performing The Analysis

The analysis stage includes:

- Clarifying Specific Details Required For Software Development
- Determining Initial Prototype Ideas:
    - What Functions Could Be The Most Suitable For The New Product
    - What USPs (Unique Selling Points) Your Future Software Should Have To Compete Well On The Market.

This way, you can define the main requirements, what tools and approaches to use, and how to reach your business goals most efficiently.

Thus, the analysis phase helps you understand your core business needs and what you should do to fulfill them.

Q.No.07: What is the main goal of White Box Testing?

Ans: White box testing techniques analyze the internal structures the used data structures, internal design, code structure, and the working of the software rather than just the functionality as in black box testing. It is also called glass box testing clear box testing or structural testing. White Box Testing is also known as transparent testing or open box testing.

Q.No.08: Which model follows a linear-sequential life cycle approach?

Ans: The Waterfall Model was the first Process Model to be introduced. It is also referred to as a **linear-sequential life cycle model**. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

The Waterfall model is the earliest SDLC approach that was used for software development. The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap.

Q.No.09: Write two advantages of Waterfall Model.

Ans: Following are two advantages of Waterfall Model:

Properly Defined: In the classical waterfall model, each stage in the model is clearly defined.

Properly Documented: Processes, actions, and results are very well documented.

Q.No.10: In which phase do we assess whether the proposed project is technically feasible?

Ans: In feasibility phase we assess the practicality of implementing a proposed project from a technological point. It involves evaluating whether the necessary technology(Hardware, Software), tools and resources are available or can be developed to support the system.

Q.No.11: What is the main focus of Black Box Testing?

Ans: Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing.

Q.No.12: Write down two disadvantages of Agile Model.

Ans: Following are disadvantages of Agile Model:

Dependency on Customer Availability: Agile greatly depends on ongoing customer and stakeholder feedback and participation. Customers who are unavailable or who don't know enough about the domain can impede development and slow it down.

Scaling Agile: While Agile works effectively for small to medium-sized teams working on relatively basic projects, scaling Agile methods to bigger teams or more complicated projects can be more difficult. As the project grows, it gets harder to maintain coordination, alignment, and communication.

Q.No.13: List down the Phases of Waterfall Model.

Ans: Waterfall Model has five phases:

- Requirements
- Design
- Development
- Testing
- Deployment
- Maintenance

Q.No.14: List down the phases of Agile Model.

Ans: Agile Model has following phases:

- Requirements gathering
- Planning
- Design
- Implementation
- Testing
- Deployment
- Maintenance

Q.No.15: What is Physical topology?

Ans: Physical topology indicates the arrangement of different elements of a network. It reflects the physical layout of devices and cables to form a connected network. It is concerned with the essentials of the network ignoring minute details like transfer of data and device type. The pattern of arrangement of nodes (computers) and network cables depends on the ease of installation and setup of the network. It affects cost and bandwidth

capacity based on a solution of devices. It takes into account the placement of nodes and the distance between them. Devices can be arranged to form a ring (Ring Topology) or linearly connected in a line called Bus Topology.

Q.No.16: Name two examples of Infrastructure as Service(IaaS).

Ans: Following are examples of Iaas:
storage, networking and virtualization.

Q.No.17: Write down two advantages of Ring topology.

Ans: Following are two advantages of Ring Topology:

Network Management: Faulty devices can be removed without affecting the entire network.

Reliable: It is more reliable network topology as communication is not dependent on a single host.

Q.No.18: Name few Public Cloud Service Providers.

Ans: Following are Public Cloud Service Providers:

1. Amazon Web Services (AWS)
2. Microsoft Azure
3. Google Cloud Platform (GCP)
4. Alibaba Cloud
5. Oracle Cloud
6. IBM Cloud (Kyndryl)
7. Tencent Cloud

Q.No.19: What is the primary advantage of symmetric encryption?

Ans: The main advantage of symmetric encryption over asymmetric encryption is that it is fast and efficient for large amounts of data.

Q.No.20: Which encryption method is suitable for encrypting large amount of data efficiently?

Ans: Both symmetric and asymmetric encryption methods can be secure when used properly. However, symmetric encryption is generally considered to be more secure for encrypting large amounts of data.

Q.No.21: What is the primary purpose of cryptography?

Ans: Cryptography ensures confidentiality by encrypting sent messages using an algorithm with a key only known to the sender and recipient. A common example of this is the messaging tool WhatsApp, which encrypts conversations between people to ensure they cannot be hacked or intercepted.

Q.No.22: What is encryption key?

Ans: In cryptography, a key is a string of characters used within an encryption algorithm for altering data so that it appears random. Like a physical key, it locks (encrypts) data so that only someone with the right key can unlock (decrypt) it. The original data is known as the *plaintext*, and the data after the key encrypts it is known as the *ciphertext*.

The formula:

plaintext+key = ciphertext

Q.No.23: What do you know about plaintext, cipher text, encryption and decryption?

Ans: Plaintext: This is the original, human-readable form of the data that you want to protect. It can be any type of digital information, such as text, files, or communication message.

Cipher Text: Cipher is an algorithm which is applied to plain text to get ciphertext. It is the unreadable output of an encryption algorithm. The term "cipher" is sometimes used as an alternative term for ciphertext. Ciphertext is not understandable until it has been converted into plain text using a key.

Encryption: Encryption is the basic building block of data security. It is the simplest and most important way to ensure a computer system's information can't be stolen and read by someone who wants to use it for malicious purposes.

Data security encryption is widely used by individual users and large corporations to protect user information sent between a browser and a server. That information could include everything from payment data to personal information.

Decryption: The conversion of encrypted data into its original form is called Decryption. It is generally a reverse process of encryption. It decodes the encrypted information so that an authorized user can only decrypt the data because decryption requires a secret key or password.

Q.No.24: What is the principle of duality in Boolean Algebra and why is it important in digital logic?

Ans: According to the duality principle, if we have postulates or if we have theorems of Boolean Algebra for any one type of operation then the operation can be converted into another type of operation.

In other words AND can be converted to OR and OR can be converted into AND
We can interchange '0 with 1', '1 with 0', '(+) sign with (.) sign' and '(.) sign with (+) sign' to perform dual operation. T
This principle ensures that if a theorem is proved using postulates of Boolean algebra, then the dual of this theorem automatically holds and there is no requirement of proving it separately.
Principle of Duality is a very important principle used in Boolean algebra. This states that starting with a Boolean relation, another Boolean relation can be derived by :

1. Changing each OR sign (+) to an AND sign(.).
2. Changing each AND sign (.) to an OR sign(+).
3. Replacing each 0 by 1 and each 1 by 0

Q.No.25: How memory circuits use logic gates? Give their significance in digital systems.
Ans: Memory circuits in digital systems are typically constructed using logic gates, which are the building blocks of digital circuits. Logic gates such as AND, OR, NOT, NAND, and NOR gates are combined in specific configurations to create memory circuits. These memory circuits store binary data in the form of 0s and 1s by utilizing the states of the logic gates to represent the data.
The significance of memory circuits in digital systems lies in their ability to store and retrieve data accurately and efficiently. Memory circuits enable computers and other digital devices to store instructions, data, and intermediate results during processing. They play a crucial role in the operation of digital systems by providing temporary or permanent storage of information, which is essential for the functioning of various applications and processes

Q.No.26: Give three uses of logic gates.
Ans: Following are the uses of logic gates:

- Logic gates form the basis of digital computers. Combinations of logic gates are used to perform arithmetic and logical operations, enabling the execution of complex tasks.
- Used in digital clocks and other devices, utilize logic gates to control the display of numbers and characters.
- Logic gates play a role in the design of communication systems, including encoding and decoding of data in communication protocols.

Q.No.27: What is the primary purpose of the SDLC?

Ans: SDLC stands for Software Development Life Cycle. It is a process followed by software development teams to design, develop, test, and deploy high-quality software applications. The purpose of SDLC is to provide a structured approach to developing software that ensures the final product meets the requirements of the stakeholders and is delivered on time and within budget.

Q.No.28: Name different phases of SDLC.

Ans: The following are phases or steps in SDLC:

- Defining the problem
- Planning phase
- Feasibility study
- Analysis phase
- Requirements Engineering
- Designing Phase
- Development/Coding phase
- Testing/Verification phase
- Deployment/Implementation phase
- Documentation phase
- Maintenance/Support phase

Q.No.29:Why feasibility study is important in SDLC? Give three reasons.

Ans:A feasibility study is a report that assesses a number of market factors that dictate project viability. The report also provides guidelines for "right-sizing" the facility concept to meet market conditions and definitions of success. Feasibility studies examine your market including potential facility users, competitors offering similar services, potential facility costs and revenues, and options for project development. In short, a feasibility study will help determine you or your community's definition of success, the sustainability of your model, and help you define your community's goals.

Reasons:

Feasibility Studies Help Define Your Goals and Objectives
Feasibility Studies Help You Develop A Plan
Feasibility Studies Help Execute That Plan

Q.No.30: How does the design phase contribute to the development of a software system?

Ans: The design phase is very important step in making software. This is where we plan how the software will be built based on what we learned during the Analysis phase. We use something called Unified Modeling Language(UML) and different design patterns to help us in this phase. We usually work on two main things :

Algorithms: These are step by step instructions that tell the computer how to do something. Its like a recipe that tells you exactly what to do to make a dish.

Flowchart: This is a visual representation of how the software will work. It shows the different steps and decisions the software will make to achieve its goals.

Q.No.31: What is the significance of testing/verification in SDLC?

Ans: Software testing is a crucial activity in the software development life cycle that aims to evaluate and improve the quality of software products. Thorough testing is essential to ensure software systems function correctly, are secure, meet stakeholders' needs, and ultimately provide value to end users.

Properly planned and executed testing is invaluable for reducing project risk, providing confidence in the software quality, meeting compliance needs, ensuring satisfied users, enabling continuous improvement, and reducing overall costs. The importance of effective testing cannot be overstated when developing and maintaining complex, reliable software systems in today's world.

Q.No.32: Give three advantages and two disadvantages of Bus Topology in networking?

Ans: Following are three advantages of Bus Topology:

- It works very efficiently well when there is a small network.
- It is easy to connect or remove devices in this network without affecting any other device.
- Very cost-effective as compared to other network topology i.e. mesh and star

Two disadvantages of Bus Topology:

- Bus topology is not good for large networks.
- If the main cable is damaged, the whole network fails or splits into two.

Q.No.33: How does Mesh topology provide redundancy in network communication?

Ans: A mesh topology has multiple connections, making it the most fault tolerant topology available. Every component of the network is connected directly to every other component. Characteristics of a mesh topology are as follows:

A mesh topology provides redundant links across the network.

If a break occurs in a segment of cable, traffic can still be rerouted using the other cables.

Q.No.34:Compare and contrast Horizontal Scalability and Vertical Scalability in cloud computing.

Ans: Horizontal Scaling: Also called 'scaling out', horizontal cloud scaling improves the cloud throughput by adding new computing infrastructure. At a basic level, scaling out can mean adding new computing nodes or machines to enhance the data processing and storage capabilities.

Vertical Scaling: Vertical cloud scaling enhances the technical specifications of existing infrastructure by adding or replacing CPU, HDD, or other components. Decommissioning existing systems and replacing them with higher capability infrastructure would also qualify as vertical scaling or 'scaling up'. The old infrastructure is either discarded for scrap, resold, or repurposed for less intensive business processes. Compared to horizontal scaling, vertical scaling can take longer and may entail a period of downtime. However, scaling up is usually cheaper than scaling out.

Q.No.35: Name three common types of cybersecurity threats.

Ans: Following are common types of cybersecurity threats:

1. Phishing
2. Ransomware
3. Insider threats

Q.No.36: What is the role of encryption in cybersecurity and how does it protect sensitive data?

Ans: The internet is awash with hackers attempting to illegally access networks, devices and data. Cyber-attacks are becoming increasingly frequent, with around a quarter of organisations having identified breaches or attacks at least once a week.

Encryption in network security and other forms of cybersecurity is vital for many reasons.

Security threats – attacks such as denial of service, malware, database invasion and unauthorised internet access are highly prevalent, but can all be averted using cyber security encryption.

Data interception – as data is passed over communication channels such as email, it can be intercepted and stolen. However, if the data is encrypted, then it will be useless to the cyber thief.

Unauthorised access – network intrusions can lead to data record leaks and loss of confidential information. Encryption in network security can, however, avoid any leaked data being accessed.

Virus attacks – when a network or other online resource comes under attack by malware, viruses or Trojan horses, the security of the system or network will be under threat, with the potential for considerable data loss. Encryption will however prevent data being misused for criminal intent.

Q.No.37: Differentiate between symmetric and asymmetric encryption method.

Ans: Difference between Symmetric and Asymmetric Encryption Methods:

Symmetric Key Encryption	**Asymmetric Key Encryption**
It only requires a single key for both encryption and decryption.	It requires two keys, a public key and a private key, one to encrypt and the other to decrypt.
The size of ciphertext is the same or smaller than the original plaintext.	The size of ciphertext is the same or larger than the original plaintext.
The encryption process is very fast.	The encryption process is slow.
It is used when a large amount of data needs to be transferred.	It is used to transfer small amount of data.
It only provides confidentiality.	It provides confidentiality, authenticity, and non-repudiation.
The length of key used is 128 or 256 bits	The length of key used is 2048 or higher

Symmetric Key Encryption	**Asymmetric Key Encryption**
In symmetric key encryption, resource utilization is low compared to asymmetric key encryption.	In asymmetric key encryption, resource utilization is high.
It is efficient as it is used for handling large amount of data.	It is comparatively less efficient as it can handle a small amount of data.
Security is lower as only one key is used for both encryption and decryption purposes.	Security is higher as two keys are used, one for encryption and the other for decryption.

Q.No.38: Why is it essential to keep your software up to date in terms of cybersecurity?
Ans: It's important to keep your software up to date because updates enhance existing features, patch security flaws, add new security features, fix bug issues and improve performance for devices.

Q.No.39: What is 2FA(Two-Factor Authentication)? Give its importance in securing user accounts.
Ans: Two-factor authentication (2FA) is a security system that requires two distinct forms of identification in order to access something.
Two-factor authentication can be used to strengthen the security of an online account, a smartphone, or even a door. 2FA does this by requiring two types of information from the user—a password or personal identification number (PIN), a code sent to the user's smartphone (called a message authentication code), or a fingerprint—before whatever is being secured can be accessed.

Examples of Two-Factor Authentication (2FA)
Apple account holders can use 2FA to ensure that accounts can only be accessed from trusted devices. If a user tries to log in to their iCloud account from a different computer, the user will need the password, but also a multi-digit code that Apple will send to one of the user's devices, such as their iPhone.

Q.No.40: What is primary purpose of firewall in network security and how it work?
Ans: A firewall acts as a network filter and based on the predefined security rules, it continuously monitors and controls the incoming and outgoing traffic. As an example, a rule can be set in the firewall of a school LAN, that a student cannot access data from the finance server, while the school accountant can access the finance server.
Q.No.41: What are the characteristics of a strong password? Give two examples.
Ans: A strong password possess the following characterstics:
Length: A strong password should be long i.e 10 to 12 digits or even more.
Complexity: It should contain a mix of uppercase and lowercase letters, numbers and special characters($,#,?) etc.
Unpredictability: Avoid guessable information like name, DOB or common phrases.
Uniqueness: Use different passwords for different accounts.
Examples of Strong passwords:

1. P@SSwoRD30)L
2. 0Open%(,%")DooR88

Give long answers to the following Extended Response Questions(ERQs)
Q.No.01: Design logic circuits for the following Boolean functions.
i) E1=$(\boldsymbol{A}+\boldsymbol{B}).(\boldsymbol{A}+\boldsymbol{B})$
ii)E2=$(\boldsymbol{A}.\boldsymbol{B})+(\boldsymbol{A}+\boldsymbol{B}).\boldsymbol{C}$
iii)E3=$(\boldsymbol{A}+\boldsymbol{B}+\boldsymbol{C})+\boldsymbol{A}.(\boldsymbol{C}+\boldsymbol{B})$
iv)E4=$(\boldsymbol{A}+\boldsymbol{B}).\boldsymbol{B}+(\boldsymbol{A}+\boldsymbol{C})$
v)E5=xyz+xyz+xyz+xyz
vi)E5=xz+xy
Q.No.02: Draw Truth tables for the Boolean functions in Q.No.01
Q.No.03: Simplify the following Boolean Functions using K-Maps method.
i)E1= $(\boldsymbol{A}.\boldsymbol{B})+(\boldsymbol{A}.\boldsymbol{B})+(\boldsymbol{A}.\boldsymbol{B})$
ii)E2=$(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})+(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})+(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})+(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})$
iii)E3=$(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})+(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})+(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})+(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})$
iv)E4=$(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})+(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})+(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})+(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})+(\boldsymbol{A}.\boldsymbol{B}.\boldsymbol{C})$
v)E5=xyz+xyz+xyz+xyz
vi)E6=xy+xy

Q.No.04: Compare and contrast the Waterfall model and Agile model in software development. Which one do you think is more suitable for modern software development and why?

Ans: Agile Model: Agile methodology is a modern approach to project management that emphasizes flexibility, collaboration, and incremental delivery. It is designed to respond to unpredictability through iterative cycles of planning, executing, and evaluating. Below are Agile Methodology Principles.

- **Iterative and Incremental**: Agile projects are divided into small iterations or sprints, typically 1-4 weeks long. Each iteration results in a potentially usable product feature.
- **Customer Collaboration**: Agile teams prioritize customer collaboration and feedback. They work closely with stakeholders to understand and deliver what the customer truly needs.
- **Adaptive to Change**: Agile embraces changes in requirements throughout the project lifecycle. It encourages continuous adaptation and flexible responses to change.
- **Empowered Teams**: Agile teams are self-organizing and cross-functional, with members from different disciplines working together to achieve project goals.
- **Frequent Delivery of Working Software**: Agile projects prioritize delivering working software frequently, typically every few weeks, to maximize customer value.
- **Continuous Improvement**: Agile promotes continuous improvement and learning. Teams regularly reflect on their processes and adjust them to improve efficiency and effectiveness.
- **Transparent and Open Communication**: Agile fosters transparent communication within teams and with stakeholders. Information is shared openly to build trust and alignment.
- **Focus on Quality**: Agile teams emphasize sustainable development and strive to maintain a constant pace. They maintain high standards of work through continuous attention to technical excellence and good design.
- **Early and Predictable Delivery**: Agile provides early and predictable delivery of products, with incremental releases that allow for more realistic planning and improved stakeholder satisfaction.
- **Close Collaboration Between Developers and Business Users**: Agile methodologies emphasize close daily cooperation between developers and business people.

Waterfall Model: It is one of the easiest and traditional model to manage. Because of its traditional development nature, each phase has specific deliverables and a review process. The waterfall model works well in smaller size projects where requirements are easily understandable.

The waterfall model is a universally accepted SDLC model. In this method, the whole process of software development is divided into various phases. The development in the waterfall model is seen as flowing steadily downwards (like a waterfall) as it is a continuous software development model. This model is named "Waterfall Model", because its diagrammatic representation resembles a cascade of waterfalls. Some important points related to the waterfall model are listed as follows -

- Waterfall model is not an ideal model to develop a large scale project size.
- The requirements in the waterfall model should be clear cut at the beginning time; otherwise, it may lead to a less effective method.
- In the waterfall model, it is hard to move back in order to make changes in the previous phase.
- The testing process in the waterfall model starts after the completion of development. So, there is a high chance of bugs to be found later in the project development.

There are several parameters to consider when choosing the right methodology for your project. No methodology can be considered better than the other; it all depends on the factors which are responsible for driving the project. The following mentioned aspects can help a team make a trade-off between the two methodologies.

- For project size and complexity, for small projects, Agile seems a desirable choice over Waterfall as it will be less chaotic while handling changes in the project.
- Requirements, if a project requires a strict protocol to be followed to generate a desirable outcome, the Waterfall model serves well in such situations, unlike Agile, which is suitable for handling constant changes in a project.
- Client involvement, if timely feedback is required from the client at different ends of the project, Agile would make a better choice in this situation in comparison to the Waterfall model.

- Pattern of delivery, Agile is involved in projects which deliver results in parts, whereas Waterfall model delivers the outcome all at once.

Q.No.05: Discuss the role of requirements engineering in SDLC. What are the challenges and benefits of gathering and managing requirements effectively?

Ans: Requirements Engineering: A systematic and strict approach to the definition, creation, and verification of requirements for a software system is known as requirements engineering. To guarantee the effective creation of a software product, the requirements engineering process entails several tasks that help in understanding, recording, and managing the demands of stakeholders.

Requirements Gathering: Requirements elicitation is the process of gathering information about the needs and expectations of stakeholders for a software system. This is the first step in the requirements engineering process and it is critical to the success of the software development project. The goal of this step is to understand the problem that the software system is intended to solve and the needs and expectations of the stakeholders who will use the system.

Several techniques can be used to elicit requirements, including:

- **Interviews**: These are one-on-one conversations with stakeholders to gather information about their needs and expectations.
- **Surveys**: These are questionnaires that are distributed to stakeholders to gather information about their needs and expectations.
- **Focus Groups**: These are small groups of stakeholders who are brought together to discuss their needs and expectations for the software system.
- **Observation**: This technique involves observing the stakeholders in their work environment to gather information about their needs and expectations.
- **Prototyping**: This technique involves creating a working model of the software system, which can be used to gather feedback from stakeholders and to validate requirements.

Requirements Verification and Validation: Requirements verification and validation (V&V) is the process of checking that the requirements for a software system are complete, consistent, and accurate and that they meet the needs and expectations of the stakeholders. The goal of V&V is to ensure that the software system being developed meets the requirements and that it is developed on time, within budget, and to the required quality.

It's important to note that V&V is not a one-time process, but it should be integrated and continue throughout the software development process and even in the maintenance stage.

Requirements management: Requirement management is the process of analyzing, documenting, tracking, prioritizing, and agreeing on the requirement and controlling the communication with relevant stakeholders. This stage takes care of the changing nature of requirements. It should be ensured that the SRS is as modifiable as possible to incorporate changes in requirements specified by the end users at later stages too. Modifying the software as per requirements in a systematic and controlled manner is an extremely important part of the requirements engineering process.

Benefits of Requirement Engineering:

- Helps ensure that the software being developed meets the needs and expectations of the stakeholders
- Can help identify potential issues or problems early in the development process, allowing for adjustments to be made before significant
- Helps ensure that the software is developed in a cost-effective and efficient manner
- Can improve communication and collaboration between the development team and stakeholders
- Helps to ensure that the software system meets the needs of all stakeholders.
- Provides an unambiguous description of the requirements, which helps to reduce misunderstandings and errors.
- Helps to identify potential conflicts and contradictions in the requirements, which can be resolved before the software development process begins.
- Helps to ensure that the software system is delivered on time, within budget, and to the required quality standards.
- Provides a solid foundation for the development process, which helps to reduce the risk of failure.

Common Challenges in eliciting requirement:

Eliciting requirements is one of the most challenging aspects of software engineering. Some common challenges include:

1. **Understanding the user's needs:** Requirements are often poorly defined and may change over time, making it difficult for engineers to understand the user's true needs.
2. **Managing stakeholders:** There may be multiple stakeholders with different goals and priorities, making it difficult to satisfy everyone's requirements.
3. **Identifying and mitigating risks:** Engineers must identify and mitigate potential risks associated with the requirements, such as security vulnerabilities or scalability issues.
4. **Handling ambiguity**: Requirements may be ambiguous, inconsistent, or incomplete, making it difficult for engineers to understand what the system should do.
5. **Keeping up with changing technology:** Requirements must be aligned with the latest technology trends and innovations, which can be difficult to predict and keep up with.
6. Maintaining a balance between feasibility, cost and time: Engineers need to balance the feasibility of implementing a requirement, the cost of implementation, and the time required to implement it.
7. **Maintaining traceability:** Engineers need to maintain traceability of requirements throughout the development process to ensure that all requirements are met and any changes are tracked.
8. **Understanding large and complex system requirements is difficult:** The word 'large' represents 2 aspects: Large constraints in terms of security, etc. due to a large number of users and a large number of functions to be implemented.
9. **Undefined system boundaries:** There might be no defined set of implementation requirements. The customer may go on to include several unrelated and unnecessary functions besides the important ones, resulting in an extremely large implementation cost that may exceed the decided budget.
10. **Customers/Stakeholders are not clear about their needs:** Sometimes, the customers themselves may be unsure about the exhaustive list of functionalities they wish to see in the software. This might happen when they have a very basic idea about their needs but haven't planned much about the implementation part.
11. **Conflicting requirements are there:** There is a possibility that two different stakeholders of the project express demands which contradict each other's

implementation. Also, a single stakeholder might also sometimes express two incompatible requirements.

12. **Changing requirements is another issue:** In the case of successive interviews or reviews from the customer, there is a possibility that the customer expresses a change in the initial set of specified requirements. While it is easy to accommodate some of the requirements, it is often difficult to deal with such changing requirements.
13. **Partitioning the system suitably to reduce complexity:** The projects can sometimes be broken down into small modules or functionalities which are then handled by separate teams. Often, more complex and large projects require more partitioning. It needs to be ensured that the partitions are non-overlapping and independent of each other.
14. **Validating and Tracing requirements:** Cross-checking the listed requirements before starting the implementation part is very important. Also, there should be forward as well as backward traceability. For eg, all the entity names should be the same everywhere, i.e., there shouldn't be a case where 'STUDENT' and 'STUDENTS' are used at separate places to refer to the same entity.

Q.No.06: Discuss various deployment or implementation methods provide real world scenarios where each deployment method would be most suitable.

Ans: Once a new system is developed (or purchased), the organization must determine the best method for implementing it. Convincing a group of people to learn and use a new system can be a challenging process. Using the new software and the business processes it gives rise to can have far-reaching effects within the organization.

There are several different methodologies an organization can adopt to implement a new system. Four of the most popular are listed below.

- **Direct Method**. In the direct implementation methodology, the organization selects a particular date that the old system will not be used anymore. On that date, the users begin using the new system, and the old system is unavailable. The advantages of using this methodology are that it is speedy and the least expensive. However, this method is the riskiest as well. If the new system has an operational problem or is not properly prepared, it could prove disastrous for the organization.
- **Pilot implementation**. In this methodology, a subset of the organization (called a pilot group) starts using the new system before the rest of the organization. This has a smaller impact on the company and allows the support team to focus on a smaller group of individuals.

- **Parallel operation**. With the parallel operation, the old and new systems are used simultaneously for a limited period of time. This method is the least risky because the old system is still being used while the new system is essentially being tested. However, this is the most expensive methodology since work is duplicated and support is needed for both systems in full.
- **Phased implementation**. In a phased implementation, different functions of the new application are used as functions from the old system are turned off. This approach allows an organization to move from one system to another slowly.

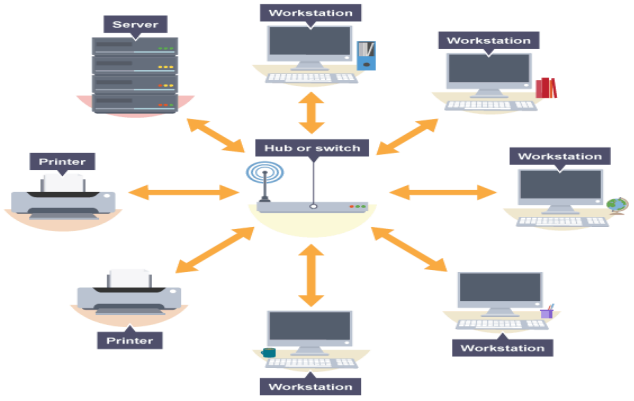
These implementation methodologies depend on the complexity and importance of the old and new systems.

Q.No.07: Explain Bus, Star and Ring network topologies. Give their advantages and disadvantages.

Ans: Star Topology: In star topology each device in the network is connected to a central device called **hub**. Unlike Mesh topology, star topology doesn't allow direct communication between devices, a device must have to communicate through hub.

If one device wants to send data to other device, it has to first send the data to hub and then the hub transmit that data to the designated device.

The **central device is known as hub** and **other devices connected to hub are called clients**. Generally Coaxial cable or RJ-45 cables are used to connect the clients to the hub.



Advantages of Star topology

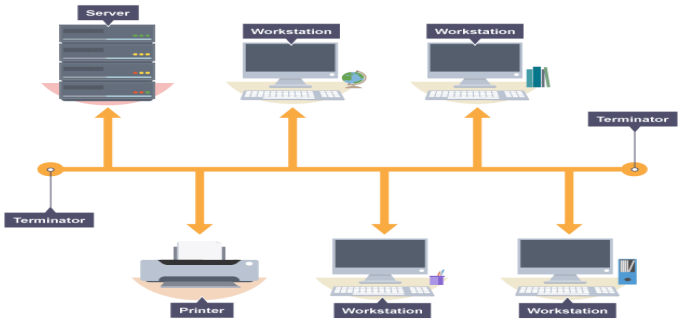
- **Less expensive:** Less expensive because each device only need one I/O port and needs to be connected with hub with one link.
- **Easier to install**
- **Cost effective:** Less amount of cables required because each device needs to be connected with the hub only.

- **Robust:** If one link fails, other links will work just fine.
- **Easy to troubleshoot:** Easy fault detection because the link can be easily identified.
- **Reliable:** Each device is separately connected to the hub, so a connection failure between a device and hub doesn't affect the connection of the other devices.

Disadvantages of Star topology

- If hub goes down everything goes down, none of the devices can work without hub.
- Hub requires more resources and regular maintenance because it is the central system of star topology.
- **Not Scalable:** There is a limit to add new devices as each device increase the load on the central unit (hub or switch). This is why it is not suitable for the large networks.

Bus Topology: In *a* bus topology, all *nodes* in the *network* are connected directly to a central cable that runs up and down the network - this cable is known as the *backbone*. *Data* is sent up and down the backbone until it reaches the correct node.



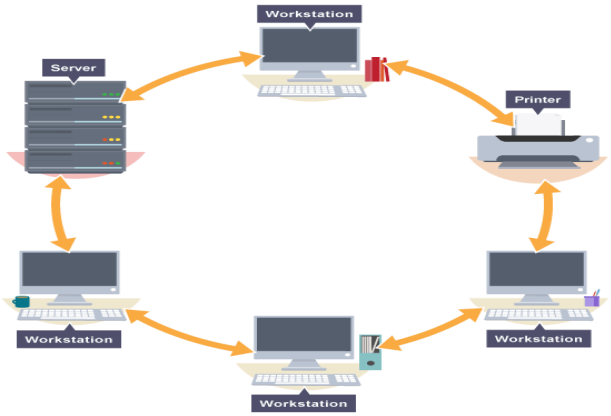
Advantages of bus topology

- **Easy installation**, each cable needs to be connected with backbone cable.
- **Less Expensive: Less cables** are required than Mesh and star topology
- **Limited failure:** The connection failure of one device doesn't affect the connections of other devices on network.

Disadvantages of bus topology

- Difficultly in fault detection.
- **Not scalable** as there is a limit of how many nodes you can connect with backbone cable.
- **Difficult to troubleshoot:** It is difficult to identify the cause of failure.
- **Data collision:** When two or more devices send the data simultaneously then there is a chance of data collision. However this can be solved by implementing CSMA techniques that we discussed above.

Ring Topology: n a ring network topology, nodes are connected in a ring or a loop. Data is sent around the ring, being passed from one computer system to the next until it reaches its destination.



Advantages and disadvantages of using a ring network topology

Having nodes arranged in a ring network topology brings some benefits:

- because data passes around the network in one direction, there are no network collisions
- adding additional nodes has very little impact on bandwidth

However, ring network topologies also have disadvantages:

- if any of the nodes fail, the ring is broken and data cannot be transmitted
- it is difficult to troubleshoot a ring network topology
- because all nodes are wired together, the network must be temporarily stopped to add additional nodes

Q.No.08: In the context of cloud computing, elaborate on the concept of scalability and reliability. How does these concepts contribute to the effectiveness of cloud services? Provide a real-world example.

Ans: Scalability in cloud computing refers to a system or application's ability to handle an increasing workload by adding more resources or nodes to the existing infrastructure. It allows businesses to quickly scale up or scale out to meet the growing demands without hampering performance.

There are three types of scaling in cloud computing:

- Vertical scaling: Adding resources to increase the capacity of a single instance.
- Horizontal scaling: Adding more instances to distribute the workload across multiple machines.
- Diagonal scaling: Combining vertical and horizontal scaling to optimize performance and capacity.

In public cloud environments like AWS, cloud scalability allows users to easily add or remove servers or resources to existing infrastructure. It allows businesses to scale up dynamically or out based on demand, ensuring optimal performance and cost efficiency.

Reliability in context of Cloud Computing: When you access an app or service in the cloud, you can reasonably expect that:

- The app or service is up and running.
- You can access what you need from any device at any time from any location.
- There will be no interruptions or downtime.
- Your connection is secure.
- You will be able to perform the tasks you need to get your job done.

Factors like these measure the reliability of your cloud offerings. In a perfect world, your system would be 100% reliable. But that is probably not an attainable goal. In the real world, things will go wrong. You will see faults from things such as server downtime, software failure, security breaches, user errors, and other unexpected incidents.

Reliability in cloud computing is important for businesses of any size. Buggy software can cause lost productivity, lost revenue, and lost trust in your brand. Before you deploy your applications to the cloud, make sure they are thoroughly tested against a variety of real-world scenarios. This helps to ensure that they are reliable and will meet customer expectations.

Q.No.09: Imagine you are responsible for the cybersecurity of a large organization. Describe a comprehensive cybersecurity strategy that includes multiple layers of defense against various threats.

Ans: Cybersecurity is an intricate field that involves a plethora of different services, tools, and technologies. It elevates network security beyond basic practices, and takes a much more comprehensive approach to network and data protection. This comprehensive approach is known as layered security.

Layered security in cybersecurity is focused on creating as many detection points as possible in order to identify and neutralize cyberthreats and breach attempts. Let's take a closer look at everything involved in creating a layered security strategy.

A cybersecurity strategy using several tactics to back up every aspect of your network's defense with others to ensure all potential vulnerabilities are covered.

Layered security in cybersecurity is all about creating secondary safety nets that bolster network security and mitigate single points of failure throughout the network. By investing in a layered security strategy, you'll minimize the gaps across your network, and have a better chance at identifying and neutralizing cyberthreats before they can cause any damage.

Different tactics to defend different threats:

Advanced Spam Filtering: Advanced spam filtering will help protect your employees from receiving dangerous phishing emails.

But don't email providers have spam filters anyway?

Well, yes, but filters that are free are often lacking in many of the filtering techniques used by advanced filters. Anyone who has a Gmail account will know that spam can still get through, in spite of its filter.

Next-Gen Antivirus and Multi-Layered Network Security: Traditional antivirus solutions lack the capabilities of next-gen antivirus software, which utilize the following technologies:

- **Machine learning:** Files are analyzed using an automated bot that can discover any malicious elements—all without any interruption to the user.
- **Behavior analysis:** Computer processes can be monitored in real-time and detect any abnormal behavior, terminating malicious processes.
- **Threat intelligence:** When a device encounters a threat, every other device under the network will be updated to counter the danger without any need for manual input.

Web Application Firewall: A web application firewall is used to stop threats against your website or applications hosted on your site.

In many cases, business applications are tied into your network, so a WAF can help protect this communication channel.

Website Backup and Restore: t's not just your networks that are vulnerable, your website is too. A solution that allows you to instantly backup and restore your site should the worst happen is absolutely vital, and yet many, many businesses have nothing to protect their sites in the event of a breach.

Multi-Factor Authentication (MFA): MFA is a simple and highly efficient way of ensuring the security of your workers' login credentials. MFA requires the user to have a traditional sign-in method (usually a password), in addition to something more personal, like a fingerprint or text message.

Security Awareness Training: Phishing relies on exploiting end users who don't know what to look for in a spam email.

To address this, it's absolutely crucial that organizations train their employees so that they won't be hoodwinked by a cybercriminal.

Additionally, a layered security strategy provides the following:

- Vigilant security structures that address the sophistication and depth of modern cybersecurity threats with redundancy and thorough monitoring.
- Multiple security solutions and tools that enhance the work of cybersecurity professionals and create a network security posture with effective redundancies.
- Continual education and training that keeps your network safe from the newest cyberthreats to surface.

There is a lot to consider when it comes to creating your cybersecurity and network security strategies, which is why it can be such a powerful defense to implement layered security tactics across your network.

Q.No.10: Explain how scalability and reliability are tested in network systems , outlining the steps involved in each process.

Ans: Testing Scalability In Network System: Testing the scalability of a network system involves assessing its ability to handle increasing levels of workload or user demand while maintaining performance, reliability and efficiency. This testing process typically involves the following steps:

Define Performance Metrics: Determine the Key Performance Indicators(KPIs) that will be used to measure the system's scalability. These metrics may include response time, throughput, resource utilization and system stability under load.

Benchmarking: Compare network performance against industry standards or competitors to assess how well it scales.
Load Testing: Conduct load testing to evaluate the network's performance under heavy traffic conditions. Tools like Apache, JMeter, LoadRunner can simulate a large number of users or devices accessing the network simultaneously.
Stress Testing: Stress testing involves pushing the network beyond its capacity to identify its breakpoints.
Analyze Test Results: Analyze the collected data to identify any bottlenecks, performance degradation or scalability limitations within the network system. Determine the system's ability to scale up or scale out in response to growing demands.
Testing Reliability In Network System: Testing the reliability of a network system involves assessing its ability to consistently deliver services and maintain connectivity without experiencing failures or interruptions. How the reliability of a network system can be tested:
Define Reliability Metrics: Determine the reliability indicators that will be used to measure the system performance. These metrics may include uptime, mean time between failures(MTBF), mean time to repair(MTTR) and error rates.
Execute Fault Injection Tests: Introduce faults or failures into the network system deliberately to observe how it respond under the adverse conditions.
Redundancy Testing: Evaluate the effectiveness of redundancy mechanism, such as load balancing, failover and backup systems. Simulate the failure of redundant components to ensure seamless failover.
Disaster Recovery Testing: Test disaster recovery plans to ensure data and services can be restored in the event of disastrous failures or data breaches.
Documentation and Reporting: Maintain thorough documentation of tests performed and their results. Provide detailed report to stakeholders.

Q.No.11: Discuss the key characteristics of cloud computing, including service models and deployment models with examples.

Ans: Following are the key characteristics of Cloud Computing:

On-Demand Self -Service: Users can provision and manage computing resources as needed, without requiring human intervention from service providers.

Flexibility: Cloud Computing lets users access data or services using internet-enabled devices (such as smartphones and laptops). Whatever you want is instantly available on the cloud, just a click away. Sharing and working on data thus becomes easy and comfortable. Many organizations these days prefer to store their work on cloud systems, as it makes collaboration easy and saves them a lot of costs and resources. Its ever-increasing set of features and services is also accelerating its growth.

Scalability: Scalability is the ability of the system to handle the growing amount of work by adding resources to the system. Continuous business expansion demands a rapid expansion of cloud services. One of the most versatile features of Cloud Computing is that it is scalable.

Broad network access: One of the most interesting features of cloud computing is that it knows no geographical boundaries. Cloud computing has a vast access area and is accessible via the internet. You can access your files and documents or upload your files from anywhere in the world, all you need is a good internet connection and a device, and you are set to go.

Resource Pooling: Instead of owning everything yourself, cloud computing lets you share resources with others in the same neighborhood, making it more efficient and cost effective.

Measured Service: Usage of computing resources is monitored, controlled and billed based on the actual usage.

Security: Cloud providers implement robust security measures to protect data and resources.

Cloud Service Models: There are three main types of service models of cloud computing. Each type of cloud computing provides different levels of control, flexibility, and management so that you'll select the proper set of services for your needs.

Infrastructure As A Service (IaaS)

It is the **most flexible** type of cloud service which lets you rent the hardware and contains the basic building blocks for cloud and IT.

It gives complete control over the hardware that runs your application (servers, VMs, storage, networks & operating systems).

It's an **instant computing** infrastructure, provisioned and managed over the internet.

IaaS gives you the very best level of flexibility and management control over your IT resources.

It is almost like the prevailing IT resources with which many IT departments and developers are familiar.

Examples of IaaS are virtual **Machines** or **AWS EC2(**Amazon Elastic Compute Cloud is a web service that provides secure, resizable compute capacity in the cloud**)**, Storage or Networking. DigitalOcean, Amazon Web Services (AWS), Microsoft Azure, Google Compute Engine (GCE), Rackspace, and Cisco Metacloud.

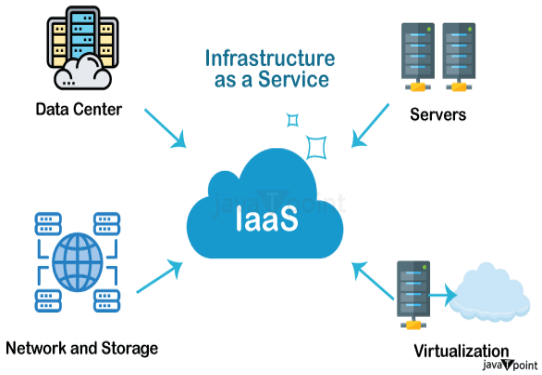


Diagram of IaaS

Platform As A Service (PaaS)

PaaS is a cloud service model that gives a ready-to-use development environment where developers can specialize in writing and executing high-quality code to make customized applications.

It helps to create an application quickly without managing the underlying infrastructure. For example, when deploying a web application using PaaS, you don't have to install an operating system, web server, or even system updates. However, you can scale and add new features to your services.

This cloud service model makes the method of developing and deploying applications simpler and it is more expensive than IaaS but less expensive than SaaS.

This helps you be more efficient as you don't get to worry about resource procurement, capacity planning, software maintenance, patching, or any of the opposite undifferentiated work involved in running your application.

Examples of PaaS: Elastic Beanstalk or Lambda from AWS, WebApps, Functions or Azure SQL DB from Azure, Cloud SQL DB from Google Cloud, or Oracle Database Cloud Service from Oracle Cloud.

Diagram of PaaS

Software As A Service (SaaS)

SaaS provides you with a complete product that is run and managed by the service provider. The software is hosted online and made available to customers on a subscription basis or for purchase in this cloud service model.

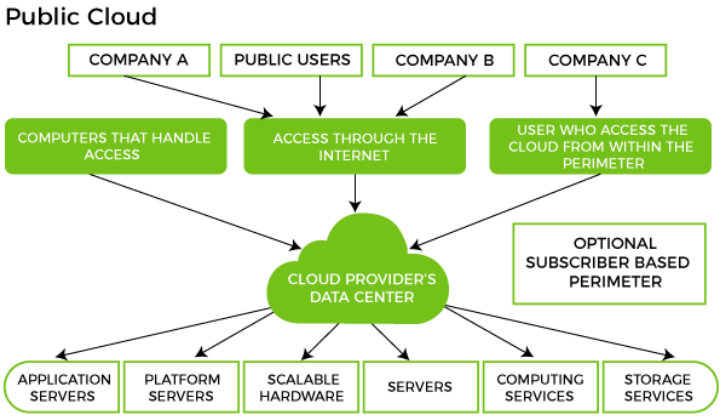
With a SaaS offering, you don't need to worry about how the service is maintained or how the underlying infrastructure is managed. It would help if you believed how you'd use that specific software.

Examples of SaaS: Microsoft Office 365, Oracle ERP/HCM Cloud, SalesForce, Gmail, or Dropbox.

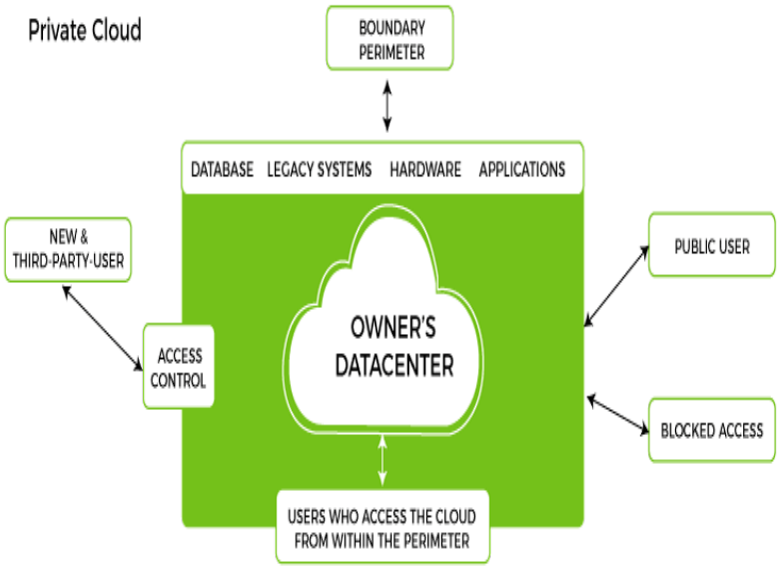
Diagram of SaaS

Cloud Deployment Models:

Public Cloud Model: As its names suggest, the public cloud is available to the general public, and resources are shared between all users. They are available to anyone, from anywhere, using the Internet. The public cloud deployment model is one of the most popular types of cloud.



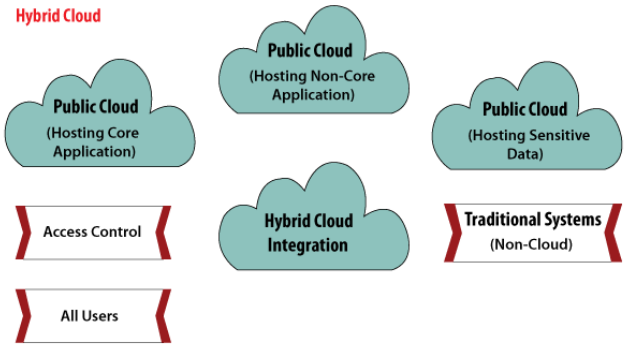
Private Cloud Model: In this model cloud resources are used exclusively by a single organization providing more control and customization over the infrastructure.



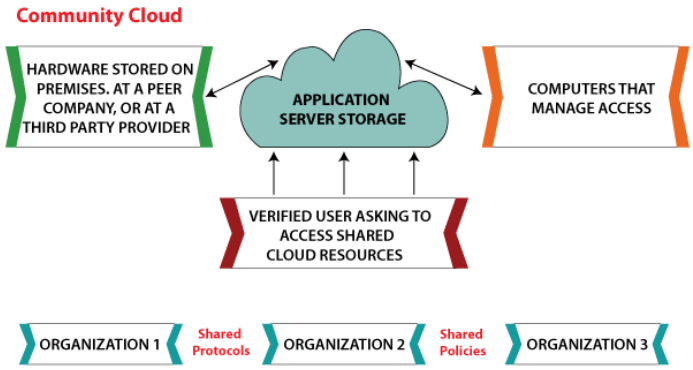
Hybrid Cloud Model:

A hybrid cloud is a combination of both public and private cloud environments that allows organizations to take advantage of the benefits of both types of clouds. It manages traffic levels during peak usage periods It can provide greater flexibility, scalability, and cost-

effectiveness than using a single cloud environment. Examples – IBM, DataCore Software, Rackspace, Threat Stack, Infinidat, etc.



Community Cloud Model: In this the cloud infrastructure is shared by several organizations with common interests. It improves security, data privacy, and reliability, making it the perfect choice for the government, banks, and universities. As this model is specially designed for a group of organizations, it lets you easily collaborate and share the data.



Extra Knowledge for Students

Do Amazon Web Services fall into which of the following cloud-computing category? (1). Platform as a Service, (2) Software as a Service, (3) Infrastructure as a Service, (4) Back-end as a Service

Correct Answer: Infrastructure as a Service Amazon Web Services falls into the Infrastructure as a Service cloud-computing category.

Which one of the following can be considered as the most complete cloud computing service model? (1) PaaS, (2) SaaS, (3) IaaS

Correct Answer: SaaS Explanation: The most complete cloud computing service model must contain the computing hardware and software, as well as the solution itself. Hence the SaaS model has all these features.

Which one of the following provides the resources or services such as virtual infrastructure, virtual machines, virtual storage, and several other hardware assets? (1) PaaS, (2) SaaS, (3) IaaS, (4) All of the above

Correct Answer: IaaS Explanation: The IaaS (Infrastructure as a service) service provider maintains all the infrastructure, while the client is responsible for several other deployment aspects.

You plan to provide Infrastructure as a Service (IaaS) resources in Azure. Which resource is an example of IaaS? (1) an Azure web app, (2) an Azure virtual machine, (3) an Azure logic app, (4) an Azure SQL database

Correct Answer: an Azure virtual machine Explanation/Reference: One can control hardware using IaaS resources and applications using PaaS resources.

Q.No.12: Explain the concept of phishing and its different forms in cyber attacks.

Ans: Phishing is a common type of cyber attack that targets individuals through email, text messages, phone calls, and other forms of communication. A phishing attack aims to trick the recipient into falling for the attacker's desired action, such as revealing financial information, system login credentials, or other sensitive information.

As a popular form of social engineering, phishing involves psychological manipulation and deception whereby threat actors masquerade as reputable entities to mislead users into performing specific actions. These actions often involve clicking links to fake websites, downloading and installing malicious files, and divulging private information, like bank account numbers or credit card information.

Types of Phishing Attacks

Phishing has evolved into more than simple credential and data theft. How an attacker lays out a campaign depends on the type of phishing. Types of phishing include:

- **Email phishing:** the general term given to any malicious email message meant to trick users into divulging private information. Attackers generally aim to steal account

credentials, personally identifiable information (PII) and corporate trade secrets. However, attackers targeting a specific business might have other motives.

- **Spear phishing:** these email messages are sent to specific people within an organization, usually high-privilege account holders, to trick them into divulging sensitive data, sending the attacker money or downloading malware.
- **Link manipulation:** messages contain a link to a malicious site that looks like the official business but takes recipients to an attacker-controlled server where they are persuaded to authenticate into a spoofed login page that sends credentials to an attacker.
- **Whaling (CEO fraud):** these messages are typically sent to high-profile employees of a company to trick them into believing the CEO or other executive has requested to transfer money. CEO fraud falls under the umbrella of phishing, but instead of an attacker spoofing a popular website, they spoof the CEO of the targeted corporation.
- **Content injection:** an attacker who can inject malicious content into an official site will trick users into accessing the site to show them a malicious popup or redirect them to a phishing website.
- **Malware:** users tricked into clicking a link or opening an attachment might download malware onto their devices. Ransomware, rootkits or keyloggers are common malware attachments that steal data and extort payments from targeted victims.
- **Smishing:** using SMS messages, attackers trick users into accessing malicious sites from their smartphones. Attackers send a text message to a targeted victim with a malicious link that promises discounts, rewards or free prizes.
- **Vishing:** attackers use voice-changing software to leave a message telling targeted victims that they must call a number where they can be scammed. Voice changers are also used when speaking with targeted victims to disguise an attacker's accent or gender so that they can pretend to be a fraudulent person.
- **"Evil Twin" Wi-Fi:** spoofing free Wi-Fi, attackers trick users into connecting to a malicious hotspot to perform man-in-the-middle exploits.
- **Pharming:** pharming is a two-phase attack used to steal account credentials. The first phase installs malware on a targeted victim and redirects them to a browser and a spoofed website where they are tricked into divulging credentials. DNS poisoning is also used to redirect users to spoofed domains.
- **Angler phishing:** using social media, attackers reply to posts pretending to be an official organization and trick users into divulging account credentials and personal information.
- **Watering hole:** a compromised site provides endless opportunities, so an attacker identifies a site used by numerous targeted users, exploits a vulnerability on the site,

and uses it to trick users into downloading malware. With malware installed on targeted user machines, an attacker can redirect users to spoofed websites or deliver a payload to the local network to steal data.

Q.No.13: Describe Denial of Service and Distributed Denial of Service attacks including their objectives and methods of execution.

Ans: A denial of service (DoS) attack is a cyber attack that aims to make a device, service, network, or other information system unavailable to legitimate users. The hacker uses a single machine and typically floods the target with an extremely high number of requests. Eventually, the target machine can no longer process normal traffic.

When thinking of a **cyber attack**, for many people what comes to mind is someone trying to access data illegally. A DoS attack is not necessarily about accessing or stealing someone else's data. The goal, in most cases, is to block users from accessing a service. Revenge, competition, extortion, and even activism are some reasons people resort to DoS attacks.

Effects of DOS attacks

- Genuine users are not able to access resources, so may not be able to find the information or carry out the actions they need.
- Businesses may not be able to carry out time critical actions.
- They may suffer reputational damage.
- Customers may choose to use a competitor.

Reasons for DOS attacks

- Financial - The attacker may demand payment to stop the attack
- Political - The attacker may wish to take down government websites to protest at government actions
- Personal - An individual may have a grievance against a company and decide to enact revenge
- Some of the common differences between DoS and DDoS are mentioned below.

Distributed Denial of Service: A distributed denial of service (DDoS) attack is a type of DoS attack that uses several distributed machines to launch the attack instead of a single machine.

DoS attacks come from one IP and are relatively easy to counter. DDoS attacks come from multiple IPs, which makes them more difficult to stop.

When the attack comes from distributed sources, it can be much harder to differentiate malicious traffic from normal traffic. As a result, DDoS attacks are harder to detect before they cause real damage. With a DoS attack, only one machine needs to be detected and stopped.

How does a DDoS attack work?

DDoS attacks are carried out with networks of Internet-connected machines.

These networks consist of computers and other devices (such as IoT devices)which have been infected with malware, allowing them to be controlled remotely by an attacker. These individual devices are referred to as bots (or zombies), and a group of bots is called a botnet.

Once a botnet has been established, the attacker is able to direct an attack by sending remote instructions to each bot.

When a victim's server or network is targeted by the botnet, each bot sends requests to the target's IP address, potentially causing the server or network to become overwhelmed, resulting in a denial-of-service to normal traffic.

Because each bot is a legitimate Internet device, separating the attack traffic from normal traffic can be difficult.

DoS	DDoS
DoS Stands for Denial of service attack.	DDoS Stands for Distributed Denial of service attack.
In Dos attack single system targets the victim system.	In DDoS multiple systems attack the victim's system.
Victim's PC is loaded from the packet of data sent from a single location.	Victim PC is loaded from the packet of data sent from Multiple locations.
Dos attack is slower as compared to DDoS.	A DDoS attack is faster than Dos Attack.
Can be blocked easily as only one system is used.	It is difficult to block this attack as multiple devices are sending packets and attacking from multiple locations.

DoS	DDoS
In DOS Attack only a single device is used with DOS Attack tools.	In a DDoS attack, The volumeBots are used to attack at the same time.
DOS Attacks are Easy to trace.	DDOS Attacks are Difficult to trace.

Q.No.14: Explain the concept of insider threats in cybersecurity including the types of individuals involved and their motivations for malicious actions.

Ans: An insider threat is a malicious activity against an organization that comes from users with legitimate access to an organization's network, applications or databases. These users can be current employees, former employees, or third parties like partners, contractors, or temporary workers with access to the organization's physical or digital assets. They can even come in the form of compromised service accounts. While the term is most commonly used to describe illicit or malicious activity, it can also refer to users who unintentionally cause harm to the business.

At a high level, there are several types of insider threats. The five most common insider threats include:

- **Malicious Insider Threats.** Malicious insiders are the people who abuse company data and assets on purpose with deliberate malicious intent. These insiders could be disgruntled employees who are motivated to sell company data for financial gain or leak sensitive data in order to cause damage to the organization. Departing employees may take data to a new company for their professional gain. Additionally, malicious insiders are increasingly recruited, bribed, or extorted by outside actors such as nation-states or ransomware groups.
- **Opportunistic Insider Threats.** Opportunistic insiders are very common and can be thought of as malicious insiders without premeditated intent. An opportunistic insider may collect sensitive information over time without initially intending to misuse the data. At a later time, the user may decide to misuse that data, such as after moving to a new company or after being fired. Both opportunistic and malicious insiders intentionally misuse data. However, the opportunistic insider is an important distinction because the user abuses data that the organization has already lost control of.

- **Negligent Insider Threats.** Negligent insiders expose data or assets by consciously breaking security policy. The intention may not be to cause harm but rather simply to perform a task in a way the user perceives as faster or easier. For example, a user who intentionally sends an important file to their personal webmail in order to work remotely without going through the company VPN and remote authentication process is negligent. Once again, the damage of such behavior can be the same regardless of the user's motivation.
- **Accidental Insider Threats.** Many users will expose data purely by accident. Modern applications make it very easy to share data, and a busy, distracted user can easily make mistakes that can take data out of the company's control. For example, a user may accidentally upload an important file to a personal Dropbox account with public permissions instead of privately in a corporate account. Or a user may inadvertently share a file with the wrong person in the company's Google Drive when they type in the recipient's name and, in a rush, send it without noticing the browser autocompleted the recipient to someone else with the same first name.
- **Compromised Insider Threats.** The compromised end user can blur the lines between an insider threat and a more traditional external threat. A compromised user occurs when a threat actor or malware is able to take control of a user's machine and/or credentials to steal data and other critical assets. In many cases, this is still considered an external threat. However, many of the underlying behaviors in which the attacker attempts to aggregate and exfiltrate sensitive data can mimic that of an insider threat. As a result, insider threat security tools can be highly valuable in preventing the loss of data and even in the ability to detect external threats.