1. **Who is stakeholder and explain types of stake holders**

A stakeholder is any person or a group of persons or an organization that is directly or indirectly effected or impacted by the proposed IT solution.

**Types of Stakeholders:**

* **Project Stakeholders:** Business Analyst, Project Manager, Development Team, Testing and Operations.
* **Business Stakeholders:** Project Manager, Business Owner, Business Sponsor, Executive Sponsor, Operation Team, SPOC, and End user.
* **Third Party Stakeholders:** Auditors, Focus Group, Manufacturer, Outsource, and Legal Specialist
* **Negative Stakeholders:** Competitors, Hacker, Development Team, Political Party and Public opinion.

1. **Explain about Component, Package and Subsystems**

**Component:** Component is Collection of Classes.

**Package:** Collection of Components which are not reusable in nature

**Subsystems:** Collection of Components which are reusable in nature

**Note:** Product Development Companies work on Subsystems and Application Development Companies work on Packages

1. **List down the reasons for project failure**

**Reasons for project failures:**

* Improper Planning
* Unclear goals
* Not enough resources
* Lack of user involvement
* Improper requirements gathering
* Leadership issues
* Ignoring risks
* Technical problems
* Unrealistic expectations

1. **Explain about MOSCOW technique**

The MoSCoW technique is a prioritization method used in project management and software development to categorize and prioritize requirements or features based on their importance.

* M – Must have
* S – Should have
* C – Could have
* W – Wont have or Would have

The basic steps involved in this prioritization technique is Identification of Requirements, Grouping the requirements, then we reach out to stakeholders to assign priorities to each and every requirement. According to priorities indicate them as Must have, Should have, Could have, Won’t or would have, then review the prioritized requirements with stake holders and ensure agreement with them.

1. **Derive Boundary Classes, Controller Classes, Entity Classes**

**Boundary Class:**

Boundary classes represent the interaction points between the system and its external entities, such as users or external systems, they are responsible for handling input and output. They encapsulate the user interface components

**Controller Class:**

Controller classes mediate between the boundary classes and the entity classes. They handle the flow of control and coordinate the system's behaviour. Controllers receive input from the boundary classes, process it, and interact with the entity classes to perform the necessary actions

**Entity Class:**

Entity classes represent the core business objects or concepts that the system manages. They encapsulate the data and business logic related to the system's domain. They often have attributes and methods related to the core business operations.

1. **what is MVC architecture**

The Model-View-Controller (MVC) framework is an architectural/design pattern that separates an application into three main logical components Model, View, and Controller. Each architectural component is built to handle specific development aspects of an application. It isolates the business logic and presentation layer from each other.

1. **Model:** The Model component corresponds to all the data-related logic that the user works with. It can add or retrieve data from the database. It responds to the controller’s request because the controller can’t interact with the database by itself. The model interacts with the database and gives the required data back to the controller
2. **View:** The View component is used for all the UI logic of the application. It generates a user interface for the user. Views are created by the data which is collected by the model component but these data aren’t taken directly but through the controller. It only interacts with the controller.
3. **Controller:** The controller is the component that enables the interconnection between the views and the model so it acts as an intermediary it just tells the model what to do. It process all the business logic and incoming requests, manipulate data using the Model component and interact with the View to render the final output.
4. **Explain about Waterfall and Agile**

**Waterfall:** This model is a sequential approach where each phase of development must be completed before moving on to the next phase. It is best suited for projects with well-defined requirements and clear project goals.

**Agile:** This methodology is based on an iterative and incremental approach which involves close collaboration between the development team and stakeholders. This method is best suited for projects with rapidly changing requirements, high risk, and complex environments.

1. **Explain about MVP**

A Minimal Viable Product is the most basic version of a product that includes only the essential features necessary to address the core needs of early users. The main purpose of an MVP is to quickly bring a functional version of the product to the market, allowing the development team to gather valuable feedback and insights from real users. This feedback is then used to iteratively enhance and improve the product.

1. **Explain Brainstorming Technique**

This is used to generate creative ideas and solutions through group discussions. It is a media where people can freely express their ideas, thoughts and suggestions without criticism or judgment. This is mostly done in an informal way with minimal documentation

The basic idea behind brainstorming is to find a conclusion for a specific problem by gathering a list of ideas spontaneously contributed by its members.

1. **Observation Technique – Explain both Active and Passive approaches**

**Observation:**

Observing, shadowing users or even doing part of their job, can provide information of existing processes, inputs and outputs. The main advantage is that it is useful if the user is not able to clearly explain what they do or their requirements for the new system. Can see ideas for improving processes or removing unnecessary activities from the new system.

Disadvantage is that this is Relatively slow, which is focused on existing processes rather than the new system processes.

Passive/Invisible:

In this approach, the business analyst observes the subject matter expert working through the business routine but does not ask questions. The business analyst writes notes about what he/she sees, otherwise stays out of the way as invisible. The BA waits until the entire process has been completed before asking any questions.

Active/Visible:

In this approach, while the business analyst observes the current process and takes notes he/she may dialog with the worker. When the BA has questions as to why something is being done as it is, questions can be asked right away, even if it breaks the routine of the person being observed.

1. **Explain about Document Analysis**

Documentation analysis is crucial techniques used in Project management and requirement gathering stage. This is done mainly to understand existing documentation and establishing baseline and reference for current project.You may have documentation about your current system which could provide some of the input for the new system requirements. Such documentation (if it exists) could include interface details, user manuals, and software vendor manuals.

In this we basically have 3 stages:

1. Prepare For Document Analysis:

Evaluate which existing system documentation are relevant and appropriate to be studied.

1. Analyse The Document:

Study the material and identify relevant business details also Document business details as well as questions to discuss with subject matter experts.

1. Post Doc Analysis Wrap – Up:

Review and confirm the selected details with subject matter experts and obtain answers to questions which are noted earlier.

1. **Explain about Sprint Planning Meeting**

**Sprint Planning:** This meeting is conducted at the beginning of each sprint, the team plans the work to be done during the sprint. The team and the product owner discuss the prioritize user stories from the product backlog. The team estimates the effort required, and together, they commit to completing a set of user stories within the sprint

1. **Explain about Daily Standup (Daily Scrum) Meeting**

**Daily standup (Daily Scrum):** This is held daily to keep the team synchronized, identify impediments, and plan the day’s work. It is a brief time-boxed meeting focused on coordination. Each team member answers 3 questions:

* What did I do yesterday?
* What will I do today?
* Are there any blockers?

1. **What is Sprint Size and Scrum Size**

**Sprint Size:** Sprint size refers to the time duration set for completing a sprint in Agile development, typically ranging from one to four weeks. It's determined based on factors such as project complexity, team capacity, and the need for frequent feedback and adjustments.

**Scrum Size:** Scrum size typically refers to the size of the Scrum team, which ideally consists of 5-9 members, including the Scrum Master and Product Owner. The size of the Scrum team is optimized to promote effective communication, collaboration, and productivity while ensuring that the team remains agile and adaptable to changing requirements and circumstances.

1. **Explain DOR and DOD**

**Definition of Ready:** Definition of Ready sets the criteria for a task or user story to be considered prepared for implementation. It ensures that all necessary information, resources, and dependencies are identified and available before the team begins work, helping to prevent delays and misunderstandings during development.

**Definition of Done:** Definition of Done outlines the criteria that a task or user story must meet to be considered complete. It includes quality standards, testing requirements, and any other criteria agreed upon by the team. Meeting the Definition of Done ensures that the work is finished and ready for delivery to the customer or end-user.

1. **Explain about Sprint review meeting**

This meeting is held at the end of each sprint, the team showcases the completed work on the features or user stories completed during the sprint. Stakeholders provide feedback, and the product owner discusses the next set of priorities

1. **Explain about Sprint Retrospective**

This is a reflection on the completed sprint to identify what went well, what could be improved, and actions for continuous improvement. The team discusses what worked out and what did not, identifying areas of improvement. Action items are established to enhance team processes in the next sprint.

1. **Explain about Backlog refinement**

Backlog refinement:

* This process involves reviewing, clarifying, and prioritizing items in the Product Backlog to ensure they are ready for selection in upcoming sprints.
* This is a regular collaborative activity that is conducted throughout the Sprint.
* The entire Scrum Team, including the Product Owner, Scrum Master and Development Team collaborates during Backlog Refinement.

1. **What are Complexity Points**

**COMPLEXITY POINTS:**

In Scrum, complexity points are a measure used to estimate the relative complexity of user stories or tasks during the sprint planning process. The purpose of assigning complexity points is to provide a high-level estimation of the effort or complexity involved in completing a user story or task.

CP is estimated by the Scrum Developers, but this need to be done from customer point of view

Planning poker is used by scrum developers to estimate complexity points.

1. **What is business value**

**BUSINESS VALUE:**

Business value is how important the user story is to the business. It is a way to prioritize work based on the impact it has on achieving business goal.

The goal is to deliver highest value features ensuring that the most critical needs are addressed early in development process.

This has the potential impact on revenue generation, cost savings and other business metrics.