**1)SWOT ANALYSIS:**

SWOT (Strengths, Weaknesses, Opportunities, Threats) analysis is a key tool for business analysts, offering a comprehensive snapshot of a business's internal and external landscape.

**Analysing Internal Factors:**

Strengths: Brand recognition, skilled workforce, innovative technology. Leverage these for advantage.

Weaknesses: Limited resources, outdated processes, communication gaps. Address these to improve efficiency.

**Analysing External Factors:**

Opportunities: Emerging markets, changing customer preferences, new partnerships. Seize these to fuel growth.

Threats: Increased competition, regulatory changes, economic downturns. Understand these to mitigate risks.

**2)USE CASE, USER STORY AND ACCEPTANCE CRITERIA:**

1. Use Case: Imagine it as a detailed script, outlining how a user interacts with a system to perform a specific action. It describes the steps the user takes, the system's responses, and the expected outcome, like a play for actors and set design.

2. User Story: Think of it as a concise wish list, written from the user's perspective. It describes what the user wants to achieve and the value they expect from the system, capturing the essence of their desired experience in an easily understandable format.

3. Acceptance Criteria: These are the checkpoints that determine if a user story is truly complete and meets the user's needs. They define the specific functionalities, performance metrics, and other requirements that must be fulfilled for the story to be considered "done" and ready for user acceptance.

**3)CONFLICTS IN A PROJECT:**

How do you handle managing team conflicts in a project?

1. Clear Communication: Ensure everyone understands project goals, roles, and expectations to minimize misunderstandings.

2. Active Listening: Listen attentively to all sides of the conflict to uncover the root causes, not just surface disagreements.

3. Open Dialogue: Facilitate respectful discussions where everyone feels heard and encouraged to share their perspectives.

4. Collaborative Solutions: Seek solutions that benefit all team members while aligning with project goals. This might involve compromise, creative problem-solving, or even revisiting project plans.

5. Documentation and Follow-up: Clearly document agreed-upon solutions, timelines, and action items. Monitor progress and be prepared to adapt if needed.

**4) PRIORITISATION METHOD:**

The Moscow method is a powerful tool for defining priorities for requirements. It allows for clear communication, stakeholder alignment, and efficient resource allocation. It follows two step process

1. Understanding the Requirements: Start by thoroughly understanding all the requirements, their purpose, and potential impact. This includes gathering information from various stakeholders through interviews, workshops, and documentation

review.

2. Categorizing Requirements:

Categorize each requirement based on the Moscow criteria:

a. Must-Haves: These are essential requirements that must be met for the project's success. They are non-negotiable and typically address core functionalities or business needs.

b. Should-Haves: These are important requirements that add significant value but can be deferred if necessary. They often enhance functionality or user experience.

c. Could-Haves: These are desirable requirements that would be nice to have but can be postponed or even eliminated without compromising the project's core objectives.

d. Won't-Haves: These are requirements that are deemed unnecessary or infeasible for the current project scope or budget. They may be considered for future iterations.

**5)GAP ANALYSIS:**

A gap analysis is a tool used to identify the difference between your current state and your desired state. It's like comparing where you are now to where you want to be, then figuring out what steps to get there.

This analysis is used across various domains, from business and project management to personal development.

Main Stages:

Define your desired state: This could be specific goals, desired performance levels, or ideal outcomes.

Assess your current state: Measure key indicators to understand where you stand. Gather data on resources, skills, performance,

Analyse the gap: Compare your current state to your desired state.

Identify areas where you fall short or lack necessary elements.

Develop an action plan: Based on the identified gaps, create strategies and steps to bridge the difference. Allocate resources and set timelines.

**6)WATERFALL AND AGILE METHODOLGY:**

Waterfall: This is sequential in nature. That means that before one step can begin; the previous step needs to have been completed. This is very similar to workflows found in the manufacturing and construction sectors.

Agile: This methodology uses an incremental approach. This means that business analysts can start off with simple versions of the project design and build the design into small modules that are further elaborated on as the project progresses.

**7) Difference between functional and non-functional requirements?**

**Answer:**

* **Functional Requirements:** These define the specific behaviours or functions of a system. They describe what the system should do. For example, “The system should allow users to log in with a username and password” or “The system must generate reports on demand.”
* **Non-Functional Requirements:** These describe the quality attributes or constraints of the system. They define how the system should perform. Examples include performance metrics like response time, scalability, security standards, and reliability.

**8)Tools to use for document business requirements:**

**Answer:** As a Business Analyst, I use various tools to document and track business requirements, including:

* **Microsoft Office (Word, Excel, PowerPoint):** For creating detailed documents, spreadsheets for analysis, and presentations for stakeholder meetings.
* **JIRA:** A popular tool for tracking user stories, tasks, and defects in Agile environments.
* **Confluence:** For collaborative documentation and sharing of requirements, specifications, and meeting notes.
* **Visio:** For creating process flow diagrams, use case diagrams, and other visual models.
* **Balsamiq or Axure:** For prototyping and wireframing UI/UX designs.
* **Trello or Asana:** For tracking tasks and progress in projects with both team and stakeholder visibility.

**9) functional specification and a technical specification:**

**Answer:**

* **Functional Specification:** This document describes the functionality that the system must provide from a user's perspective. It focuses on what the system should do, without detailing how it will be implemented. Examples include user stories, use cases, and acceptance criteria.
* **Technical Specification:** This document provides a detailed description of how the system will be implemented from a technical perspective. It covers system architecture, data structures, algorithms, security measures, and integration points.

**10) Handle Change request to business requirements during the project lifecycle:**

1. **Assessing the Impact:** Analyses how the changes will affect the scope, timeline, and budget.
2. **Communicating with Stakeholders:** Keep stakeholders informed about the potential impacts of the changes.
3. **Updating Documentation:** Ensure that requirements documents, user stories, and specifications are updated to reflect the changes.
4. **Reprioritizing Work:** Work with the product owner and stakeholders to adjust priorities as necessary.
5. **Managing Change Control:** Follow a formal change control process (if applicable) to ensure that changes are tracked and approved.

**11) SPRINT BACKLOG AND PRODUCT BACKLOG:**

|  |  |
| --- | --- |
| **SPRINT Backlog** | PRODUCT BACKLOG |
| A list of tasks and items to be completed in the current sprint. | A prioritized list of features, enhancements, and bug fixes for the entire product. |
| Focused on a subset of the product features for a specific sprint. | Includes the entire scope of the product, including all desired features and enhancements. |
| Owned by the Development Team. | Owned by the Product Owner. |
| Short-term, typically for the duration of one sprint (usually 1-4 weeks). | Long-term, represents the overall roadmap and vision for the entire product lifecycle. |
| Contains detailed tasks, user stories, and work items for the current sprint. | Contains high-level user stories, features, and epics, with a focus on business value. |
| Derived from the Product Backlog, prioritized for the sprint's goals. | Prioritized by the Product Owner based on business value and stakeholder needs. |
| Can be modified during the sprint, but typically only during the Sprint Planning or Daily Scrum. | Continuously updated and refined (groomed) based on stakeholder feedback and changing business priorities. |

**12)PRODUCT GROOMING:**

Product grooming, also known as backlog grooming or refinement, is the process of reviewing and prioritizing tasks in a product backlog. It involves refining user stories, defining requirements more clearly, and breaking down larger tasks into smaller, manageable ones. The goal is to ensure that the product backlog is up to date, with well-defined tasks that are ready for future sprints or development cycles. This helps teams stay aligned, focused on the most important features, and ensures smoother product development.

**13)List down the reasons for project failure:**

**ANSWER:**

**1. Lack of Clear Objectives**

* When a project does not have well-defined goals, scope, or outcomes, it can lead to confusion, misdirection, and unmet expectations.

**2. Poor Planning**

* Insufficient or inadequate planning, including unrealistic timelines, budget overruns, and unanticipated risks, often leads to project failure.

**3. Ineffective Communication**

* Miscommunication between team members, stakeholders, and management can cause misunderstandings and delays, making it difficult to address issues proactively.

**4. Inadequate Resources**

* Lack of necessary resources, whether financial, human, or technological, can hinder a project's progress and completion.

**5. Unrealistic Expectations**

* Setting overly ambitious or unattainable goals, such as demanding deadlines or inflated budgets, can cause the project to spiral out of control.

**6. Scope Creep**

* When the project's scope is continually expanded or changed without proper control, it can lead to confusion, delays, and resource depletion.

**7. Inexperienced Team**

* A project team that lacks the skills, experience, or training necessary to execute the project efficiently may cause delays, errors, or even project collapse.

**8. Lack of Stakeholder Engagement**

* If stakeholders are not adequately involved or engaged throughout the project, it may lead to misaligned expectations and the failure to meet their needs.

**14)DECISION ANALYSYIS:**

Decision analysis is a systematic approach to making decisions under uncertainty, involving the evaluation of different alternatives based on their potential outcomes and risks. It uses models like decision trees, cost-benefit analysis, or simulations to assess various scenarios. The goal is to make informed, rational choices that maximize value or minimize risk. This method is commonly used in business, finance, and project management to support critical decision-making processes.

**15)ROLE OF BUSINESS ANALYST IN PROJECT:**

The role of a Business Analyst (BA) in a project is to bridge the gap between stakeholders and the development team, ensuring that business requirements are clearly understood and translated into technical solutions. BAs gather and analyse requirements, define project scope, and prioritize tasks to ensure alignment with business goals. They also facilitate communication, manage stakeholder expectations, and help identify potential risks or issues early in the project. Throughout the project, BAs ensure that the delivered product or solution meets the business needs and provides value to the organization.

**16) WHAT IS REQUIREMNET:**

A requirement is a specific need or condition that a product, service, or system must fulfill to meet the objectives or expectations of stakeholders. In the context of project management or product development, it defines what the solution should achieve or what features it should include. Requirements can be functional (what the system should do) or non-functional (how the system should perform, such as reliability, speed, or security). Properly defined requirements are essential for guiding the design, development, and delivery of a successful solution.

**TYPES OF REQUIREMNET:**

a) Business requirement

b) stakeholder requirement

c)solution requirement: functional and non-functional requirement

**17) The bond between requirements and a Business Analyst (BA)**:

1. **Requirement Gathering**: A BA plays a key role in identifying, gathering, and documenting the needs of stakeholders, users, and the business. They ensure that all relevant requirements are captured accurately and comprehensively.
2. **Requirement Analysis**: Once the requirements are gathered, the BA analyzes them to understand their feasibility, relevance, and alignment with business objectives. They help prioritize requirements based on business value and urgency.
3. **Clarification & Communication**: The BA acts as a liaison between stakeholders (e.g., business owners, customers) and the development team. They ensure that the requirements are clearly communicated and understood by both sides, helping to clarify any ambiguities.
4. **Requirement Validation**: Throughout the project, the BA ensures that the requirements are being properly implemented and validate that the final deliverables align with the initial requirements. This ensures that the end solution meets business needs and stakeholder expectations.

**18) SDLC:**

The Software Development Life Cycle (SDLC) is a structured approach to software development, guiding the process from concept to delivery. It begins with **planning**, where project scope, objectives, and resources are defined. **Requirement gathering** follows, where business and technical needs are collected from stakeholders. In the **design phase**, the software architecture, user interfaces, and system components are outlined. The **development phase** involves writing the code and creating the necessary system components. During **testing**, the software is evaluated to ensure it meets requirements and is free from defects. After successful testing, **deployment** involves releasing the software for use. Finally, **maintenance** involves ongoing support, updates, and bug fixes to ensure the software remains functional and aligned with evolving needs. Each phase builds upon the previous one, ensuring structured and efficient software delivery.

**19) V-MODEL:**

* Verification and Validation: Each development phase has a corresponding testing phase, ensuring that the system is verified against its design and validated against user requirements.
* Early Detection of Issues: Testing starts early, making it easier to detect and fix issues in the early stages of development.
* Clear Structure: The model has a clear, systematic approach to development and testing, reducing risks of missed requirements or overlooked defects.
* Rigid and Sequential: The V-Model, like the Waterfall model, is quite rigid, with each phase needing to be completed before moving to the next. This can make it challenging to accommodate changes once development begins.

The V-Model is particularly useful in projects where requirements are well understood upfront and unlikely to change significantly throughout the development process. It helps ensure a high level of quality assurance through continuous verification and validatio**n.**

**20) The Spiral Model:**

The Spiral Model is most beneficial for large, complex projects where requirements are expected to change over time, or when there are significant risks involved. It is commonly used in projects where the final product cannot be fully defined at the beginning and where frequent feedback from users or stakeholders is critical.

In summary, the Evolutionary Spiral Model is a flexible, risk-aware development methodology that allows for continuous iteration and refinement. It is especially useful for large-scale, complex projects that require frequent adjustments and stakeholder involvement throughout the development process.

Advantages of the Spiral Model:

* Risk Mitigation: It provides a systematic approach to managing risks and ensures that high-risk areas are addressed early on.
* Flexibility: The model accommodates changing requirements and evolving project goals, making it ideal for projects with uncertain or shifting needs.
* Continuous Feedback: Stakeholders can see and provide feedback on each iteration, which helps ensure that the final product meets their needs.
* Iterative Refinement: Each cycle produces a more complete version of the product, allowing for incremental progress.

**21) Data Flow Diagram (DFD):**

A Data Flow Diagram (DFD) is a visual representation used to illustrate the flow of data within a system. It shows how data moves through processes, where it's stored, and how it interacts with external entities (like users or other systems). DFDs help to understand the system's functionality and its components without getting into technical details.

**22)Process Mapping:**

Process Mapping is a visual technique used to document and analyse the flow of activities within a business process. It involves creating diagrams that represent the steps, decisions, and interactions involved in completing a specific task or workflow. The primary goal is to understand how processes work, identify inefficiencies, and find opportunities for improvement. By using standardized symbols (like ovals, rectangles, diamonds, and arrows), process maps help clarify roles, responsibilities, and data flow. Process mapping is widely used in business analysis, process optimization, and quality management to ensure processes are efficient and effective.

**23) Business Rules Analysis:**

Business Rules Analysis is the process of identifying, defining, and analyzing the rules that govern business operations and decision-making. These rules dictate how an organization operates, sets constraints, and guides actions based on specific conditions. Business rules help ensure consistency, standardization, and compliance across various processes and systems. This analysis involves working with stakeholders to document these rules clearly and identifying any gaps or conflicts. By understanding business rules, organizations can streamline operations, improve decision-making, and ensure alignment with regulatory requirements.

**24) Risk Analysis:**

Risk Analysisis the process of identifying, assessing, and prioritizing potential risks that could negatively impact a project, organization, or system. It involves evaluating the likelihood of various risks occurring and the potential consequences they could have. By understanding risks, businesses can develop strategies to mitigate, avoid, or transfer them, reducing their impact. Risk analysis helps in making informed decisions, managing uncertainties, and ensuring that resources are allocated effectively. It’s an essential tool in both project management and strategic planning to safeguard against unforeseen challenges.

**25) Business Case Development:**

Business Case Development is the process of creating a structured document that justifies the initiation of a project or investment. It outlines the objectives, benefits, costs, risks, and potential returns to help stakeholders make informed decisions. A well-developed business case includes a clear problem statement, proposed solution, and a detailed analysis of the expected outcomes, such as financial benefits, operational improvements, or strategic advantages. It serves as a tool for securing approval, guiding project execution, and ensuring that resources are allocated to initiatives that align with business goals.