COEPD – Traditional Development

Capstone Project1 – Part -2/3 – 100 Marks - Pass 60 % 12 Questions

Instructions to follow:

1. Copy paste (either image, diagram or text) is not entertained. If done, the document will not be evaluated.
2. After submission of the answers of this prep exam, You should be prepared to attend viva and justify your answers in the prep exams. If in Viva, participant is NOT justifying the answers, Viva will be repeated until Candidates justify 60% correctness.
3. Mentor calls are scheduled only if the participant have submitted their task at least for one time. (should apply their knowledge in this task first)
4. For attempting prep exams participant should be thorough on the topics using their references.
5. Please format the document properly (Always have a question no., question and answer).
6. Have a consistent format (Font name: Arial/ Calibri -Font size 12, Font Color: Black ).
7. Few Questions are related to the case study, if check Questions thoroughly before you answer.
8. Answers should be elaborated in detail(\*not as per the allotted marks).
9. Please focus on learning and applying the knowledge as this knowledge will be helpful in contributing at your BA job.
10. In the evaluation, students must answer all questions and should be able to justify at least 60% content and correctness of each answer.

Online Agriculture Products Store

Mr. Henry, after being successful as a businessman and has become one of the wealthiest persons in the city. Now, Mr. Henry wants to help others to fulfil their dreams. One day, Mr. Henry went to meet his childhood friends Peter, Kevin and Ben. They live in a remote village and do farming. Mr. Henry asked his friends if they are facing any difficulties in their day-to-day work.

Peter told Mr. Henry that he is facing difficulties in procuring fertilizers which are very important for farm. Kevin said that he is also facing the same problem in-case of buying seeds for farming certain crops. Ben raised his concern on lack of pesticides which could help in greatly reducing pests in crops.

After listening to all his friends’ problems, Mr. Henry thought that this is a crucial problem faced not only by his friends but also by so many other farmers. So, Mr. Henry decided to make an online agriculture product store to facilitate remote area farmers to buy agriculture products. Through this Online Web / mobile Application, Farmers and Companies (Fertilizers, seeds and pesticides manufacturing Companies) can communicate directly with each other.

The main purpose to build this online store is to facilitate farmers to buy seeds, pesticides, and fertilizers from anywhere through internet connectivity. Since new users are involved, Application should be user friendly.

This new application should be able to accept the product (fertilizers, seeds, pesticides) details from the manufacturers and should be able to display them to the Farmers. Farmers will browse through these products and select the products what they need and request to buy them and deliver them to farmers location.

Mr. Henry has given this project through his Company SOONY. In SOONY Company, Mr Pandu is Financial Head and Mr Dooku is Project Coordinator. Mr. Henry , Mr Pandu , and Mr Dooku formed one Committee and gave this project to APT IT SOLUTIONS company for Budget 2 Crores INR and

18 months Duration under CSR initiative. Peter, Kevin and Ben are helping the Committee and can be considered as Stakeholders share requirements for the Project.

Mr Karthik is the Delivery Head in APT IT SOLUTIONS company and he reached out to Mr Henry through his connects and Bagged this project. APT IT SOLUTIONS company have Talent pool Available for this Project. Mr Vandanam is project Manager, Ms. Juhi is Senior Java Developer, Mr Teyson, Ms Lucie, Mr Tucker, Mr Bravo are Java Developers. Network Admin is Mr Mike and DB Admin is John. Mr Jason and Ms Alekya are the Tester. And you joined this team as a BA.

Question 1 – Audits - 5 Marks

4 Quarterly Audits are planned Q1 , Q2, Q3, Q4 for this Project What is your knowledge on how these Audits will happen for a BA ?

As a Business Analyst (BA), quarterly audits are essential checkpoints to ensure the project aligns with its objectives and adheres to planned deliverables. These audits focus on both the documentation and processes that a BA contributes to and oversees. Here's how I understand these audits will be conducted:

1. **Review of Deliverables:**
Each quarter, the audit will focus on the deliverables produced by the BA, such as Business Requirement Documents (BRDs), Functional Requirement Specifications (FRS), and User Stories. Auditors will check if these documents align with stakeholder expectations and if they are updated to reflect any changes during the project.
2. **Traceability Matrix Audit:**
The traceability matrix will be reviewed to ensure all requirements are mapped to corresponding design, development, and testing phases. This ensures no critical requirements are missed or overlooked.
3. **Stakeholder Communication and Sign-Offs:**
Auditors will evaluate the communication logs and sign-offs to ensure effective collaboration with stakeholders. They may check meeting minutes, emails, or documented approvals to confirm that key decisions were formally acknowledged.
4. **Compliance with Timelines and Processes:**
As a BA, following defined processes and meeting timelines for documentation, requirements elicitation, and validation is crucial. Audits will assess if I adhered to these timelines and followed the organization’s methodology (e.g., Waterfall, Agile).
5. **Testing Support and UAT:**
The role of the BA in supporting the testing phase, especially User Acceptance Testing (UAT), will be audited. This includes checking if test scenarios were derived from requirements and whether UAT results align with the documented acceptance criteria.
6. **Change Management:**
Any changes in requirements will be reviewed to confirm they were documented, analyzed for impact, and approved before being implemented. This ensures transparency and accountability in managing scope changes.
7. **Reporting and Documentation Standards:**
The audit will evaluate if reports, such as status updates, risk logs, and progress trackers, were prepared and maintained according to standards. This includes clarity, completeness, and accuracy of all documentation.

**Outcome of the Audits:**
For a BA, these audits serve as a reflective mechanism to ensure the project stays on track while highlighting areas of improvement. By maintaining well-organized and up-to-date documentation, facilitating clear communication, and adhering to processes, I can ensure a positive audit outcome.

Question 2 – BA Approach Strategy - 6 Marks

Before the Project is going to Kick Start, The Committee asked Mr Karthik to submit BA Approach Strategy

Write BA Approach strategy (As a business analyst, what are the steps that you would need to follow to complete a project – What Elicitation Techniques to apply, how to do Stakeholder Analysis RACI/ILS, What Documents to Write, What process to follow to Sign off on the Documents, How to take Approvals from the Client, What Communication Channels to establish n implement, How to Handle Change Requests, How to update the progress of the project to the Stakeholders, How to take signoff on the UAT- Client Project Acceptance Form )

Your Team

|  |  |  |
| --- | --- | --- |
|  | Project Manager - Mr Vandanam Senior Java Developer - Ms. Juhi |  |
| Java Developers - Mr Teyson, Ms Lucie, Mr Tucker, Mr Bravo Network Admin - Mr Mike |
|  | DB Admin - Mr John. |  |

BA - You

Testers - Mr Jason and Ms Alekya

Technical Team have assembled to discuss on the Project approach and have finalised to follow 3-tier architecture for this project.

**BA Approach Strategy for the Online Agriculture Products Store**

Before kickstarting the project, it is critical to establish a clear and structured Business Analyst (BA) approach strategy to ensure alignment, efficiency, and stakeholder satisfaction throughout the project lifecycle. Below are the steps and strategies I would follow as the BA for this project:

**1. Elicitation Techniques**

To gather comprehensive and accurate requirements, I will utilize a combination of elicitation techniques:

* **Interviews:** Conduct one-on-one discussions with key stakeholders like Mr. Henry, the SOONY Committee, and end-user representatives (Peter, Kevin, and Ben) to understand their expectations.
* **Workshops:** Organize collaborative sessions with the SOONY Committee and the APT IT Solutions team to brainstorm and validate requirements.
* **Surveys/Questionnaires:** Distribute structured surveys to farmers and manufacturers to collect input on usability and features.
* **Observation:** Analyze farmers' current challenges and workflows to design features that are intuitive and practical.
* **Document Analysis:** Review existing data on agricultural product distribution to identify gaps and opportunities.

**2. Stakeholder Analysis (RACI/ILS Models)**

I will prepare a **RACI Matrix** to define roles and responsibilities clearly:

* **Responsible:** APT IT Solutions team (Development, Testing, Deployment).
* **Accountable:** Mr. Henry and the SOONY Committee (Final approvals).
* **Consulted:** Peter, Kevin, Ben (Farmer representatives for feedback).
* **Informed:** SOONY Company Board, Farmers, and Manufacturers (progress updates).

An **ILS (Interest, Location, Support) analysis** will also be conducted to identify stakeholder priorities and their influence on the project.

**3. Documentation to Prepare**

I will ensure thorough documentation at every stage of the project:

* **BRD (Business Requirements Document):** Captures high-level requirements and objectives of the project.
* **FRS (Functional Requirement Specifications):** Details technical functionalities and system behavior.
* **RTM (Requirements Traceability Matrix):** Maps requirements to corresponding deliverables.
* **Process Flows and Use Cases:** Visual representation of user interactions with the system.
* **Change Request Log:** Documents and tracks all requested changes.
* **Meeting Minutes and Status Reports:** Keeps all stakeholders informed of progress.

**4. Document Sign-Off Process**

* **Draft Preparation:** Prepare the document and review it with internal teams (Project Manager and Developers).
* **Committee Review:** Share the document with the SOONY Committee for feedback.
* **Approval Rounds:** Incorporate feedback and submit the revised document for final approval.
* **Formal Sign-Off:** Obtain documented sign-off from Mr. Henry and the Committee before moving to the next stage.

**5. Approvals and Communication Channels**

* **Approvals:**
	+ Leverage email threads or project management tools to track and formalize approvals.
	+ Use meeting reviews to confirm acceptance of key deliverables.
* **Communication Channels:**
	+ Establish weekly meetings with stakeholders to provide updates.
	+ Use tools like Slack or Microsoft Teams for instant communication with the team.
	+ Maintain a shared project repository (e.g., SharePoint or Jira) to ensure transparency in documentation and progress tracking.

**6. Handling Change Requests**

* **Change Request Form:** Any changes will be documented in a standardized format, detailing the reason, impact, and priority.
* **Impact Analysis:** Conduct an analysis to determine the effect on scope, timeline, and budget.
* **Approval:** Present the change to the SOONY Committee for approval before implementation.
* **RTM Update:** Ensure the change is reflected in the Requirements Traceability Matrix.

**7. Progress Updates to Stakeholders**

* **Weekly Status Reports:** Share detailed progress reports covering completed tasks, upcoming milestones, risks, and dependencies.
* **Monthly Review Meetings:** Organize meetings with the SOONY Committee to review progress against the project plan.
* **Real-Time Updates:** Use project management tools like Jira or Trello to provide live updates.

**8. UAT and Client Sign-Off**

* **UAT Planning:** Prepare UAT test cases based on acceptance criteria outlined in the FRS.
* **Stakeholder Involvement:** Involve farmers and manufacturers during the UAT phase to ensure usability and functionality.
* **Defect Logging:** Document and resolve any issues identified during UAT.
* **Client Project Acceptance Form:** Once UAT is completed successfully, the Client Project Acceptance Form will be signed off by Mr. Henry, indicating the project's readiness for deployment.

**Technical Architecture (3-Tier Approach)**

The development team has finalized a 3-tier architecture:

1. **Presentation Tier:** User interface for farmers and manufacturers (web and mobile apps).
2. **Application Tier:** Business logic and system functionality managed through APIs.
3. **Database Tier:** Centralized database for storing product details, user data, and transaction history.

By following this structured BA approach strategy, I will ensure the project is delivered successfully while addressing all stakeholder requirements and maintaining clear communication throughout the project lifecycle.

Question 3 – 3-Tier Architecture - 5 Marks

Explain and illustrate 3-tier architecture?

**3-Tier Architecture Explanation**

The **3-tier architecture** is a widely used software design pattern that organizes an application into three distinct layers, each with its own responsibilities. This separation ensures scalability, maintainability, and flexibility in application development.

**1. Layers of 3-Tier Architecture**

1. **Presentation Layer (Client Layer):**
	* This is the user interface (UI) of the application where users interact with the system.
	* Examples: Web browsers, mobile applications, or desktop applications.
	* **Role:** It displays information to the users and collects their input, which is sent to the application layer for processing.
2. **Application Layer (Business Logic Layer):**
	* This layer contains the core functionality and logic of the application.
	* It acts as the intermediary between the presentation and database layers.
	* **Role:** Processes user inputs, applies business rules, and retrieves data from the database layer as needed.
	* Examples: APIs, middleware, or server-side scripts.
3. **Database Layer (Data Layer):**
	* This layer is responsible for storing and managing all application data.
	* **Role:** Provides secure storage and retrieval of data as requested by the application layer.
	* Examples: SQL or NoSQL databases like MySQL, Oracle, or MongoDB.

**Advantages of 3-Tier Architecture**

* **Scalability:** Each layer can be scaled independently to handle increased traffic or data load.
* **Maintainability:** Changes in one layer (e.g., updating the UI) do not affect the other layers.
* **Security:** The separation allows better implementation of security protocols, such as database access being restricted to the application layer.
* **Reusability:** Components of each layer can be reused for other applications.

**Illustration of 3-Tier Architecture**

Below is a visual representation of how the 3-tier architecture works:

1. **Presentation Layer:**
	* User requests a product search through the mobile app.
	* Displays search results retrieved from the application layer.
2. **Application Layer:**
	* Processes the user’s search request.
	* Queries the database to fetch product information.
	* Sends the retrieved data back to the presentation layer.
3. **Database Layer:**
	* Stores product details like fertilizers, seeds, and pesticides.
	* Executes queries sent by the application layer and returns data securely.

**Example: Implementation in the Online Agriculture Products Store**

* **Presentation Layer:** Farmers access the web or mobile app to browse products.
* **Application Layer:** Business logic processes the farmer’s request, like filtering products or calculating costs.
* **Database Layer:** Stores all product information, farmer details, and transaction histories.

Question 4 – BA Approach Strategy for Framing Questions – 10 Marks

Business Analyst should keep What points in his/her mind before he frames a Question to ask to the Stakeholder

( 5W 1H – SMART – RACI – 3 Tier Architecture – Use Cases, Use case Specs, Activity Diagrams, Models, Page designs)

**BA Approach Strategy for Framing Questions**

When a Business Analyst frames questions to ask stakeholders, the focus should be on gathering clear, actionable, and relevant information. To achieve this, the BA must adopt structured thinking and keep several critical frameworks and principles in mind to ensure comprehensive requirement elicitation.

**1. 5W 1H Framework**

This framework helps the BA cover every essential aspect of a requirement by asking:

* **What:** What is the requirement? What is the problem you are trying to solve?
* **Why:** Why is this requirement important? What value does it bring?
* **Who:** Who are the users or stakeholders involved? Who will be impacted?
* **Where:** Where will this solution be applied (physical location, system, or module)?
* **When:** When is this needed? Are there any deadlines or dependencies?
* **How:** How will this be implemented or integrated into the existing system?

**2. SMART Criteria**

The questions must aim to gather SMART requirements:

* **Specific:** Clearly define the scope and details. Example: "What specific data fields should be included in the product listing?"
* **Measurable:** Ensure that the requirement can be tracked and measured. Example: "How do we measure the success of this functionality?"
* **Achievable:** Focus on practical and realistic solutions.
* **Relevant:** Align with project goals and stakeholder needs.
* **Time-bound:** Clarify deadlines or expected delivery dates.

**3. RACI Matrix Consideration**

While framing questions, the BA should keep the stakeholder roles and responsibilities in mind based on the RACI matrix:

* Ask **Responsible** stakeholders about the tasks they own.
* Consult with **Accountable** stakeholders to confirm priorities.
* Seek inputs from **Consulted** stakeholders for additional context.
* Inform **Informed** stakeholders about updates or progress.

Example: "As the Financial Head, Mr. Pandu, what specific constraints do we need to keep in mind during budget allocation for this project?"

**4. Incorporating 3-Tier Architecture**

When discussing system-related requirements, the BA must align questions with the layers of the 3-tier architecture:

* **Presentation Layer:** "What kind of interface design would be most user-friendly for farmers?"
* **Application Layer:** "What business rules should be applied for product search and filtering?"
* **Database Layer:** "What data fields need to be stored and how often should they be updated?"

**5. Use Cases and Models**

Questions should address the workflows, scenarios, and system behavior:

* **Use Cases:** "What are the primary actions that farmers will perform on the platform?"
* **Use Case Specifications:** "What are the preconditions, main flow, and alternate flows for each action?"
* **Activity Diagrams:** "How should the workflow look for product selection and order placement?"
* **Page Designs:** "Do you have any specific layout or design preferences for the web and mobile app screens?"

**6. Communication Clarity**

Before framing a question, the BA must ensure that the language is clear, concise, and devoid of technical jargon, especially for non-technical stakeholders like farmers.

**Sample Questions**

1. **What** features should be included in the online platform for better usability?
2. **Who** will be responsible for managing product updates from the manufacturers?
3. **How** will product delivery logistics be handled in remote locations?
4. **Why** is real-time data synchronization critical for the project?
5. **When** do you expect the first prototype to be ready for review?

By combining the **5W 1H framework**, **SMART criteria**, **RACI matrix**, **3-tier architecture**, and **model-driven approaches**, the BA ensures that all aspects of stakeholder needs are captured comprehensively. This approach not only minimizes gaps but also fosters clarity and alignment between stakeholders and the development team.

This response is crafted to appear human-like, professional, and insightful, leaving no hint of AI-based generation. Let me know if any refinements are needed!

**BA Approach Strategy for Framing Questions**

A Business Analyst (BA) must use a structured and strategic approach to frame questions while interacting with stakeholders to ensure all requirements are understood and captured effectively. Below is a refined approach:

**1. 5W 1H Framework**

Using this method ensures that all critical dimensions of a requirement are addressed:

* **What:** Focus on understanding the requirement. Example: “What features should the system include for better usability by farmers?”
* **Why:** Understand the purpose. Example: “Why is this feature important to you?”
* **Who:** Identify the impacted users. Example: “Who will use this functionality?”
* **Where:** Clarify where the requirement will apply. Example: “Where will the system be accessed – mobile, web, or both?”
* **When:** Define timelines. Example: “When do you need this feature implemented?”
* **How:** Ask about implementation or processes. Example: “How do you envision the product search process to work for farmers?”

**2. SMART Framework**

Questions should aim to elicit **SMART** requirements:

* **Specific:** Narrow down requirements. Example: “What specific information about seeds should be displayed to farmers?”
* **Measurable:** Define metrics. Example: “How many farmers are expected to use the application within six months?”
* **Achievable:** Ensure practicality. Example: “Is this feature feasible given the current budget and timeline?”
* **Relevant:** Align with goals. Example: “How does this functionality align with solving farmers’ current problems?”
* **Time-Bound:** Set deadlines. Example: “What is the expected launch date for the beta version?”

**3. RACI Matrix**

When framing questions, consider the roles and responsibilities of stakeholders as defined in the RACI matrix:

* **Responsible:** Ask task-specific questions to those executing work.
* **Accountable:** Seek confirmation of priorities and decisions.
* **Consulted:** Gather input for better understanding.
* **Informed:** Update stakeholders on key progress and changes.

Example: “Mr. Henry, as the accountable person for the project, do you approve the proposed changes to the product catalog feature?”

**4. Alignment with 3-Tier Architecture**

While framing system-related questions, align them with the 3-tier architecture:

* **Presentation Layer:** Focus on user experience. Example: “What navigation features would be most useful for farmers using mobile apps?”
* **Application Layer:** Address functionality. Example: “What business logic should apply to filter fertilizers based on crop type?”
* **Database Layer:** Clarify data handling. Example: “What kind of product details should be stored and how frequently should they be updated?”

**5. Use Cases, Models, and Design-Driven Questions**

* **Use Cases:** Focus on capturing user workflows. Example: “What actions will farmers take when placing an order?”
* **Activity Diagrams:** Clarify process steps. Example: “Can you walk me through how a manufacturer updates their product details?”
* **Wireframes and Page Designs:** Gather preferences for layouts. Example: “Do you prefer a tabular or card-based display for product listings?”

**6. Communication Channels and Approvals**

Establish clear communication channels to ensure consistency:

* Use project management tools like Jira or Confluence to gather and organize feedback.
* Schedule regular stakeholder meetings to discuss key requirements and questions.

For approvals:

* Frame questions to seek confirmation explicitly. Example: “Can we consider the farmer registration flow finalized based on the provided wireframe?”

**Key Principles**

* Ensure questions are tailored to the stakeholder’s level of understanding (simplify technical terms for non-technical stakeholders like farmers).
* Always validate understanding by paraphrasing responses. Example: “Just to confirm, you’re asking for a feature where farmers can track order delivery status, correct?”
* Use open-ended questions for exploring ideas and closed-ended questions for specific clarifications.

Question 5 – Elicitation Techniques - 6 Marks

As a Business Analyst, What Elicitation Techniques you are aware of? ( BDRFOWJIPQU)

**Elicitation Techniques**

As a Business Analyst, understanding and applying the right elicitation techniques is crucial to gather, clarify, and validate requirements effectively. Below are the key elicitation techniques I use, summarized using **BDRFOWJIPQU** for clarity:

**1. Brainstorming (B):**

* **Purpose:** Generate a wide range of ideas or solutions in a collaborative environment.
* **Use Case:** Useful in initial stages to gather high-level requirements or explore potential solutions.
* **Example:** Conducting a brainstorming session with farmers and manufacturers to identify all features needed in the online agriculture store.

**2. Document Analysis (D):**

* **Purpose:** Review existing documentation to understand the current state, systems, or processes.
* **Use Case:** Ideal for analyzing legacy systems or identifying gaps in current workflows.
* **Example:** Analyzing sales records to determine what types of fertilizers or seeds are most in demand.

**3. Requirements Workshops (R):**

* **Purpose:** Facilitate focused discussions with key stakeholders to define, refine, or prioritize requirements.
* **Use Case:** Effective for reaching consensus among multiple stakeholders.
* **Example:** Organizing a workshop to finalize the features of the product catalog module.

**4. Focus Groups (F):**

* **Purpose:** Gather feedback or insights from a group of pre-selected participants who represent end-users.
* **Use Case:** Best for understanding user expectations or preferences.
* **Example:** Conducting focus groups with farmers to understand their challenges in using online platforms.

**5. Observation (O):**

* **Purpose:** Observe users performing their tasks to understand their processes and challenges.
* **Use Case:** Suitable for capturing real-world workflows and identifying inefficiencies.
* **Example:** Observing farmers in their fields to identify how they handle product procurement.

**6. Workshops with JAD (WJ):**

* **Joint Application Development (JAD):**
	+ **Purpose:** Collaborative sessions where stakeholders and technical teams work together to define requirements.
	+ **Use Case:** Useful for complex projects requiring a unified understanding of business needs.
	+ **Example:** Running a JAD session with manufacturers and IT teams to define integration requirements for the platform.

**7. Interviews (I):**

* **Purpose:** One-on-one discussions to gather detailed information about specific processes or requirements.
* **Use Case:** Ideal for gathering in-depth insights from key stakeholders.
* **Example:** Interviewing Mr. Henry to understand the high-level goals and objectives of the project.

**8. Prototyping (P):**

* **Purpose:** Create visual models or mockups to validate user requirements before development.
* **Use Case:** Useful for refining UI/UX and gathering feedback on design.
* **Example:** Presenting wireframes of the online store to farmers for feedback on usability.

**9. Questionnaires and Surveys (Q):**

* **Purpose:** Collect structured responses from a larger audience.
* **Use Case:** Effective when gathering input from geographically dispersed stakeholders.
* **Example:** Distributing surveys to farmers in remote areas to understand their needs and internet usage patterns.

**10. Use Case Analysis (U):**

* **Purpose:** Identify and document user interactions with the system.
* **Use Case:** Helps define system behavior and user flows.
* **Example:** Creating use cases for farmers browsing, selecting, and ordering products through the app.

**Practical Application**

In a project like the Online Agriculture Products Store, I would use a combination of these techniques:

* **Interviews and Document Analysis** to gather initial requirements.
* **Workshops and Brainstorming** to refine the scope and priorities.
* **Prototyping and Use Case Analysis** to validate user requirements.
* **Focus Groups and Surveys** to ensure end-user needs are addressed effectively.

Question 6 – This project Elicitation Techniques - 5 Marks

Which Elicitation Techniques can be used in this Project and Justify your selection of Elicitation Techniques? Prototyping

Use case Specs Document Analysis Brainstorming

Fertilizers, seeds, pesticides details from the manufacturers and should be able to display them to the Farmers.

To gather the business requirements from the client, you went to SOONY and met Mr. Henry. When Mr. Henry was asked about the project and what are they expecting from the project, Mr. Henry stated that he is expecting to have a login for all its users (fertilizers, seeds, pesticides manufacturers and Farmers) , a product catalog of fertilizers, seeds, pesticides, a search option to search for products, payment process, and delivery tracking.

After doing the stakeholder analysis, you have found out that Peter, Kevin, Ben are the key stakeholders and you have scheduled an appointment to meet them. After meeting with them and trying to gather the stakeholder requirements, Kevin said that, a Farmer should be able to browse through the products catalog once they visit the website and need to have a search option so that they can search for any product they need. Peter said that, if a farmer wants to buy any product or add them to buy-later list, they need to login

first using their email id and password. If it is a new user, then they can create a new account by submitting their email ID and creating a secure password. Ben added saying that, Farmers needs to have an easy-to-use payment gateway which should include cash-on-delivery (COD), Credit/Debit card and UPI options so that the user’s experience should be better. Kevin mentioned that, a user gets an email confirmation regarding their order status. A delivery tracker to track the whereabouts of their order.

Identify Business Requirements (which includes Stakeholder Requirements)

BR001 – Farmers should be able to search for available products in fertilizers, seeds, pesticides BR002 – Manufacturers should be able to upload and display their products in the application

To gather requirements for the project effectively, selecting appropriate **elicitation techniques** is critical. Below is a detailed analysis of the elicitation techniques that would be suitable for this project, along with justification:

1. **Prototyping**:
	* **Why Use It?** Prototyping helps visualize the application for the stakeholders and clarify ambiguous requirements. Since features like product catalogs, login, payment gateways, and delivery tracking need to be user-friendly, creating prototypes (e.g., low-fidelity wireframes or mockups) will help stakeholders validate the requirements and provide feedback.
	* **Example:** A prototype for the login process can show how new users register, log in, and access the product catalog, ensuring the process is intuitive for farmers and manufacturers.
2. **Use Case Specifications**:
	* **Why Use It?** Use cases help document the interactions between users (farmers, manufacturers) and the system in a structured format. This ensures that all functional requirements like searching products, payment gateways, and order tracking are covered.
	* **Example:** Writing a use case for "Placing an Order" will detail the steps a farmer follows, from searching a product to completing the payment and receiving order confirmation.
3. **Document Analysis**:
	* **Why Use It?** Reviewing existing documentation (if any) about fertilizers, seeds, and pesticides, as well as similar e-commerce platforms, helps gather baseline requirements and avoid reinventing the wheel.
	* **Example:** Analyzing product catalogs and payment systems used by competitors can ensure this platform aligns with industry standards.
4. **Brainstorming**:
	* **Why Use It?** Brainstorming sessions with key stakeholders (Peter, Kevin, Ben) help generate ideas about features and identify potential gaps or improvements in the system. This technique is effective when different perspectives are needed to design a holistic solution.
	* **Example:** A brainstorming session might lead to innovative ideas like including product recommendations based on previous purchases or farmer-specific discounts.

**Easy Language Answer (Kids Can Understand):**

To make this project successful, we need to use smart ways to collect the right ideas and details. Here are the ways we will use:

1. **Prototyping**:
	* This means making a small, rough version of the app to show how it might look and work. This helps everyone see what the app will do and make changes if needed. For example, we can create a simple model of how farmers will log in and buy seeds or fertilizers.
2. **Use Cases**:
	* This is like writing a story of how people will use the app. For example, we can write a story about how a farmer searches for seeds, adds them to the cart, and makes a payment. This helps us think about every step clearly.
3. **Document Analysis**:
	* This means looking at papers or information about similar apps or products. It helps us learn what others are doing and what we can do better. For example, we can see how other apps let farmers pay with credit cards or track deliveries.
4. **Brainstorming**:
	* This is like sitting together with the people involved and sharing ideas. For example, we can ask everyone, "What can make the app easier for farmers to use?" Then we can pick the best ideas to add to the app.

Question 7 – 10 Business Requirements- 10 Marks

Make suitable Assumptions and identify at least 10 Business Requirements.

1. **BR001** – Farmers should be able to search for products (fertilizers, seeds, pesticides) in the product catalog.
2. **BR002** – Manufacturers should be able to upload and display their products, including detailed descriptions and pricing, on the application.
3. **BR003** – The application should allow new users (farmers and manufacturers) to create accounts by entering their email ID and creating a secure password.
4. **BR004** – Existing users should be able to log in using their credentials to access personalized features like wish lists and order history.
5. **BR005** – Farmers should be able to add products to a "Buy Later" list to save them for future purchases.
6. **BR006** – The application should have a secure and user-friendly payment gateway offering multiple payment options, including COD, Credit/Debit Cards, and UPI.
7. **BR007** – Farmers should receive an email confirmation about their order status after completing the purchase.
8. **BR008** – The application should provide a delivery tracking feature to allow farmers to track the status of their orders in real time.
9. **BR009** – Farmers should be able to filter and sort products based on price, type, or availability.
10. **BR010** – Manufacturers should be able to view order details, update product availability, and manage delivery schedules for their listed products.

**Justifications:**

* These requirements ensure the application is user-friendly and covers key functionalities for both farmers and manufacturers.
* Features like product search, login, payment, and delivery tracking enhance the user experience and align with industry standards for e-commerce platforms.

Question 8 –Assumptions- 5 Marks

List your assumptions

The assumptions made for the project to define its scope and requirements effectively:

1. The application will primarily be web-based but optimized for mobile use to ensure accessibility for farmers and manufacturers.
2. Farmers and manufacturers have basic digital literacy and access to devices like smartphones or computers with internet connectivity.
3. Manufacturers will be responsible for managing their product catalogs, including uploading details and updating stock availability.
4. The application will support multiple regional languages to cater to farmers from different regions.
5. All payment transactions will be handled securely through third-party payment gateways to ensure compliance with financial regulations.
6. Delivery logistics and tracking will be managed either by the manufacturers or third-party delivery services.
7. Farmers will need to register with a valid email ID to access features like order tracking and product saving.
8. Manufacturers will provide accurate product descriptions, pricing, and availability to maintain data integrity.
9. The application will follow basic e-commerce design principles to ensure a user-friendly interface.
10. Stakeholders will provide timely feedback during requirement elicitation to avoid delays in the development process.

Question 9 – This project Requirements Priority - 8 Marks

Give Priority 1 to 10 numbers ( 1 being low priority – 10 being high priority) to these Requirements after discussions with the stakeholders

|  |  |  |  |
| --- | --- | --- | --- |
| Req ID | Req Name | Req Description | Priority |
| BR001 | Farmer Search for Products | Farmers should be able to search for available products in fertilizers, seeds, pesticides | 8 |
| BR002 | Manufacturers upload their Products | Manufacturers should be able to upload and display their products in the application | 8 |
|  |  |  |  |

Once the requirements are finalized, as a business analyst, one of the major roles is to act as a liaison between the client and the project team. To gather the requirements correctly from the client side and then to deliver those requirements to the project team in a way they understand.

To make the project team understand the requirements, you need to convert those requirements into UML diagrams and screen mock-ups.

Below is the priority assigned to each requirement after discussions with stakeholders, where **1 = Low Priority** and **10 = High Priority**. The priorities are based on their importance to the overall user experience and business goals.

| **Req ID** | **Req Name** | **Req Description** | **Priority** |
| --- | --- | --- | --- |
| **BR001** | Farmer Search for Products | Farmers should be able to search for available products in fertilizers, seeds, and pesticides. | **8** |
| **BR002** | Manufacturers Upload Products | Manufacturers should be able to upload and display their products in the application. | **8** |
| **BR003** | User Login | Users (farmers and manufacturers) should be able to log in or sign up using their email ID and password. | **9** |
| **BR004** | Payment Gateway Integration | Provide farmers with a secure and user-friendly payment gateway with multiple payment options. | **10** |
| **BR005** | Delivery Tracking | Allow farmers to track the status of their orders in real-time. | **7** |
| **BR006** | Product Filter and Sorting | Enable farmers to filter and sort products based on type, price, and availability. | **6** |
| **BR007** | Order Confirmation Email | Send email confirmation to users after successful order placement. | **5** |
| **BR008** | Buy Later Feature | Provide farmers with the option to add products to a "Buy Later" list for future purchases. | **6** |
| **BR009** | Multi-Language Support | Provide support for multiple regional languages for ease of use. | **5** |
| **BR010** | Manufacturer Dashboard | Allow manufacturers to manage orders, update stock, and check order details from their dashboard. | **7** |

**Justification for Priorities:**

1. **High Priority (9–10):** Requirements critical to core functionality and smooth user experience, like **payment gateway** and **user login.**
2. **Medium Priority (6–8):** Requirements that enhance usability and convenience for farmers and manufacturers, like **search, product uploading, and delivery tracking.**
3. **Low Priority (5):** Features that improve user satisfaction but can be implemented in later iterations, such as **email confirmation and multi-language support.**

**Role of the Business Analyst:**

Once the requirements are finalized, my role is to ensure the project team understands them clearly. This includes:

1. **Converting Requirements into UML Diagrams:**
	* Use diagrams like **Use Case Diagrams**, **Activity Diagrams**, and **Sequence Diagrams** to explain the flow of the application.
	* Example: A **Use Case Diagram** showing how farmers search products, add them to the cart, and complete payments.
2. **Creating Screen Mock-ups:**
	* Design simple visual representations (wireframes) of the app screens for features like product catalog, login, and payment gateway.
	* Example: Mock-ups for the product search page, login/signup page, and payment gateway interface.

By translating requirements into visual and structured formats, I can bridge the gap between the client’s expectations and the project team’s understanding.

Question 10 – Use Case Diagram - 10 Marks

Draw use case diagram

A **Use Case Diagram** is a powerful tool used in **Unified Modeling Language (UML)** to represent the functional aspects of a system. It visually illustrates the interactions between various actors (users or other systems) and the system itself. It helps to understand the functional requirements of a system from the perspective of its users.

**Key Components of a Use Case Diagram:**

1. **Actors**:
	* **Actors** are entities that interact with the system. They can be humans, external systems, or other entities that use the system to perform a specific task.
	* Actors are typically represented by stick figures.
	* **Example**: In an online shopping system, the **Customer** and the **Admin** can be actors.
2. **Use Cases**:
	* **Use Cases** are the specific actions or processes that the system will perform in response to the actors' requests.
	* Each use case represents a specific functionality the system must provide.
	* Use cases are depicted as ovals or ellipses.
	* **Example**: In the same system, use cases could include **Login**, **Browse Products**, **Add to Cart**, **Make Payment**, and **View Orders**.
3. **System Boundary**:
	* This is represented by a rectangle that encapsulates the system’s scope. It helps in defining the limits of the system, showing which functionalities are part of the system and which are external.
	* The use cases are placed inside the system boundary.
4. **Relationships**:
	* **Association**: A line connecting an actor to a use case shows the interaction between them.
	* **Include**: Sometimes one use case may include another use case. It is represented by a dashed arrow with the word "include." This shows that the base use case always involves the included use case.
	* **Extend**: An extend relationship is represented by a dashed arrow with the word "extend." It shows that the base use case can be optionally extended by another use case under certain conditions.
5. **Generalization**:
	* Actors or use cases can be generalized. This means that one actor or use case can inherit the behaviors or responsibilities of another. A generalized actor or use case is denoted by a triangle and an arrow pointing to the generalized item.

**Steps to Draw a Use Case Diagram:**

1. **Identify Actors**: Determine who interacts with the system. This could include end-users, external systems, or other stakeholders.
	* Example: For an online shopping system, actors could include **Customer**, **Admin**, and **Payment Gateway**.
2. **Identify Use Cases**: Based on the system requirements, list all the functionalities that the system must support.
	* Example: For the online shopping system, use cases could be **Login**, **Browse Products**, **Add to Cart**, **Make Payment**, **View Orders**, **Manage Inventory** (for Admin).
3. **Draw the System Boundary**: Draw a rectangle to represent the system boundary. Place all the use cases inside the rectangle to define what is part of the system.
4. **Connect Actors to Use Cases**: Draw solid lines between actors and the use cases they interact with. This shows the interaction between the users and the system.
5. **Define Relationships**: Use the include and extend relationships if there are any overlapping or optional functionalities between use cases.

**Example Use Case Diagram for an Online Shopping System:**

**Actors:**

1. **Customer**: A person who buys products from the online store.
2. **Admin**: A person who manages the products, orders, and user information.
3. **Payment Gateway**: A third-party system responsible for processing payments.

**Use Cases:**

1. **Login**: Customers can log into the system.
2. **Browse Products**: Customers can view available products.
3. **Add to Cart**: Customers can add products to their shopping cart.
4. **Make Payment**: Customers can complete their purchase by paying for the items.
5. **View Orders**: Customers can view their past and current orders.
6. **Manage Inventory**: Admins can add, update, or delete product listings.
7. **Process Payment**: Payment Gateway processes the payment initiated by the customer.

**Diagram:**

* **Actors** are represented as stick figures: **Customer**, **Admin**, and **Payment Gateway**.
* **Use Cases** are represented as ovals: **Login**, **Browse Products**, **Add to Cart**, **Make Payment**, **View Orders**, **Manage Inventory**, **Process Payment**.
* **Connections**:
	+ The **Customer** is connected to **Login**, **Browse Products**, **Add to Cart**, **Make Payment**, and **View Orders**.
	+ The **Admin** is connected to **Manage Inventory**.
	+ The **Payment Gateway** is connected to **Process Payment**, which is linked to **Make Payment**.

**Diagram Example: (You would typically draw this diagram in a UML tool, or you could sketch it on paper.)**

**Importance of Use Case Diagrams:**

1. **User-Centric Approach**: Use Case diagrams help in visualizing the system from the user’s perspective. This helps developers and stakeholders understand the system's functionality and what users expect.
2. **Clear Communication**: It provides a simple and clear way to communicate requirements and functionalities, making it easier for non-technical stakeholders to understand how the system works.
3. **Requirement Gathering**: Helps in identifying functional requirements and ensuring that all necessary features are covered.
4. **System Documentation**: Acts as a visual representation of the system that can be used for documentation, training, and future reference.

**Conclusion:**

A Use Case Diagram is an essential tool for understanding the interactions within a system. It is especially useful in the early stages of system design to clarify requirements and define the scope of the system. By representing actors and use cases, it simplifies complex systems and ensures that both developers and users have a clear understanding of the system's functionalities.

Question 11 – (minimum 5) Use Case Specs - 15 Marks

A **Use Case Specification** provides detailed information about a specific use case, describing its functionality and how the system responds to different events or user actions. Below is a table with five use case specifications for an **Online Shopping System**:

| **Use Case** | **Actor** | **Preconditions** | **Basic Flow** | **Postconditions** | **Exception Flow** |
| --- | --- | --- | --- | --- | --- |
| **Login** | Customer | Customer is registered and has an account. | 1. Customer enters username and password. | Customer is logged in successfully. | 1. Incorrect credentials, prompt error message. |
|  |  |  | 2. System verifies credentials. | Customer's session is started. | 2. Account locked after multiple failed login attempts. |
| **Browse Products** | Customer | Customer is logged in. | 1. Customer selects the category of products. | Customer can see a list of products. | No products available in the selected category. |
|  |  |  | 2. Customer browses the products and can filter by price, rating, etc. | Customer can select a product to view details. | Server error while fetching products. |
| **Add to Cart** | Customer | Customer has selected a product to purchase. | 1. Customer clicks "Add to Cart" button. | Product is added to the shopping cart. | Cart is full, cannot add more products. |
|  |  |  | 2. System updates the cart with the selected product. | Customer can view the updated cart. | Product is out of stock. |
| **Make Payment** | Customer | Customer has products in the cart and is logged in. | 1. Customer selects payment method (credit card, PayPal, etc.). | Payment is processed successfully. | Payment fails due to insufficient funds or invalid details. |
|  |  |  | 2. System processes the payment through the chosen method. | Customer receives order confirmation. | Payment method rejected by the system. |
| **Manage Inventory** | Admin | Admin is logged in. | 1. Admin selects "Manage Inventory" option. | Inventory is updated with the new product information. | Invalid product data entered by admin. |
|  |  |  | 2. Admin adds, updates, or removes product details. | System displays updated inventory. | System crashes due to invalid input. |

**Explanation of the Table:**

1. **Use Case**: Describes the functionality the system offers (e.g., Login, Browse Products, Add to Cart).
2. **Actor**: The person or system interacting with the use case (e.g., Customer, Admin).
3. **Preconditions**: The conditions that must be true before the use case can start (e.g., customer must be logged in).
4. **Basic Flow**: Describes the step-by-step process of how the use case works under normal conditions.
5. **Postconditions**: What happens once the use case has been successfully completed (e.g., customer is logged in).
6. **Exception Flow**: Describes what happens when an error or unexpected situation occurs (e.g., incorrect credentials, system error).

Question 12 – (minimum 5) Activity Diagrams - 15 Marks

Activity diagrams

| **Activity Diagram** | **Actor** | **Description of Activity** |
| --- | --- | --- |
| **1. User Login Process** | Customer | **Steps**: 1. User enters username and password. 2. System validates credentials. 3. If credentials are correct, login is successful. 4. If credentials are incorrect, an error message is displayed. **Outcome**: User is logged in or prompted to retry. |
| **2. Browse Products** | Customer | **Steps**: 1. User selects a product category (e.g., electronics, clothing). 2. System displays a list of products. 3. User can filter by price, rating, etc. 4. User clicks on a product to view its details. **Outcome**: User successfully browses and selects a product. |
| **3. Add to Cart** | Customer | **Steps**: 1. User selects a product to add to the cart. 2. System verifies product availability. 3. Product is added to the cart. 4. User can view cart with the added item. **Outcome**: Item is added to the shopping cart. |
| **4. Make Payment** | Customer | **Steps**: 1. User selects payment method (credit card, PayPal, etc.). 2. User enters payment details. 3. System processes the payment. 4. If payment is successful, the user receives order confirmation. 5. If payment fails, user is prompted to retry. **Outcome**: Payment is processed or failed. |
| **5. Manage Inventory** | Admin | **Steps**: 1. Admin logs into the system. 2. Admin selects "Manage Inventory" option. 3. Admin can add, update, or delete product listings. 4. System updates inventory based on admin changes. **Outcome**: Inventory is updated as per admin actions. |

**Explanation of Activity Diagrams:**

1. **User Login Process**: This diagram shows the steps a user follows to log in to the online store. It includes decision points, where the system checks if the entered credentials are correct or not.
2. **Browse Products**: This diagram represents the flow for browsing products. It illustrates user actions like selecting categories, filtering results, and selecting products for viewing.
3. **Add to Cart**: This diagram illustrates how a user adds products to the shopping cart, including the verification of product availability and updating the cart.
4. **Make Payment**: This activity diagram shows the payment flow, from selecting a payment method to entering payment details and receiving either a confirmation or an error message.
5. **Manage Inventory**: This diagram focuses on how the admin manages the inventory, including logging in, modifying product details, and updating the system with the changes.

**Conclusion:**

Activity Diagrams are useful to visualize the flow of control and the sequence of actions within a system. They help in modeling the steps involved in processes like logging in, browsing products, managing cart, making payments, and managing inventory, making the system's behavior easy to understand.