Q1: 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?

 **BA Responsibilities in Quarterly Audits**

**Q1 Audit (Requirement Gathering & Analysis Phase)**

**Objective:** Validate initial requirements and stakeholder alignment.

**BA Responsibilities:**

* Ensure **Business Requirement Document (BRD)** & **Software Requirement Specification (SRS)** are complete and approved.
* Verify **Stakeholder Sign-offs** on requirements.
* Check that requirements are mapped to business goals (Traceability Matrix).
* Ensure all requirement changes are documented (Change Log) Participate in **audit meetings** with the Project Manager & Committee.

 **Q2 Audit (Design & Early Development Phase)**

 **Objective:** Ensure design aligns with requirements and early development follows specifications.

**BA Responsibilities:**

* Verify that the **Functional Specification Document (FSD)** and **UI/UX Prototypes (Axure, Figma)** are accurate.
* Confirm that developers are following **approved system designs**.
* Ensure system **workflows and process diagrams** match stakeholder expectations.
* Review **Requirement Traceability Matrix (RTM)** to ensure no missed requirements.
* Validate that **early development versions** are aligned with business objectives.

**Q3 Audit (Testing & System Integration Phase)**

 **Objective:** Ensure test cases align with business requirements and defects are managed efficiently.

**BA Responsibilities:**

* Verify that **Test Cases** cover all business scenarios.
* Ensure **User Acceptance Testing (UAT) Plan** is well-defined and documented.
* Review defect logs from **Testers (JIRA/Azure DevOps)** to confirm traceability to requirements.
* Ensure **Test Reports** are aligned with functional requirements.
* Validate that **non-functional requirements (performance, security, usability)** are being tested.

**Q4 Audit (UAT, Deployment & Post-Implementation Phase)**

**Objective:** Validate the system’s readiness for go-live and ensure user feedback is addressed.

**BA Responsibilities:**

* Facilitate **UAT (User Acceptance Testing)** sessions and document feedback.
* Ensure any **last-minute requirement changes** are documented and approved.
* Validate that **training documents and user manuals** are ready for end-users.
* Confirm that the **final system aligns with business goals**.
* Participate in **Post-Implementation Review (PIR)** meetings to analyze project success.

 **Key Audit Artifacts a BA Must Maintain**

* **Business Requirement Document (BRD)**
* **Software Requirement Specification (SRS)**
* **Requirement Traceability Matrix (RTM)**
* **Change Logs (CRs & Approvals)**
* **Process & Data Flow Diagrams**
* **Test Case Documentation & UAT Reports**
* **Stakeholder Sign-off Records**

Q2 BA Approach Strategy - 6 Marks

A "BA approach strategy" refers to a planned method or roadmap that a business analyst (BA) uses to conduct their analysis activities within a project, outlining the specific techniques, timing, and deliverables they will focus on to achieve the project objectives, essentially acting as a guide to navigate the business analysis process effectively.

**Business Analyst Approach Strategy for the SOONY Online Agriculture Product Store**

**Prepared by:** Gautham M
**Business Analyst, APT IT SOLUTIONS**
**Submitted to:** Mr. Karthik (Delivery Head)
**Date:** [18/02/2025]

 **BA Approach to Project Execution**

This document outlines the **Business Analysis (BA) approach strategy** for the SOONY Online Agriculture Product Store. The project will follow the **V-Model SDLC** and implement a **3-tier architecture**. As a **Business Analyst (BA),** I will ensure effective **requirement gathering, documentation, stakeholder management, and change control** throughout the project lifecycle.

**1️. Requirement Elicitation Techniques**

To gather detailed business and technical requirements, I will use the following techniques:

* **Interviews** – One-on-one discussions with Mr. Henry, Mr. Pandu, Mr. Dooku, and farmer representatives (Peter, Kevin, Ben).
* **Workshops** – Conduct group meetings with client stakeholders and development teams.
* **Surveys & Questionnaires** – Collect feedback from potential farmer users and manufacturing companies.
* **Document Analysis** – Review existing agricultural product sales and procurement reports.
* **Prototyping** – Create UI wireframes using **Axure** for validation before development.
**Observation** – Understand how farmers currently procure fertilizers, seeds, and pesticides.

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

 **RACI Matrix (Responsibility Assignment Chart)**

| **Stakeholder** | **Responsible (R)** | **Accountable (A)** | **Consulted (C)** | **Informed (I)** |
| --- | --- | --- | --- | --- |
| **Mr. Henry (Business Owner)** |  | A | C | I |
| **Mr. Pandu (Financial Head)** |  | A | C | I |
| **Mr. Dooku (Project Coordinator)** |  | A | C | I |
| **Peter, Kevin, Ben (Farmer Reps)** |  |  | C | I |
| **Mr. Karthik (Delivery Head)** | A |  | C | I |
| **Mr. Vandanam (Project Manager)** | R | A | C | I |
| **Business Analyst (You)** | R |  | C | I |
| **Developers (Ms. Juhi, Mr. Teyson, Ms. Lucie, Mr. Tucker, Mr. Bravo)** | R |  |  | I |
| **Testers (Mr. Jason, Ms. Alekya)** | R |  |  | I |
| **DB & Network Admins (Mr. John, Mr. Mike)** | R |  |  | I |
| **Farmers (End Users)** |  |  | C | I |
| **Manufacturing Companies** |  |  | C | I |

**Influence-Interest Stakeholder (ILS) Model** will also be used to categorize stakeholders based on power and interest in the project.

**3️. Key Documents to Prepare & Sign-off Process**

| **Document** | **Purpose** | **Owner** | **Approval Required By** |
| --- | --- | --- | --- |
| Business Requirement Document (BRD) | Captures high-level business needs | BA | Mr. Henry, Mr. Pandu, Mr. Dooku |
| Software Requirement Specification (SRS) | Converts business needs into detailed system requirements | BA & Developers | Project Manager, Client Committee |
| Functional Specification Document (FSD) | Defines UI/UX, workflows, and system behaviors | BA & UI Designers | Project Manager, Client Committee |
| Requirement Traceability Matrix (RTM) | Ensures requirements are mapped to design and test cases | BA & Testers | Project Manager |
| Test Plan & Test Cases | Defines test strategy and scenarios for validation | Testers | Project Manager |
| User Acceptance Testing (UAT) Plan | Defines acceptance criteria for end users | BA & Testers | Client Committee |
| Change Request Form (CRF) | Tracks modifications and approvals | BA | Project Manager, Client Committee |
| UAT Sign-Off & Project Acceptance Form | Confirms final system approval before deployment | BA & Testers | Client Committee |

 **Sign-Off Process:**
✔ BA drafts and reviews documents internally.
✔ Documents are shared with stakeholders for feedback.
✔ Final approval from the **client committee (Mr. Henry, Mr. Pandu, Mr. Dooku).**
✔ Signed-off documents are stored in a centralized repository.

**4️. Communication Channels for the Project**

 **Internal Communication (APT IT SOLUTIONS Team)**

* **Daily Standup Meetings** (Scrum-style for updates).
* **Weekly Progress Meetings** with Mr. Vandanam (PM).
* **Slack & Microsoft Teams** for quick discussions.
* **JIRA / Azure DevOps** for task tracking.

 **External Communication (Client & Stakeholders)**

* **Weekly Status Reports** to the client.
* **Bi-Weekly Stakeholder Meetings** for updates and approvals.
* **Formal Email Updates** after each milestone.
* **Client Feedback & Review Sessions** via Zoom/MS Teams.

**5.Change Request (CR) Handling Process**

 **Process:**

1️. **Client Requests Change** → BA captures details in a **Change Request Form (CRF).**
2️. **Impact Analysis** → BA & Dev Team evaluate scope, cost, and timeline impact.
3️. Approval **from Client Committee** (Only major changes require sign-off).
4️. **Update RTM & Documentation** → Ensure traceability.
5️. **Implement Change** → Developers update the system.
6️. **Retest & Deploy** → Testers validate the changes before release.

**Change Tracking Tool:** JIRA / Azure DevOps

**6️ Progress Reporting to Stakeholders**

* **Daily Progress Updates** – Internal updates for development & testing teams.
* **Weekly Status Reports** – Shared with the **client committee.**
* **Monthly Review Meetings** – Review project health and milestones.
* **Quarterly Audits (Q1, Q2, Q3, Q4)** – Ensure compliance with business goals.

**7️ UAT & Final Client Acceptance Process**

 **Steps to Complete UAT & Sign-Off:**
✔ Conduct **User Acceptance Testing (UAT)** with key users (Farmers, Suppliers).
✔ Document **feedback & defects** and address them.
✔ Obtain **UAT Sign-Off Form** from **Mr. Henry, Mr. Pandu, Mr. Dooku.**
✔ **Project Acceptance Form** signed by the client, confirming successful delivery.

 **Summary of BA Approach Strategy**

* **Elicitation Techniques:** Interviews, Workshops, Surveys, Prototyping
* **Stakeholder Analysis:** RACI & ILS Models for role clarity
* **Documentation & Sign-offs:** BRD, SRS, RTM, FSD, UAT Plans
* **Communication Channels:** Meetings, Reports, JIRA, Teams
* **Change Request Handling:** CRF, Impact Analysis, Client Approval
* **Progress Updates:** Daily, Weekly, Monthly, Quarterly Audits
* **UAT & Final Sign-Off:** UAT Testing, Defect Resolution, Client Sign-Off

Q3: Explain and illustrate 3-tier architecture?

The **3-Tier Architecture** is a structured approach to application development that separates the system into three independent layers. Each layer has a distinct role, making the system **scalable, maintainable and efficient.**

**Layers of 3-Tier Architecture**

1️. Application or Client Layer

* This is the front-end of the application where users interact.

This is the **user interface (UI)** that allows Farmers and Manufacturers to interact with the system. Manufacturers can **list products, manage inventory, and process orders**.

* It includes web browsers, mobile apps, or desktop applications. Farmers can **browse products, order, pay, and track deliveries**.
* Technologies used: HTML, CSS, JavaScript, React, Angular, Vue.js.

2️. Business Logic Tier

* Handles the core logic and processes of the application. This layer processes **business logic** and acts as a bridge between the UI and the database.
* Responsible for validations, calculations, and communication between the UI and database. It ensures **secure authentication, order management, payment processing, and inventory updates**.
* Technologies used: Java (Spring Boot), .NET, Python (Django), Node.js, Payment gateway.

3️. Database Layer

* Stores, retrieves, and manages data securely. This layer is responsible for **storing and managing data** related to users, products, orders, and transactions.
* Ensures data consistency, security, and efficient querying. It handles **data consistency, security, and backup**.
* Technologies used: MySQL, PostgreSQL, MongoDB, Oracle DB. AWS cloud.

Q4: BA Approach Strategy for Framing Questions

**Key Considerations for a Business Analyst Before Framing Questions for Stakeholders**

As a **Business Analyst (BA)**, asking the right questions is critical to gathering accurate and complete requirements. Before framing a question for stakeholders, a BA should consider the following:

**1️. 5W 1H Approach**

A structured way to frame questions using:

* **Who** – Who are the users of this system? Who will be impacted?
* **What** – What problem are we solving? What features are needed?
* **When** – When should this be delivered? When is it needed?
* **Where** – Where will this solution be used? (Mobile, Web, Cloud, etc.)
* **Why** – Why is this requirement important? Why is this a priority?
* **How** – How will this feature work? How will it impact business goals?

**Example:**
*"What challenges do farmers face when ordering products? How do they currently purchase fertilizers?"*

**2. SMART Approach (Specific, Measurable, Achievable, Relevant, Time-bound)**

* **Specific** – Define clear and detailed objectives.
* **Measurable** – Identify how success will be measured.
* **Achievable** – Ensure the requirement is realistic.
* **Relevant** – Align with business needs and goals.
* **Time-bound** – Define deadlines and priorities.

**Example:**
*"What specific product categories should be included in the online store? How will we measure the success of the ordering process?"*

**3. RACI Matrix (Responsible, Accountable, Consulted, Informed)**

Before asking questions, a BA should identify who is:

* **Responsible** – Who performs the task?
* **Accountable** – Who makes final decisions?
* **Consulted** – Who provides input or feedback?
* **Informed** – Who needs to be kept updated?

 **Example:**
*"Mr. Henry (Accountable), can you confirm the final list of products? Mr. Pandu (Consulted), do you have any budget concerns?"*

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

A BA should ensure that requirements align with the system's architecture:

* **Application Layer:** What UI elements are required? (Forms, filters, buttons)
* **Business Logic Layer:** What workflows and processes need automation?
* **Database Layer:** What data should be stored? (Orders, products, payments)

**Example:**
*"How should the product search feature be designed? Should farmers be able to filter by category?"*

**5. Use Cases & Specifications**

* Identify the **actors** (users) and **use cases** (features).
* Define **system interactions** and how users will complete actions.
* Ensure clarity in **use case specifications** (preconditions, postconditions, flow).

 **Example:**
*"Can you walk me through the process of how a farmer currently orders products? What happens if a product is out of stock?"*

**6. Activity Diagrams & Models**

* Use **activity diagrams** to understand workflows.
* Consider **ER diagrams, BPMN, flowcharts, wireframes** to model requirements visually.

**Example:**
*"Can you confirm the steps in processing an order from checkout to delivery?"*

**7. Page Designs & User Experience (UX)**

* Ensure **user-friendly UI** for Farmers and Manufacturers.
* Define **navigation, filters, and accessibility**.
* Consider **mobile responsiveness** for rural users.

**Example:**
*"Should farmers be able to track orders in real-time? Would a chatbot help in resolving queries?"*

Q5. As a Business Analyst, What Elicitation Techniques you are aware of? ( BDRFOWJIPQU)

**B - Brainstorming**

* Used to generate **new ideas, solutions, and features**.
* Encourages open discussion among stakeholders.
* Helps identify **hidden requirements.**

 **Example:** Identifying innovative features for SOONY's online agriculture store.

**D - Document Analysis**

* Reviewing **existing documentation** (SOPs, policies, past project reports).
* Helps understand **business rules and processes**.
* Useful for regulatory and compliance-driven projects.

 **Example:** Studying SOONY's financial reports to align budget constraints.

**R – Reverse engineering**

* Understanding an **existing system by analyzing its components and functionalities**.
* Helps in **migrating systems or improving outdated applications**.
* Identifies **gaps in current processes**.

 **Example:** Examining a competitor’s agriculture e-commerce platform to refine SOONY’s features. Or other e-commerce platforms.

**F - Focus Groups**

* Gathering insights from a **targeted group of end-users**.
* Helps in understanding **user expectations and challenges**.
* Efficient for identifying **UI/UX improvements**.
* **Example:** Interviewing farmers in different regions to understand internet connectivity issues.

**O - Observations (Job Shadowing)**

* Observing **how users perform tasks in real-life scenarios**.
* Helps detect **pain points and inefficiencies**.
* Useful when users **can’t clearly articulate their needs**.

 **Example:** Watching how farmers currently buy seeds and fertilizers offline.

**W - Workshops**

* A structured session involving **business & technical teams**.
* Helps define **system requirements, UI, and process flows**.
* Encourages real-time feedback and decision-making.

 **Example:** Conducting a JAD session to finalize the SOONY app’s workflow.

**J – Joint Application Development**

* Facilitates structured meetings between business and IT teams.
* Helps resolve requirement conflicts and clarify expectations.
* Encourages collaborative decision-making.

Example: A JAD session between SOONY’s business team and developers to finalize database requirements.

**I – Interview**

* One-on-one or group discussions with **key stakeholders**.
* Helps uncover **specific business, functional, and non-functional requirements**.
* Can be **structured, semi-structured, or unstructured**.

 **Example:** Interviewing SOONY’s financial head, Mr. Pandu, to understand the project's budget constraints.

**P - Prototyping**

* Creating a **mock-up or wireframe** of the application.
* Helps stakeholders visualize the final product.
* Reduces misunderstandings in requirements.

 **Example:** Designing UI wireframes for the SOONY app before finalizing development.

 **Q - Questionnaires & Surveys**

* Used for collecting **quantitative and qualitative data**.
* Helps in decision-making based on large user responses.
* Useful for gathering feedback **from geographically dispersed stakeholders**.

 **Example:** Sending a survey to farmers asking about their preferred payment methods.

**U - Use Cases & User Stories**

* Defining **how users interact** with the system.
* Helps developers and testers understand **expected behaviour**.
* Clarifies **business goals through real-world scenarios**.

 **Example:** Creating a use case for "Farmer places an order for seeds".

Q6. Which Elicitation Techniques can be used in this Project and Justify your selection of Elicitation Techniques? Prototyping, Use case Specs, Document Analysis, Brainstorming

**Prototyping**

* Since the application must be user-friendly, creating wireframes and prototypes will help visualize the UI/UX.
* Enables stakeholders to review the design before development, reducing rework.
* Ensure the checkout and payment process is seamless.

Justification in Project:

* Helps farmers and manufacturers understand how the product catalogue, cart, and checkout process will work.
* Allows quick feedback loops before full-scale development.
* Creating wireframes for the login page, product catalogue, cart, and order tracking module for stakeholders to review

**Use Case Specifications**

* Defining how each type of user (Farmer, Manufacturer, Admin) interacts with the system.
* Ensures that business and functional requirements are mapped to actual system behaviour.
* Helps create test cases for validation.

Example Usage:

* Use case: "Farmer places an order"
	+ Actors: Farmer, System, Manufacturer.
	+ Pre-condition: Farmer logs in and selects a product.
	+ Steps: Add to cart → Choose payment option → Confirm purchase → Receive order confirmation.

**Document Analysis**

Since fertilizer, seed, and pesticide manufacturers already have product catalogs, business documents, and regulatory requirements, analysing these documents will help in understanding:

* Product details (types, pricing, availability).
* Compliance and regulations related to selling agricultural products.
* Existing sales and logistics workflows to integrate into the new system.

Example Usage:

* Examining the overall transport solutions for rural areas and their management can assist in adopting delivery methods from companies that are implementing them.

**Brainstorming**

* Engaging key stakeholders (Mr. Henry, Peter, Kevin, and Ben) in a brainstorming session will help:
* Identify pain points faced by farmers in procurement.
* Generate innovative ideas for features, such as personalized recommendations and bulk-order discounts.
* Prioritize essential features like product search, login system, payment options, and delivery tracking.

Example Usage:

* Conducting a brainstorming session with farmers and manufacturers to explore how an automated bidding system could benefit them.
* Conducting a brainstorming session with stakeholders to identify key functionalities like COD payment options and product filtering.

Q7: Identify Business Requirements (which includes Stakeholder Requirements)

A **BRD (Business Requirement Document)** is a formal document that outlines the business needs, objectives, and high-level requirements of a project. It serves as a bridge between business stakeholders and the development team, ensuring everyone has a clear understanding of what needs to be achieved.

**BR001** – Farmers should be able to search for available products in fertilizers, seeds, and pesticides.

**BR002** – Manufacturers should be able to upload and display their products in the application.

**BR003** – Farmers should be able to browse through the product catalogue .

**BR004** – Farmers must log in to add products to the **buy-later list** or make a purchase.

**BR005** – New users should be able to **register** by submitting their email ID and creating a secure password.

**BR006** – The system should allow **secure authentication and login** for users.

**BR007** – The application should include a **search feature** for products based on keywords, category, and price range.

**BR008** – Farmers should be able to add items to the **shopping cart and proceed to checkout**.

**BR009** – The application should support **multiple payment methods**, including:

* Cash on Delivery (COD)
* Credit/Debit Cards
* UPI Transactions

**BR010** – Farmers should receive an **email confirmation** regarding their order status.

**BR011** – The system should have a **delivery tracking** feature to show the order’s status and estimated arrival time.

**Q8: List your Assumptions**

As a Business Analyst (BA), making assumptions is necessary to fill gaps in information, ensure smooth project execution, and mitigate risks. Assumptions help in defining the scope, feasibility, and requirements of a project when all details are not explicitly provided.

* **Login Access:** User should be able to login using Facebook or Gmail account.
* **Internet Access:** Farmers and manufacturers will have stable internet access to use the online platform effectively.
* **User Familiarity:** Farmers and manufacturers have basic knowledge of using web and mobile applications.
* **Device Compatibility:** The platform will be accessible via desktops, smartphones, and tablets.
* **Product Availability:** Manufacturers will regularly update and maintain the availability of fertilizers, seeds, and pesticides.
* **Delivery Feasibility:** Delivery services will be available to transport products from manufacturers to farmers in remote locations.
* **Secure Transactions:** The payment gateway will support secure transactions via COD, UPI, and Credit/Debit cards. Assuming the user has some kind of quick payment options
* **Regulatory Compliance:** The application will comply with local agricultural and e-commerce regulations.
* **Scalability:** The system will be scalable to handle an increasing number of users and transactions.
* **Multi-Language Support:** The platform may need localization features for farmers who prefer regional languages.
* **Customer Support Availability:** A dedicated support team will be available for resolving user issues.
* **Authentication and Authorization:** Each user will have a unique login, and role-based access control (RBAC) will be implemented.
* **Hosting and Maintenance:** APT IT Solutions will provide hosting and technical support post-deployment.
* **Data Privacy:** User information, including payment details, will be securely stored and managed.
* **User Onboarding:** A simple and guided onboarding process will be provided for new users.
* **Order Tracking:** Farmers can track their orders in real-time through the application

**Q9: This project Requirements Priority.**

Prioritization Techniques are commonly used by Business Analysts to determine the priority of requirements in a structured way. For the Online Agriculture Product Store Project, the MoSCoW Method would be the best approach because: MoSCoW clearly separates critical vs. optional features.

**1. MoSCoW Method (Must-Have, Should-Have, Could-Have, Won’t-Have)**

* **Must-Have (M)** → Essential for the project to function (e.g., Login, Payment Gateway, Order Tracking,).
* **Should-Have (S)** → Important but not critical (e.g. Buy-Later List, E-mail confirmations, Invoices sent to E-mail, Guest checkout).
* **Could-Have (C)** → Nice-to-have features (.Multi-Language Support, Recently Purchased, Suggestions, AI-Based Product Recommendations, Help Chat Bot.)
* **Won’t-Have (W)** → Out of scope for now but could be considered in future releases.

Priority Numbering for Online Agriculture Product Store

| **Req ID** | **Requirement Name** | **Priority (1-10)** | **Reason for Priority** |
| --- | --- | --- | --- |
| **BR001** | Farmers/manufactures/admin Login for Purchase/Add products/Management | 10 | Ensures security and personalized access |
| **BR002** | Manufacturers Upload Their Products | 9 | Essential for product availability |
| **BR003** | Farmers Browse Product Catalogue | 8 | Enhances user experience, not mandatory for transactions |
| **BR004** | Farmer Search for Products | 10 | Key feature to find products quickly |
| **BR005** | New User Account Creation | **9** | Required for onboarding & transactions |
| **BR006** | Multiple Payment Options | **10** | Essential for completing purchases |
| **BR007** | Email Confirmation for Orders | 9 | Important for order transparency |
| **BR008** | Order Tracking System | **10** | Critical for customer satisfaction |
| **BR009** | Farmer wants to buy any product or add them to buy-later list, | 8 | Improves accessibility but not urgent |
| **BR010** | Customer Support Availability | **7** | Needed for issue resolution |

Q10: Draw use case diagram :

A Use Case Diagram is an essential tool in software development, especially for Business Analysts (BAs), because it helps visualize the system’s functional requirements from the perspective of its users. Here’s why we need it:

* + 1. Understanding User Interactions
		2. Requirement Clarity & Validation
		3. Communication with Stakeholders
		4. Basis for Other Diagrams & Development
		5. Helps Developers & Testers
		6. Supports Change Management



Q11: Prepare use case specs for all use cases:

A **Use Case** is a detailed description of how a **user (actor)** interacts with a **system** to achieve a specific goal. It defines the **functional requirements** of the system and ensures that all possible user interactions are accounted for.

|  |  |
| --- | --- |
| Use case ID | UC1 |
| Use case name | User Login |
| Created By | MG |
| Date created | 18/02/2025 |
| Actor | Farmers, Manufacturers, Admin |
| Description | Users must login to access the platform |
| Pre- Condition | The users must have a registered account The user must have an active internet connection |
| Post – Condition | The user is successfully logged in will be directed to the previous page of login. |
| Normal Flow of events | User enters email and password.System validates credentials.If valid, login is successful.User is redirected to the homepage. |
| Alternate flow | invalid credentials: The system shows an error message and prompts for re-entry.Will be directed to the password /login recovery page |
| Assumptions | Farmers have access to Gmail or Facebook to use it for registration. |

|  |  |
| --- | --- |
| Use case ID | UC2 |
| Use case name | Browse Products |
| Created By | MG |
| Date created | 18/02/2025 |
| Actor | Farmers  |
| Description | Farmers can browse Fertilizers, seeds and pesticides |
| Pre- Condition | The user must be logged in.Manufacturers must have uploaded products. |
| Post – Condition | The user is successfully logged in will be directed to the previous page of login |
| Normal Flow of events | The farmer navigates to the product catalog.The system displays available products with details. |
| Alternate flow | No products available: The system displays a message: “No products found.” |
| Assumptions |  |

|  |  |
| --- | --- |
| Use case ID | UC3 |
| Use case name | Search Products |
| Created By | MG |
| Date created | 18/02/2025 |
| Actor | Farmers  |
| Description | Farmers can search for specific products using keywords. |
| Pre- Condition | Products exit in the catalog. |
| Post – Condition | The system displays relevant information of the search product |
| Normal Flow of events | The farmer enters a search keyword.The system fetches and displays matching products.. |
| Alternate flow | No matches available: The system displays a message: “No results found.” |
| Assumptions | Farmers are looking for a specific product and the manufacture is there on the database of the platform |

|  |  |
| --- | --- |
| Use case ID | UC4 |
| Use case name | Upload Products |
| Created By | MG |
| Date created | 18/02/2025 |
| Actor | Manufacturers |
| Description | Manufacturers add new products to the catalogue. |
| Pre- Condition | Manufactures must be logged in. |
| Post – Condition | The product is added in and displayed in the catalogue. |
| Normal Flow of events | Manufacturer selects "Upload Product."Manufacturer enters product details.The system saves the product. |
| Alternate flow | **Incomplete details:** The system prompts the user to enter missing information |
| Assumptions | Farmers are looking for a specific product and the manufacture is there on the database of the platform |

|  |  |
| --- | --- |
| Use case ID | UC5 |
| Use case name | Manage users |
| Created By | MG |
| Date created | 18/02/2025 |
| Actor | Admin |
| Description | Admin can manage user accounts |
| Pre- Condition | Admin must be logged in |
| Post – Condition | Users can be added/removed |
| Normal Flow of events | Admin selects user management.Adds/Removes users.System updates records. |
| Alternate flow | User not found: system shows an error |
| Assumptions | Searched party is registered with the platform |

Q12: Activity Diagrams

 1.Simple login activity diagram:



 2. Branch and Merge.

 

 3. Fork and join

 

4. Nested Loop:



5. Swim Lane

 