

CAPSTONE PROJECT (PART-2)

Online Agriculture Products Store Project

Introduction:

This case study revolves around a Corporate Social Responsibility (CSR) initiative taken by Mr. Henry, a successful businessman who wanted to give back to society by helping farmers from remote areas. After a casual conversation with his childhood friends Peter, Kevin, and Ben—who are farmers—Mr. Henry became aware of the challenges they face in procuring essential agricultural products like fertilizers, seeds, and pesticides. These problems highlighted a gap in the agricultural supply chain, especially in rural areas where access to quality products is limited.

To address this issue, Mr. Henry proposed the idea of building an Online Agriculture Products Store, which would serve as a digital platform enabling direct communication between farmers and manufacturing companies. Through this platform, farmers would be able to browse and purchase required products conveniently from their mobile devices or computers.

The project is funded by Mr. Henry's company, SOONY, with a budget of ₹2 Crores and a planned duration of 18 months. The project is being executed by APT IT SOLUTIONS, a technology company selected for their skilled talent pool and relevant experience. A dedicated team comprising developers, testers, network and database admins, along with project managers and business analysts, has been formed to bring this idea to life.

The main goal of this initiative is to simplify access to essential agricultural inputs for farmers and contribute to their growth and productivity. The project not only has commercial potential but also a strong social impact, making it a significant example of how technology can be used to solve real-world problems.

Team Overview:

For the successful execution of the Online Agriculture Products Store project, a collaborative team structure has been established, involving various stakeholders, technical experts, and support roles from both the client and service provider sides.

Below is the breakdown of the team involved in the project:

Client Side (SOONY Company)

- **Mr. Henry** – He is the initiator of the project and also one of the wealthiest businessmen in the city. He came up with this idea as part of a CSR initiative to help remote farmers, including his childhood friends.
- **Mr. Pandu** – Financial Head at SOONY. He is responsible for overseeing the financial management and budget allocation for the project.
- **Mr. Dooku** – Project Coordinator from SOONY. He is involved in supervising the project progress and coordination from the client's side.
- **Committee Members** – Mr. Henry, Mr. Pandu, and Mr. Dooku form a decision-making committee for this project.
- **Stakeholder Farmers** – Peter, Kevin, and Ben (friends of Mr. Henry) are farmers from a remote village and are acting as key stakeholders. They are actively involved in sharing requirements and feedback based on actual farming needs.

Vendor Side (APT IT SOLUTIONS)

APT IT SOLUTIONS was selected as the execution partner for this CSR project. The team structure from their side includes:

- **Mr. Karthik** – Delivery Head. He played an important role in bagging the project for APT IT SOLUTIONS through his professional network and will oversee overall delivery.
- **Mr. Vandanam** – Project Manager. He is responsible for planning, execution, and ensuring the project is completed within budget and timeline.
- **Ms. Juhi** – Senior Java Developer. She leads the technical development efforts and helps guide junior developers.
- **Java Developers** – Mr. Teyson, Ms. Lucie, Mr. Tucker, and Mr. Bravo are part of the Java development team working on core features of the application.
- **Mr. Mike** – Network Administrator. He handles networking and infrastructure-related aspects to support the application.
- **Mr. John** – Database Administrator. He is in charge of designing and managing the project's database.
- **Mr. Jason and Ms. Alekya** – QA Testers. They are responsible for testing the application and ensuring its quality before deployment.
- **My Role – Business Analyst (BA)** – I have joined this project as the Business Analyst. I'm responsible for gathering and analyzing requirements from stakeholders, preparing

documentation like SRS and use cases, and ensuring the developed solution meets the business needs.

Question 1: Audits

An **audit** is a **formal, systematic, and independent evaluation** carried out to determine whether activities, processes, or deliverables:

1. Comply with defined standards, policies, or requirements
2. Are traceable, accurate, and complete
3. Meet the intended business objectives

It is considered part of the **Governance and Monitoring domain**.

- Audits verify whether **business analysis activities and outputs** (requirements, models, traceability, approvals, UAT, etc.) are being performed **according to agreed processes, organizational standards, and regulatory guidelines**.
- They help ensure that **the BA's work contributes to value delivery** and that no requirement gaps, compliance issues, or uncontrolled changes (scope creep) exist.

Types of Audits in a BA Context:

- **Process Audit** → Did the BA follow proper elicitation, documentation, validation techniques?
- **Compliance Audit** → Do requirements and deliverables meet **organizational, regulatory, or BABOK V3-aligned standards**?
- **Traceability Audit** → Can every requirement be traced from **business need** → **solution** → **test case** → **delivery**?
- **Quality Audit** → Are requirements clear, unambiguous, testable, and aligned with business goals?

In this 18-month Agriculture E-commerce project, **4 quarterly audits (Q1, Q2, Q3, Q4)** will be conducted. Each audit checks **compliance and progress** from a BA standpoint:

Q1 Audit (Months 1–3) – *Requirement Gathering & Planning*

- Check if **elicitation activities** were performed systematically.
- Validate **stakeholder engagement** (Farmers, Sponsor, CSR body, Developers).

-
- Ensure **Business Process Model (BPM), SWOT analysis, requirement prioritization (Hundred Dollar Method)** are documented and baselined.
 - Confirm **traceability** of business needs → requirements.

Q2 Audit (Months 4–9) – *Development & Parallel Testing Begins*

- Review **requirements lifecycle management** – are changes tracked and approved?
- Audit **requirements documentation (BRD, SRS, user stories, acceptance criteria)**.
- Confirm BA's involvement in **supporting developers and testers** during d1–d2, t1–t2 cycles.
- Ensure **compliance with BABOK elicitation techniques** (interviews, workshops, observation).

Q3 Audit (Months 10–15) – *UAT Preparation & Execution*

- Check if **requirements are validated** with stakeholders (esp. farmers & sponsor).
- Confirm BA's role in **UAT planning, facilitation, and defect triaging**.
- Ensure BA's **timesheets and deliverables** (traceability matrix, use cases, workflows) are maintained.
- Verify **stakeholder approvals** for solution scope and test cases.

Q4 Audit (Months 16–18) – *Deployment & Closure*

- Verify BA's role in **final solution evaluation** – does the solution meet business objectives?
- Audit **knowledge transfer & training** documentation for farmers/users.
- Review BA's contribution to **benefits realization analysis**.
- Confirm BA compliance with **organizational governance** and BABOK standards.

Question 2: BA Approach Strategy

Before the project initiation, a well-structured Business Analysis (BA) Approach is essential to ensure alignment with organizational goals, stakeholder expectations, and delivery timelines. According to **BABOK v3 – Business Analysis Planning and Monitoring**, the BA must define

how business analysis will be performed, the deliverables to be produced, and the processes for stakeholder engagement and approvals.

Stakeholder Analysis and Roles- (RACI/ILS)

- **Identify Stakeholders:**
 - Client representatives (SOONY company, Farmer groups, End customers).
 - Internal team (PM – Mr. Vandanam, Developers – Ms. Juhi, Mr. Teyson, Ms. Lucie, Mr. Tucker, Mr. Bravo, Testers – Mr. Jason & Ms. Alekya, Network Admin – Mr. Mike, DB Admin – Mr. John).
- **RACI Matrix:**
 - **Responsible:** BA (requirements elicitation, documentation), Developers/Testers (execution).
 - **Accountable:** Project Manager (overall delivery), Client sponsor (final approvals).
 - **Consulted:** Domain SMEs, Farmers, End users.
 - **Informed:** Wider project stakeholders, support teams.
- **Influence/Level of Support (ILS):** Map stakeholders by influence (decision-making power) vs. support (commitment level) to prioritize engagement strategies.

Elicitation Techniques-

- **Workshops** (farmer representatives, client team, internal developers).
- **Interviews** (with sponsor & SME).
- **Surveys/Questionnaires** (farmers/end-users in remote areas).
- **Document Analysis** (previous CSR reports, farmer schemes).
- **Prototyping** (wireframes/mock-ups in Figma/Framer to validate UI/UX early).

Documentation to be Produced-

- **Business Requirements Document (BRD)** – captures high-level business goals.
- **Functional Requirements Specification (FRS)** – detailed system requirements (aligned with 3-tier architecture: Presentation, Application, Database).

-
- **System Requirement Specification (SRS)** – technical translation for developers.
 - **Process Models** – BPMN diagrams for workflows (ordering, payment, logistics).
 - **RACI & Stakeholder Register**.
 - **Change Request Log**.
 - **UAT Plan & Client Acceptance Form**.

Process for Document Sign-off & Approvals-

- Draft document prepared by BA → Internal Review (PM, Tech Leads) → Client walkthrough sessions.
- Iterative feedback captured → Final version baseline stored in document repository.
- **Formal sign-off** through digital approval (email confirmation or e-signatures) from client sponsor.

Communication Channels-

- **Weekly Status Meetings** – with PM and technical leads.
- **Monthly Steering Committee Reviews** – progress with client sponsor.
- **Collaboration Tools** – MS Teams (meetings, chat), JIRA (backlog tracking), Confluence (documentation repository).
- **Audit Reviews (Q1–Q4)** – reporting alignment with client/governance body.

Change Request (CR) Handling-

- Formal **CR Template** maintained.
- Impact analysis performed (scope, cost, timeline).
- Approval sought from PM + Client before execution.
- Updated baseline requirements stored in the **Requirements Traceability Matrix (RTM)**.

Progress Reporting-

- **Weekly Dashboards** prepared by BA and shared with the PM.
- **Monthly Milestone Reports** presented to stakeholders.

-
- Requirements coverage tracked via **RTM** to ensure traceability from business needs → UAT.

UAT & Client Sign-off-

- BA prepares **UAT Plan** with testers (Mr. Jason & Ms. Alekya).
- UAT executed in client's presence with agreed test cases.
- Issues logged and tracked till closure.
- Final **Client Project Acceptance Form** signed-off to confirm project closure.

Question 3: 3-Tier Architecture

The **3-Tier Architecture** is a **client-server software architecture pattern** that separates an application into three logical layers: **Presentation Tier, Application Tier, and Data Tier**. This separation ensures scalability, maintainability, and security.

Presentation Tier (Client Layer)-

- The **front-end/UI layer** that interacts directly with the end-users (farmers, buyers).
- Handles input (forms, search, login) and displays output (product listings, order status).
- **Technologies:** Web browsers, mobile apps, HTML, CSS, JavaScript, React.

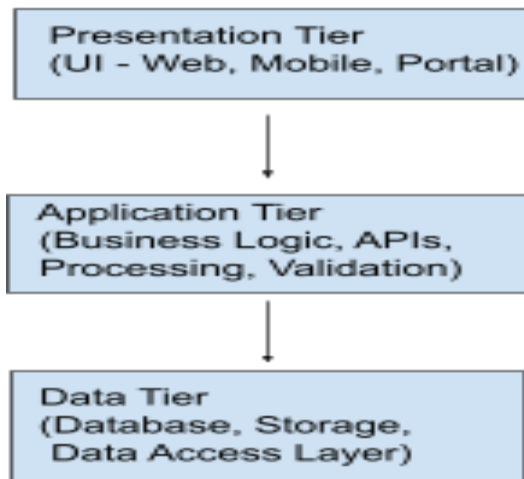
Application Tier (Business Logic Layer)-

- The **middle layer** that processes user requests.
- Implements the business rules, validations, and workflows.
- Ensures data is processed correctly before storage/retrieval.
- Technologies: Java (as per project team), Spring Boot, APIs.

Data Tier (Database Layer)-

- Responsible for **data storage, retrieval, and management**.
- Stores farmer details, product catalog, orders, payments, logistics.
- Ensures data integrity, security, and backups.

-
- Technologies: SQL/Oracle/MySQL, handled by DB Admin (Mr. John).



Benefits of 3-Tier Architecture-

- **Separation of concerns** → easier to maintain.
- **Scalability** → each layer can be scaled independently.
- **Security** → database is isolated from direct user access.
- **Reusability** → business logic can be reused across multiple UIs (web/mobile).

Question 4: BA Approach Strategy for Framing Questions

When a Business Analyst frames questions to ask stakeholders, the goal is to ensure **clarity, completeness, and relevance** of the elicited requirements.

Apply the 5W & 1H Framework-

Before framing a question, we have to ensure it addresses:

- **Who** – Who will use the system? Who are the stakeholders impacted?
- **What** – What functionality is required? What business rules apply?
- **When** – When will the process occur? What timelines or frequency?
- **Where** – Where will the system be used (location, environment)?
- **Why** – Why is this requirement needed? What is the business value?

-
- **How** – How should the requirement be implemented from the user’s perspective?

This ensures holistic elicitation without leaving gaps.

Ensure SMART Principle-

Each question framed should help uncover requirements that are:

- **Specific** – clear, focused (e.g., “What data should farmers provide during registration?”)
- **Measurable** – tied to measurable outputs (e.g., “How many product categories should be supported?”)
- **Achievable** – realistic and feasible within the project scope.
- **Relevant** – aligned to business goals (e.g., market access for farmers).
- **Time-bound** – defined within timelines (e.g., “By when should orders be processed?”).

Consider RACI Matrix Implications-

Questions should clarify **roles and responsibilities**:

- **Responsible** – Who will perform the task?
- **Accountable** – Who will approve/own it?
- **Consulted** – Who provides input?
- **Informed** – Who needs updates?

This ensures stakeholders are properly identified before seeking inputs.

Leveraging System Architecture Context (3-Tier Architecture)-

When framing questions, the BA should align them to the architecture:

- **Presentation Layer** – Questions on user interface, usability, and design preferences.
- **Application Layer** – Questions on business logic, workflows, rules, and integration.
- **Data Layer** – Questions on data storage, access, reporting, and security.

Use Cases & Use Case Specifications-

-
- Questions should uncover **actor goals, preconditions, triggers, and flows**.
 - Example: *“What happens if a farmer tries to upload a product without setting a price?”*
 - Helps refine **use case diagrams** and identify exceptions.

Activity Diagrams & Models-

- Questions should clarify **process flows, decision points, and alternate scenarios**.
- Example: *“What steps should be triggered after an order is placed?”*
- Supports building **activity diagrams and BPM models**.

Page Designs & UI Considerations-

- Ask questions related to layout, navigation, and accessibility.
- Example: *“Should farmers be able to filter products by location or category on the homepage?”*

Question 5:Elicitation Techniques

Elicitation is the practice of **drawing out, collecting, and discovering information** from stakeholders and other sources to understand needs, requirements, and constraints of a project.

It is not just asking questions but involves **engaging, observing, and analyzing stakeholders and systems** to uncover true requirements.

Elicitation Techniques (BDRFOWJIPQU)-

Brainstorming:

Brainstorming is an elicitation technique used to generate a wide range of ideas and solutions in a group setting. It is particularly useful for encouraging creativity, exploring possibilities, and uncovering requirements that might not surface through direct questioning.

Document Analysis:

Document analysis involves reviewing existing documentation such as policies, user manuals, reports, and contracts to identify relevant requirements. This technique provides a baseline understanding of processes and helps the BA build on existing knowledge rather than starting from scratch.

Requirement Workshops:

Workshops are structured sessions where multiple stakeholders collaborate to identify, refine, and prioritize requirements. This technique is efficient because it encourages discussion, alignment, and consensus-building within a controlled setting.

Focus Groups:

Focus groups bring together selected stakeholders or users to gather their opinions, preferences, and expectations. This technique is often used to understand user perspectives, validate product features, or capture market-driven requirements.

Observation:

Observation, also known as job shadowing, requires the BA to watch stakeholders as they perform their daily tasks. This technique helps uncover hidden needs and challenges that stakeholders themselves may not be consciously aware of or may forget to mention in interviews.

Walkthroughs:

A walkthrough is a step-by-step review of processes, requirements, or prototypes with stakeholders. It ensures mutual understanding, validates assumptions, and identifies any gaps or errors in requirements before moving further.

Joint Application Development (JAD) Sessions:

JAD sessions are facilitated workshops that bring business and IT stakeholders together to quickly gather and validate requirements. These sessions are highly collaborative and often reduce the time needed for requirement gathering.

Interviews:

Interviews are one-on-one or group discussions with stakeholders. They allow the BA to gather detailed insights, clarify ambiguities, and capture specific needs directly from the source in a conversational setting.

Prototyping:

Prototyping involves building mock-ups, wireframes, or models to visualize requirements. This

technique allows stakeholders to interact with early designs, provide feedback, and refine requirements before actual development begins.

Questionnaires and Surveys:

Questionnaires and surveys help collect structured information from a larger group of stakeholders. They are useful when time is limited or when feedback is required from distributed teams.

User Stories:

User stories are short, structured descriptions of requirements, usually expressed in the format: *As a [user], I want [requirement], so that [business value]*. This technique keeps the focus on user needs and ensures that requirements deliver tangible business outcomes.

Question 6: Elicitation Techniques in this project

The elicitation techniques listed in our assignment (**Prototyping, Use Case Specifications, Document Analysis, and Brainstorming**) are well-chosen and absolutely valid for this project. They align perfectly with the type of system being built and the diversity of stakeholders.

Prototyping: Prototyping is a requirements elicitation technique in which a preliminary version of the system (a prototype) is built to demonstrate functionality, user interfaces, or workflows. It allows stakeholders to interact with a model of the system early in the project, thereby clarifying ambiguous requirements and identifying gaps in understanding. It reduces the risk of miscommunication and ensures stakeholder expectations are aligned with system design.

- Prototyping is very effective for this project as it helps stakeholders like farmers and manufacturers visualize how the product catalog, search functionality, payment gateway, and delivery tracker will appear and function. Since many stakeholders may not be technical, interactive prototypes and wireframes will allow them to provide clear feedback on usability and layout before development begins.

Use Case Specifications: Use case specification is a structured way of capturing functional requirements by describing interactions between users (actors) and the system to achieve specific goals. A use case is defined as “a description of a system’s behavior as it responds to a request from one of its stakeholders.” Use case specifications provide detailed steps, preconditions, postconditions, and exceptions, making them a practical tool for documenting stakeholder requirements in a precise manner.

- Use case specifications are useful to capture how different users—farmers, manufacturers, and administrators—will interact with the system. For example, use cases can describe how a farmer registers, searches products, adds items to a cart, makes payments, or tracks delivery. Similarly, manufacturers’ use cases will capture how

they upload and manage product details. These structured scenarios help ensure that all functional requirements are documented in a detailed manner.

Document Analysis: Document analysis is the review and study of existing documentation to gather requirements and background knowledge about a system or domain. It involves examining materials such as business plans, process documents, regulations, or product descriptions to identify requirements that may not be explicitly stated by stakeholders. It ensures accuracy, regulatory compliance, and alignment with organizational goals.

- Document analysis is important because there will be existing reference material such as manufacturer product details, catalogs, pricing sheets, and government regulations related to fertilizers, seeds, and pesticides. By analyzing these documents, the BA can ensure accurate product information, compliance with standards, and alignment with existing business processes.

Brainstorming: Brainstorming is a creative group elicitation technique used to generate a broad set of ideas and requirements within a limited time. It is a technique that encourages free thinking, avoids criticism, and promotes collaboration to maximize the generation of potential solutions. In requirements elicitation, brainstorming enables diverse stakeholders to contribute perspectives, leading to richer and more innovative requirement sets.

- Brainstorming sessions with stakeholders such as Mr. Henry, Peter, Kevin, and Ben will help uncover additional requirements that might not surface through interviews alone. For instance, the idea of adding a “buy-later list,” COD option, or an email confirmation system can emerge from collective discussions. Brainstorming also helps reconcile differing stakeholder viewpoints and ensures innovative ideas are captured early.

But as a Business Analyst, I’d also consider that **no single set of techniques fits all situations**. In practice, a blend works best depending on stakeholders’ profiles, project complexity, and clarity of requirements. I would **also consider adding two more techniques**:

Workshops: Workshops are structured, collaborative sessions where stakeholders and business analysts work together to elicit, refine, and prioritize requirements. They are “facilitated meetings where stakeholders and development teams collaborate to produce high-quality requirements in a compressed time frame.” Workshops are valuable because they promote consensus-building, immediate feedback, and alignment among diverse stakeholder groups.

- Bringing together farmers, manufacturers, and SOONY’s internal team in a facilitated session can speed up requirement clarification and resolve conflicts early.

Observation / Job Shadowing: Observation, also known as job shadowing, is an elicitation technique in which the business analyst observes stakeholders in their actual work environment to understand their tasks, processes, tools, and challenges. It is “a technique used to elicit requirements by conducting an assessment of processes and activities in their natural context.” This method is particularly effective in capturing tacit knowledge and identifying pain points that stakeholders may not be able to articulate explicitly.

-
- Since farmers and manufacturers have unique workflows, observing how they currently search for, buy, and manage agricultural products can reveal real-world challenges that stakeholders may not articulate in interviews.

Conclusion: Therefore, the use of Prototyping, Use Case Specifications, Document Analysis, and Brainstorming provides a strong foundation for eliciting requirements in this project.

Prototyping brings clarity to stakeholders by visualizing the system, **use case specifications** capture the detailed step-by-step interactions, **document analysis** ensures accuracy from existing product and compliance records, and **brainstorming** fosters creativity and consensus.

In practice, this combination could be further strengthened by techniques such as **workshops** and **observation**, which allow deeper engagement with farmers and manufacturers who may not always articulate their needs clearly. Together, these approaches ensure that both business and stakeholder requirements—such as catalog browsing, secure logins, product uploads, payment options, and delivery tracking—are comprehensively gathered, validated, and documented.

Business & Stakeholder Requirements:

- **Business requirements** are higher-level statements of the goals, objectives, or needs of the enterprise. They describe why the initiative is undertaken and what value it should deliver to the business.
- **Stakeholder requirements** describe the needs of stakeholders that must be met in order to achieve the business requirements. They bridge the gap between business requirements and solution requirements.

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 catalog without login.

BR004 – Farmers should be able to create a new account by registering with their email ID or their phone number and a secure password.

BR005 – Registered users (Farmers and Manufacturers) should be able to log in using their email ID/ phone number and password.

BR006 – Farmers should be able to add products to the cart for purchase.

BR007 – Farmers should be able to add products to a "buy-later" list after login.

BR008 – The application should provide multiple payment options, including Cash on Delivery (COD), Credit/Debit Card, and UPI.

BR009 – Farmers should receive an email confirmation/ an SMS regarding their order placement and order status.

BR010 – Farmers should be able to track the delivery status of their orders using a delivery tracker.

BR011 – The system should ensure that all login, registration, and payment processes are secure to protect user data.

Question 7 & 8: 10 Business Requirements and Assumptions

BR001 – Unified Product Catalog

The system must provide a single, centralized catalog for fertilizers, seeds, and pesticides so that farmers can easily browse and compare products.

Assumption: *Farmers may not have time to visit multiple vendors physically, hence a unified catalog saves effort and cost.*

BR002 – Manufacturer Product Uploads

Manufacturers must be able to upload and update their product information (price, description, availability, images) directly in the platform.

Assumption: *Manufacturers need autonomy to manage their product listings without depending on intermediaries.*

BR003 – Secure User Authentication

The platform must provide secure login and registration for all user types (farmers and manufacturers) to ensure safe access and protect personal data.

Assumption: *Cybersecurity is a critical concern for protecting sensitive data like contact details and payment information.*

BR004 – Search and Filter Functionality

The system should allow farmers to search and filter products by category, price range, brand, or availability.

Assumption: *Farmers need quick access to specific products instead of scrolling through long lists.*

BR005 – Shopping Cart and Buy-Later Option

Farmers should be able to add items to a shopping cart or save them to a buy-later list before finalizing purchases.

Assumption: *Farmers may not make instant purchase decisions and need flexibility to revisit products.*

BR006 – Multiple Payment Options

The system must support diverse payment methods including Cash on Delivery (COD), UPI, and Debit/Credit cards.

Assumption: *Many farmers in rural areas prefer COD or UPI due to limited banking access.*

BR007 – Order Confirmation and Notifications

The platform should send email/SMS notifications for order confirmation, shipment, and delivery updates.

Assumption: *Farmers may have limited internet access, so SMS is a reliable backup channel along with email.*

BR008 – Delivery Tracking System

The application must provide a real-time delivery tracker so that farmers can monitor their order status.

Assumption: *Transparency in delivery builds trust and reduces farmer anxiety about shipment delays.*

BR009 – User-Friendly Interface

The system should be designed with simple navigation, clear icons, and support for local languages to make it farmer-friendly.

Assumption: *Many farmers may not be highly tech-savvy, so simplicity and accessibility are essential.*

BR010 – Reporting and Analytics for Manufacturers

The system should provide manufacturers with dashboards showing product performance, sales trends, and farmer feedback.

Assumption: *Manufacturers will want insights to improve their supply chain and marketing decisions.*

Question 9: Requirements Priority of this project

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 Products	Manufacturers should be able to upload and display their products in the application	8
BR003	User Registration & Login	Farmers and manufacturers should be able to register, login, and manage their accounts securely	10
BR004	Product Catalog	Farmers should be able to browse through a structured product catalog of fertilizers, seeds, and pesticides	9
BR005	Add to Buy-Later List	Farmers should be able to add selected products to a "Buy Later" list before purchase	7
BR006	Secure Payment Gateway	Payment gateway should support COD, Credit/Debit Cards, and UPI for convenience	10
BR007	Order Confirmation Notification	Farmers should receive email confirmation regarding order placement and payment	8
BR008	Delivery Tracking	Farmers should be able to track the delivery status of their orders	9
BR009	Role-Based Access Control	Different logins (farmers, manufacturers, admins) should have role-specific access	7
BR010	Product Reviews & Ratings	Farmers should be able to view and provide feedback on purchased products	6

Reasoning for priorities:

10 → Core functions like login (BR003) and payment (BR006) — critical to project success.

9 → Browsing catalog (BR004) and delivery tracking (BR008) — highly important for usability.

8 → Search (BR001), manufacturer uploads (BR002), and order confirmation (BR007) — very important but slightly less than payments/login.

7 → Buy-later list (BR005) and role-based access (BR009) — useful but not absolutely critical for MVP.

6 → Reviews/ratings (BR010) — good to have but lower priority.

Question 10: Use Case Diagram

A **Use Case Diagram** is a type of **behavioral diagram** defined by the **Unified Modeling Language (UML)** that represents the **functional requirements** of a system from the user's perspective. It shows the **interactions between external actors (users or other systems)** and the system itself, capturing the **goals (use cases)** that actors want to achieve.

Use case diagrams help to **visualize system boundaries**, identify **stakeholders' requirements**, and provide a **high-level view of the system functionality** without going into technical implementation details.

- **Purpose:**
 - Describes *what* the system should do, not *how* it will be implemented.
 - Focuses on functional requirements.
- **Actors:**
 - External entities (people, systems, organizations) that interact with the system.
 - Can be *primary actors* (initiating interaction) or *secondary actors* (support roles).
- **Use Cases:**
 - Functionalities or services the system provides to fulfill an actor's goal.
 - Represented as ovals.
- **System Boundary:**
 - Shown as a rectangle enclosing all use cases.
 - Defines what is inside (system responsibility) vs. outside (external actors).
- **Relationships:**
 - > **Association** (line between actor and use case) → interaction.

-
- >**Include** → mandatory sub-functionality.
 - >**Extend** → optional/conditional behavior.
 - >**Generalization** → inheritance between actors or use cases.

- **Advantages:**
 - Simple and easy to understand for both technical and non-technical stakeholders.
 - Helps in requirements gathering, validation, and communication.
 - Acts as a foundation for creating detailed use case specifications.

Question 11: Use Case Specs

A **use case specification** is a detailed textual description of the interactions between a system and its actors to achieve a specific goal. It captures the flow of events, preconditions, postconditions, and exceptions related to a use case, serving as a blueprint for system behavior.

Each use case spec typically includes:

1. **Use Case ID** – Unique identifier
2. **Use Case Name** – Descriptive title
3. **Actors** – Who is involved (primary and secondary)
4. **Preconditions** – What must be true before use case begins
5. **Postconditions** – Expected result at the end
6. **Trigger** – Event that starts the use case
7. **Main Flow (Basic Flow)** – Step-by-step normal interaction
8. **Alternate Flows (Extensions)** – Variations or exceptions
9. **Priority** – High/Medium/Low
10. **Assumptions** – Any underlying assumptions

EXAMPLES:

Use Case 1: **Farmer Browses Products**

- **ID:** UC01
- **Actors:** Farmer
- **Preconditions:** Farmer has logged into the application.

-
- **Postconditions:** Products are displayed to the farmer.
 - **Trigger:** Farmer clicks “Browse Products.”
 - **Main Flow:**
 1. Farmer selects “Product Catalog.”
 2. System displays available categories (Seeds, Fertilizers, Pesticides).
 3. Farmer clicks on a category.
 4. System shows the list of products under that category.
 - **Alternate Flow:** If no products available, system displays “No products found.”
 - **Priority:** High

Use Case 2: Farmer Searches Products

- **ID:** UC02
- **Actors:** Farmer
- **Preconditions:** Farmer has access to search functionality.
- **Postconditions:** Relevant product list is shown.
- **Trigger:** Farmer enters keyword in search bar.
- **Main Flow:**
 1. Farmer enters keyword (e.g., “organic fertilizer”).
 2. System checks product database.
 3. System displays matching results.
- **Alternate Flow:** If no results, system suggests similar products.
- **Priority:** High

Use Case 3: Manufacturer Uploads Products

-
- **ID:** UC03
 - **Actors:** Manufacturer
 - **Preconditions:** Manufacturer account verified.
 - **Postconditions:** Products uploaded successfully.
 - **Trigger:** Manufacturer clicks “Upload Product.”
 - **Main Flow:**
 1. Manufacturer enters product details (name, price, description).
 2. System validates input.
 3. System uploads product to database.
 - **Alternate Flow:** If invalid details, system shows error.
 - **Priority:** High

Use Case 4: Farmer Places an Order

- **ID:** UC04
- **Actors:** Farmer
- **Preconditions:** Farmer is logged in and product is in stock.
- **Postconditions:** Order is successfully placed.
- **Trigger:** Farmer clicks “Buy Now.”
- **Main Flow:**
 1. Farmer selects product and quantity.
 2. Farmer adds to cart.
 3. Farmer proceeds to checkout.
 4. System generates order confirmation.
- **Alternate Flow:** If product is out of stock, system displays “Unavailable.”

-
- **Priority:** High

Use Case 5: Admin Manages Users

- **ID:** UC05
- **Actors:** Admin
- **Preconditions:** Admin has system access.
- **Postconditions:** User accounts updated.
- **Trigger:** Admin selects “User Management.”
- **Main Flow:**
 1. Admin views list of users (farmers, manufacturers).
 2. Admin approves, suspends, or deletes accounts.
- **Alternate Flow:** If unauthorized access, system denies action.
- **Priority:** Medium

Question 12: Activity Diagrams

An **Activity Diagram** is a **behavioral diagram in UML** that represents the **dynamic aspects of a system** by modeling the **workflow of activities and actions**. It describes the **sequence of operations, decision points, and parallel processes** that occur within a system to achieve a specific functionality.

It is often compared to a **flowchart**, but it is more powerful since it supports **concurrency, object flow, and synchronization**.

1. **Purpose:**
 - Shows how the system behaves in terms of workflows.
 - Captures the flow of activities from start to end.
 - Useful in business process modeling and system design.
2. **Elements:**

-
- **Initial Node** → solid black circle (starting point).
 - **Activities/Actions** → rounded rectangles (tasks performed).
 - **Decision Nodes** → diamonds (branching conditions).
 - **Fork/Join Nodes** → parallel processes (concurrent activities).
 - **Final Node** → bullseye symbol (end of workflow).
 - **Transitions/Flows** → arrows showing sequence.

3. Advantages:

- Provides a **clear understanding of business workflows**.
- Helps identify **bottlenecks** and **dependencies**.
- Bridges the gap between **business process models** and **system design**.

Here are **5 diagrams** that can be drawn based on our case study: (each representing one major flow)

1. User Registration & Login

- Start → Enter Email/Password → [Decision: New User?] → Register → Verification → Login → End.

2. Browsing & Searching Products

- Start → Open Product Catalog → Select Category (Fertilizers/Seeds/Pesticides) → Search by Keyword → Display Results → Select Product → End.

3. Adding to Cart & Buy Later List

- Start → Select Product → [Decision: Buy Now or Buy Later?] → Add to Cart OR Add to Wishlist → End.

4. Payment Process

- Start → Checkout → Select Payment Method (COD / UPI / Card) → Process Payment → [Decision: Success/Failure] → Confirm Order OR Retry → End.

5. Delivery Tracking

- Start → User Logs in → Select “Track Order” → Fetch Order Details → Display Order Status (Packed → Shipped → Out for Delivery → Delivered) → End.

