**CAPSTONE PROJECT 1 PART 2**

**Q-1:** 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?

**Answer:**

**Audit and its importance:**

Audits are systematic examinations and evaluations of various aspects of a project, process, or organization to ensure compliance with standards, regulations, and best practices. They provide an objective assessment of whether the operations, processes, and controls are functioning effectively.

**Types of Audits:**

**Internal Audits:**

* Conducted by the organization's own audit team.
* Aim to evaluate internal controls, processes, and compliance with internal policies.
* Help in identifying areas for improvement and ensuring adherence to organizational standards.

**External Audits:**

* Conducted by external independent auditors.
* Focus on compliance with external regulations and standards.
* Often used for financial audits, regulatory compliance, and certification purposes.

**Audits from a Business Analyst's (BA) Perspective:**

As a Business Analyst perspective, audits will focus on the documentation, processes, and methodologies employed in gathering, analysing, and managing requirements. Here's what auditors will typically look for during each quarterly audit:

**Quarter 1 Audit Report: Requirement Gathering Phase (6 weeks)**

Checklist:

1. **BRD Template:**

* Ensure the Business Requirements Document (BRD) template is standardized and includes sections such as project overview, objectives, scope, requirements, assumptions, constraints, and approval.
* Verify the template has been populated with all required information specific to the project.

2. **Elicitation Result Report:**

* Confirm that elicitation sessions (interviews, workshops, surveys) have been conducted with all key stakeholders, including Peter, Kevin, Ben, and other farmers, as well as representatives from fertilizer, seed, and pesticide companies.
* Check that the results of these sessions have been documented thoroughly.

3. **Duplicate Requirement Report:**

* Review the requirement list to identify and eliminate any duplicates.
* Ensure a report of identified duplicates and actions taken to resolve them is maintained.

4. **Grouping of Functionality/Feature Client Sign-Off:**

* Categorize requirements into functional groups such as user authentication, product catalog, order processing, payment gateway, and delivery tracking.
* Obtain client sign-off for each grouped functionality/feature, ensuring agreement and alignment on scope and priorities.

5. **Email Communication:**

* Archive all relevant email communications between the project team, stakeholders, and clients.
* Ensure these communications cover key discussions, decisions, and confirmations regarding requirements.

|  |  |
| --- | --- |
| Stage | **Quarter 1 Audit Report** |
| **(Requirement Gathering Phase)** |
| Completed | 6 weeks |
| Checklist | BRD Template |
| Elicitation Result Report |
| Duplicate Requirement Report |
| Grouping of Functionality/Feature Client Sign-Off |
| Email Communication |

**Quarter 2 Audit Report: Design Phase (10 weeks)**

Checklist:

1. **High-Level Design (HLD) Document:**

* + Verify that the HLD document is complete and includes architecture diagrams, data flow diagrams, and system component details.
  + Ensure the HLD aligns with the BRD.

2. **Low-Level Design (LLD) Document:**

* + Confirm that LLD documents are created for each module detailing the specific technical implementation.
  + Ensure the LLD provides sufficient detail for developers to begin coding.

3. **Design Review Sign-Off:**

* + Conduct design review meetings with key stakeholders and document feedback.
  + Obtain formal sign-off from the client and stakeholders on the finalized design.

4. **Prototyping and Wireframes:**

* + Develop prototypes and wireframes for critical user interfaces.
  + Validate these prototypes with stakeholders to ensure they meet user expectations and requirements.

5. **Email Communication:**

* + Archive all email communications regarding design discussions, feedback, and approvals.

|  |  |
| --- | --- |
| Stage | **Quarter 2 Audit Report (Design Phase)** |
| Completed | 10 weeks |
| Checklist | High-Level Design (HLD) Document |
| Low-Level Design (LLD) Document |
| Utilization tools |
| Design Review Sign-Off |
| Prototyping and Wireframes |
| Stakeholder MOM |
| Email Communication |

**Quarter 3 Audit Report: Development Phase (16 weeks)**

Checklist:

1. **Code Review Reports:**

* + Ensure code review sessions are conducted regularly and documented.
  + Address any identified issues or bugs promptly.

2. **Unit Test Cases and Results:**

* + Verify that unit test cases are written for each module.
  + Ensure that unit test results are documented and reviewed.

3. **Integration Plan:**

* + Confirm that an integration plan is in place detailing how different modules will be integrated.
  + Validate the plan with all relevant team members.

4. **Version Control Documentation:**

* + Ensure version control systems are used and maintained.
  + Document all code commits and changes thoroughly.

5. **Email Communication:**

* + Archive all email communications regarding development progress, issues, and resolutions.

|  |  |
| --- | --- |
| Stage | **Quarter 3 Audit Report** |
| **(Development Phase)** |
| Completed | 16 weeks |
| Checklist | Code Review Reports |
| JAD session Report |
| Unit Test Cases and Results |
| Integration Plan |
| Version Control Documentation |
| BA ad Developer MOM |
| Email Communication |

**Quarter 4 Audit Report: Testing and Deployment Phase (12 weeks)**

Checklist:

1. **Test Plan and Test Cases:**

* + Verify the creation and approval of a comprehensive test plan covering functional, non-functional, performance, and security testing.
  + Ensure test cases are detailed and mapped to requirements.

2. **Test Execution Reports:**

* + Document the execution of test cases and track defects.
  + Ensure all critical and high-priority defects are resolved.

3. **User Acceptance Testing (UAT):**

* + Coordinate UAT sessions with key stakeholders.
  + Document UAT feedback and ensure all issues are addressed before final sign-off.

4. **Deployment Plan:**

* + Develop a detailed deployment plan including rollback procedures.
  + Validate the plan with all relevant stakeholders.

5. **Email Communication:**

* + Archive all email communications regarding testing results, UAT feedback, and deployment planning.

|  |  |
| --- | --- |
| Stage | **Quarter 4 Audit Report** |
| **(Testing and Deployment Phase)** |
| Completed | 12 weeks |
| Checklist | Test Plan and Test Cases |
| Test Execution Reports |
| User Acceptance Testing (UAT) |
| Deployment Plan |
| Email Communication |

**Key Areas Auditors Will Focus On for a BA**

**Requirements Management:**

1. Quality and completeness of requirements documentation.
2. Processes for managing requirement changes.
3. Traceability of requirements through the project lifecycle.

**Stakeholder Engagement:**

1. Effectiveness of stakeholder identification and engagement.
2. Communication records and meeting minutes.

**Documentation and Deliverables:**

1. Accuracy and completeness of design documents, test cases, and other BA deliverables.
2. Adherence to project timelines and milestones.

**Change Control:**

1. Processes for handling changes in requirements and scope.
2. Documentation of changes and their impact on the project.

**Collaboration and Communication:**

1. Interaction with other project team members, especially developers and testers.
2. Responsiveness to queries and issues raised during development and testing phases.

**Compliance and Standards:**

1. Adherence to organizational and project-specific standards and methodologies.
2. Use of templates and tools as per the organization’s guidelines.

**Q-2: BA Approach Strategy**

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

The Business Analyst (BA) approach strategy is a structured method used by Business Analysts to identify business needs, gather requirements, and deliver solutions that align with the goals and objectives of the organization.

**1. Project Kick-off and Initial Planning:**

* **Objective:** Establish a clear understanding of the project scope, goals, and stakeholders.
* **Activities:**
  1. Conduct a project kick-off meeting with the committee, project manager, and key stakeholders.
  2. Define project objectives, scope, deliverables, timelines, and success criteria.

**2. Elicitation Techniques:**

* **Objective:** Gather comprehensive requirements from stakeholders.
* **Techniques:**
  1. **Interviews:** Conduct interviews with Peter, Kevin, Ben, and other key stakeholders to gather detailed requirements.
  2. **Workshops:** Organize workshops with farmers and manufacturers to understand their needs and challenges.
  3. **Surveys/Questionnaires:** Distribute surveys to gather input from a broader audience of farmers.
  4. **Observation:** Visit farms and manufacturing facilities to observe current processes and identify pain points.
  5. **Document Analysis:** Review existing documentation related to farming products, procurement processes, and current tools used.

**3. Stakeholder Analysis and RACI Matrix:**

* **Objective:** Identify and categorize stakeholders and define their roles and responsibilities.
* **Activities:**
  1. Stakeholder Identification: List all stakeholders including Mr. Henry, Mr. Pandu, Mr. Dooku, Peter, Kevin, Ben, the development team, and end-users (farmers and manufacturers).
  2. RACI Matrix: Create a RACI (Responsible, Accountable, Consulted, Informed) matrix to define roles and responsibilities.

|  |  |  |  |
| --- | --- | --- | --- |
| **Task/Activities** | **Responsible** | **Accountable** | **Consulted** |
| Requirement Gathering | BA | PM | Stakeholder |
| Design | BA, Sr Developer | PM | Development Team |
| Development | Developers | PM | BA |
| Testing | Tester | Tester/Lead | BA, Developers |
| Deployment | Network Admin | PM | BA, Developers |
| UAT | Tester/BA | PM | Stakeholders |

**4. Documentation:**

**Objective:** Create and manage comprehensive project documentation.

**Documents:**

**Business Requirements Document (BRD):** Detailed documentation of all business requirements.

**Functional Requirements Document (FRD):** Detailed functional specifications based on business requirements.

**Technical Specifications Document:** Created by the development team with input from the BA.

**Test Plan and Test Cases:** Detailed testing approach and test cases.

**User Manuals and Training Guides:** Documentation to help end-users understand the system.

**5. Sign-off Process:**

**Objective:** Ensure all documents are reviewed, approved, and signed off by relevant stakeholders.

**Process:**

Review sessions with stakeholders for each document.

Incorporate feedback and make necessary revisions.

Obtain formal sign-off on the documents through email confirmations or signed approval forms.

**6. Client Approvals and Communication Channels:**

**Objective:** Maintain clear and consistent communication with the client and stakeholders.

**Channels:**

**Regular Meetings:** Weekly status meetings with the project team and bi-weekly meetings with the client.

**Email Updates:** Regular email updates on project progress.

**Project Management Tools:** Use tools like JIRA or Trello to track progress and issues.

**Reports:** Monthly progress reports to the client and stakeholders.

**7. Handling Change Requests:**

**Objective:** Manage changes to project scope or requirements efficiently.

**Process:**

**Change Request Form:** Use a standardized form for stakeholders to submit change requests.

**Impact Analysis:** Assess the impact of the change on project scope, timeline, and budget.

**Approval:** Obtain approval from the project manager and client for any changes.

**Update Documentation:** Reflect changes in the BRD, FRD, and other relevant documents.

**8. Progress Updates and Stakeholder Communication:**

**Objective:** Keep stakeholders informed about project progress and any issues.

**Activities:**

**Weekly Updates:** Send weekly status updates to stakeholders.

**Monthly Reports:** Provide detailed monthly progress reports including milestones achieved, upcoming tasks, and any risks or issues.

**Stakeholder Meetings:** Conduct regular meetings with key stakeholders to discuss progress and address any concerns.

**9. UAT and Client Project Acceptance:**

**Objective:** Ensure the system meets the client's requirements and obtain final acceptance.

**Activities:**

**UAT Planning:** Create a detailed UAT plan and schedule.

**UAT Execution:** Facilitate UAT sessions, gather feedback, and document any issues.

**Issue Resolution:** Work with the development team to resolve any issues identified during UAT.

**Client Acceptance Form:** Prepare a Client Project Acceptance Form for final sign-off once all issues are resolved and the client is satisfied.

**Q-3: 3-Tier Architecture**

**Explain and illustrate 3-tier architecture?**

3-tier architecture is a client-server architecture pattern that separates the user interface, functional process logic, and data storage layers into three distinct modules, each of which can be developed, maintained, and scaled independently. This architecture promotes modularity, reusability, and scalability of applications.

**1. Presentation Tier (User Interface Layer):**

**Purpose:** This is the topmost level of the application responsible for displaying the user interface and handling user interactions.

**Components:** Web browsers, mobile applications, or desktop applications.

**Functions:**

Manages the presentation logic.

Captures user inputs and sends them to the business logic layer.

Receives and displays the data to the users.

**2. Business Logic Tier (Application Layer):**

**Purpose:** This middle layer contains the core functionality of the application and business logic.

**Components:** Application server, business logic server.

**Functions:**

Processes user inputs received from the presentation layer.

Applies business rules, computations, and validations.

Manages data transfer between the presentation and data layers.

**3. Data Tier (Database Layer):**

**Purpose:** This bottom layer is responsible for data storage and management.

**Components:** Database servers, data storage systems.

**Functions:**

Stores, retrieves, and manages data.

Ensures data integrity and security.

Provides data to the business logic layer upon request.

**Example Scenario: Online Agriculture Products Store**

**Application Layer:**

* + The farmers access the online store through a web browser or a mobile app.
  + They can browse through different products (fertilizers, seeds, pesticides), view product details, and place orders.

**Business Logic Layer:**

* + The application server processes user requests such as searching for products, adding items to the cart, and checking out.
  + It applies business rules, such as verifying product availability and calculating total costs including taxes and discounts.

**Data Layer:**

* + The database server stores information about products, user accounts, orders, and transactions.
  + When a farmer searches for a product, the business logic tier queries the database to retrieve the relevant data and sends it back to the presentation tier for display.

**Illustration of 3-tier Architecture:**

**Application** **Layer**

Web Browser

Mobile App

**Business Logic Layer**

Product/Order/Payment Logic

Business Logic Server

**Data Layer**

Database Server

Data Storage Systems

**Q4: BA Approach Strategy for Framing Questions**

Before framing questions to ask stakeholders, a Business Analyst (BA) should consider several frameworks and strategies to ensure comprehensive, clear, and actionable information is gathered.

1. **5W 1H:**

This framework helps in ensuring that all necessary aspects of a question are covered. It is an Question tool where we write down or frame a question.

**Who:** Identifies the stakeholders involved or affected.

Example: Who will be using the online agriculture product store?

**What:** Defines what is being discussed or required.

Example: What functionalities are most important for the farmers?

**When:** Specifies the timeline or deadlines.

Example: When do you need the application to be ready?

**Where:** Determines the location or context.

Example: Where will the application be used (e.g., on farms, in offices)?

**Why:** Understands the reasons or goals behind requirements.

Example: Why do you need a specific feature (e.g., real-time inventory updates)?

**How:** Explores the methods or processes.

Example: How do you currently procure fertilizers, and how do you envision this changing with the online store?

**2. SMART:**

Ensures that the requirements and questions are Specific, Measurable, Achievable, Relevant, and Time-bound. It is a Validation tool.

**Specific:** Be clear and precise.

Example: What specific types of fertilizers do you use regularly?

**Measurable:** Ensure that the requirement can be quantified or assessed.

Example: How many transactions do you expect to process monthly?

**Achievable:** Check that the requirement is realistic.

Example: Can the farmers realistically use a mobile app given their current technology access?

**Relevant:** Make sure the requirement is pertinent to the project goals.

Example: Are bulk order discounts relevant to your purchasing process?

**Time-bound:** Define the timeframe for implementation or achievement.

Example: When do you need the bulk order feature implemented by?

**3. RACI:**

Identifies the roles and responsibilities in the project.

**Responsible:** Who will carry out the task?

Example: Who is responsible for maintaining the product catalogue?

**Accountable:** Who is ultimately accountable for the task?

Example: Who will approve the final design of the user interface?

**Consulted:** Who needs to be consulted before a decision or action?

Example: Who should we consult to validate the pesticide procurement process?

**Informed:** Who needs to be informed about decisions or actions?

Example: Who needs to be informed about new features added to the application?

**4. 3-Tier Architecture:**

Consider the structure and flow of data across the presentation, business logic, and data tiers.

**Application Layer:** How will the user interface be designed and what interactions will be required?

Example: What pages and features do farmers need in the mobile app?

**Business Logic Layer:** What business rules and processes will the application enforce?

Example: What are the rules for bulk purchasing and discount eligibility?

**Data Layer:** What data needs to be stored and retrieved, and how will this data be managed?

Example: What data fields are necessary for tracking pesticide inventory?

**5. Use Cases and Use Case Specifications:**

Define the functional requirements and interactions of the system.

**Use Cases:** Describe specific scenarios in which the system interacts with users or other systems.

Example: Use Case: "Purchase Fertilizer" – A farmer selects a fertilizer, adds it to the cart, and completes the purchase.

**Use Case Specifications:** Detail the basic flow (positive flow) and alternative flows (negative flow).

**Basic Flow (Positive):**

* + Farmer logs into the application.
  + Searches for fertilizer.
  + Adds fertilizer to cart.
  + Proceeds to checkout.
  + Confirms purchase.

**Alternative Flow (Negative):**

* + Farmer logs into the application.
  + Searches for fertilizer.
  + Adds fertilizer to cart.
  + Checkout fails due to payment error.
  + Farmer receives an error message and retries the payment or contacts support.

**Q5:** **Elicitation Techniques**

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

Elicitation techniques are crucial for gathering requirements from stakeholders and ensuring that the final solution aligns with their needs and expectations. Here is a comprehensive overview of various elicitation techniques, including those encapsulated by the acronym BDRFOWJIPQU:

There are 11 elicitation techniques to apply accordingly: -

**I. Brainstorming:**

* + This technique is used to generate new ideas and find a solution for a specific issue.
  + The members included for brainstorming can be domain experts, subject matter experts.
  + Multiple ideas and information give us a repository of knowledge and we can choose from different ideas.

**II. Document Analysis:**

* + During this step of the requirements elicitation process, business analysts review existing documentation at hand, with the intent of identifying requirements for changes or improvements.
  + Examples of document analysis sources include pre-existing project plans, system specifications, process documentation, market research dossiers, customer feedback, meeting minutes, and user manuals.
  + Document analysis is performed before scheduling more in-depth requirements elicitation sessions or interviews with stakeholders.

**III. Reverse engineering:**

* + In this Technique, any outdated documentation in an existing system, can be reversed to understand what the system does.
  + His is an elicitation technique that can extract implemented requirements from the system.
  + There are two types of reverse engineering techniques.
* Black box reverse engineering: The system is studied without examining its internal structure (function and composition of software).
* White box reverse engineering: The inner workings of the system are studied (analysing and understanding of software code).

**IV. Focus Group:**

* + By using a focus group, you can get information about a product, service from a group.
  + The Focus group includes subject matter experts. The objective of this group is to discuss the top issue and provide information. A moderator manages this session.

**V. Observation:**

* + An excellent elicitation technique helps understand requirements based on observations related to process flows and work environments of stakeholders.
  + Observation requires a business analyst to go and look at the work – for example, observing the business processes in scope of the project.
  + The elicitation technique observation is an effective means of understanding how a user does their job by assessing their work environment.

**VI. Workshops:**

* + Workshops comprise a group of users or stakeholders working together to identify requirements.
  + A requirement workshop is a structured way to capture requirements.
  + Workshops are used to scope, discover, define, and prioritize requirements for the proposed system.

**VII. JAD (Joint Application Development):**

* + This technique is more process-oriented and formal as compared to other techniques.
  + These are structured meetings involving end-users, PMs, SMEs.
  + This is used to define, clarify, and complete requirements.

**VIII. Interviews:**

* + An interview is a systematic approach to elicit information from a person or group of people.
  + This is the most common technique used for requirement elicitation.
  + Interview techniques should be used for building strong relationships between business analysts and stakeholders.
  + In this technique, the interviewer directs the question to stakeholders to obtain information. One to one interview is the most commonly used technique.

**IX. Prototyping:**

* + Prototyping is used to identify missing or unspecified requirements.
  + In this technique, frequent demos are given to the client by creating the prototypes so that client can get an idea of how the product will look like.
  + Prototypes can be used to create a mock-up of sites, and describe the process using diagrams.

**X. Questionnaire & Surveys:**

* + For Survey/Questionnaire, a set of questions is given to stakeholders to quantify their thoughts.
  + After collecting the responses from stakeholders, data is analysed to identify the area of interest of stakeholders. Questions should be based on high priority risks.

**XI. Use case specs:**

* + Use cases are an effective and widely used technique for eliciting software requirements.
  + The use-case approach focuses on the goals that users have with a system, rather than emphasizing system functionality.
  + This technique combines text and pictures to provide a better understanding of the requirements.
  + The use cases describe the ‘what’, of a system and not ‘how’. Hence, they only give a functional view of the system.
  + The components of the use case design include three major things – Actor, Use cases, use case diagram.

**Q6: This project Elicitation Techniques:**

The following elicitation techniques would be most effective for the Online Agricultural Product Store project:

**1. Prototyping**

**Purpose:** To create a working model or mock-up of the application to demonstrate key features such as product catalogue, search functionality, and payment gateway.

**Justification**:

Helps stakeholders (Peter, Kevin, Ben) visualize the system interface.

Enables feedback on UI/UX and functional aspects early in the development.

Useful for refining requirements related to usability and user flow.

**2. Use Case Specifications**

**Purpose**: To capture detailed interactions between users (Farmers and Manufacturers) and the system.

**Justification:**

Helps in documenting scenarios like login, product search, purchase, and delivery tracking.

Ensures all functional requirements, including alternative and exception flows (e.g., payment failures), are covered.

Provides clarity to the development team for implementation.

**3. Document Analysis**

**Purpose**: To analyse existing documents like user manuals, policies, or previous system specifications (if any).

**Justification**:

Useful to gather information on standard product catalogue structures or delivery tracking systems.

Helps in understanding industry-specific details (e.g., standard data for fertilizers, seeds, and pesticides).

**4. Brainstorming**

**Purpose**: To generate ideas and gather diverse inputs from stakeholders (Peter, Kevin, Ben).

**Justification**:

Encourages stakeholders to suggest innovative features (e.g., advanced filters for product search).

Facilitates discussion about potential challenges or improvements for the user experience.

**Best Technique for Each Stakeholder Group**

| **Stakeholder Group** | **Best Elicitation Technique** | **Reason** |
| --- | --- | --- |
| Farmers (Peter, Kevin) | **Prototyping, Brainstorming** | Easy visualization and ideation for usability improvements. |
| Manufacturers (Ben, etc.) | **Document Analysis, Prototyping** | Ensures clarity in product uploads and compliance with guidelines. |
| Mr. Henry (Client) | **Use Case Specs, Prototyping** | Provides structured scenarios and visual feedback for approval. |

**Recommendation:**

* Start with Prototyping: Create mock-ups of critical features and gather feedback from farmers and manufacturers.
* Develop Use Case Specs: Write detailed use cases for login, product search, payment, and order tracking.
* Conduct a Brainstorming Workshop: Explore additional features and workflows with stakeholders.
* Perform Document Analysis: Review existing product documentation and industry standards.

**Q7: 10 Business** **Requirements**:

1. BR001- The platform should have a product catalogue that includes all fertilizers, seeds and pesticides from different manufacturer and vendors.
2. BR002- The platform should allow farmers to search for products by name, category and brand.
3. BR003- The platform should have a login feature for all users including farmers, manufacturers and vendors.
4. BR004- the platform should allow new users to create an account by submitting their E- mail id and creating a secure PW.
5. BR005- The platform should have a user-friendly interfaces and easy navigation for a better user experience.
6. BR006- The platform should have a payment gateway that includes COD, credit/debit cards and UPI options.
7. BR007- The platform should send E-mail confirmation regarding order status to users.
8. BR008- The platform should have a delivery tracker to track the whereabouts of the order.
9. BR009- The platform should have scalable to accommodate future growth and expansion.
10. BR010- The platform should have a secure infrastructure to protect user data and prevent data breaches.

**Q8: Assumptions:**

1. The project for an e-commerce platform for fertilizers, seeds and pesticides targeted towards farmers.
2. The platform will have a product catalogue and will allow users to search fertilizers, seeds and pesticides.
3. This platform will have login feature for farmers, Manufacturers and vendors and allow new users to create account by submitting their E-mail id and creating a secure password.
4. The platform will have a payment gateway that includes COD, credit/debit card and UPI options.
5. The platform will send E-mail confirmations regarding order status and delivery tracker to track product.
6. The platform will have user-friendly interfaces and easy navigation for a better user experience.

**Q9: This project Requirements Priority:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement** | **Description** | **Priority** | **Justification** |
| **BR001** | Product catalogue including fertilizers, seeds, and pesticides from various manufacturers/vendors. | **10** | Core functionality; without this, the platform cannot fulfil its purpose. |
| **BR002** | Search feature by name, category, and brand. | **9** | Enhances usability and ensures users can find products quickly, crucial for user experience. |
| **BR003** | Login feature for all users, including farmers, manufacturers, and vendors. | **9** | Ensures user-specific access and management, foundational for platform security and operations. |
| **BR004** | Account creation for new users by submitting email ID and creating a secure password. | **8** | Essential for on boarding new users but can follow after initial core functionalities are built. |
| **BR005** | User-friendly interface and easy navigation. | **10** | Critical for adoption, especially given farmers may have limited technical skills. |
| **BR006** | Payment gateway with COD, credit/debit cards, and UPI options. | **8** | Important for completing transactions; COD can serve as a fall back if online payment is delayed. |
| **BR007** | Email confirmation regarding order status. | **7** | Nice-to-have; adds convenience but can be implemented after core order functionalities. |
| **BR008** | Delivery tracker to track the whereabouts of the order. | **6** | Enhances user experience but is less critical during the initial implementation phase. |
| **BR009** | Scalable platform to accommodate future growth. | **9** | Vital for long-term sustainability and should be considered during the architectural design phase. |
| **BR010** | Secure infrastructure to protect user data and prevent data breaches. | **10** | Critical to build trust and comply with data protection laws. |

**Q10: Use Case Diagram**

Use Case diagram provides a high-level overview of the interactions between users and the system, capturing the essential functionalities required for the online agriculture products store.

**Use Case Diagram for Online Agriculture Products Store**

**Actors:**

1. Farmer

2. Manufacturer

3. Admin

4. System

**Use Cases:**

1. Sign-up/Login

2. Search Products

3. View product Catalogue

4. Add to cart

5. Place order

6. Process payment

7. Manage Product

8. Tracking

9. Product Delivery Notification

**Diagram:**



**Q11: Prepare use case specs for all use cases:**

|  |  |
| --- | --- |
| Use Case ID | UC001 |
| Use Case Name | Buying a product |
| Actors | Customer, Seller |
| Description | This use case describes how users can make purchase via App |
| Pre - Condition | User should have been registered into the application |
| Post - Condition | Successfully able to login the Account |
| Basic Flow | Step 1: User create and account and login.  Step 2: User search for a product from the search bar.  Step 3: same product and related product option from different manufacture will be appeared on the screen.  Step 4: User select one product, selects the size and quantity of the product and click on "buy now option".  Step 5: System will take to another page, where total price calculation will be displayed along with the products added to cart.  Step 6: User click on "Place order button".  Step 7: User need to to choose the mode of the payment.  Step 8: User need to enter the banking details and make payment.  Step 9: User will receive order confirmation on email along with the tracking id. Step 10: Basic flow end here. |
| Alternate Flows | Step 1: User is not able to login and redirected to forgot "Username/Password" page.  Step 2: If you user is not able to get the right information, he can request for a call from customer care.  Step 3: once he get connected with the customer care he will explain the issue to the customer care representative,  Step 4: Customer care will send a link to reset password to his email account.  Step 5: User will go to that link and system will take to new page, where user will be able to change new password  Step 6: User will be put a new password.  Step 7: System will ask to reconfirm the password.  Step 8: User will be able to login the account now. |
| Exceptions | If internet connectivity lost while doing this use case, system displays " check with your internet connectivity " |
| Frequency of use | High |
| Assumptions | It is assumed that the customer is registered.  It is assumed that the customer has the computer knowledge.  It is assumed that the customer has a suitable device to use the APP. |

|  |  |
| --- | --- |
| Use Case ID | UC002 |
| Use Case Name | Exchange of product |
| Actors | Customer, Seller |
| Description | This use case describes how users can exchange a purchased product. |
| Pre - Condition | User should have purchased a product before in order to make a exchange. |
| Post - Condition | Successfully able to exchange the product |
| Basic Flow | Step 1: User login to account via credentials.  Step 2: User click on "Account" .  Step 3: System takes to different page with other details.  Step 4: User select option "Exchange" among those options.  Step 5: System will take to another page, where recently ordered products will be displayed on the screen.  Step 6: User has to choose the product which he wants to exchange.  Step 7: User will get another option where he will be asked- "different size in same product" or "want to buy another product"  Step 8: User need to choose one of the options and take action according to chosen option. Step 9: Once the product is chosen, user will have to click on button "Exchange".  Step 10: User will get the confirmation on email. |
| Alternate Flows | Step 1: User couldn't find the size which he wanted.  Step 2: User call customer care agent to get a solution  Step 3: Agent suggested to wait for the size to be restocked and gave a tentative date or go for similar products.  Step 4: Agent shares the link of similar products to the registered email of the customer.  Step 5: User choose the product  Step 6: User will be put a new password.  Step 7: System will ask to reconfirm the password. Step 8: User will be able to login the account now. |
| Exceptions | If internet connectivity lost while doing this use case, system displays " check with your internet connectivity " |
| Frequency of use | High |
| Assumptions | It is assumed that the customer is registered.  It is assumed that the customer has the computer knowledge.  It is assumed that the customer has a suitable device to use the APP. |

|  |  |
| --- | --- |
| Use Case ID | UC003 |
| Use Case Name | Return of product |
| Actors | Customer, Seller |
| Description | This use case describes how users can return a purchased product. |
| Pre - Condition | User should have purchased a product before in order to make a return. |
| Post - Condition | Successfully able to exchange the product |
| Basic Flow | Step 1: User login to account via credentials.  Step 2: User click on "Account"  Step 3: System takes to different page with other details.  Step 4: User select option "Return" among those options.  Step 5: System will take to another page, where recently ordered products will be displayed on the screen.  Step 6: User has to choose the product which he wants to return. Step 7: User will get another option where he will be asked to provide the bank account number for amount of the returned product to be credited.  Step 8: User need to enter the account number and submit.  Step 9: User will get the confirmation on email. |
| Alternate Flows | Step 1: User didn't get the amount in his account within the TAT.  Step 2: User call customer care agent to ask payment status.  Step 3: Payment was stuck due to a technical glitch.  Step 4: User was shared complaint form to be filled.  Step 5: Once form submitted, user received another TAT on the email of amount to be credited.  Step 6: User get the payment id in registered email |
| Exceptions | User put the incorrect bank account. |
| Frequency of use | High |
| Assumptions | It is assumed that the customer has a valid bank account number.  It is assumed that the customer has good internet connectivity.  It is assumed that the customer has computer knowledge. |

|  |  |
| --- | --- |
| Use Case ID | UC004 |
| Use Case Name | Update the delivery address |
| Actors | Customer, Seller |
| Description | This use case describes how users can update address. |
| Pre - Condition | User should have a valid deliverable postal address. |
| Post - Condition | Successfully able to update address. |
| Basic Flow | Step 1: User login to account via credentials.  Step 2: User click on "Account".  Step 3: System takes to different page with other details.  Step 4: User select option "Update" among those options.  Step 5: System will take to another page, where mandatory fields like; Apt number, landmark, pin code, city name will be displayed and has to be field.  Step 6: User need to click on "submit" button.  Step 7: User can use the updated address for products delivery. |
| Alternate Flows | Step 1: User is not able to update the address.  Step 2: User will refresh the page.  Step 3: User gets error again while submitting details.  Step 4: User use live chat box  Step 5: User is asked to not leave blank any star marked field.  Step 6: after updating all mandatory field, address was successfully submitted. |
| Exceptions | User put the incorrect address details like; pin exceeds the maximum number of digits |
| Frequency of use | High |
| Assumptions | It is assumed that the customer has a valid postal address It is assumed that the customer has good internet connectivity.  It is assumed that the customer has computer knowledge.  It is assumed, customer understands, what details has to be put in every field. |

|  |  |
| --- | --- |
| Use Case ID | UC005 |
| Use Case Name | Update the new contact number |
| Actors | Customer, Seller |
| Description | This use case describes how users can update/ change new phone number |
| Pre - Condition | User should have a new contact number. |
| Post - Condition | Successfully able to change contact number. |
| Basic Flow | Step 1: User login to account via credentials. Step 2: User click on "Account" . Step 3: System takes to different page with other details. Step 4: User select option "Manage your Account" among those options. Step 5: System will take to another page, where personal details will be displayed. 6: User has to click Mobile number Step 7: User will get a red popup button "CHANGE". Step 8: OTP will be sent to existing updated number Step 9: once number is verified with the OTP user put. User can update new contact number. Step 10: New contact number is successfully updated. |
| Alternate Flows | Step 1: User didn't get the OTP in registered existing number. Step 2: User restarts the phone. Step 3: User raised a ticket with the customer care Step 4: User was shared issue ticket number in the registered email. Step 5: Issue got fixed with the help of support team Step 6: contact number is successfully changed. |
| Exceptions | User put the incorrect phone number. |
| Frequency of use | Low |
| Assumptions | It is assumed that the customer has a valid phone number. It is assumed that the customer has good phone network to receive OTP. It is assumed that the customer has checked the message inbox for OTP. |

1. Activity Diagrams:

* Activity diagram are one of the 5 diagram in the UML for modelling the dynamic aspects of the system. An activity is essentially a flowchart showing the flow of control from activity to activity.
* In this online case study, we use certain elements for the use case model.
* Control flow an arrow showing the direction of the workflow.
* Activity indicates a step in the process it is a unit of work done by the system or a consistent achieved.
* Connector is a trigger attached to control flow when the guard condition is true.







**END.**