*Capstone project Part -2*

1. *Four quarterly Audit reports*

An **audit** is a careful review or examination of something, like a process, system, or financial record, to make sure it is accurate, follows rules, and works as expected. The goal is to identify any issues or areas for improvement.

|  |  |
| --- | --- |
| stage | Quarter 1- Audit report -requirement gathering phase  |
| completed | 1-10 week  |
| check list  | Stakeholder Involvement: |
|   | Requirements Documentation |
|   | Elicitation Techniques Used: |
|   | Challenges Identified |
|  |  |
|  |  |
| stage | Quarter 2- Audit report -requirement analysis |
| completed | week 16 -23 |
| check list  | Requirements Analysis Quality: |
|   | Stakeholder Validation |
|   | System Design Considerations |
|   | Risk Analysis |
|  |  |
|  |  |
| stage | Quarter 1- Audit report-devolopment  |
| completed | week 40-68 |
| check list  | Feature Implementation |
|   | Code Quality and Compliance |
|   | Integration with External Systems |
|   | Development Progress |
|  |  |
|  |  |
| stage | Quarter 1- Audit report-Testing  |
| completed | week 58-78 |
| check list  | Test Coverage |
|   | Bug Detection and Resolution |
|   | User Acceptance Testing |
|   | Performance Testing |

1. *BA approach strategy*

The **BA (Business Analyst) Approach Strategy** is a structured plan or methodology that outlines how a Business Analyst will approach and execute their responsibilities in a project. It defines the processes, tools, techniques, and activities that will be used to gather, analyze, and manage business requirements, ensuring that they align with the needs and objectives of the stakeholders.

### **1. Which Elicitation Technique to Apply**

To gather requirements effectively, I will apply the following elicitation techniques based on the project’s needs:

* **Interviews**: Conduct one-on-one interviews with key stakeholders such as the farmers, manufacturers, and administrators to understand their challenges, needs, and expectations from the system.
* **Workshops**: Host workshops with cross-functional teams (including the stakeholders) to collaboratively define business requirements, priorities, and potential solutions.
* **Surveys and Questionnaires**: Distribute surveys to farmers to gather data on their preferences, product needs, and issues they face in their daily operations.
* **Prototyping**: Develop prototypes for key features like the product catalog or order management system. This will allow stakeholders to visualize and provide feedback before the final system is developed.
* **Document Analysis**: Review existing documentation, such as business plans or agricultural product catalogs, to understand the current state and identify gaps or areas for improvement.
* **Observation**: Observe farmers' current workflows and challenges in purchasing agricultural products to ensure the system is designed to meet their real-world needs.

### **2. How to Do Stakeholder Analysis (RACI)**

Stakeholder analysis helps identify key stakeholders and their responsibilities. The **RACI matrix** is an effective tool for this:

* **RACI Definition**:
	+ **R**: Responsible (the person who performs the task)
	+ **A**: Accountable (the person ultimately accountable for the task)
	+ **C**: Consulted (people who need to be consulted before a decision is made)
	+ **I**: Informed (people who need to be kept informed of progress)

### **3. What Documents to Write**

The following documents will be prepared and managed throughout the project:

1. **Business Requirements Document (BRD)**: Defines the business goals, objectives, and requirements of the project.
2. **Use Case Specifications**: Detailed specifications of the use cases for the system.
3. **Functional Specifications Document (FSD)**: Describes the system’s functionality and the features that will be implemented.
4. **Non-Functional Requirements Document (NFRD)**: Describes performance, security, scalability, and other non-functional aspects of the system.
5. **System Design Document (SDD)**: Includes details on the architecture, database design, and overall system design.
6. **Test Cases and Test Plans**: Developed based on the requirements to ensure that the system is tested thoroughly.
7. **User Acceptance Testing (UAT) Plan**: A document outlining the approach and criteria for UAT.
8. **Training Manuals/Documentation**: To help users and administrators understand how to use the system.
9. **Project Status Reports**: Regular updates to keep stakeholders informed on progress.
10. **Post-Implementation Report**: Summarizes the project, outlining what went well and lessons learned.

### **4. What Process to Follow to Sign Off on Documents**

* **Document Drafting**: Once the document (e.g., BRD, FSD) is drafted, it will be shared with relevant stakeholders for review.
* **Feedback and Revisions**: Gather feedback from stakeholders and make necessary revisions to the document.
* **Formal Review**: Conduct a formal review meeting with stakeholders (e.g., Mr. Henry, Mr. Pandu, Mr. Dooku, farmers) to discuss the document and its alignment with business objectives.
* **Approval and Sign-Off**: After final revisions, the document will be presented for sign-off by the appropriate stakeholders (e.g., Sponsor, Project Coordinator).
* **Archiving**: Once signed off, the document will be archived for reference and future audit.

### **5. How to Take Approvals from the Client**

1. **Draft Deliverables**: Present initial drafts of key documents (e.g., BRD, Use Case Specs, Functional Specifications).
2. **Review Sessions**: Organize regular review meetings to ensure that the client’s expectations are met and any changes are discussed early.
3. **Feedback Collection**: Actively collect and address any feedback from the client.
4. **Final Review**: Once all changes are made, present the final document for approval.
5. **Formal Approval**: Obtain client signatures or email confirmation as formal approval of the documents.
6. **Track Approvals**: Maintain a record of all approvals, including dates and the approvers.

### **6. What Communication Channels to Establish and Implement**

To ensure effective communication, I will establish the following channels:

* **Email**: For formal communication, document sharing, and approvals.
* **Instant Messaging (Slack/Teams)**: For quick discussions, status updates, and queries.
* **Weekly Meetings (Online/Offline)**: To update stakeholders on project progress, gather feedback, and resolve any issues.
* **Project Management Tools (Jira/Asana/Trello)**: For tracking project tasks, milestones, and issues.
* **Documentation Repository (Confluence/SharePoint)**: To store and share documents, requirements, designs, and reports.
* **User Feedback Forms**: To gather input from end-users like farmers and manufacturers.

### **7. How to Handle Change Requests**

* **Change Request Submission**: Any stakeholder requesting a change will formally submit a change request document, specifying the details of the change and the reason.
* **Impact Analysis**: The BA and development team will analyze the impact of the change on scope, timeline, and resources.
* **Stakeholder Discussion**: Present the impact analysis to the stakeholders to decide whether the change should be approved or rejected.
* **Approval and Update**: If approved, update the relevant project documents and the project plan. If not approved, inform the stakeholder of the decision.
* **Implementation**: Once approved, the change will be incorporated into the system or processes.

### **8. How to Update the Progress of the Project to Stakeholders**

* **Weekly Status Meetings**: Provide detailed updates during weekly project meetings.
* **Project Management Tool Updates**: Regular updates on Jira or Trello to reflect the current status of tasks and milestones.
* **Monthly Progress Reports**: Summarize the project's achievements, risks, and next steps in monthly reports.
* **Email Updates**: Send weekly or bi-weekly email summaries of progress to all stakeholders, including achievements, risks, and upcoming milestones.
* **Risk and Issues Log**: Track and share the log of risks and issues with stakeholders to ensure transparency.

### **9. How to Take Signoff on the UAT - Client Project Acceptance Form**

1. **Prepare UAT Plan**: Define the UAT criteria, the scope of testing, and the responsibilities of the users.
2. **Conduct UAT Testing**: Perform the testing phase with actual end-users (farmers) to validate that the system meets the business requirements.
3. **Obtain Client Feedback**: Gather feedback from the client regarding the UAT results and address any discrepancies or concerns.
4. **Prepare UAT Sign-Off Form**: Create a formal UAT Client Project Acceptance Form, which includes details of tested features, issues resolved, and the final approval status.
5. **Sign-Off Meeting**: Conduct a meeting with the client to review UAT results and obtain their sign-off.
6. **Formal Sign-Off**: Have the client formally sign the UAT acceptance form, which indicates their approval for the project completion.

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

 **3-Tier Architecture** is a software design pattern that divides an application into three separate layers

* Application layer – application layer is also called presentation layer. It handles the display interface components like screen and pages . it has responsibilities like collect inputs , display the information . example – e commerce website
* Business logic layer – business logic layer is intermediary between the presentation layer and data storage layer. This layers holds core logic of application. It processes the requests from the presentation layer. And display the information to the presentation layer

Example- in online order. Process payments order details.

* Data base layer – this is the last and bottom layer of architecture and its responsible for holding the data and storing it. Example- the user data

*4.BA approach strategy for framing the question.*

* 5W1H framework- who, what, when, where, why and how

Example- "Who are the main users of the system?", "What are the most critical features you need?", "How do you currently handle product orders?"

* SMART – specific, measurable, attainable, relevant, and time bound
* RACI charts- RACI charts help define and clarify roles and responsibilities within a team by outlining who is responsible, accountable, consulted and informed for each task.
* UML- unified modeling language is a standardized way of diagramming and modeling software systems to aid in design, development and communication between team members.

*5. Write about the various elicitation techniques*

**Elicitation techniques** are methods used by Business Analysts to gather information, requirements, and insights from stakeholders, users, and other sources. These techniques help in understanding the needs and expectations of the project to ensure the solution meets the desired outcomes.

B - Brainstorming

Definition: A group activity where stakeholders come together to generate ideas, solutions, or requirements for a specific problem or need. It is a creative and open-ended approach to solving problems without immediate judgment.

Purpose: To generate a broad range of ideas and possible solutions in a short amount of time.

D - Document Analysis

Definition: Reviewing existing documents, records, reports, and any relevant written materials to gather information about the project, processes, or business requirements.

Purpose: To extract valuable information from existing resources, such as user manuals, system documentation, and business plans, that might clarify requirements or inform the project.

R - Reverse Engineering

Definition: Analyzing existing systems or products to understand their structure, behavior, and requirements. This is often done when documentation is lacking, or when working with legacy systems.

Purpose: To extract underlying business rules, data models, or system structures that can inform the new project or system design.

F - Focus Groups

Definition: A group discussion involving a select group of users or stakeholders to explore their opinions, perceptions, and requirements for a product or system.

Purpose: To collect feedback from targeted groups to understand their needs, expectations, and reactions to proposed solutions.

O - Observation

Definition: Watching how users interact with systems, perform tasks, or use products in their natural environment to identify problems or improvement opportunities.

Purpose: To gather real-time insights into how work is done and to identify gaps between current processes and what users need or expect.

W - Workshops

Definition: Collaborative sessions involving multiple stakeholders, often led by a facilitator, where participants work together to define, refine, and prioritize requirements.

Purpose: To gather in-depth input from various stakeholders, ensure alignment, and quickly resolve misunderstandings or disagreements.

J - Joint Application Development (JAD)

Definition: A structured workshop involving stakeholders, developers, and users to collaboratively define system requirements and design solutions in a focused and time-bound setting.

Purpose: To achieve consensus on requirements, design, and solution features in a collaborative and efficient manner.

I - Interviews

Definition: One-on-one discussions between the Business Analyst and stakeholders to gather detailed information, clarify needs, and uncover hidden requirements.

Purpose: To obtain detailed, specific information from stakeholders, especially when dealing with complex or sensitive topics.

P - Prototyping

Definition: Creating a preliminary model or mock-up of the system or product to visualize and test key features before full development.

Purpose: To gather feedback early in the process, refine requirements, and make adjustments before finalizing the system design.

Q - Questionnaires/Surveys

Definition: A method of collecting information from a large group of stakeholders or users by using structured questions (either open-ended or closed-ended) to gather data on their needs, preferences, or experiences.

Purpose: To collect quantitative or qualitative data from a wide audience quickly and efficiently, especially when the group is large or geographically dispersed.

U - Use Cases

Definition: Scenarios that describe how a user interacts with the system to achieve a specific goal. Use cases define the system’s expected behavior in response to user actions.

Purpose: To specify detailed interactions between users (actors) and the system, helping to clarify functional requirements and the user experience.

6*. Which elicitation technique is suitable for this project?*

For the **Online Agriculture Product Store** project, the most suitable elicitation technique would be **Prototyping**.

### **Justification**:

**Prototyping** is ideal for this project for the following reasons:

1. **Visualizing User Interface**: Since the project involves an online store that farmers will use, creating prototypes (such as wireframes or mockups) will allow stakeholders (farmers, manufacturers, and administrators) to visualize how the application will look and function. This is especially important because farmers may not be familiar with complex systems, and seeing a prototype helps them provide valuable feedback on the user interface and experience.
2. **Early Feedback**: With prototyping, you can gather feedback early in the project lifecycle. As the application is developed, farmers and manufacturers can interact with prototypes and suggest changes or improvements before full development begins. This helps in refining the product and ensures that it meets the stakeholders' expectations.
3. **User-Centric Design**: The farmers, who are the primary users, may have diverse technological expertise. Prototyping allows them to engage directly with a tangible version of the app, making it easier to understand the flow and functionality. This reduces the risk of miscommunication and ensures the system will be user-friendly.
4. **Iterative Improvement**: Prototyping promotes an iterative approach, where the design and features can evolve based on ongoing feedback. This is particularly important in a project where the user needs to be at the center of the design, and adjustments are likely to be needed as the application is refined.
5. **Engagement with Stakeholders**: It fosters collaboration and ongoing engagement with stakeholders, as they can see and interact with an evolving version of the product, ensuring alignment with business goals and user needs.

Another suitable technique can be used in this project -

Use Case Specifications are also ideal for this project for the following reasons:

1. Clear User-System Interaction: Use cases help in defining how different types of users (farmers, manufacturers, and admins) interact with the system. For example, a use case might describe how a farmer browses products, places an order, or makes a payment. This ensures that all functional requirements are clearly defined from the user's perspective.
2. Defining Requirements: This technique allows for detailed documentation of system behaviors. By creating use case specifications, we can outline specific actions, system responses, and outcomes, which are crucial for both developers and testers. It helps ensure that the application functions as expected for all user roles.
3. Understanding Business Goals: By defining use cases, we can directly link business processes to the system’s functionality. For example, if one of the business goals is to allow manufacturers to easily upload product information, the use case would specify how they interact with the system to achieve this goal. This keeps the development aligned with business objectives.
4. Handling Multiple Stakeholders: Since the project involves multiple stakeholders with different roles, use case specifications help document requirements from different perspectives. Whether it’s a farmer browsing products or an admin managing orders, use cases make it easier to define and manage diverse needs in the project.
5. Testing & Validation: Use cases provide a clear set of actions and expected outcomes, which are beneficial during testing. Testers can use the scenarios described in use case specifications to validate that the system behaves as intended.

*7and 8. Assumptions and business requirements for this project.*

Assumptions for the Online Agriculture Product Store Project

Assumption 1: User Accessibility
The application will be accessible via both web and mobile platforms to cater to farmers in remote areas with varying internet access.

Assumption 2: Internet Connectivity
It is assumed that farmers have intermittent internet access, so the app will be designed to be lightweight and work in low-bandwidth environments.

Assumption 3: Product Availability Management
Manufacturers (fertilizers, seeds, and pesticides) will have regular updates about product availability and can easily manage their stock through the application.

Assumption 4: Payment Methods
The application will support multiple payment methods, including online payment gateways and Cash on Delivery (COD), given the varied financial accessibility in rural areas.

Assumption 5: Language Support
The application will support multiple regional languages to ensure inclusivity for farmers with limited proficiency in English.

Assumption 6: User Training
Farmers may require basic training on how to use the application, and an easy-to-understand user guide or tutorial will be provided.

Assumption 7: Delivery Integration
The application will integrate with delivery services capable of delivering agricultural products to remote locations.

Assumption 8: Data Security
The application will adhere to best practices for data security to ensure the protection of sensitive user and transaction data.

Assumption 9: Mobile Device Compatibility
Farmers will mostly access the platform through smartphones, so the mobile version will be optimized for low-end devices.

Assumption 10: Regulatory Compliance
The application will comply with all local regulations related to e-commerce and agricultural product distribution.

Business Requirements for the Online Agriculture Product Store Project

Business Requirement 1: Product Catalog Management
The system should allow manufacturers to upload and manage their product catalogs (fertilizers, seeds, pesticides), including descriptions, prices, stock levels, and images.

Business Requirement 2: User Registration and Authentication
The application should provide a secure registration and authentication process for farmers, manufacturers, and administrators. This will include features like email/phone verification and password recovery.

Business Requirement 3: Search and Filter Functionality
The system should allow farmers to search for products by category (e.g., fertilizers, seeds, pesticides) and apply filters like price, rating, or brand.

Business Requirement 4: Product Ordering and Cart Management
Farmers should be able to add products to a shopping cart, modify quantities, and review their orders before placing them.

Business Requirement 5: Order Tracking and Delivery Status
The system should provide real-time tracking of orders, showing the current status (e.g., "Processing," "Shipped," "Delivered") and an estimated delivery time.

Business Requirement 6: Multiple Payment Options
The system should support multiple payment methods, including credit/debit cards, mobile wallets, and Cash on Delivery (COD), to accommodate various payment preferences.

Business Requirement 7: Order History and Invoices
The application should maintain a history of previous orders and allow farmers to download invoices for accounting and record-keeping.

Business Requirement 8: Admin Dashboard for Order and Inventory Management
Administrators should have access to a dashboard where they can view and manage orders, update inventory levels, and manage user accounts (farmers and manufacturers).

Business Requirement 9: Ratings and Reviews
Farmers should be able to rate and leave reviews for products they have purchased, helping other farmers make informed purchasing decisions.

Business Requirement 10: Customer Support Integration
The system should provide customer support features, including live chat, email support, and a FAQ section, to assist farmers with any questions or issues they encounter while using the platform.

*9. Give the priority to the requirements 1-10( low to high)*

### **Business Requirements for the Online Agriculture Product Store Project**

(Assigned Priority from 1 to 10)

**Business Requirement 1: Product Catalog Management**
**Priority: 9**
Rationale: Managing product catalogs is crucial as it directly impacts the availability of products for farmers. Accurate product details (like descriptions, prices, and stock levels) are essential for the functioning of the online store.

**Business Requirement 2: User Registration and Authentication**
**Priority: 10**
Rationale: User registration and authentication are critical for securing the platform, ensuring only authorized users (farmers, manufacturers, and admins) access the system. This is a fundamental requirement for the system to function

**Business Requirement 3: Search and Filter Functionality**
**Priority: 8**
Rationale: Searching and filtering products is a key feature for user experience, allowing farmers to find relevant products quickly. This helps reduce friction for users browsing through the catalog.

**Business Requirement 4: Product Ordering and Cart Management**
**Priority: 10**
Rationale: Enabling farmers to place orders and manage the cart is one of the core functionalities of the platform. Without this, the e-commerce system cannot function properly.

**Business Requirement 5: Order Tracking and Delivery Status**
**Priority: 8**
Rationale: Order tracking provides transparency for the users, which is crucial for managing customer expectations. It's vital but comes after ensuring product availability and order placement features are functional.

**Business Requirement 6: Multiple Payment Options**
**Priority: 9**
Rationale: Providing multiple payment methods, including Cash on Delivery, ensures that a wide range of farmers can complete their purchases, especially considering rural areas may have limited access to online banking.

**Business Requirement 7: Order History and Invoices**
**Priority: 7**
Rationale: While important for farmers to maintain order history and invoices for tracking purchases, this is slightly less urgent than ensuring the basic functionalities like ordering, payment, and catalog management.

**Business Requirement 8: Admin Dashboard for Order and Inventory Management**
**Priority: 9**
Rationale: The admin dashboard is essential for managing orders and inventory efficiently. This helps ensure smooth operations for administrators and manufacturers, making it a high priority.

**Business Requirement 9: Ratings and Reviews**
**Priority: 6**
Rationale: While helpful in building trust and improving customer experience, this can be considered a lower priority compared to core functionalities like product management, ordering, and payment options.

**Business Requirement 10: Customer Support Integration**
**Priority: 7**
Rationale: Providing support is important for addressing farmer issues and concerns, but it is slightly less urgent compared to ensuring the store’s basic functionality and user authentication.

|  |  |
| --- | --- |
| Business Requirement | Priority |
| Business Requirement 2: User Registration and Authentication | 10 |
| Business Requirement 4: Product Ordering and Cart Management | 10 |
| Business Requirement 5: Order Tracking and Delivery Status | 8 |
| Business Requirement 3: Search and Filter Functionality | 8 |
| Business Requirement 1: Product Catalog Management | 9 |
| Business Requirement 6: Multiple Payment Options | 9 |
| Business Requirement 8: Admin Dashboard for Order and Inventory Management | 9 |
| Business Requirement 7: Order History and Invoices | 7 |
| Business Requirement 10: Customer Support Integration | 7 |
| Business Requirement 9: Ratings and Reviews | 6 |

*10. Draw a use case*



1. *prepare use specs for all use cases*

### ****1. Browse Products****

**Actor(s):** Farmer
**Preconditions:** Farmer is logged in.
**Postconditions:** Displays available products.
**Basic Flow:**

1. Farmer logs in.
2. System displays a list of products.
3. Farmer browses and selects a product.

### ****2. Search for Products****

**Actor(s):** Farmer
**Preconditions:** Farmer is logged in.
**Postconditions:** Displays search results.
**Basic Flow:**

1. Farmer enters search criteria.
2. System displays matching products.
3. Farmer selects a product.

### ****3. Select Product****

**Actor(s):** Farmer
**Preconditions:** Farmer is logged in.
**Postconditions:** Displays product details.
**Basic Flow:**

1. Farmer selects a product.
2. System shows detailed information.

### ****4. Place Order****

**Actor(s):** Farmer
**Preconditions:** Farmer is logged in, has products in cart.
**Postconditions:** Order is placed and sent to the manufacturer.
**Basic Flow:**

1. Farmer reviews cart.
2. System asks for address/payment.
3. Payment processed and order confirmed.

### ****5. Track Order****

**Actor(s):** Farmer
**Preconditions:** Farmer is logged in, has placed orders.
**Postconditions:** Displays order status.
**Basic Flow:**

1. Farmer checks order history.
2. System displays order status and tracking.

### ****6. Upload Product Details****

**Actor(s):** Manufacturer
**Preconditions:** Manufacturer is logged in.
**Postconditions:** Product is added to catalog.
**Basic Flow:**

1. Manufacturer enters product details.
2. System uploads product to catalog.

### ****7. Update Product Information****

**Actor(s):** Manufacturer
**Preconditions:** Manufacturer is logged in, has products uploaded.
**Postconditions:** Product details updated.
**Basic Flow:**

1. Manufacturer selects product.
2. Manufacturer updates product info.
3. System validates and updates product.

### ****8. Manage Products****

**Actor(s):** Admin
**Preconditions:** Admin is logged in, has permissions.
**Postconditions:** Catalog is updated.
**Basic Flow:**

1. Admin adds, updates, or removes products.
2. System processes changes.
3. *Prepare 5 activity diagram*
* User registration-



* search products



1. Add products to cart



1. Making a payment



1. Delivery
2. 