Question – 1 Functional requirements

Functional requirement define specific behaviour, functions or operations of the system. They describe what the system should do, outlining necessary tasks, actions or activities it must perform to achieve its objectives.

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | **Requirement Name** | **Requirement Description** | **Priority** |
| FR0001 | Farmer registration | Farmer must be able to register with the application | 8 |
| FR0002 | Farmer search for products | Farmers must be able to search available products | 9 |
| FR0003 | Manufacturer registration | Manufacturer must be able to register | 8 |
| FR0004 | Product upload | Manufacturer must be able to upload products with details like price, stock and description | 8 |
| FR0005 | Product catalogue | A categorized catalogue must display all the available products to the farmers. | 9 |
| FR0006 | Payment gateway | Farmers must be able to make secured payment using multiple payment options. | 10 |
| FR0007 | Notification | Farmers must be receiving confirmation details on email as well as Text message. | 8 |
| FR0008 | Delivery tracking | Farmers must be able to track their orders in real time | 9 |
| FR0009 | Order management | Farmers must be able to view their order history and make changes in order. | 8 |
| FR0010 | Wish list facility | Farmers must be able to wish list product for future consideration | 7 |
| FR0011 | Review and rating | User must be able to give product review and rating | 8 |
| FR0012 | User authentication | User must be authenticated before placing order | 10 |
| FR0013 | Mobile friendly design | System must support mobile usage | 9 |
| FR0014 | Dashboard for manufacturer | Manufacturer should have dashboard to manage products and sales data | 8 |
| FR0015 | Report generation | Manufacturer must be able to generate report on sales, inventory and farmer feedbacks. | 7 |
| FR0016 | Support ticketing | Farmers and manufacturers must be able to raise ticket on concerning issue. | 9 |
| FR0017 | Multi language support | System must be able to support multiple languages for better accessibility. | 5 |
| FR0018 | Product recommendation | System should recommend product to user based on past purchases | 6 |
| FR0019 | Data backup | All the critical data must be backed up automatically. | 9 |
| FR0020 | Data security | System should comply with security standards | 10 |

Non-Functional requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | **Requirement name** | **Requirement Description** | **Priority** |
| NFR0101 | Page loading time | Each page should load within 2 seconds | 9 |
| NFR0102 | Scalability | System should handle at least 10000 user concurrently | 9 |
| NFR0103 | Availability | The system should have up time of 99.9% | 10 |
| NFR0104 | Response time | System should not take more than 1 second for response | 10 |

Question-2 Wireframes and prototypes.









Question-3 Note on tools (Balsmiq & MS Visio)

Balsmiq

Balsmiq is a rapid wire-framing tool designed for creating low-fidelity mock-ups of user interfaces. It is widely used by UX designers, product managers, and developers to visualize and communicate design ideas effectively.

Features;

* Drag and drop interface: Simple, intuitive interface for building mock-ups quickly.
* Low-fidelity design: Focuses on functionality and layout, avoiding distractions of high fidelity details.
* Pre-built UI elements: Offers a library of UI components like buttons, forms, and menus.
* Collaboration: Allows for team feedback and collaboration through comments and annotations.
* Export options: Mock-ups can be exported as images or PDFs for sharing and presentation.

Uses

* Designing wireframes for websites, mobile apps, and software interfaces.
* Gathering early feedback on designs.
* Demonstrating workflows and layouts to stakeholders.

Microsoft Visio

It is a software application that allows users to create wide range of visual diagrams. It is used by business analysts, IT professionals and architects for its versatility and advanced features. It is primarily used for visualizing business processes, system architecture, and project planning.

Key features of Microsoft Visio:

* Extensive templates: Supports flowcharts, organizational charts, network diagrams, UML diagrams and more.
* Integration: seamlessly integrates with other Microsoft Office tools and services like SharePoint and OneDrive.
* Custom shapes: Allows users to create and customize shapes to suit specific needs.
* Collaboration: Supports real time collaboration and sharing among team members.
* Data Linking: Connects diagrams to external data sources, enabling dynamic updates.
* Who uses MS Visio:

 Business analyst to visualize business processes, workflows and system designs and to draw UML diagrams.

IT professionals: Create network diagrams, system architecture maps and data flow diagrams.

Project managers: To illustrate project timelines and dependencies.

Comparison:

|  |  |  |
| --- | --- | --- |
| Aspect  | Balsamiq | Visio |
| Purpose  | UI/UX wire framing | Diagramming and flowcharting |
| complexity | Easy to use ,minimal learning |  |
| Output style | Low-fidelity mock-ups | High-fidelity diagrams |
| Target audience  | Designers, product managers | Business analysts, IT professionals. |

Question – 4 Requirement traceability Matrix

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Requirement ID | Requirement Name | Requirement description | Design | D1 | T1 | T2 | T3 | T4 | UAT |
| FR0001 | Farmer registration | Farmers should be able to register with the application |  |  |  |  |  |  |  |
| FR0002 | Farmer search for products | Farmers must be able to search for available products in fertilizers, seeds and pesticides |  |  |  |  |  |  |  |
| FR0003 | Product details display | Farmers should be able to view detailed description of products |  |  |  |  |  |  |  |
| FR0004 | Add to cart | Farmers should be able to add selected product to shopping cart |  |  |  |  |  |  |  |
| FR0005 | Place order | Farmers should be able to place orders for selected products |  |  |  |  |  |  |  |
| FR0006 | Payment Integration  | System should allow farmers to make payment online  |  |  |  |  |  |  |  |
| FR0007 | Delivery address management | Farmers should be able to manage and save multiple delivery addresses |  |  |  |  |  |  |  |
| FR0008 | Order tracking | Farmers should be able to track the status of their order |  |  |  |  |  |  |  |
| FR0009 | Manufacturer registration | Manufacturer must be able to register with application |  |  |  |  |  |  |  |
| FR0010 | Product upload | Manufacturer should be able to upload product details |  |  |  |  |  |  |  |
| FR0011 | Communication module  | The system should allow communication between farmers and manufacturers. |  |  |  |  |  |  |  |
| FR0012 | Notification system  | The system should notify farmers and manufacturers of important updates |  |  |  |  |  |  |  |
| FR0013 | Reviews and ratings  | Farmers should be able to leave review and rating on purchased products |  |  |  |  |  |  |  |
| FR0014 | Search filters | Farmers should be able to filter products based on categories and price |  |  |  |  |  |  |  |
| FR0015 | Multi-language support  | Application should support multiple languages |  |  |  |  |  |  |  |
| FR0016 | Inventory management | Manufacturer should be able to manage inventory details |  |  |  |  |  |  |  |
| FR0017 | Reports and analytics | System should generate reports for both farmers and manufacturers |  |  |  |  |  |  |  |
| FR0018 | Role based access | The application should implement role based access control |  |  |  |  |  |  |  |
| FR0019 | Customer support | Farmers and manufacturers should be able to access support through system |  |  |  |  |  |  |  |
| FR0020 | FAQ section | FAQ section should be available for common queries |  |  |  |  |  |  |  |
| NFR0101 | Page loading time | Each page should load within 2 second |  |  |  |  |  |  |  |
| NFR0102 | Security  | Application should comply with industry security standards |  |  |  |  |  |  |  |
| NFR0103 | Scalability  | The system should be scalable to handle traffic |  |  |  |  |  |  |  |
| NFR0104 | Mobile responsiveness | System should be responsive for mobile and tablet devices |  |  |  |  |  |  |  |
| NFR0105 | Uptime  | System should have up time of 99.9% |  |  |  |  |  |  |  |

D1: Completed Design

T1-T4: Testing Phases

UAT: User Acceptance Testing

Question – 5: Test case documents

Here are 10 test case examples to validate key functionalities of the application.

|  |  |
| --- | --- |
| Test case 1 | Farmer registration  |
| Test case ID | TC001 |
| Test Objective | Validate that a farmer can successfully register with the application |
| Preconditions  | None  |
| Test steps | 1. Open the application.
2. Enter valid personal details (name, address, phone and email).
3. Set user name and password.
4. Submit registration form.
 |
| Expected results  | Farmer is successfully registered and a confirmation message is displayed. |

|  |  |
| --- | --- |
| Test case 2 | Manufacture product upload |
| Test case ID | TC002 |
| Test Objective | Verify that manufacturer can upload the product details (seeds, fertilizers and pesticides). |
| Preconditions  | Manufacturer must be logged in. |
| Test steps | 1. Login as a manufacturer.
2. Navigate to the "Add Product" page.
3. Enter product details (name, category, price, quantity, description).
4. Upload an image of the product.
5. Click "Submit".
 |
| Expected results  | Product is successfully added to the catalogue and visible to farmers. |

|  |  |
| --- | --- |
| Test case 3 | Product Search by Farmer |
| Test case ID | TC003 |
| Test Objective | Ensure farmers can search for products using keywords or categories. |
| Preconditions  | Farmers must be logged in  |
| Test steps | 1. Login as a farmer.
2. Go to the "Search" page.
3. Enter a keyword (e.g., "fertilizer") in the search bar.
4. Select a category from the dropdown.
5. Click "Search".
 |
| Expected results  | A list of relevant products is displayed |

|  |  |
| --- | --- |
| Test case 4 | Add Product to Cart |
| Test case ID | TC004 |
| Test Objective | Validate that farmers can add products to their shopping cart. |
| Preconditions  | Farmer must be logged in and a product search must be completed. |
| Test steps | 1. Search for a product.
2. Click on a product from the search results.
3. Click "Add to Cart".
 |
| Expected results  | Product is successfully added to the cart, and the cart count is updated. |

|  |  |
| --- | --- |
| Test case 5 | Place an Order |
| Test case ID | TC005 |
| Test Objective | Ensure farmers can place an order successfully. |
| Preconditions  | Product must be added to the cart. |
| Test steps | 1. Navigate to the cart.
2. Click "Checkout".
3. Enter delivery details (address, phone number).
4. Select a payment method and complete the transaction.
 |
| Expected results  | Order is placed successfully, and a confirmation message is displayed. |

|  |  |
| --- | --- |
| Test case 6 | Product Delivery Tracking |
| Test case ID | TC006 |
| Test Objective | Verify that farmers can track their order status. |
| Preconditions  | Farmer must have placed an order. |
| Test steps | 1. Login as a farmer.
2. Navigate to the "My Orders" page.
3. Select an order to view its details.
 |
| Expected results  | Order status (e.g., "Processing", "Shipped", and “Delivered") is displayed. |

|  |  |
| --- | --- |
| Test case 7 | Update Farmer Profile |
| Test case ID | TC007 |
| Test Objective | Validate that farmers can update their profile information. |
| Preconditions  | Farmer must be logged in. |
| Test steps | 1. Login as a farmer.
2. Navigate to the "My Profile" page.
3. Edit personal details (e.g., phone number, address).
4. Save changes.
 |
| Expected results  | Profile is successfully updated, and changes are reflected. |

|  |  |
| --- | --- |
| Test case 8 | Manufacturer Registration |
| Test case ID | TC008 |
| Test Objective | Verify that manufacturers can register with the application. |
| Preconditions  | None. |
| Test steps | 1. Open the application.
2. Navigate to the "Register" page.
3. Select "Manufacturer" as the account type.
4. Enter details (name, company, address, email, phone).
5. Submit the registration form.
 |
| Expected results  | Manufacturer is successfully registered, and a confirmation message is displayed. |

|  |  |
| --- | --- |
| Test case 9 | Product Reviews |
| Test case ID | TC009 |
| Test Objective | Ensure farmers can leave reviews for purchased products. |
| Preconditions  | Farmer must have completed a purchase. |
| Test steps | 1. Login as a farmer.
2. Navigate to the "My Orders" page.
3. Select a completed order.
4. Leave a review and rating for a product.
 |
| Expected results  | Review is successfully saved and visible to others. |

|  |  |
| --- | --- |
| Test case 10 | Application Accessibility |
| Test case ID | TC010 |
| Test Objective | Ensure the application adheres to accessibility standards (WCAG 2.1). |
| Preconditions  | None  |
| Test steps | 1. Navigate through the application using keyboard navigation.
2. Test with a screen reader.
3. Check colour contrast and font sizes.
 |
| Expected results  | The application is accessible to all users, including those with disabilities. |

These test cases cover critical functionalities of the online agriculture store.

Question- 6: DB Design



Question 7:

What is Data Flow Diagram (DFD)?

A Data Flow Diagram is a graphical representation of the flow of data within a system. It illustrates how data moves from one component to another, including the processes that transform the data, data stores and external entities. DFDs are used to model a system at various levels of detail, helping stakeholders understand how the system operates.

Key Components of a DFD:

1. Processes: Represented by circles or ovals, these transform inputs into outputs (Place order).
2. Data flows: Represented by arrows, these show the movement of data between entities, processes and data stores.
3. Data stores: represented by parallel lines or rectangles, these indicate where data is stored.
4. External entities: Represented by rectangles, these are sources or destinations of data external to the system (farmer, manufacturer).

Data flow for a farmer placing an order:

1. A farmer browses products (fertilizers, seeds and pesticides) listed by manufacturers.
2. The farmer selects the products, places an order and provide delivery details.
3. The system stores the order details and sends a notification to the manufacturer.

DFD Diagram Representation:

1. External Entity:
* Farmer: Initiates the process by placing an order.
1. Processes:
* Browse products: The farmer browses through the available products.
* Place order: The farmer selects products and places an order.
* Send order notification: The system notifies the manufacturer about the order.
1. Data stores:
* Product catalogue: stores product details provided by manufacturers.
* Order database: stores the farmers order details.
1. Data flows:
* From farmer to browse products: search criteria.
* From browse products to farmer: product list
* From farmers to place order: order details.
* From place order to send order notification: order conformation.
* From send order notification to manufacturer: order details.



Question – 8 Change request

Handling change requests in a project requires a structured approach to ensure that the changes are analysed, approved and implemented effectively without disrupting the overall timeline and budget.

Acknowledge the change request

Initiate the process: accept the change request from the stakeholder (in this case, due to changes in the government taxation structure).

Document the request: Record the details of the change in a change request form

Including:

* The nature of change
* Who raised the request
* The reason for the change (compliance with new government regulations)

Analyse the Impact of the change

* Assess scope: Determine how the change will affect the existing functionality of the online store (pricing, tax calculations and invoices)
* Identifying dependencies: Identify modules, systems and processes impacted by the change (backend logic for tax calculation, database fields, UI changes).
* Estimate effort and cost: collaborate with the project team (developers, tester and DB admin) to estimate the effort, time and cost required to implement the change.

Feasibility evaluation

* Budget and timeline impact: analyse if the change can be accommodated within the 2 crore INR budget and 18-month timeline.
* Business justification: Assess the necessity of the change. In this case, compliance with tax regulations makes it a mandatory change.
* Risks and benefits: evaluate the risks and benefits of implementing the change.

Seek approval

* Present to stakeholders: Communicate the findings to the committee (Mr.Henry, Mr Pandu and Mr.Dooku) and other stakeholders like Peter, Kevin and Ben.
* Obtain approval: Secure formal approval for the change from the stakeholders before proceeding.

Plan for Implementation

* Update the requirements document: Revise the functional and technical specifications to reflect the new tax structure.
* Communicate with the team: Inform the project manager (Mr. Vandanam) and team members about the approved change.
* Create a change implementation plan: Define tasks, timeline and resource allocation.
* Implement and Test the changes:
* Develop the changes: Developers (Juhi, Teyson, Lucie, Tucker, and Bravo) implement the changes.
* Database updates: The DB admin (John) updates relevant tables for tax calculations.
* Network admin check: Mike ensures the changes are reflected across integrated systems.
* Testing: Testers (Jason and Alekya) validate the new tax structure for accuracy and functionality across scenarios.

Communicate the update

* Stakeholder communication: Notify stakeholders about the successful implementation of the change.
* User training: provide training or updated user manuals to ensure farmers and companies understand the changes.

Document and close the change request

* Record the changes: Update all project document to include the new tax structure details.
* Close the request: Mark the change request as completed in the change log.

Best practices

* Maintain clear and transparent communication throughout the process.
* Ensure thorough testing to avoid defects post-implementation.
* Minimize disruption to ongoing project activities by effectively planning and scheduling the change.

 By following these steps, the project can adapt to the new taxation structure while staying aligned with the goals and constraints of the initiative.

Question 9- Change request VS an Enhancement

This is a change request

Reasons:

* Change in Scope: The original requirement for the application was focused on enabling farmers to purchase agricultural products (fertilizers, seeds and pesticides) from manufacturers. The new requirement introduces a different functionality. Allowing farmers to sell their crop yields and enabling auction system.
* Impact on the existing system: This request alters the scope of the project by adding seller functionality, inventory management for farmers, a public marketplace, and auction features. These are new modules and were not part of the original project scope.
* Significant efforts and resources: Implementing these features will require additional analysis, design, development and testing possibly more budget and time, which is characteristic of change request.

Role of Business analyst

Acknowledge the request:

* Thank Ben and Kevin for bringing up the suggestion.
* Clarify that their idea is valuable but represent a change from the original project scope.

Document the change request:

Prepare a formal change request document detailing:

* New features: selling crop yields, adding products, displaying to the public, and the auction system.
* Justification for the change: helping farmers generate income and increase market opportunities.
* Stakeholders involved
* Risk, benefits and expected outcomes.

Analyse the Impact:

* Impact on the timeline and budget: Collaborate with the project team to estimate how much additional time, cost and resources are needed to implement these features.
* Impact on existing features: evaluate whether the changes will interfere with or delay the development of already planned functionalities.
* Technical feasibility: Discuss with developers, DB admin, and Network admin to ensure feasibility of the auction system and seller features.

Communicate with the committee:

* Present the findings to the committee (Mr. Henry, Mr. Pandu and Mr. Dooku).
* Highlight how the change aligns with the overall goal of the project (supporting farmers) but also discuss potential challenges.

Obtain approval:

* If the committee approves the change, update the project scope, timeline and budget accordingly.
* If the change is not approved, explain the decision to Ben and Kevin and suggest revisiting the idea as a Phase 2 features post-completion of the original project.

Update the project plan:

* If approved, update the requirements document, work breakdown structure and timelines to include the new functionality.
* Clearly define the priority of the new features relative to existing deliverables.

Treating this as a change request ensures proper evaluation, avoids scope creep and allows all stakeholders to agree on the next steps without impacting the project’s success.

Question- 10:

Estimation of man hour for project

Understand the scope

The project involves:

* Developing an online agriculture product store for web and mobile platforms
* Features include:
* User registration for farmers and manufacturers.
* Product listing by manufacturers.
* Product browsing and purchasing by farmers.
* Order management and delivery tracking
* Additional considerations:
* User-friendly interface
* Database setup and integration
* Testing and deployment

Break down the work

1. Requirement gathering and analysis: Conducted by BA and stakeholders estimated Time is 120hrs.
2. UI/UX Design Wireframes, prototypes and final designs estimated time: 200hrs
3. Backend development APIs for product management, user management, order management etc., estimated time is 600hrs.
4. Frontend development
* Web application development (React/Angular). Estimated time is 400hrs.
* Mobile application development (Flutter/React Native). Estimate time is 400hrs.
1. Database design and development: Schema creation, database setup and integration. Estimated time is 200hrs.
2. Integration: Connect backend, frontend and database. Estimated time is 200hrs.
3. Testing and QA: Functional integration and performance testing. Estimated time is 300hrs.
4. Deployment and support: Deployment on servers/cloud and post deployment support. Estimated time is 120hrs.

Estimated Total Man-Hours

|  |  |
| --- | --- |
| Phase  | Estimated Man-Hours |
| Requirement gathering | 120 |
| UI/UX Design | 200 |
| Backend development | 600 |
| Frontend development | 800 |
| Database design | 200 |
| Integration | 200 |
| Testing and QA | 300 |
| Deployment and support | 120 |
| Total | 2540 |

Consider buffer time

Add a buffer (20% of total time) for unforeseen challenges. Buffer = 2540\*0.2= 508 hours.

Final estimation

Total Man-hours= 2540+508= 3048hrs.

Resource allocation

With team of ten people including developers, testers, designers and managers the project would take approximately: 3048/10= 305hr per team member over 18 months.

Question-11 UAT

As a business analyst facilitating User Acceptance Testing and ensuring the projects closure involves several steps:

Preparing for UAT

Understand the scope and plan for UAT

Collaborate with stakeholders to finalize the scope of UAT based on requirement and agreed acceptance criteria.

Prepare a UAT plan outlining:

* Test objectives.
* Timeline and milestones.
* Roles and responsibilities.
* Testing environment and tools required.

Set up the testing environment

* Coordinate with the development, testing and network tams to ensure that the testing environment mirrors the production environment.
* Populate the UAT environment with appropriate test data.

Prepare UAT Test cases

* Create UAT test cases and scripts in collaboration with stakeholders and end users. These test cases should align with the agreed requirements.
* Share the test cases with the client for approval.

Coordinate UAT Logistics

* Schedule testing sessions with the client and stakeholders.
* Provide training or walkthroughs of the application to help users understand its functionality.

Conducting UAT

1. Kick-off UAT
* Host a kick-off meeting to brief users on the testing process, objectives and expectations.
* Ensure all stakeholders are aligned on the process.
1. Monitor UAT Execution
* Actively support users during testing, addressing any questions or issues.
* Document feedbacks, issues and defects reported by users.
1. Track and resolve issues
* Collaborate with the development and testing teams to resolve defects.
* Prioritize critical issues to ensure they are addressed within the UAT timeline.
1. Obtain sign-off for individual features

After users validate that specific features meet the requirements, obtain written or electronic approval for those features.

Completing UAT and Project Closure

1. Final sign-off
* Once all the issues identified during UAT are resolved, obtain final user acceptance sign-off from the client, confirming that the application meets the agreed requirements.
1. Deployment to production
* Coordinate with the deployment team to transition the application to the production environment.
* Notify stakeholders of the go-live plan.

Process for project closure

1. Conduct a closure meeting
* Host a project closure meeting with the client, stakeholders and project team to review:
* Project deliverables
* Lessons learned
* Any remaining action items or follow-ups.
1. Handover documentation
* Provide all project documentation to the client, including:
* Requirements documentation.
* User manuals or training materials
* Technical documentation
* Final UAT results and approvals
1. formal closure
* Obtain a project closure signoff from the client, acknowledging that all deliverables have met and accepted.

UAT acceptance process summary

1. Prepare UAT plan and environment.
2. Define UAT test cases and get client approval.
3. Conduct UAT session with stakeholders.
4. Address and resolve defects reported during UAT
5. Obtain final sign-off from the client.
6. Deploy to production and transition ownership to the client.

These steps must be followed to ensure smooth UAT process and a structured closure of the project while maintaining client satisfaction.

Question-12 Project closure document

A project closure document is a formal record that signifies the successful completion of the project. It provides a summary of projects objectives, deliverables, outcomes and lessons learned. It ensure that all stakeholders agree that the project has met its goal and no further work is pending.

Key components of a project closure document

1. Project summary
* Brief description of the project
* Objectives and goals achieved
* Project duration and budget utilization.
1. Deliverables and final status
* List of final deliverables (Online agriculture store).
* Confirmation that all deliverables have been handed over to the client.
* System testing and deployment details.
1. Stakeholder Acknowledgment
* Sign-off from the key stakeholders (Mr. Henry, SOONY Company, APT IT solutions).
* Acknowledgment that the project meets their requirements.
1. Financial summary
* Total budget allocated (2crores INR).
* Actual expenditure.
* Any remaining funds or additional costs incurred.
1. Lessons learned
* Challenges faced during the project.
* Solution implemented.
* Best practices for future projects.
1. Risk and issues resolved
* Major risks encountered and mitigation strategies.
* Pending issues (if any) and recommendations for future improvement.
1. Future recommendations
* Suggestions for future enhancements.
* Maintenance and support plans.
1. Approval and sign-off
* Signature of key stakeholders (Mr. Henry, Mr. Pandu, Mr. Dooku and APT IT Solutions team).
* Final acknowledgment that the project is successfully completed.

Purpose of the project closure document

* Ensures that all project objectives are met.
* Provides a historical reference for similar future projects.
* Confirms a formal acceptance from stakeholders.
* Helps in transitioning from project development to maintenance.