**Question 1 –**

**Functional Requirements**

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

**Non Functional Requirements**

* Non-functional requirements will describe the qualities and attribute of a system
* Focusing on how the system performs rather than specific behavior or function.

|  |  |  |
| --- | --- | --- |
| **Req ID** | **Requirement Name** | **Requirement Description** |
| FR001 | Farmer Registration | Farmer should be able to register with application |
| FR002 | Farmer Login | Farmer should be able login using their email & password |
| FR003 | Manufacturer Registration | Manufacturer should be able to register with application |
| FR004 | Manufacturer Login | Manufacturer should be able to log in and manage their listings |
| FR005 | Search For Product | Farmers should be able to search for seeds, pesticides and fertilizers |
| FR006 | Product Catalog Display | Farmer should be able to browse through product listings |
| FR007 | Product Filtering & Sorting | Farmer should be able to filter product by category, price, brand etc. |
| FR008 | Product Details Page | Clicking on a product should display detailed information |
| FR009 | Add to Cart | Farmers should be able to add products to a shopping cart |
| FR010 | Buy Later Feature | Farmers should be able to add products to a "buy later" list. |
| FR011 | Order Placement | Farmers should be able to place orders through the application. |
| FR012 | Payment Processing | The system should support multiple payment methods. |
| FR013 | Order Confirmation | Farmers should receive an email confirmation upon placing an order. |
| FR014 | Delivery Tracking | Farmers should be able to track their order status. |
| FR015 | Product Review & Rating | Farmers should be able to review and rate products. |
| FR016 | Order History | Farmers should be able to view past orders. |
| FR017 | Manufacturer Product Upload | Manufacturers should be able to upload product details. |
| FR018 | Customer Support | Farmers and manufacturers should be able to contact customer support |
| FR019 | Multi-Language Support | The application should support multiple languages for accessibility |
| FR020 | Admin Panel | Admins should be able to oversee transactions and manage users |

**Question 2–Minimum 5 page designs**

**1. Home Page**

* **Header:**
	+ Logo on the left
	+ Navigation bar (Home, Products, About, Contact, Login)
	+ Search bar in the center
	+ Cart icon on the right
* **Main Section:**
	+ Banner with featured products
	+ Categories: Fertilizers, Seeds, Pesticides
	+ Recommended products grid
* **Footer:**
	+ Quick links, Contact details, Social media icons

### 2. Login/Sign-up Page

* **Login Section:**
	+ Email field
	+ Password field
	+ Login button
	+ Forgot password link
* **Sign-up Section:**
	+ Email field
	+ Create password field
	+ Confirm password field
	+ Register button
* **Alternative Login:**
	+ Google and Facebook sign-in options

### 3. Product Catalog Page

* **Header:**
	+ Search bar, Filters, Categories dropdown
* **Product Listings:**
	+ Grid of product cards (Image, Name, Price, Buy Now button)
	+ Sorting options (Price, Popularity, Ratings)
* **Pagination or Load More Button**

### 4. Product Details Page

* **Product Image on Left**
* **Product Details on Right:**
	+ Name, Price, Manufacturer Details
	+ Description
	+ Add to Cart button, Buy Now button
	+ Reviews & Ratings
	+ Related Products Section

### 5. Checkout & Order Tracking Page

* **Checkout Section:**
	+ Shipping Address Form
	+ Payment Methods (COD, UPI, Credit/Debit Card)
	+ Order Summary
	+ Place Order Button
* **Order Tracking:**
	+ Order ID field
	+ Tracking status (Pending, Shipped, Delivered)
	+ Estimated delivery date

**Question 3 – Tools (Visio, Balsamiq)**

* **Microsoft Visio: -** It’s a diagramming and vector graphics application used to create diagrams, flowcharts, and other visual representations of complex information.
* **Balsamiq**: - It’s a rapid wire framing tool used to create mockups and prototype of user interfaces.
* **Axure: -** It’s a more advance prototyping tool used to create high-fidelity, interactive wireframes and prototypes for web and mobile applications.

**Question 4 – RTM**

It is a document to track the requirements throughout the project lifecycle, ensuring that they are met and no requirements are overlooked.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Req ID | Requirement Description | Source | Design | Development | Testing | Status |
| REQ-001 | New User Registration | Peter | User Registration Doc | Complete | Passed | Complete |
| REQ-002 | User Login (Farmers & Manufacturers) | Mr. Henry | UI Design Doc 1 | Complete | Passed | Complete |
| REQ-003 | Product Catalog (Fertilizers, Seeds, Pesticides) | Mr. Henry, Kevin | Database Schema 1 | Complete | Passed | Complete |
| REQ-004 | Product Search | Kevin | Search Functionality Doc | Complete | Passed | Complete |
| REQ-005 | Add to Buy-Later List | Peter | UI Design Doc 2 | Complete | Passed | Complete |
| REQ-006 | Payment Gateway (COD, Card, UPI) | Peter | Payment Integration Doc | In Progress | In Progress | In Progress |
| REQ-007 | Order Confirmation Email | Ben | Email Template Doc | Complete | Passed | Complete |
| REQ-008 | Delivery Tracking | Ben | Tracking API Integration | In Progress | Not Started | In Progress |
| REQ-009 | Product Details from Manufacturers | Mr. Henry | Data Input Form Design | Complete | Passed | Complete |
| REQ-010 | Display Products to Farmers | Mr. Henry | UI Design Doc 3 | Complete | Passed | Complete |

**Question 5 – 10 Test Case Documents**

Test case document is a detailed outline by tester to ensure that a software application or system is working as expected.

**Test Case 1: Valid User Login (Farmer)**

* **Test Case ID:** TC\_001
* **Test Case Name:** Valid Farmer Login
* **Description:** Verify that a registered farmer can log in with valid credentials.
* **Preconditions:** Farmer account exists.
* **Steps:**
	1. Navigate to the login page.
	2. Enter valid email and password.
	3. Click "Login."
* **Expected Result:** User is successfully logged in and redirected to the home page.
* **Actual Result:** (Pass/Fail)
* **Tester:** Jason

**Test Case 2: Invalid User Login**

* **Test Case ID:** TC\_002
* **Test Case Name:** Invalid User Login
* **Description:** Verify system behavior with incorrect login credentials.
* **Preconditions:** None
* **Steps:**
	1. Navigate to the login page.
	2. Enter invalid email and/or password.
	3. Click "Login."
* **Expected Result:** Error message displayed, user not logged in.
* **Actual Result:** (Pass/Fail)
* **Tester:** Alekya

**Test Case 3: Product Search**

* **Test Case ID:** TC\_003
* **Test Case Name:** Product Search
* **Description:** Verify product search functionality.
* **Preconditions:** Product catalog exists.
* **Steps:**
	1. Navigate to the product catalog page.
	2. Enter a valid product name in the search field.
	3. Click "Search."
* **Expected Result:** Relevant products are displayed in the search results.
* **Actual Result:** (Pass/Fail)
* **Tester:** Jason

**Test Case 4: Add to Buy-Later List**

* **Test Case ID:** TC\_004
* **Test Case Name:** Add to Buy-Later
* **Description:** Verify adding a product to the buy-later list.
* **Preconditions:** User is logged in. Product exists in catalog.
* **Steps:**
	1. Browse the product catalog.
	2. Click "Add to Buy-Later" for a product.
* **Expected Result:** Product is added to the user's buy-later list.
* **Actual Result:** (Pass/Fail)
* **Tester:** Alekya

**Test Case 5: New User Registration**

* **Test Case ID:** TC\_005
* **Test Case Name:** New User Registration
* **Description:** Verify new user registration process.
* **Preconditions:** None
* **Steps:**
	1. Navigate to the registration page.
	2. Enter required information (email, password, etc.).
	3. Click "Register."
* **Expected Result:** User account is created, confirmation message displayed.
* **Actual Result:** (Pass/Fail)
* **Tester:** Jason

**Test Case 6: Payment Gateway - COD**

* **Test Case ID:** TC\_006
* **Test Case Name:** Payment Gateway - COD
* **Description:** Verify Cash on Delivery payment option.
* **Preconditions:** User is logged in, has items in cart.
* **Steps:**
	1. Proceed to checkout.
	2. Select "Cash on Delivery" as payment method.
	3. Confirm order.
* **Expected Result:** Order is placed successfully with COD as payment.
* **Actual Result:** (Pass/Fail)
* **Tester:** Alekya

**Test Case 7: Order Confirmation Email**

* **Test Case ID:** TC\_007
* **Test Case Name:** Order Confirmation Email
* **Description:** Verify order confirmation email is sent.
* **Preconditions:** User has placed an order.
* **Steps:**
	1. Place an order.
* **Expected Result:** User receives an order confirmation email.
* **Actual Result:** (Pass/Fail)
* **Tester:** Jason

**Test Case 8: Delivery Tracking**

* **Test Case ID:** TC\_008
* **Test Case Name:** Delivery Tracking
* **Description:** Verify delivery tracking functionality.
* **Preconditions:** Order has been shipped.
* **Steps:**
	1. Access the delivery tracking page (via link in email or on website).
	2. Enter tracking number.
* **Expected Result:** Delivery status and information are displayed.
* **Actual Result:** (Pass/Fail)
* **Tester:** Alekya

**Test Case 9: Manufacturer Product Submission**

* **Test Case ID:** TC\_009
* **Test Case Name:** Manufacturer Product Submission
* **Description:** Verify manufacturers can submit product details.
* **Preconditions:** Manufacturer account exists.
* **Steps:**
	1. Log in as a manufacturer.
	2. Fill out the product details form.
	3. Submit the form.
* **Expected Result:** Product details are saved in the system.
* **Actual Result:** (Pass/Fail)
* **Tester:** Jason

**Test Case 10: Product Display to Farmers**

* **Test Case ID:** TC\_010
* **Test Case Name:** Product Display to Farmers
* **Description:** Verify products are displayed correctly to farmers.
* **Preconditions:** Products have been submitted by manufacturers.
* **Steps:**
	1. Log in as a farmer.
	2. Browse the product catalog.
* **Expected Result:** Product details are displayed correctly.
* **Actual Result:** (Pass/Fail)
* **Tester:** Alekya

**Question 6 – DB Design**

**DB Schema: -** It is a blueprint that outlines the structure of a database, including its tables, fields, relationships, constrains, and other characteristics.

**ER Diagram**: - it is a visual representation of the relationships between entities in a database. It depicts the entities (such as tables), attributes (properties or fields), and relationships between them.

**Question 7 – Data Flow Diagram**

Data Flow Diagram (DFD) is a graphical representation of the flow of data within a system. It visually shows how data moves from one process to another, how it stored, and where it ends up.

It helps analysts and designers understand the flow of data within a system, identify potential bottlenecks or inefficiencies, and communicate system requirements to stakeholders.

--------------------------------------------------------------------------------------------------------------------

Order Confirmed

Place Order & Payment

FARMER

Brows/ Search Product

Add to cart /Buy Later

Product Catalog

Product Database

Cart / Buy Later

Payment Gateway

Order Details (Database)

**Explanation:**

1. **Farmer:** Interacts with the system.
2. **Browse/Search Products:** Farmer searches or browses the product catalog.
3. **Product Catalog (Database):** Stores product information.
4. **Product Details (Database):** Provides detailed information about selected products.
5. **Add to Cart/Buy Later:** Farmer adds desired products to their cart or buy-later list.
6. **Cart/Buy Later (Database):** Stores the farmer's cart and buy-later items.
7. **Place Order & Payment:** Farmer proceeds to checkout, provides payment information.
8. **Payment Gateway (External):** Handles payment processing (e.g., UPI, card).
9. **Order Confirmed (System/Email):** System confirms the order, sends confirmation email.
10. **Order Details (Database):** Stores order information

**Question 8 – Change Request**

Change requests are a formal proposal to alter a system, product or project.

1. **Identify the Change:** Document the specific changes required to the tax structure (e.g., new tax rates, calculations).
2. **Impact Assessment:** Analyze the impact of the change on the system (e.g., which modules are affected, how much development effort is needed).
3. **Create a Change Request Form:** Fill out a formal change request form detailing the change, its impact, and justification.
4. **Submit for Approval:** Submit the change request to the change control board (CCB) – in this case, Mr. Henry, Mr. Pandu, and Mr. Dooku.
5. **Prioritization:** The CCB will prioritize the change request based on its urgency and impact.
6. **Implementation:** If approved, the development team will implement the changes.
7. **Testing:** Thorough testing will be conducted to ensure the changes are implemented correctly and don't introduce any new issues.
8. **Deployment:** The changes will be deployed to the production environment.

**Question 9 – Change Request vs. Enhancement**

This is an **enhancement**. It's a significant new feature, not just a modification of existing functionality.

Here's my response:

"Ben and Kevin, thank you for your suggestions. The ability for farmers to sell their crops through the platform is a valuable addition. However, because this involves substantial new functionality (adding product listings by farmers, auction system), it's considered an *enhancement* rather than a simple change request. We'll need to follow the enhancement request process, which includes:

1. **Detailed Requirements Gathering:** We'll need to discuss the specifics of how farmers will list their products, the auction process, payment mechanisms, etc.
2. **Feasibility Study:** We'll assess the technical feasibility and effort required.
3. **Cost Estimation:** We'll estimate the cost of developing this new feature.
4. **Prioritization:** This enhancement will be prioritized along with other project tasks.
5. **Implementation (if approved):** The development team will implement the feature.

**Question 10 – Estimations**

Man hours are the required effort of the resources to complete a project.

There are 3 types of projects:-

* Small – Upto 500 hrs.
* Medium – Upto 1000 hrs.
* Large – Upto 1500 hrs.

This is a medium- size project with reasonably experienced team. I am assuming 18 months translate to roughly 360 working days (excluding weekends and holidays) with average of 8 working hours per day.

Let’s assume core functionality (login, product catalog, and search payment) takes 12000 man hours. New farmer selling feature might take another 4000 hours. Testing and bug fixing could take 20% of the development 3200 man hours. Project management, meetings, etc might add another 1900 hours.

Final Estimate: - 12000+4000+3200+1900 = 21100 man-hours.

**Question 11 – UAT (User Acceptance Test)**

**UAT Process:**

**Planning: -** In this step, blue prints are made to implement UAT testing for every feature that needs to test and minimum standard for accepting the test.

**Designing: -** Here the test cases are designed to hide all the possibilities of software packages in a real world environment.

**UAT Tester: -** A testing team consist of an end user that meet the criteria for implementing testing, they should know the test cases to run and understand the functionalities.

**Bug Fixing: -** Whatever bugs are found in the UAT Testing, the development team should work on them and make it software error free.

**Sign Off: -** After removing all the bugs, the testing team indicates acceptance of the completion of the bugs. In this phase, all the stakeholders come to a conclusion that the software is ready to GO LIVE and sign it off.

**Project Closure:**

1. **Final Deliverables:** Ensure all deliverables (source code, documentation, etc.) are handed over to the client.
2. **Project Documentation:** Complete all project documentation, including the project closure report.
3. **Knowledge Transfer:** Conduct knowledge transfer sessions with the client's team to ensure they can maintain and operate the system.
4. **Post-Project Review:** Conduct a post-project review to identify lessons learned and best practices for future projects.
5. **Formal Closure Meeting:** Hold a formal project closure meeting with all stakeholders to officially close the project.
6. **Archiving:** Archive all project-related documents and artifacts.