**Question 1– Functional Requirements- Identify minimum 20 functional requirement**

**Functional Requirment:-**

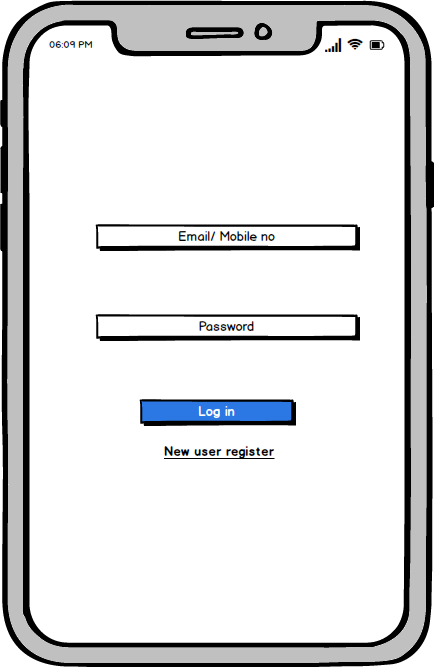
|  |  |  |  |
| --- | --- | --- | --- |
| **Req ID** | **Req Name** | **Req Description** | **Priority** |
| FR 01 | User Registration | User should be able to create an account by providing basic details such as name, email ID, phone number, Password | 10 |
| FR 02 | Product Catalog | The application should be able to display a comprehensive list of all available products including fertilizers, seeds and pesticides with detailed information | 10 |
| FR 03 | Mobile App | The application should have a mobile app version for users to access it from their mobile devices | 2 |
| FR 04 | Search functionality | Users should be able to search for products based on various criteria like name, type, brand, price range, etc | 9 |
| FR 05 | Product Details | Users should be able to view detailed information about each product, including price, quantity, description, images and reviews | 9 |
| FR 06 | Shopping Cart | User details be able to add products to their cart and view their cart details, including total price and quantity | 7 |
| FR 07 | Checkout Process | Users should be able to complete the checkout process by providing delivery address, payment details and confirmation of order | 8 |
| FR 08 | Order Cancellation | Cancellation User should be able to cancel their order and request a refund as per the company's policies | 7 |
| FR 09 | Order Tracking | Users should be able to track their order status, including confirmation, processing, shipping and delivery. | 7 |
| FR 10 | User Profile | Users should be able to manage their profile | 7 |
| FR 11 | Order History | Users should be able to view their previous order history, including details like order date, order status and product details | 6 |
| FR 12 | Product Reviews | Users should be able to rate and review products they have purchased which will be visible to other users | 6 |
| FR 13 | Product Comparison | Users should be able to compare products based on various parameters such as price, quality and features | 5 |
| FR 14 | News letter subscription | Users shold be able to subscribe to the newsletter to receive updates about new products, discounts and promotions | 5 |
| FR 15 | Wish list | Users should be able to add products to their wish list for future purchase | 4 |
| FR 16 | Product Recommendation | Users should be provided with personalized product recommendations based on their search and purchase history | 4 |
| FR 17 | Multiple payment options | Users should be able to pay for their orders through multiple payment options such as credit/debit cards, net banking and wallets | 3 |
| FR 18 | Customer support | Users should be able to contact customer support for any queries, complaints or feedback | 3 |
| FR 19 | multi languages Support | The application should support multiple languages for users from different regions | 2 |
| FR 20 | SE Optimization | The application should be optimized for search engine to improve its visibility and ranking in search results | 1 |

**Non Functional requirement:-**

|  |  |  |  |
| --- | --- | --- | --- |
| **Req ID** | **Req Name** | **Req Description** | **Priority** |
| NFR 01 | Page Loading Time | Each Page should load within 2 seconds time | 9 |
| NFR 02 | Authentication | Users should be required to authenticate using username and password | 10 |
| NFR 03 | Data encryption | Sensitive data should be encrypted | 8 |
| NFR 04 | Access control | Only authorized users should have access to certain functionalities or data | 7 |
| NFR 05 | Availability | The system should be available 99.9% of the time | 9 |
| NFR 06 | Browser compatibility | The system should be compatible with the latest versions of popular web browsers (Chrome, Firefox, Safari, Edge). | 8 |
| NFR 07 | Operating system compatibility | The system should be compatible with Windows, macOS, and Linux operating systems | 9 |
| NFR 08 | Efficient resource usage | The system should optimize resource usage to minimize costs (e.g., cloud service costs) | 5 |
| NFR 09 | Traceability | Changes made to the system configuration or code should be logged and traceable to specific users or processes. | 4 |
| NFR 10 | Consistency | The user interface should have a consistent layout and design across all pages | 6 |

**Question 2–Minimum 5 page designs**

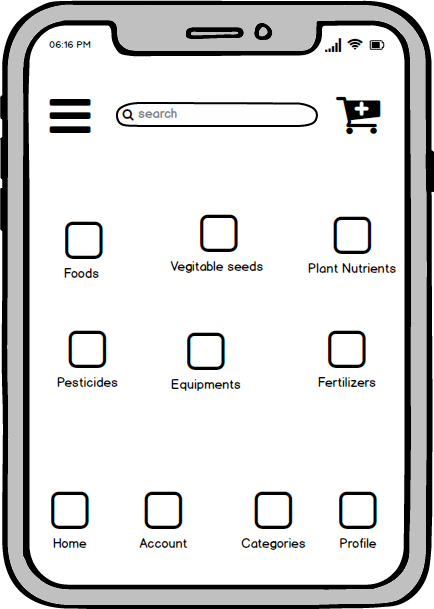
**1-Log in Page**

****

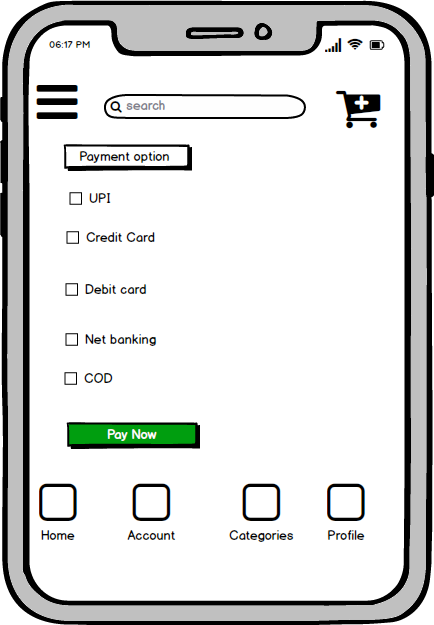
**2- Registration Page**

****

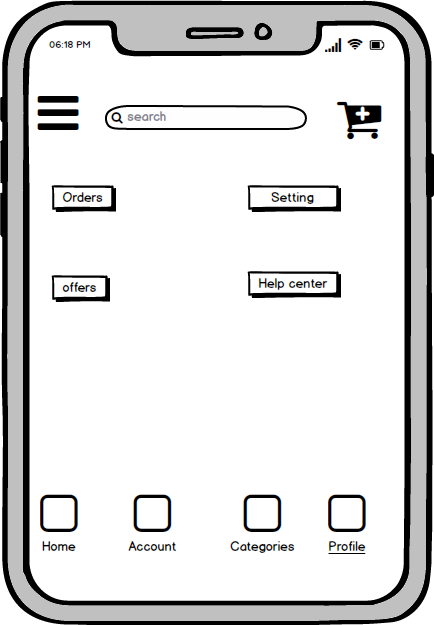
**3 Main Page**

****

**4- Payment page**

****

**5- Profile page**

****

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

**Microsoft Visio:**

Microsoft Visio is a versatile diagramming tool that allows users to create a wide range of diagrams, including flowcharts, organizational charts, and network diagrams. With its extensive library of shapes and customization options, Visio enables users to visually represent complex information and processes. It offers collaboration features, data linking capabilities, and seamless integration with other Microsoft Office applications.

**Key components and features of Microsoft Visio:**

**Shapes:** Visio provides a wide range of pre-defined shapes for various purposes such as flowcharts, network diagrams, electrical diagrams, organizational charts, and more. Users can drag and drop these shapes onto the canvas to create diagrams.

**Templates:** Visio offers numerous templates tailored for specific types of diagrams. These templates come with pre-defined shapes, connectors, and settings optimized for their respective diagram types, making it easier for users to get started.

**Stencil:** In Visio, stencils are collections of related shapes grouped together for easy access. Each template typically comes with its own set of stencils containing shapes specific to the diagram type. Users can also create custom stencils to organize their frequently used shapes.

**Connectors:** Visio provides various types of connectors to link shapes together and illustrate relationships in diagrams. Users can customize the appearance of connectors, such as line style, arrowheads, and endpoints.

**Text and Formatting Tools:** Users can add text to shapes and connectors to provide additional information or labels in their diagrams. Visio offers a range of formatting tools for text, including font styles, sizes, colors, and alignment options.

**Themes and Styles:** Visio allows users to apply themes and styles to their diagrams to enhance visual consistency and appeal. Themes can be applied to change the overall look and feel of a diagram, while styles can be used to customize the appearance of individual shapes.

**Grid and Guides:** Visio provides gridlines and guides to help users align and position shapes precisely on the canvas. Gridlines can be customized in terms of spacing and visibility, while guides can be dragged onto the canvas to serve as alignment aids.

**Data Linking:** Visio offers the ability to link shapes and diagrams to external data sources such as Excel spread sheets, databases, and SharePoint lists. This allows users to create dynamic diagrams that update automatically based on changes in the underlying data.

**Collaboration and Sharing:** Visio supports collaboration features such as commenting, reviewing, and co-authoring, allowing multiple users to work on the same diagram simultaneously. Diagrams can also be shared and published in various formats, including PDF, image files, and web pages.

**Integration with Other Microsoft Products:** Visio integrates seamlessly with other Microsoft Office applications such as Word, Excel, PowerPoint, and SharePoint. Users can embed Visio diagrams into Office documents or publish them to SharePoint for easy access and sharing.

**Balsamiq:**

Balsamiq is a popular wireframing tool used for creating low-fidelity prototypes. It focuses on simplicity and sketch-like designs to quickly visualize and communicate design ideas. With its drag-and-drop interface and pre-built UI elements, Balsamiq allows users to rapidly iterate and gather feedback on the basic structure and layout of a digital product

**Mockup Editor:** The core component of Balsamiq is its mockup editor, where users can drag and drop various UI elements (such as buttons, text boxes, images, and icons) onto a canvas to create wireframes and mockups of their designs. The editor provides a simple and intuitive interface for building prototypes quickly.

**UI Library:** Balsamiq comes with a comprehensive library of pre-built UI components and symbols that users can use in their designs. These components cover a wide range of UI elements commonly found in web and mobile applications, including navigation bars, form controls, tables, and more

**Customization Options:** Users can customize the appearance of UI elements in Balsamiq, such as adjusting colors, fonts, sizes, and styles, to match the desired look and feel of their designs. This allows for quick iteration and experimentation during the wireframing process.

**Templates:** Balsamiq offers a variety of pre-designed templates for different types of projects and applications, including websites, mobile apps, desktop software, and more. These templates provide a starting point for users to kickstart their designs and speed up the prototyping process.

**Linking and Interactivity:** Balsamiq allows users to create clickable prototypes by linking mockup screens together to simulate user interactions and navigation flows. This feature enables stakeholders to experience the user interface firsthand and provide feedback on usability and functionality.

**Version Control and Collaboration:** Balsamiq offers built-in version control and collaboration features that allow multiple team members to work on the same project simultaneously. Users can share their designs with others, track changes, leave comments, and iterate on designs collaboratively

**Desktop and Web Versions:** Balsamiq is available in both desktop and web-based versions, catering to different user preferences and workflow requirements. The desktop version offers offline access and a standalone application, while the web version provides browser-based access and real-time collaboration capabilities

**Question 4– RTM**

To tackle the situation and provide Mr. Henry and Peter with the current status of the project, I will prepare a Requirements Traceability Matrix (RTM). The RTM helps in tracking the requirements and their progress throughout the project. Here's an example of an RTM:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Req ID** | **Req Name** | **Req Description** | **Design** | **D1** | **T1** | **D2** | **T2** | **D3** | **T3** | **UAT** |
| FR001 | User Registration | User should be able to create an account by providing basic details such as name, email ID, phone number, Password | User will register | Done | Done | Done | Done | Done | Done | Done |
| FR002 | Product Catalog | The application should be able to display a comprehensive list of all available products including fertilizers, seeds and pesticides with detailed information | Done | Done | Done | Done | Done | Done | Done | Done |
| FR003 | Search functionality | Users should be able to search for products based on various criteria like name, type, brand, price range, etc | Done | Done | Done | Done | Done | Done | Done | Done |
| FR004 | SE Optimization | The application should be optimized for search engine to improve its visibility and ranking in search results | Done | Done | Done | Done | Done | Done | Done | Done |
| FR005 | Shopping Cart | User details be able to add products to their cart and view their cart details, including total price and quantity | Done | Done | Done | Done | Done | Done | Done | Done |
| FR006 | Checkout Process | Users should be able to complete the checkout process by providing delivery address, payment details and confirmation of order | Done | Done | Done | Done | Done | Done | Done | Done |
| FR007 | Order Tracking | Users should be able to track their order status, including confirmation, processing, shipping and delivery. | Done | Done | Done | Done | Done | Done | Done | Done |
| FR008 | User Profile | Users should be able to manage their profile | Done | Done | Done | Done | Done | Done | Done | Done |
| FR009 | Order History | Users should be able to view their previous order history, including details like order date, order status and product details | Done | Done | Done | Done | Done | Done | Done | Done |
| FR010 | Product Comparison | Users should be able to compare products based on various parameters such as price, quality and features | Done | Done | Done | Done | Done | Done | Done | Done |
| FR011 | Multiple payment options | Users should be able to pay for their orders through multiple payment options such as credit/debit cards, net banking and wallets | Done | Done | Done | Done | Done | Done | Done | Done |
| FR012 | multi languages Support | The application should support multiple languages for users from different regions | Done | Done | Done | Done | Done | Done | Done | Done |
| NF001 | Page Loading Time | Each Page should load within 2 seconds time | Done | Done | Done | Done | Done | Done | Done | Done |
| NF002 | Authentication | Users should be required to authenticate using username and password | Done | Done | Done | Done | Done | Done | Done | Done |
| NF003 | Data encryption | Sensitive data should be encrypted | Done | Done | Done | Done | Done | Done | Done | Done |

**Question 5–10 Test Case Documents- Prepare 10 Test Case Documents**

**Test case document – 1**

|  |  |
| --- | --- |
| **Test Case ID** | TC001 |
| **Test Case Document** | Farmer Registration |
| **Test Case Description** | Verify that a farmer can successfully register with the application using valid credentials |
| **Test Steps:** | Enter valid farmer details in the registration form. |
| Click on the "Register" button. |
| **Expected Result** | The farmer should be successfully registered and redirected to the login page |

**Test case document – 2**

|  |  |
| --- | --- |
| **Test Case ID** | TC002 |
| **Test Case Document** | Farmer Login |
| **Test Case Description** | Verify that a farmer can log in with their registered emailand password. |
| **Test Steps:** | Enter valid farmer email and password in the login form. |
| Click on the "Login" button. |
| **Expected Result** | The farmer should be successfully logged in and redirected tothe home page |

**Test case document – 3**

|  |  |
| --- | --- |
| **Test Case ID** | TC003 |
| **Test Case Document** | Product Search |
| **Test Case Description** | Verify that a farmer can search for products in the catalog. |
| **Test Steps:** | Enter a product keyword in the search bar. |
| Click on the search button. |
| **Expected Result** | The system should display relevant products matching thesearch keyword |

**Test case document – 4**

|  |  |
| --- | --- |
| **Test Case ID** | TC004 |
| **Test Case Document** | Add Product to Cart |
| **Test Case Description** | Verify that a farmer can add a product to their cart for purchase. |
| **Test Steps:** | Browse through the product catalog. |
| Select a product |
| .Click on the "Add to Cart" button. |
| **Expected Result** | The product should be successfully added to the farmer's cart. |
|  | **Test case document – 5** |
| **Test Case ID** | TC005 |
| **Test Case Document** | Payment Process |
| **Test Case Description** | Verify that the payment process functions correctl |
| **Test Steps:** | Proceed to checkout from the cart. |
| Enter payment details (e.g., credit card information, UPI, or selectcash-on-delivery option). |
| Click on the "Pay Now" button |
| **Expected Result** | The payment should be processed successfully, and the farmer should receive an order confirmation. |

**Test case document – 6**

|  |  |
| --- | --- |
| **Test Case ID** | TC006 |
| **Test Case Document** | Order Tracking |
| **Test Case Description** | Verify that a farmer can track the status of their order |
| **Test Steps:** | Go to the order tracking section. |
| Enter the order ID or relevant details. |
| Click on the "Track Order" button. |
| **Expected Result** | The system should display the current status and location of the farmer's order. |
|  | **Test case document – 7** |
| **Test Case ID** | TC007 |
| **Test Case Document** | Manufacturer Login |
| **Test Case Description** | Verify that a manufacturer can log in with their registered email and password |
| **Test Steps:** | Enter valid manufacturer email and password in the login form. |
| Click on the "Login" button |
|  |
| **Expected Result** | The manufacturer should be successfully logged in and redirected to their dashboard. |
|  | **Test case document – 8** |
| **Test Case ID** | TC008 |
| **Test Case Document** | Product Listing by Manufacturer |
| **Test Case Description** | Verify that a manufacturer can list their products in the catalog. |
| **Test Steps:** | Access the product listing section in the manufacturer dashboard. |
| Enter product details, pricing, and other relevant information. |
| Click on the "List Product" button. |
| **Expected Result** | The product should be successfully listed in the catalog under the manufacturer's name. |

**Test case document – 9**

|  |  |
| --- | --- |
| **Test Case ID** | TC009 |
| **Test Case Document** | Update Product Details |
| **Test Case Description** | Verify that a manufacturer can update the details of their listed products. |
| **Test Steps:** | Access the product details section in the manufacturer dashboard. |
| Select a product for update |
| Modify the product information. |
| Click on the "Update" button. |
| **Expected Result** | The product details should be successfully updated and reflected in the catalog |

**Test case document – 10**

|  |  |
| --- | --- |
| **Test Case ID** | TC010 |
| **Test Case Document** | Logout |
| **Test Case Description** | Verify that a user (farmer or manufacturer) can log out from their account. |
| **Test Steps:** | Click on the "Logout" button or link. |
|  |
| **Expected Result** | The user should be successfully logged out and redirected to the login page. |

**Question 6– DB Design**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Products |  | Shopping Order |  | Deliveries |  |
|  | Product Id |  | Order ID |  | Delivery id |  |
|  | Catagary id |  | Customer ID |  | order id |  |
|  | Product Name |  | date |  | customer id |  |
|  |  |  |  |  | product id |  |
|  |  |  |  |  | date |  |
|  | Categories |  |  |  |  |  |
|  | Category ID |  | Customer |  |  |  |
|  | Category Name |  | Customer ID |  |  |  |
|  | Category typr |  | Name |  |  |  |
|  |  |  | Contact |  | Transaction report |  |
|  |  |  | Address |  | report id |  |
|  |  |  |  |  | customer id |  |
|  | Seller |  |  |  | order id |  |
|  | Seler id |  | Payment |  | product id |  |
|  | Product Id |  | Payment id |  | payment id |  |
|  | Name |  | Customer id |  | date |  |
|  | contact add |  | data |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**Question7–Data Flow Diagram**



**Question 8– Change Request**

Handling change requests in a project involves a systematic process to ensure that changes are effectively managed while minimizing disruption to the project's scope, timeline, and resources. Here are the steps typically followed to handle change requests

1. Change Request Identification: Identify and document the change request, including the specific details of the requested change, the reason for the change, and its potential impact on the project.

2. Change Impact Analysis: Assess the impact of the change on various aspects of the project, such as scope, timeline, cost, resources, and risks. Evaluate the feasibility and implications of implementing the change.

3. Change Evaluation: Review the change request with key stakeholders, including project sponsors, clients, and relevant team members. Discuss the potential benefits, risks, and trade-offs associated with the change. Consider the project's objectives, constraints, and priorities in the evaluation process.

4. Change Prioritization: Prioritize the change request based on its urgency, impact, and alignment with project goals. Determine whether the change is critical and must be implemented immediately or can be scheduled for a future phase or release.

5. Change Approval: Obtain formal approval from the appropriate stakeholders, such as project sponsors or change control boards. Ensure that all stakeholders are in agreement regarding the change and its implications.

6. Change Implementation: Incorporate the approved change into the project plan, including any necessary adjustments to the scope, schedule, budget, or resources. Communicate the change to the project team and other relevant stakeholders. Update project documentation, such as requirements, design, and test plans, to reflect the approved change.

7. Change Communication: Communicate the approved change to all relevant parties, including team members, clients, and other stakeholders. Clearly explain the reasons for the change, its impact on the project, and any adjustments to expectations or deliverables.

8. Change Tracking and Documentation: Track and document all approved changes, including the rationale, approvals, and implemented modifications. Maintain a change log or change register to ensure transparency and accountability throughout the project.

9. Change Control and Monitoring: Continuously monitor the impact of implemented changes on the project's progress, risks, and quality. Maintain open lines of communication with stakeholders to address any concerns or issues related to the approved changes. Monitor the project's overall alignment with the revised scope, timeline, and objectives.

**Question9– Change Request Vs an Enhancement**

A change request and an enhancement are both terms used in software development and project management, but they refer to slightly different things. In this case, the request to allow farmers to add their crop yields, display them to the general public, and enable selling through the application represents an enhancement because it introduces new functionality that goes beyond the initial scope of the project. Additionally, the introduction of an auction system for crop yields adds another layer of functionality to the application. To address this enhancement request, I would follow the standard process for handling new requirements

**Change Request:**

•A change request is typically a formal proposal to alter a product, system, or project in some way.

•It often arises when there's a need to modify something that has already been defined or implemented.

•Change requests can involve fixing defects, addressing issues, or making adjustments to meet new requirements.

•They are usually submitted when there's a deviation from the initial plan or specification, and they often require approval from stakeholders before implementation

**Enhancement:**

•An enhancement, on the other hand, refers to an improvement or addition to a product or system that goes beyond its original specifications.

•Enhancements are typically intended to add new features, improve existing functionality, or enhance user experience.

•Unlike change requests, enhancements are often proactive and driven by a desire to make the product better rather than fixing something that's broken.

•While change requests are often reactive, responding to identified problems or changes in requirements, enhancements are more about adding value or staying competitive in the market

**Question 10– Estimations**

Estimating the number of man-hours required for the requested enhancements (adding crop yields, displaying them to the public, and implementing an auction system) would depend on various factors, including the complexity of the features, the size of the existing system, the development team's expertise, and the development methodology used. Without specific details about the project, it's challenging to provide an accurate estimation. However, I can offer a general guideline based on industry standards and experience:

1. Requirement Gathering and Analysis: 10-20 man-hours

2. This includes meetings with stakeholders, gathering detailed requirements, analyzing the impact, and documenting the enhancements.

3. Design and Architecture: 20-40 man-hours

4. This involves designing the system components, database structure, and user interface for the new features. It also includes identifying the necessary changes to accommodate the enhancements.

5. Development and Coding: 40-80 man-hours

6. The actual development of the new features, including backend and front end coding, integration with existing modules, and implementation of the auction system.

7. Testing and Quality Assurance: 20-40 man-hours

8. This phase involves writing test cases, performing unit testing, integration testing and ensuring the proper functioning and stability of the added features.

9. Deployment and User Acceptance Testing (UAT): 10-20 man-hours

10.Deploying the updated system to a testing environment, conducting user acceptance testing, and resolving any issues identified during UAT.

11.Documentation and Training: 10-20 man-hours

12.Documenting the new features, updating user manuals or guides, and providing training or support materials for farmers and users.

It's important to note that these estimations are rough figures and can vary significantly depending on the complexity and scale of the enhancements, the team's productivity, and other project-specific factors. It's recommended to involve the development team in the estimation process to get a more accurate assessment based on their expertise and knowledge of the project.

**Question 11– UAT**

To handle the situation of testing the final product and successfully completing it, the business analyst can follow these steps:

1. UAT Planning: Prepare a plan for User Acceptance Testing (UAT) in consultation with the client. This plan should include the scope of testing, test scenarios, test data, and timelines.

2. Test Environment Setup: Ensure that the required test environment is set up and available for the client to perform testing. This may include providing access to the testing environment, necessary test accounts, and any additional resources needed for testing.

3. Test Execution: Coordinate with the client to execute the planned test scenarios. Monitor the testing progress, provide support for any questions or issues that arise, and track the test results.

4. Defect Management: If any defects are identified during UAT, work closely with the client to understand the issues, document them, and track their resolution. Collaborate with the development team to address the reported defects and verify their fixes.

5. UAT Sign-off: Once the client has completed testing and is satisfied with the product's functionality, obtain their formal sign-off or approval. This indicates that the client has accepted the final product and is ready to move forward with its deployment

Regarding the process to close the project, it typically involves the following steps:

1. Final Documentation: Ensure that all project-related documentation is complete, including requirements, design documents, test cases, and user manuals. Review and update these documents to reflect the final product.

2. Project Review: Conduct a project review meeting with key stakeholders, including the client, to discuss the overall project performance, achievements, and lessons learned. Gather feedback and suggestions for improvement.

3. Project Closure Report: Prepare a project closure report summarizing the project's objectives, deliverables, timeline, budget, and overall success. Include any important metrics or performance indicators.

4. Handover or Deployment: Coordinate with the necessary teams, such as deployment or operations, to ensure a smooth transition of the final product to the production environment. Provide any necessary training or documentation to support the deployment process.

5. Post-Project Evaluation: After the product is deployed and operational, conduct a post-project evaluation to assess its performance, gather user feedback, and identify any areas for further improvement.

UAT Acceptance Process:

The User Acceptance Testing (UAT) Acceptance process involves the following steps:

1. Test Planning: Define the scope of UAT and identify the key features or functionalities to be tested. Prepare test scenarios and test cases based on user requirements.

2. Test Execution: Perform the planned test scenarios, following the test cases provided. Validate the system's behavior against the expected outcomes and verify that it meets the user's acceptance criteria.

3. Defect Reporting: If any issues or defects are identified during UAT, document them in a structured manner, including detailed steps to reproduce the problem. Communicate the issues to the development team for resolution.

4. Defect Resolution: Collaborate with the development team to address the reported defects. Verify the fixes provided by the development team and retest the affected areas.

5. Sign-off: Once all test scenarios have been executed, defects have been resolved, and the system meets the user's acceptance criteria, provide formal sign-off or approval. This signifies that the client accepts the product as meeting their requirements.

6. UAT Closure: Document the UAT results, including the test execution summary, any outstanding issues, and the overall assessment of the product. Communicate the closure of UAT to all stakeholders involved in the testing process.

The UAT Acceptance process ensures that the final product meets the client's expectations and is ready for deployment. It serves as a final validation before the project is considered complete and ready for closure.

**Question 12–** **Project Closure Document-**

A project closure document is a comprehensive report that summarizes the entire project's lifecycle, outcomes, and lessons learned. It serves as a formal record of the project's completion and provides important information for future reference. The document typically includes the following sections:

1. Project Overview: This section provides an overview of the project, including its objectives, scope, and stakeholders involved. It summarizes the project's purpose and sets the context for the closure report.

2. Project Achievements: Here, the document highlights the key achievements and deliverables of the project. It outlines the successful completion of milestones, tasks, and any significant accomplishments that were achieved.

3. Project Timeline and Budget: This section provides an overview of the project timeline, highlighting the start and end dates, major phases, and milestones. It also includes information on the project's budget, including any significant deviations or changes.

4. Lessons Learned: The lessons learned section reflects on the project's successes and challenges. It includes a comprehensive analysis of what worked well and what could have been improved. It highlights valuable insights and recommendations for future projects.

5. Stakeholder Feedback: This section gathers feedback from key stakeholders involved in the project. It includes their opinions, suggestions, and any concerns they may have expressed. The feedback helps in assessing the overall satisfaction and identifying areas for improvement.

6. Risks and Issues: The closure document discusses the risks and issues encountered throughout the project. It outlines the actions taken to mitigate these risks and resolve any issues that arose during the project's lifecycle

7. Project Performance: This section evaluates the project's performance against the defined objectives and success criteria. It assesses factors such as scope adherence, timeline adherence, budget performance, quality of deliverables, and customer satisfaction.

8. Project Sign-off: The closure document includes formal sign-off or approval from key stakeholders, indicating their acceptance and satisfaction with the project's outcomes. This signifies the official closure of the project

.9. Project Documentation: This section provides an overview of the project documentation, including the list of documents produced, their location, and accessibility for future reference.

10.Next Steps and Recommendations: The closure document outlines any recommended actions or next steps following the project's closure. It may include suggestions for further improvements, additional tasks, or follow-up activities.

The project closure document serves as a final report that captures the project's journey, outcomes, and key learnings. It provides a reference for future projects, helps in evaluating project success, and facilitates knowledge transfer to stakeholders involved in the project.

