**Capstone project-1 Part-3**

1. **Identify minimum 20 functional requirements**

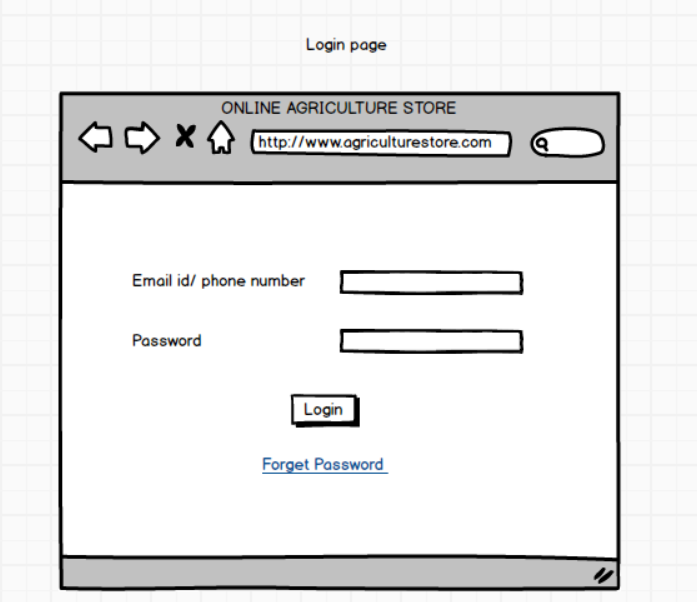
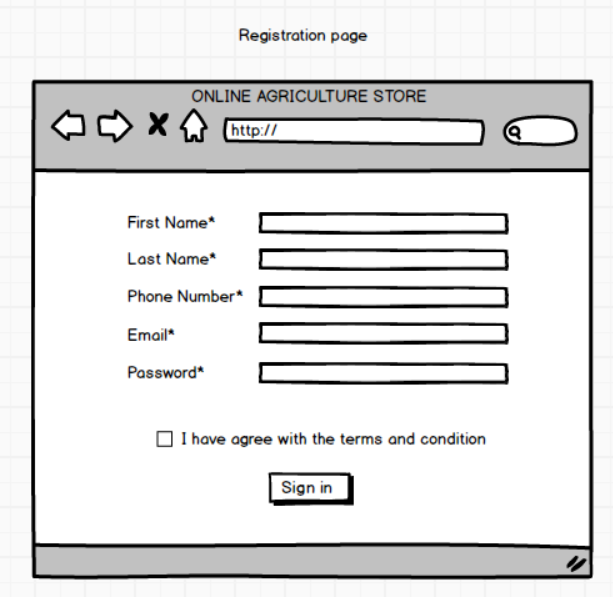
* The functional requirements show the functionalities that describe how the system will achieve the business requirements where the main focus is to show features, capabilities, and behaviour of the system.

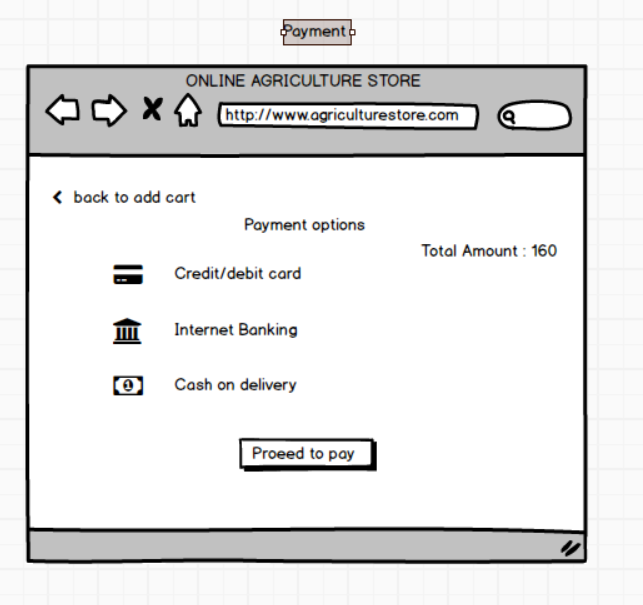
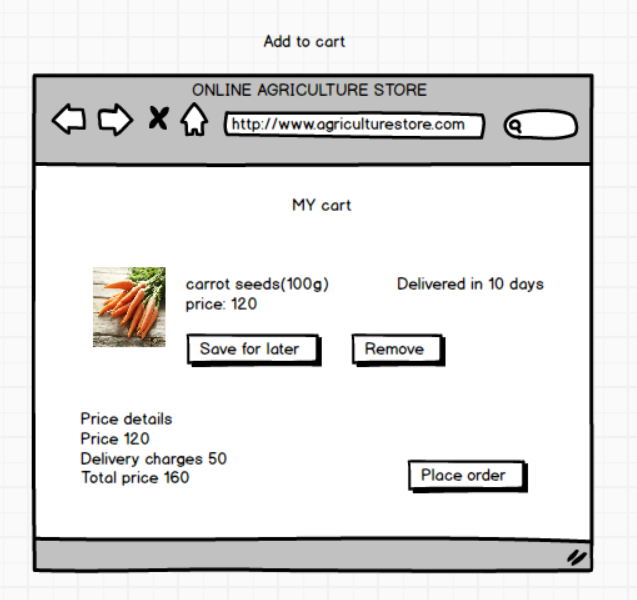
The Functional requirement

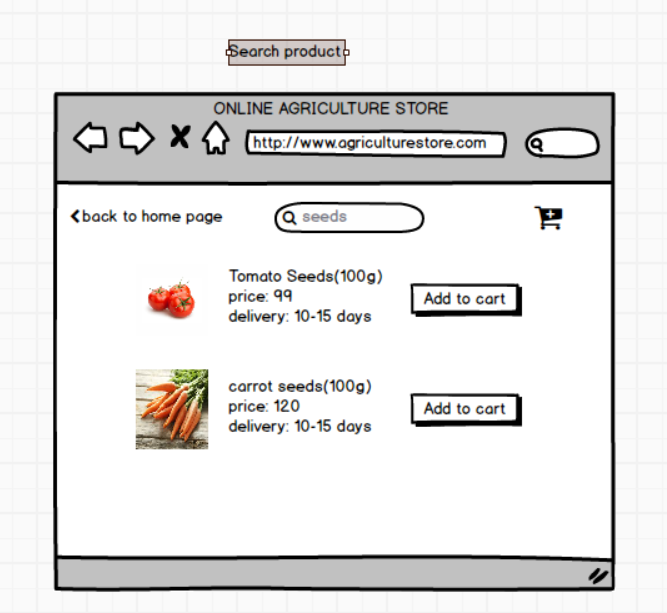
|  |  |  |  |
| --- | --- | --- | --- |
| **Req ID** | **Req Name** | **Req Description** | **Priority** |
| FR001 | User registration | The user should be able to register themselves on the application | 8 |
| FR002 | User Login | The user should able to login to their account with the help of their login credentials | 8 |
| FR003 | Uploading product details | Manufacture should able to upload their product on the application | 7 |
| FR004 | Product search | The user should be able to search all the product uploaded by the manufacturer | 8 |
| FR005 | Search Filter | User should able to use filter while searching product as per the requirement | 7 |
| FR006 | Product details | User should be able to see all the details about the product and the manufacturer uploaded in the application | 8 |
| FR007 | Adding product to cart | User should able to add the product in the cart as per their preference | 9 |
| FR008 | Payment method | The application should able to show different option to make payment | 9 |
| FR009 | Check out process | The application should allow user to proceed to check out from the shopping cart | 7 |
| FR010 | Order Confirmation | The application should able to send the confirmation of the order by email/ or message | 6 |

Non-Functional Requirement: It describes the qualities and attributes of a system, focusing on how the system performs rather than specific behaviours or Functions.

|  |  |  |  |
| --- | --- | --- | --- |
| **Req ID** | **Req Name** | **Req Description** | **Priority** |
| NFR001 | Usability | The application should be user friendly | 9 |
| NFR002 | Performance | The application must load the page in seconds | 8 |
| NFR003 | Security | The application should have secure user authentication | 8 |
| NFR004 | Compatibility | The application should be compatible with major web browser | 9 |
| NFR005 | Auto readable | OTPs must be auto readable by the application | 6 |
| NFR006 | Session time out | Session time out must be up to 2mins | 7 |
| NFR007 | Refresh | Pages must be refresh after every 30 seconds | 6 |
| NFR008 | Language support | The application should support multiple language | 9 |
| NFR009 | Maintenance | The system shall be easy to update and maintain | 8 |
| NFR010 | Delivery tracker | The delivery tracking should update the order status in real time | 7 |

1. Make wireframe and prototypes





1. Make a note of the Tools, which you are using for above concepts

Balsamiq is a lightweight wireframing tool that emphasizes speed and simplicity, making it ideal for quickly drafting user interface concepts. Its hand-drawn style and intuitive drag-and-drop interface allow designers, product managers, and developers to focus on the overall layout and structure without getting bogged down by fine details. Although it excels in facilitating brainstorming sessions and early-stage prototyping, its low-fidelity nature means that it lacks the advanced interactive features and visual polish required for high-fidelity presentations.

Axure RP is a robust tool designed for creating detailed, interactive prototypes that simulate real-world functionality. It supports dynamic content, conditional logic, and a wide range of built-in widgets, enabling users to develop complex prototypes that can be tested and refined with realistic interactivity. While Axure RP offers extensive capabilities for advanced UI/UX design, it also comes with a steeper learning curve and a higher price point, making it more suited to professional environments and enterprise-level projects where in-depth functionality is crucial.

Microsoft Visio, though primarily known as a diagramming tool, offers versatility through its extensive shape libraries and seamless integration with other Microsoft Office applications. It is commonly used to create flowcharts, network diagrams, and wireframes, allowing users to visualize processes and systems effectively. However, while Visio is powerful for general diagramming and business process visualization, it is not specifically tailored for UI/UX wireframing and may be less intuitive for designers seeking specialized prototyping features.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Req ID** | **Req Name** | **Design** | **D1** | **T1** | **D2** | **T2** | **D3** | **T3** | **D4** | **T4** | **UAT** |
| NFR001 | Usability | Completed | In complete | In complete | Incomplete | Incomplete | Incomplete | Incomplete | Incomplete | Incomplete | Incomplete |
| NFR002 | Performance | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| NFR003 | Security | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| NFR004 | Compatibility | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| NFR005 | Auto readable | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| NFR006 | Session time out | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| NFR007 | Refresh | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| NFR008 | Language support | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| NFR009 | Maintenance | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| NFR010 | Delivery tracker | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |

1. A business analyst’s key responsibilities are to keep track of the requirements and make sure that no requirement is missed. Mr. Henry and peter have approached you regarding the current status of the project. How will you tackle this situation?a

Requirement Traceability Matrix (RTM) - A Requirement Traceability Matrix (RTM) is a document used in business analysis, project management, and software development to ensure that all project requirements are being met throughout the project lifecycle. It helps track the relationship between requirements and various project elements, ensuring that each requirement is linked to corresponding deliverables, test cases, and design specifications

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Req ID** | **Req Name** | **Design** | **D1** | **T1** | **D2** | **T2** | **D3** | **T3** | **D4** | **T4** | **UAT** |
| FR001 | User registration | Completed | In complete | In complete | Incomplete | Incomplete | Incomplete | Incomplete | Incomplete | Incomplete | Incomplete |
| FR002 | User Login | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| FR003 | Uploading product details | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| FR004 | Product search | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| FR005 | Search Filter | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| FR006 | Product details | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| FR007 | Adding product to cart | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| FR008 | Payment method | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| FR009 | Check out process | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |
| FR010 | Order Confirmation | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed | Completed |

1. Prepare 10 Test case document

Test Case Documents - Test Case Documents are structured documents used in software testing to define and outline the specific conditions, steps, and expected results for verifying a particular feature or functionality of a software application. They are essential for ensuring that software meets its requirements and behaves as expected in different scenarios

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | TC01 | | Test case name | User Registration |
| Project ID | PRJ01 | | Project name | Online Agriculture store |
| PM ID | PM001 | | PM name | Mr. Vandana |
| Test Strategy ID | TS001 | | Tester ID | T001 |
| Test Plan ID | TP001 | | Tester name | Ms. Alekya |
| Test schedule ID | TSCH001 | | Date of test | Date to be determined |
| Scenario | Verify that the user can register successfully | | | |
| Link to that page | Registration page | | | |
|  |  | | | |
| Input data | User Credential 1 | | User Credential 2 | User Credential 3 |
| Email: [User1@email.com](mailto:User1@email.com) | | Email: [User2@email.com](mailto:User2@email.com) | Email: [User3@email.com](mailto:User3@email.com) |
| Password: Sha@123 | Password: Sha@1234 | | Password: Sha@1235 |
| Expected behaviour | User is able to register successfully | | | |
| Actual behaviour |  | | | |
| Comments |  | | | |
| Result(pass/Fail) |  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | TC02 | Test case name | User login |
| Project ID | PRJ02 | Project name | Online Agriculture store |
| PM ID | PM002 | PM name | Mr. Vandana |
| Test Strategy ID | TS002 | Tester ID | T002 |
| Test Plan ID | TP002 | Tester name | Mr. John |
| Test schedule ID | TSCH002 | Date of test | Date to be determined |
| Scenario | Verify that the user able to login the application Successfully | | |
| Link to that page | Login Page | | |
| Input data | User Credential 1 | User Credential 2 | User Credential 3 |
| Email: [User1@email.com](mailto:User1@email.com) | Email: [User2@email.com](mailto:User2@email.com) | Email: [User3@email.com](mailto:User3@email.com) |
| Password: Sha@123 | Password: Sha@1234 | Password: Sha@1235 |
| Expected behaviour | User is able to login to the application successfully | | |
| Actual behaviour |  | | |
| Comments |  | | |
| Result(pass/Fail) |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | TC03 | Test case name | Product Uploaded by manufacturer |
| Project ID | PRJ03 | Project name | Online Agriculture store |
| PM ID | PM003 | PM name | Mr. Vandana |
| Test Strategy ID | TS003 | Tester ID | T003 |
| Test Plan ID | TP003 | Tester name | Mr. John |
| Test schedule ID | TSCH003 | Date of test | Date to be determined |
| Scenario | Manufacturer must be able to upload their product on the application | | |
| Link to that page |  | | |
| Input data | Product details available | | |
| Expected behaviour | Manufacturer must be able to upload the details and also the customer must able to see the order and buy | | |
| Actual behaviour |  | | |
| Comments |  | | |
| Result(pass/Fail) |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | TC04 | Test case name | Add to cart |
| Project ID | PRJ04 | Project name | Online Agriculture store |
| PM ID | PM004 | PM name | Mr. Vandana |
| Test Strategy ID | TS004 | Tester ID | T004 |
| Test Plan ID | TP004 | Tester name | Mr. John |
| Test schedule ID | TSCH004 | Date of test | Date to be determined |
| Scenario | User should be able to add selected product to the cart | | |
| Link to that page |  | | |
| Input data | Product ID, Product availability, product details and price | | |
| Expected behaviour | Once login user should be able to search the product and add the required product in the cart for purchase | | |
| Actual behaviour |  | | |
| Comments |  | | |
| Result(pass/Fail) |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | TC05 | Test case name | Payment |
| Project ID | PRJ05 | Project name | Online Agriculture store |
| PM ID | PM005 | PM name | Mr. Vandana |
| Test Strategy ID | TS005 | Tester ID | T005 |
| Test Plan ID | TP005 | Tester name | Mr. John |
| Test schedule ID | TSCH005 | Date of test | Date to be determined |
| Scenario | User must be able to choose the payment option and do the payment | | |
| Link to that page |  | | |
| Input data | Card details, UPI details, payment methods | | |
| Expected behaviour | User should be able to choose the payment from net banking, card or UPI | | |
| Actual behaviour |  | | |
| Comments |  | | |
| Result(pass/Fail) |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | TC06 | Test case name | Order confirmation |
| Project ID | PRJ06 | Project name | Online Agriculture store |
| PM ID | PM006 | PM name | Mr. Vandana |
| Test Strategy ID | TS006 | Tester ID | T006 |
| Test Plan ID | TP006 | Tester name | Mr. John |
| Test schedule ID | TSCH006 | Date of test | Date to be determined |
| Scenario | User should get text or email for the order confirmation | | |
| Link to that page |  | | |
| Input data | Add to the cart and done with the payment | | |
| Expected behaviour | Once the payment is done user should get the detail of the product purchased along with the prices via email or text message | | |
| Actual behaviour |  | | |
| Comments |  | | |
| Result(pass/Fail) |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | TC07 | Test case name | Cancel order |
| Project ID | PRJ07 | Project name | Online Agriculture store |
| PM ID | PM007 | PM name | Mr. Vandana |
| Test Strategy ID | TS007 | Tester ID | T007 |
| Test Plan ID | TP007 | Tester name | Mr. John |
| Test schedule ID | TSCH007 | Date of test | Date to be determined |
| Scenario | User must able to cancel the wrong order | | |
| Link to that page |  | | |
| Input data | User should have purchased the product have order id, payment details | | |
| Expected behaviour | User should be able to cancel the order which is done by mistake from the cancel order page | | |
| Actual behaviour |  | | |
| Comments |  | | |
| Result(pass/Fail) |  | | |

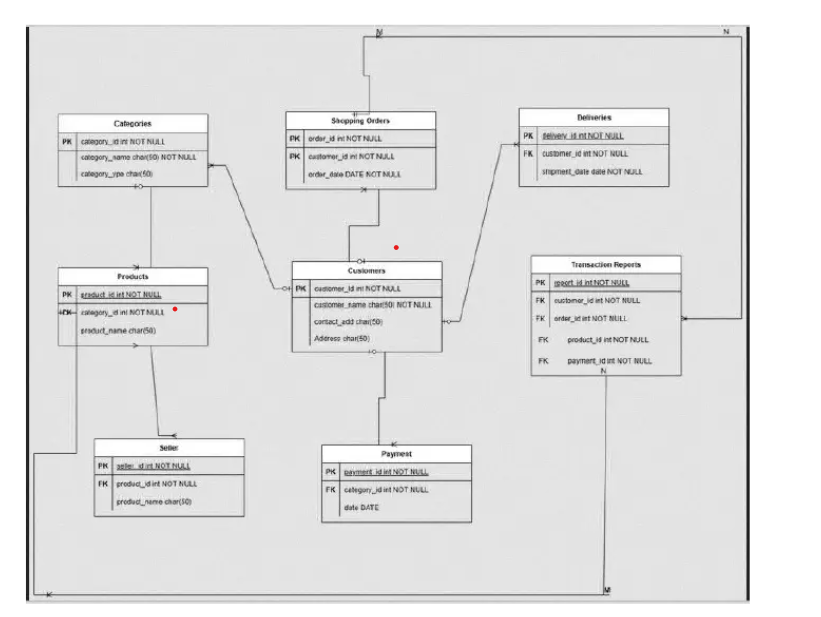
|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | TC08 | Test case name | Return of the order |
| Project ID | PRJ08 | Project name | Online Agriculture store |
| PM ID | PM008 | PM name | Mr. Vandana |
| Test Strategy ID | TS008 | Tester ID | T008 |
| Test Plan ID | TP008 | Tester name | Mr. John |
| Test schedule ID | TSCH008 | Date of test | Date to be determined |
| Scenario | User must be able to return the product which they did not like | | |
| Link to that page |  | | |
| Input data | Product already ordered and return window is open | | |
| Expected behaviour | User must be able to return the product from the return order page | | |
| Actual behaviour |  | | |
| Comments |  | | |
| Result(pass/Fail) |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | TC09 | Test case name | Session out timing |
| Project ID | PRJ09 | Project name | Online Agriculture store |
| PM ID | PM009 | PM name | Mr. Vandana |
| Test Strategy ID | TS009 | Tester ID | T009 |
| Test Plan ID | TP009 | Tester name | Mr. John |
| Test schedule ID | TSCH009 | Date of test | Date to be determined |
| Scenario | Application must be time out after being idle for a certain amount time | | |
| Link to that page |  | | |
| Input data | login id and password | | |
| Expected behaviour | User should be time out after 30 sec of being idle and ask the user id and password to login again | | |
| Actual behaviour |  | | |
| Comments |  | | |
| Result(pass/Fail) |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | TC010 | Test case name | Order tracking process |
| Project ID | PRJ010 | Project name | Online Agriculture store |
| PM ID | PM0010 | PM name | Mr. Vandana |
| Test Strategy ID | TS0010 | Tester ID | T010 |
| Test Plan ID | TP0010 | Tester name | Mr. John |
| Test schedule ID | TSCH0010 | Date of test | Date to be determined |
| Scenario | User must be able to track the product which they have purchased | | |
| Link to that page |  | | |
| Input data | User must have already ordered the product | | |
| Expected behaviour | User must be able to track the order through the application to get the live location of the product | | |
| Actual behaviour |  | | |
| Comments |  | | |
| Result(pass/Fail) |  | | |

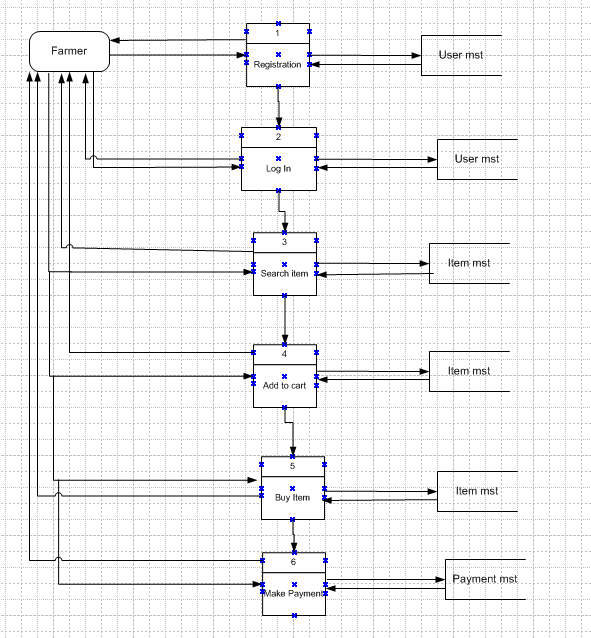
1. DB Design?

* DB Schema – DB Schema is a blueprint that outlines the structure of the database, including its tables, fields, relationships, constraints and other characteristics.
* Entity Relationship Diagram – An Entity Relationship Diagram is a visual representation of the relationships between entities in a database. It depicts the entities (tables), attributes (properties/fields) and relationship between them



1. What is a data flow diagram? Draw a data flow diagram to represent the in-flow and out-flow of data when a Farmer is placing an order for the product

The data flow diagram is a graphical representation of the flow of data within the system. It visually shows how data moves from one process to another, how it stored and where it ends up. It help analyst and designer to understand the flow of data within the system, identify potential bottlenecks or inefficiency and communicate system requirement to stakeholders



1. 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
* 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.
* 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.
* 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.
* 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.
* Change Approval: Obtain formal approval from the appropriate stakeholders, such as project sponsors or change control boards. Ensure that all stakeholders agree regarding the change and its implications.
* 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.
* 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.
* 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.
* 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

1. Change Request Vs an Enhancement? As the project is in process, Ben and Kevin have contacted you. The reason is to inform you that they want the Farmers to sell their crop yields through this application i.e. Farmers should be able to add their crop yields or products and display to general public and should be able to sell them. They also want to introduce Auction system for their Crop yields. As a BA, what will be your response? Is this a change request or enhancement

* Change Requests - A Change Request (CR) is a formal proposal to modify a project, product, process, or system. It is typically used in project management, software development, and various industries to handle modifications and improvements. A Change Request can arise at any point during a project's lifecycle, whether it's due to new requirements, unforeseen issues, or the need for enhancements.

Enhancement - Enhancements refer to improvements or additions made to the software after its initial release. These changes are aimed at increasing the software’s value, performance, or functionality based on evolving business needs, user feedback, or technological advances. Enhancements can be part of the regular software maintenance cycle or occur as part of a new version or update.

As a Business Analyst, my response to Ben and Kevin's request would be to classify it as an enhancement rather than a change request. A change request typically involves modifications to existing functionality or requirements, while an enhancement introduces new features or capabilities that were not initially specified. 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

1. Come up with estimations – How many Manhours required

* Estimation in Software Development Life Cycle refers to the process of predicting the resources (time, cost, effort) required to complete a software development project or specific tasks within that project. Estimation is acritical part of the planning phase of SDLC, as it helps in setting realistic expectations, managing resources efficiently, and determining the project's feasibility.

Man-hours estimation in software development refers to the process of estimating the total amount of human effort required to complete a software development task, project, or feature. It is typically expressed in terms of the number of hours that a team or individual will spend working on the project. This estimation is crucial for planning, budgeting, and scheduling the project. It helps determine resource requirements, assess project timelines, and evaluate project feasibility.,

Types of projects on the basis of Man-Hours:

1.Small project – Upto 500 Man-Hours

2.Medium project – Upto 1000 Man-Hours

3.Large project – Upto 1500 Man-Hours

Manhours Required = Total hours working per day x Total number of members x Total numbers of days worked over the specific period.

Number of Working Hours a day = 8 hours

Number of Resources = 12Time period provided = 18 months = 547 days = 78 weeks (Including Weekends and Public Holidays)

Assuming Weekends = 156Assuming Public Holidays = 10Total = 166

Total working days = 547-166 = 381 working days

Hence, Estimated Manhours = 8 hours \* 12 resources \* 381 days =36,576 Manhours required.

1. Project has finally completed all the stages i.e., design, development, testing etc. Now, it is the role of a business analyst to contact the client for testing of the final product and have to successfully complete it. How are you going to handle this situation? And once it is done, what will be the process to close the project? Explain UAT Acceptance process

* User Acceptance Testing (UAT) is the final phase of software testing, where the end-users validate whether the system meets their business requirements before it goes live. This testing ensures that the application is user-friendly, functional, and aligns with real-world scenarios. UAT is conducted in a production-like environment with real or sample data to simulate actual user interactions. If the system passes UAT, it is considered ready for deployment.
* **Planning & Preparation**  
  The UAT process begins with planning, where the objectives, scope, and acceptance criteria are defined. Business analysts, QA teams, and end-users collaborate to create a structured approach for testing. Test cases and scenarios are designed based on business requirements to ensure comprehensive coverage. Additionally, the UAT environment is set up, including required data, system configurations, and access permissions, to ensure a seamless testing process.
* **Test Execution**  
  During this phase, end-users and business stakeholders execute test cases to verify the system’s functionality, usability, and compliance with business needs. Testers follow predefined scenarios and record their observations. If any issues arise, they document them in a structured format, including steps to reproduce and expected vs. actual results. The focus here is to validate real-world business processes rather than technical aspects.
* **Defect Reporting & Resolution**  
  All defects identified during UAT are reported in a defect tracking system. Each issue is categorized based on severity and priority, and the development team works on necessary fixes. Once the defects are resolved, testers retest the system to confirm that the issues have been properly addressed. This iterative process continues until all critical defects are resolved, ensuring a smooth user experience.
* **Sign-Off & Acceptance**  
  Once all major issues are fixed, the business team reviews the test results and assesses whether the system meets the acceptance criteria. If the system is found to be satisfactory, key stakeholders provide formal approval through a UAT sign-off document. This approval indicates that the application is ready for production deployment. Any pending minor issues are documented for future enhancements or post-deployment fixes.

1. Explain Project closure document

* A Project Closure Document is a formal report that signifies the completion of a project. It serves as the final review and approval before officially closing the project, ensuring that all deliverables have been met, stakeholders are satisfied, and lessons learned are documented for future reference.

Points to be included in the project closure document are project overview, Achievements, lesson learned, Quality assurance, Resource utilization, Risk management and challenges

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Points to be included** | **Details** | **Reference link** |
| **1** | **Did the client sign off on UAT testing** |  |  |
| Date of Sign off | 16/02/2025 |  |
| Name of the resources |  |  |
| **2** | **Objectives of the project** |  |  |
| User Friendliness | Achieved |  |
| Customer Satisfaction | ROI in 6 to 12 months |  |
| More category | Achieved |  |
| **3** | **Functionalities worked on** |  |  |
| Secured payment process | Achieved |  |
| Categories | Achieved |  |
| **4** | **Infrastructure** |  |  |
| Software installed | Achieved |  |
| Laptop purchased | Achieved |  |
| **5** | **Funding** |  |  |
| Amount approved | 2 Crore |  |
| Amount used | 2 Crore |  |
| **6** | **Overall project information** |  |  |
| Escalation | 30 |  |
| Customer Satisfaction | High |  |
| **7** | **Value of the company** |  |  |
| Positive/Negative | Company has gained successful integration of process, increase turn over by 25% and efficiency by 20% |  |