**Online Agriculture Products Store**

**Question 1**. **BPM** : Identify Business Process Model for Online Agriculture Store – (Goal, Inputs, Resources, Outputs, Activities, Value created to the end Customer)

**Ans**-

**Goal**:-

1. To facilitate the farmers to buy seeds, pesticides, and fertilizers from anywhere through internet connectivity.
2. Farmers and manufacturing companies can connect directly with each other using online web application with ease and by selling products manufacturing company can make money.

**Inputs:**

1. Get Product details from the manufacturers and display them to the farmers through the application.
2. Farmers will browse through these agriculture products.

**Resources**: People (Peter, Kevin and Ben), Mobile, Internet.

**Outputs**-

1. Farmers and Companies (Fertilizers, seeds and pesticides manufacturing Companies) can communicate directly with each other using online web application.
2. Using online web application facilitate farmers to buy seeds, pesticides, and fertilizers from anywhere through internet connectivity.

**Activities**-

1. Accept product details from manufactures.
2. Display those product details on online web application.
3. Farmers will browse through these products.
4. Farmers will select the products what they need and request to buy them.
5. Deliver the requested products to farmers location.

**Value Created-**

1. Free access to browse agriculture products.
2. Direct communication between farmers and companies (Fertilizers, seeds and pesticides manufacturing companies).

**Question 2** – Mr. Karthik is doing SWOT analysis before he accepts this project. What Aspects he Should consider as Strengths, as Weaknesses, as Opportunity and as Threats.

**Ans**-

**Strengths**- To get the project and complete it within given time and budget in unique manner. Skilled technical people.

**Weaknesses**- Unavailability of resources

**Opportunity**- If this type of online web/Mobile application is launching in the market for the first time then its good opportunity to create more such applications in market and it will help in enhancements of the IT company.

**Threats**-

1. Poor Network Connectivity.
2. Insufficient Budget.
3. Farmers not able to use Smart phones.
4. Someone launches same application before we launch our application.

**Question 3** – Feasibility study

Mr. Karthik is trying to do feasibility study on doing this project in Technology (Java), Please help him with points (HW SW Trained Resources Budget Time frame) to consider in feasibility Study.

**Ans**-

1. Hardware- 10 Laptops or Desktops
2. Software- Java software installed on 5 machines, testing tools installed on 3 machines, Documentation and UML tools installed on 2 machines.
3. Trained Resources- 3 trained Java software developer, 1 tester.
4. Budget- 1.5 Crores
5. Time Frame- 16 Months

**Question 4** – Gap Analysis

Mr. Karthik must submit Gap Analysis to Mr. Henry to convince to initiate this project. What points (compare AS-IS existing process with TO-BE future Process) to showcase in the GAP Analysis

**Ans**-

**AS-IS**-

 Currently there is no online web application available in the market to buy agriculture products online with ease for farmers at remote area.

**TO-BE-**

1. Web application should able to accept agriculture product details from manufactures.
2. Web application should able to display those product details and farmers should able to browse them.
3. Farmers should be able to select the products as per their need and request to buy them through web application.
4. Web application should able to deliver the requested products to farmers on their location.

**Question 5** – Risk Analysis

List down different risk factors that may be involved (BA Risks And process/Project Risks.

**Ans**-

**BA Risks-**

1. Incomplete gathering of requirement -If BA fails to collect business requirements in proper manner, then all business requirements will not get collected and it is one of the hugest risks.
2. Failure at conducting meetings- If BA fails to schedule various meetings with business stakeholders as well as technical team, then there will be lack of communication and information. Effective communication is the most important way for success of any project.
3. Lack of Communication about change In requirement (if applicable) : If the BA fails to communicate and document from time to time with the technical team regarding requirement changes that will be a risk.

**Project Risks-**

1. Unavailability of Resources- If resources are not available, then it is difficult to complete project withing given timeframe.
2. Insufficient Budget-If we run out of budget, then it is difficult to complete the project within sanctioned budget of project. This is most crucial part of project risk.
3. Unavailability of Higher Authorities- If higher authorities are not available on time to guide team or to take any decision then this might also consider as a risk.

**Question 6** – Stakeholder Analysis (RACI Matrix)

Perform stakeholder analysis (RACI Matrix) to find out the key stakeholders who can take Decisions and Who are the influencers.

**Ans-**

**RACI Matrix-**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activities** | **BA** | **UI****Designer** | **Project Manager** | **SME** | **Developer** | **Tester** | **Client** | **NW****Admin** | **DB****Admin** | **Operation****Team/Production Team** |
| Planning & Requirement Gathering | R |  | A | C | I | I | I | R | R |  |
| Designing | R | R | A | C | I |  | I | R | R |  |
| Prototyping |  | R | A |  | I |  | I |  |  |  |
| Coding |  |  | A |  | R | I | I |  |  |  |
| Testing | R |  | A |  | I | R | I |  |  |  |
| Deployment |  |  | A |  |  |  | I |  |  |  |
| Maintenance |  |  | A |  |  |  | I |  |  | R |

**Question 7** – Business Case Document

Help Mr. Karthik to prepare a business case document

**Ans-**

Business Case will have following information

1. Why is this project initiated?

The main purpose to initiate this project is to facilitate farmers to buy seeds, pesticides, and fertilizers from anywhere through internet connectivity.

1. What are the current problems?
* There is no online agriculture products store to buy seeds, fertilizers and pesticides online with ease.
* Remote area farmers are facing issues like procuring fertilizers, buying seeds, lack of pesticides.
1. With this project how many problems could be solved?
* This online agriculture product store will be able to accept the product (fertilizers, seeds, pesticides) details from the manufacturers and should be able to display them to the Farmers.
* Farmers will browse through these products and select the products what they need and request to buy them and deliver them to farmers location.
* So, using this web application we will able to solve various issues of farmers like procuring fertilizers, buying seeds, lack of pesticides.
* All agriculture products will be delivered to farmers doorstep.
1. What are the resources required?
* Stakeholders to share requirements for the project
* Project Manager
* Java Developers Team
* Testing Team
* Subject Matter Experts (SME’s)
* Network Admin and DB Admin
* Operational Team
* BA
1. How much organizational change is required to adopt this technology?
* There will be no organizational change required to adopt this technology, because this web application will build from scratch. There is no existing system/app in this scenario.
1. How to identify stakeholders?
* Find Stakeholders who have information regarding the current issues faced by remote area farmers so they can help to suggest solution and will contribute more in the project.
* Find stakeholders who have already worked with similar issues.
* Find stakeholders who are passionate and enthusiastic to deal with farmers issues.

**Question 8** – Four SDLC Methodologies

The Committee of Mr. Henry, Mr. Pandu, and Mr. Dooku and Mr. Karthik are having a discussion on Project Development Approach. Mr. Karthik explained to Mr. Henry about SDLC. And four methodologies like Sequential Iterative Evolutionary and Agile. Please share your thoughts and clarity on Methodologies.

**Ans-**

**1) Sequential**-

1. Sequential model is one of the simple and easy to understand model of SDLC.
2. It is very easy to use as compared to other models of SDLC.
3. This model is suitable for smaller projects where duration is less and requirements are clearly understood.
4. In sequential model, there are 7 phases and each phase must be completed before beginning of next phase.
5. This model contains Requirement Gathering, Analysis, Design, Implementation, Testing, Deployment, Maintenance phase.
6. At the end of each phase review process takes place to determine project is on right path or not.
7. This model is easy to manage due to rigidity because each phase has specific deliverables and a review process.
8. Only one phase processed and completed at a time.
9. This model doesn’t work well with large and complex projects where requirements are changing.
10. In this model working software is produced very late during the life cycle.
11. Adjusting scope during the life cycle can kill a project.
12. There is high amount of risk and uncertainty in this type of model.

**2) Iterative**-

1. Iterative model is one of the easiest to implement SDLC models.
2. This model is started based on initial requirements and first version of software is released.
3. More features are added to the base software product with ongoing iterations until final product is created.
4. Phases in Iterative Model-
* Requirements Phase: In the requirements phase of software development, the system related information is gathered and analyzed.
* Design Phase: In the Design phase, the software solution is prepared to meet the necessities for the design. The system design may be a new one or the extension of a previous build iteration.
* Implementation and Test: In the implementation as well as a test phase, the system is developed by coding and building the user interface and modules which are then incorporated and tested.
* Review Phase: The review phase is where the software is estimated and checked as per the current requirement. Then, further requirements are reviewed discussed and reviewed to propose for an update in the next iteration.
1. Using this model, system requirements can be classified and understood well.
2. Primary necessities of the system can be defined.
3. This model is also useful when there are high risks in the system characteristic and goals.
4. Situations where resources with required skill sets are not accessible, and the system needs to be developed on a contract basis, choosing this model is a suitable decision.
5. Produces working software rapidly and before time throughout the SDLC.

**3) Evolutionary-**

1. Evolutionary model is combination of incremental, iterative and prototyping model.
2. In this model the requirement is broken down into different functional units.
3. These functional units can also be referred to as module. These modules can be incrementally built and delivered.
4. Here at the beginning the core module of the software product is developed.
5. New functionality is built, added to existing one and released as new version.
6. Each successive version is capable of performing more functions in comparison to its previous versions.
7. It is a very suitable model because of user feedback and other factors that make the model very suitable for the development of complete Software.
8. The user feedback is very helpful for the development of the next stage because after the completion of one stage we get the feedback to the user, the user feedback is very essential for the development of the next phase.
9. If you want any changes in the software requirements, all changes will be applied within a time.
10. It is very useful in a large project where you can easily find a module for step-by-step implementation. The evolutionary model is used when the users need to start using the many features instead of waiting for the complete software.
11. The evolutionary model is also very useful in object-oriented software development because all the development is divided into different units.
12. There are fewer chances of errors because all the modules are well seen.
13. It helps to reduce the risk of software project and cost of development.
14. Minimize serious problems during testing.
15. The delivery of full software can be late due to different changes by customers during development.
16. It is difficult to divide the problem into several parts, that would be acceptable to the customer which can be incrementally implemented and delivered.

**4) Agile-**

1. The Agile methodology is a practice that encourages continuous development and testing throughout the software development lifecycle of a project.
2. Agile methodologies attempt to produce the proper product through small cross-functional self-organizing teams that produce small pieces of functionality on a regular basis, allowing for frequent customer input and course correction as needed.
3. Agile development is not that tough when broken down to its core concepts. While the number of meetings involved may appear unnecessary, it saves a lot of time by optimizing development tasks and avoiding the errors that can occur during the planning stages.
4. Four main values-
* Individuals and interactions over processes and tools.
* Working software over comprehensive documentation.
* Customer collaboration over contract negotiation.
* Responding to change over following a plan.

     E. Twelve Principles of Agile

* Satisfy the customer through early and continuous delivery of valuable software.
* Welcome changing requirements, even late in development. Agile processes harness change for customer's competitive advantage.
* Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
* Business people and developers must work together daily throughout the project.
* Build projects around motivated individuals.
* The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
* Working software is the primary measure of progress.
* Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace.
* Continuous attention to technical excellence and good design enhances agility.
* Simplicity-the art of maximizing the amount of work not done is essential.
* The best architectures, requirements, and designs emerge from self-organizing teams.

At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly

**Question 9** – Waterfall RUP Spiral and Scrum Models

They discussed models in SDLC like waterfall RUP Spiral and Scrum. You put forth you're understanding on these models

**Ans-**

**1) Waterfall Model-**

1. Waterfall Model is also referred to as a linear-sequential life cycle model. It is very simple to understand and use.
2. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.
3. Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project.
4. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.
5. The sequential phases in Waterfall model are −
* **Requirement Gathering and analysis** − All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
* **System Design** − The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
* **Implementation** − With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
* **Integration and Testing** − All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
* **Deployment of system** − Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
* **Maintenance** − There are some issues which come up in the client environment. To fix those issues, patches are released. Also, to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.
1. Advantages of the Waterfall Model are as follows −
* Simple and easy to understand and use
* Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a review process.
* Phases are processed and completed one at a time.
* Works well for smaller projects where requirements are very well understood.
* Clearly defined stages.
* Process and results are well documented.
1. Disadvantages of the Waterfall Model are as follows −
* No working software is produced until late during the life cycle.
* High amounts of risk and uncertainty.
* Not a good model for complex and object-oriented projects.
* Poor model for long and ongoing projects.
* Not suitable for the projects where requirements are at a moderate to high risk of changing. So, risk and uncertainty are high with this process model.
* It is difficult to measure progress within stages.
* Cannot accommodate changing requirements.
* Adjusting scope during the life cycle can kill the project.

**2) RUP (Rational Unified Process)-**

1. The Rational Unified Process (RUP) is an iterative software development process framework created by the Rational Software Corporation, which was acquired by IBM in February 2003.
2. RUP is based on a set of building blocks, or content elements, describing what is to be produced, the necessary skills required and the step-by-step explanation describing how specific development goals are to be achieved. The main building blocks, or content elements, are the following:
3. Roles (who) -A Role defines a set of related skills, competencies and responsibilities.
4. Work Products (what) -A Work Product represents something resulting from a task, including all the documents and models produced while working through the process.
5. Tasks (how) - A Task describes a unit of work assigned to a Role that provides a meaningful result.
6. Within each iteration, the tasks are categorized into nine disciplines:

Six "engineering disciplines"-

* Business Modeling
* Requirements
* Analysis and Design
* Implementation
* Test
* Deployment

Three supporting disciplines -

* Configuration and Change Management
* Project Management
* Environment
1. Four Project life Cycle Phases
* Inception: agreement among the team and customer as to what will be built
* Elaboration: agreement within the team as to the architecture and design needed to deliver the agreed system behavior
* Construction: the iterative implementation of a fully functional system
* Transition: delivery, defect correction, and tuning to ensure customer acceptance
1. Six best practices
* Develop iteratively, with risk as the primary iteration driver
* Manage requirements
* Employ a component-based architecture
* Model software visually
* Continuously verify quality
* Control changes

**3) Spiral Model-**

1. The spiral model gives more emphases placed on risk analysis.
2. The spiral model has four phases: Planning, Risk Analysis, Engineering and Evaluation. A software project repeatedly passes through these phases in iterations (called Spirals in this model). The baseline spiral, starting in the planning phase, requirements are gathered, and risk is assessed. Each subsequent spiral builds on the baseline spiral.
3. Requirements are gathered during the planning phase. In the risk analysis phase, a process is undertaken to identify risk and alternate solutions. A prototype is produced at the end of the risk analysis phase.
4. Software is produced in the engineering phase, along with testing at the end of the phase. The evaluation phase allows the customer to evaluate the output of the project to date before the project continues to the next spiral.
5. In the spiral model, the angular component represents progress, and the radius of the spiral represents cost.
6. Advantages-
* High amount of risk analysis.
* Good for large and mission-critical projects.
* Software is produced early in the software life cycle.

G. Disadvantages

* Can be a costly model to use.
* Risk analysis requires highly specific expertise.
* Project's success is highly dependent on the risk analysis phase.
* Doesn't work well for smaller projects.

**4) Scrum Model-**

1. Scrum can be implemented either at the beginning of the project or when you sense that project is falling behind schedule. This model exercises full Admin Power.
2. **Scrum Team**: Project resources are grouped as Scrum teams which comprises of BAs, Developers, Testers. Each Team size will on average be 7-8.
3. **Scrum Master**: He will monitor the performance of the team within the sprint. Team will raise all their issues to scrum master and he will run to look for answers. This role can be played by any person in team normally BA's plays this role.
4. **Product burndown**: It shows how much work was left to do at the beginning of each sprint.
5. **Sprint:** This is the period that team decides to deliver their objective. Normally a sprint period will before 2 weeks but may extend to 4 weeks
6. **Meetings**:
* Sprint Planning Meeting -This happens at the beginning of each sprint and team decides on what they will be delivering in the sprint.
* Daily Scrum Meeting - This happens each day where team will just answer 3 questions:
* What did u do today?
* what will u do tomorrow?
* Are there any impediments that is slowing or stopping u?
* Sprint Review Meeting - This happens at the end of the sprint where team will demo the completed stories to product owner and get it cleared.
* Sprint Retrospective Meeting - This happens at the end of the sprint where team will answer these 3 questions:
* What went well in the sprint?
* what did not go well?
* What are the required areas of improvements in next sprint?

  **5. V Model**

1. V-Model also referred to as the Verification and Validation Model. In this, each phase of SDLC must complete before the next phase starts. It follows a sequential design process same as the waterfall model. Testing of the device is planned in parallel with a corresponding stage of development.
2. **Verification:** It involves a static analysis method (review) done without executing code. It is the process of evaluation of the product development process to find whether specified requirements meet.
3. **Validation:** It involves dynamic analysis method (functional, non-functional), testing is done by executing code. Validation is the process to classify the software after the completion of the development process to determine whether the software meets the customer expectations and requirements.
4. So, V-Model contains Verification phases on one side of the Validation phases on the other side. Verification and Validation process is joined by coding phase in V-shape. Thus, it is known as V-Model.

 When the APT IT SOLUTIONS company got the project to make this online agriculture product store, there is a difference of opinion between a couple of SMEs and the project team regarding which methodology would be more suitable for this project. SMEs are stressing on using the V model and the project team is leaning more onto the side of waterfall model. As a business analyst, which methodology do you think would be better for this project?

**Ans-** V Model

**Question 10** – Waterfall Vs V-Model

Write down the differences between waterfall model and V model.

**Ans-**

|  |  |
| --- | --- |
| **Waterfall Model**  | **V Model**  |
| The cost of Waterfall model is low.  | V-model is expensive.  |
| Simplicity of Waterfall model is simple.  | Simplicity of V-model is Intermediate  |
| Flexibility of Waterfall model is Rigid.  | Flexibility of V-model is Little flexible.  |
| Waterfall model is a sequential execution process.  | It is also a sequential execution process.  |
| Waterfall model’s steps move in a linear way.  | V-model’s steps don’t move in linear way  |
| Re-usability of Waterfall model is Limited.  | V-model can be Re-use for some extent.  |
| User involvement in Waterfall model is only in beginning.  | User involvement in V-model is also only in beginning.  |
| In Waterfall model testing activities start after the development activities are over.  | In V-model testing activities start with the first stage.  |
| Guarantee of success through Waterfall model is low.  | Guarantee of success through V-model is high.  |
| Waterfall model is a continuous process.  | V-model is a simultaneous process.  |
| Software made using Waterfall model, the number of defects is less in comparison of software made using V-model.  | Software made using V-model, the number of defects is greater in comparison of software made using Waterfall model.  |
| Requirement specification in Waterfall model is necessary in beginning.  | Requirement specification in V-model is also necessary in beginning.  |
| Waterfall model is less used now-a-days in software engineering.   | V-model is widely used in software engineering.  |

**Question 11** – Justify your choice

As a BA, state your reason for choosing one model for this project

**Ans-**

As a BA, I will refer to choose V Model considering following points-

1. V-Model also referred to as the Verification and Validation Model. In this, each phase of SDLC must complete before the next phase starts. It follows a sequential design process same as the waterfall model. Testing of the device is planned in parallel with a corresponding stage of development.
2. **Verification:** It involves a static analysis method (review) done without executing code. It is the process of evaluation of the product development process to find whether specified requirements meet.
3. **Validation:** It involves dynamic analysis method (functional, non-functional), testing is done by executing code. Validation is the process to classify the software after the completion of the development process to determine whether the software meets the customer expectations and requirements.
4. So, V-Model contains Verification phases on one side of the Validation phases on the other side. Verification and Validation process is joined by coding phase in V-shape. Thus, it is known as V-Model.
5. There are the various phases of Verification Phase of V-model:
* **Business requirement analysis:** This is the first step where product requirements understood from the customer's side. This phase contains detailed communication to understand customer's expectations and exact requirements.
* **System Design:** In this stage system engineers analyze and interpret the business of the proposed system by studying the user requirements document.
* **Architecture Design:** The baseline in selecting the architecture is that it should understand all which typically consists of the list of modules, brief functionality of each module, their interface relationships, dependencies, database tables, architecture diagrams, technology detail, etc. The integration testing model is carried out in a particular phase.
* **Module Design:** In the module design phase, the system breaks down into small modules. The detailed design of the modules is specified, which is known as Low-Level Design
* **Coding Phase:** After designing, the coding phase is started. Based on the requirements, a suitable programming language is decided. There are some guidelines and standards for coding. Before checking in the repository, the final build is optimized for better performance, and the code goes through many code reviews to check the performance.
1. There are the various phases of Validation Phase of V-model:
* **Unit Testing:** In the V-Model, Unit Test Plans (UTPs) are developed during the module design phase. These UTPs are executed to eliminate errors at code level or unit level. A unit is the smallest entity which can independently exist, e.g., a program module. Unit testing verifies that the smallest entity can function correctly when isolated from the rest of the codes/ units.
* **Integration Testing:** Integration Test Plans are developed during the Architectural Design Phase. These tests verify that groups created and tested independently can coexist and communicate among themselves.
* **System Testing:** System Tests Plans are developed during System Design Phase. Unlike Unit and Integration Test Plans, System Tests Plans are composed by the client's business team. System Test ensures that expectations from an application developer are met.
* **Acceptance Testing:** Acceptance testing is related to the business requirement analysis part. It includes testing the software product in user atmosphere. Acceptance tests reveal the compatibility problems with the different systems, which is available within the user atmosphere. It conjointly discovers the non-functional problems like load and performance defects within the real user atmosphere.
1. Advantages of V-Model:
* Easy to Understand.
* Testing Methods like planning, test designing happens well before coding.
* This saves a lot of time. Hence a higher chance of success over the waterfall model.
* Avoids the downward flow of the defects.
* Works well for small plans where requirements are easily understood.

**Question 12** – Gantt Chart

The Committee of Mr. Henry, Mr. Pandu, and Mr. Dooku discussed with Mr. Karthik and finalized on the V Model approach (RG, RA, Design, D1, T1, D2, T2, D3, T3, D4, T4 and UAT) Mr. Vandanam is mapped as a PM to this project. He studies this Project and Prepares a Gantt chart with V Model (RG, RA, Design, D1, T1, D2, T2, D3, T3, D4, T4 and UAT) as development process and the Resources are PM, BA, Java Developers, testers, DB Admin, NW Admin.

**Ans-**



**Question 13** – Fixed Bid Vs Billing

The Committee of Mr. Henry, Mr.Pandu and Mr.Dooku is now discussing about the funds and how to release the funds for development. They were studying Fixed Bid model and Billing Model. Share your knowledge on Fixed Bid model and Billing Model.

**Ans-**

**A. Fixed Bid Model-**

* A fixed-price contract is a type of contract wherein the payment does not depend on the resources or the time spent. It involves setting fixed price for the product, service or result defined in the contract.
* There are milestones in this project.
* We can track this projects progress with the help of milestones.
* Fixed Bid models are most suitable for long-term projects and those that have a high value to the client organization.

**B. Billing Model-**

* The Billing model works on a completely different principle than the Fixed-Price model. In this model, rather than pay a fixed sum right at the start, you pay the software team for the hours of work needed to finish a given project and for all of the materials they use.
* This type of cooperation model is useful when you cannot accurately estimate how much the project will cost or how long it will take to complete, so there’s no set price or rigid deadlines for the team.
* There are no milestones in this type of model.
* We can track each and everyday work in this model with the help of time and work.

**Question 14,15,16,17,18,19,20** – Timesheets

Please share Sample Timesheets of a BA in various SDLC Stages RG, RA, Design, D1, T1, D2,T2, D3, T3, D4, T4 and UAT, Deployment n Implementation

|  |
| --- |
| **Design Timesheet of a BA** |
|  |  |  |
| **Task** | **Estimated Time** | **Description** |
| Requirement Gathering & Analysis | 8 hours | Review and clarify requirements with stakeholders to ensure full understanding before passing them to the development team. |
| UI/UX Design Inputs | 6 hours | Provide feedback on wireframes, user interfaces, and user experience designs to ensure they meet business requirements. |
| Functional Design Specification | 10 hours | Create or review functional design documents and work closely with the development team to ensure the technical design aligns with the requirements. |
| Collaboration with Developers/Designers | 12 hours | Work with Java developers and designers to ensure that the system architecture and design align with business objectives. |
| Review of Design Artifacts | 6 hours | Review design documents (system architecture, database design, etc.) to ensure business goals are met. |
| Documentation | 8 hours | Finalize design documents and prepare a Business Requirement Document (BRD) or Functional Specification Document (FSD). |
| **Total Time for Design Phase** | **50 hours** |   |

|  |
| --- |
| **Development Timesheet of a BA** |
|  |  |  |
| **Task** | **Estimated Time** | **Description** |
| Requirement Clarification with Developers | 10 hours | Help developers understand the requirements, provide clarifications on business rules, use cases, and expected behavior. |
| User Stories/Use Case Review | 8 hours | Review and ensure that user stories and use cases are being properly implemented and aligned with the business needs. |
| Review of Sprint/Daily Standup calls. | 12 hours | Participate in daily standups or sprint meetings to track development progress and provide necessary updates or clarifications. |
| Feedback on Features/Modules | 8 hours | Provide feedback on developed features, ensuring they align with the requirements. |
| Coordination with Project Manager | 6 hours | Ensure that the development progress is on track, and report any deviations to the PM, especially if the business scope changes. |
| Documentation for Updates | 4 hours | Update business-related documentation to reflect any changes in requirements due to the development phase. |
| **Total for Development Phase** | **48 hours** |   |

|  |
| --- |
| **Testing Timesheet of a BA** |
|  |  |  |
| **Task** | **Estimated Time** | **Description** |
| Test Case Review & Approval | 10 hours | Review and approve test cases to ensure that they properly validate the system against the business requirements. |
| Functional Testing Assistance | 8 hours | Assist testers in functional testing by answering questions and verifying that the features meet the business objectives. |
| Participate in Test Execution | 12 hours | Participate in executing tests and documenting any business-related bugs or issues that arise. |
| Review Test Results | 8 hours | Review and analyze the results of testing to confirm that all business requirements have been satisfied. |
| Issue Reporting & Follow-up | 8 hours | Report bugs/issues discovered during testing and follow up with developers to ensure they are resolved. |
| Final Testing Feedback | 6 hours | Provide final feedback on test results to ensure the product aligns with business needs and that all features are working as expected. |
| **Total for Testing Phase** | **52 hours** |   |

|  |
| --- |
| **UAT Timesheet of a BA** |
|  |  |  |
| **Task** | **Estimated Time** | **Description** |
| UAT Planning | 10 hours | Define the scope of UAT, identify key stakeholders (farmers, manufacturers), and create UAT plans based on business requirements. |
| Coordinate with Stakeholders for UAT | 12 hours | Coordinate with stakeholders (Peter, Kevin, Ben) to ensure they understand UAT goals and their role in testing the system. |
| UAT Test Case Review & Approval | 8 hours | Review UAT test cases created by testers to ensure they reflect business requirements accurately. |
| Conduct UAT | 10 hours | Facilitate UAT by assisting stakeholders with testing, answering questions, and ensuring the system aligns with user expectations. |
| Collect Feedback from Stakeholders | 6 hours | Gather feedback from farmers and manufacturers regarding their experience during UAT, documenting issues and areas for improvement. |
| Issue Resolution Follow-up | 8 hours | Track issues raised during UAT, follow up with developers to ensure timely resolution, and communicate with stakeholders on the fixes. |
| Final UAT Sign-off | 4 hours | Ensure that all critical issues are addressed and obtain final sign-off from stakeholders that the system meets their needs. |
| **Total for UAT Phase** | **58 hours** |   |

|  |
| --- |
| **Deployment & Implementation Timesheet of a BA** |
|  |  |  |
| **Task** | **Estimated Time** | **Description** |
| Deployment Planning | 8 hours | Coordinate with the project manager and technical team to plan for deployment, ensuring all business requirements are covered in the final deployment. |
| User Training Materials | 10 hours | Create or review user manuals and training materials to help farmers and manufacturers get acquainted with the platform. |
| End-User Support during Deployment | 12 hours | Assist in troubleshooting and resolving issues as end-users start using the system in a live environment. |
| Stakeholder Communication | 6 hours | Communicate with stakeholders about deployment schedules, changes, and the transition process. |
| Post-Deployment Monitoring | 6 hours | Monitor the deployment to ensure the system is working as expected and all business goals are met after going live. |
| Final Documentation | 8 hours | Prepare final documentation on the project, including lessons learned, post-implementation reviews, and feedback from users. |
| **Total for Deployment & Implementation** | **50 hours** |   |