**QUESTION-1**

**BUSINEE PROCESS MODELING**

**GOAL**- Online app to bridge farmers and manufacturers, enabling farmers to purchase products easily.

**INPUTS**- Customer data, customer address, product choice of customer, payment data, agricultural companies’ data, product details,

**RESOURCES**- Internet connection, Mobile/system, partnership with manufacture company, product inventory (pesticides,seeds,fertilizer),delivery team, project development team

**OUTPUTS**- User friendly application. Delivery of products to buyers, sales revenue.

**ACTIVITIES**- Receiving of orders, notify customer,validate payment details, shipping of orders.

**VALUE**- Customer satisfaction, on time delivery, efficiency, quality and budget friendly product.

**QUESTION-2**

**SWOT ANALYSIS**

**Strength**- Innovative idea as for rural farmers, direct interaction between buyers and sellers without intermediary.

**Weakness**- poor internet connectivity in some rural areas, dependency on manufacturer, inventory issues, technical illiteracy among the farmers, lack of competitive strength.

**Opportunity**- Expanding product range, expanding regional range, educational features.

**Threats**-**Technological Risks**: Potential technical issues, bugs, or cybersecurity threats could disrupt the platform.

**Economic Factors**: Rising costs of agricultural products or delivery services could impact affordability for farmers.

**QUESTION -3**

**FEASIBILITY STUDY**- It is an analysis and evaluation of a proposed project or system to determine whether it is technically, economically and operationally viable. It helps decision- makers understand if a project is practical, achievable and worth invsting it.

**The feasibility for this project includes**

1. Hardware feasibity- servers, networks infrastructure of high speed and reliable connectivity, VPN for secure access.
2. Software Feasibility- development tools, like java, testing tools, database tools like MySQL.
3. Human resource
4. Breakdown of project budget
5. Time frame

**QUESTION-4**

**GAP ANALYSIS**- It is a comparison of the current state and the desired future state of an organization in order to identify difference that needs to be addressed.

GAP analysis revolves around- where we are and where we want to be?

For the stated case study, we can sate the below GAP ANALYSIS comparing the AS-IS (Existing process) with TO-BE (FUTURE PROCESS).

|  |  |  |  |
| --- | --- | --- | --- |
| CATEGORY | AS IS | TO BE | GAP IDENTIFIED |
| **PRODUCT ACCESS** | LIMITED TO LOCAL MARKETS AND FEW CHOICE OF PRODUCTS AVAILABILITY | FARMERS CAN ACCESS TO A WIDE RANGE OF PRODUCT THROUGH ONLINE AND ACCESS TO OTHER GEOGRAPHICAL AREAS | LIMITED PRODUCT AVAILABILITY AND BOUND TO A PARTICULAR GEOGRAPHY |
| **PRODUCT QUALITY** | NO ASSURANCE OF PRODUCTS | ACCES TO LICENSED LISTED MANUFACTURER THROUGH ONLINE APP | TRUST AND LACK OF QUALITY ASSURANCE |
| **COST OF PRODUCTS** | INVOLVEMENT INTERMEDIARIES IN SUPPLY CHAIN | PRODUCST ARE AVAILABLE AT COMPETITIVE RATES SINCE THE PLATFORM ELIMINATES INTERMEDIARIES | REDUCTION OF COST- HIGH PROFIT MARGIN |
| **TECHNICAL SUPPORT** | MINIMAL GUIDANCE ON USING PRODUCT EFFECTIVELY | MANUFACTURERS CAN EASILY PROVIDE GUIDANCE THROUGH ONLINE | FARMERS LACK TECHNICAL ADVICE TO MAXIMISE PRODUCT EFFECTIVENESS |
| **ON TIME DELIVERY** | FARMERS NEED TO TRAVEL LONG FOR EVERY SINGLE PRODUCT | ANYTIME ANYWHERE DELIVERY | TIME CONSUMING |
| **CONNECTIVITY** | NO UPDATES ON CURRENT OR NEW PRODUCTS OR DISCOUNT AVAILABLE | UPDATED PRODUCT CATLOG | POOR COMMUNICATIO AND POOR UPDATES. |
|  |  |  |  |

**QUESTION-5**

**RISK ANALYSIS**

**INTERNAL RISK**- Lack of required expertise, difficulties in application development, testing issues, payments procedure issues, compliance, dependency on vendors.

**EXTERNAL RISK**- Market risk, environmental risk.

**BA Risk-** inadequate requirement gathering, communication gap, inadequate knowledge.

**PROJECT RISK** - Scope risk, budget over runs, technology issues.

**QUESTION-6**

**RACI MATRIX (STAKEHOLDER ANALYSIS**)

RACI matrix is a project management tool used to clarify and communicate roles and responsibilities within a project team. It helps ensure that tasks and activities are assigned to the right people, reducing confusion and increasing efficiency.

**R (Responsible):** The person or team responsible for completing the task or activity.

**A (Accountable):** The person or team accountable for the outcome of the task or activity.

**C (Consulted**): The person or team consulted or informed during the task or activity.

**I (Informed**): The person or team informed or notified of the task or activity’s progress.

|  |  |  |
| --- | --- | --- |
| RACI MATRIX | NAME | DESIGNATION |
| RESPONSIBLE | JUHI | SENIOR JAVA DEVELOPER |
| MS LUCIE | JAVA DEVELOPER |
| MR TUCKER | JAVA DEVELOPER |
| MR BRUNO | JAVA DEVELOPER |
| MR JASON | TESTER |
| MS ALEKYA | TESTER |
| ACCOUNTABLE | MR VARDHAN | PROJECT MANAGER |
| CONSULTED | MR MIKE | NETWORK ADMIN |
| JOHN | DB ADMIN |
| MR DOOKU | PROJECT COORDINATOR |
| INFORMED | MR PANDU | FINANCIAL HEAD |
| PETER | COMMITTE |
| KEVIN | COMMITTE |
| BEN | COMMITTE |

**QUESTION-7**

**BUSINESS CASE DOCUMENT**

Business case is prepared by Sr Bas, solution architecture, business architecture and senor business managers. Once it is prepared, it is submitted to the management of IT Company. Management will analyze the facts and figures and considers the feasibility of the project. Management will call for an investor meeting and explain to the investors about this project on how much investment is required, risk factor, how return will come. If the investors show interest in invsting in this project, then the IT Company will go ahead and accepts SOW from the client.

Here’s a Business case for the case study

1. **Why is this project initiated**?

* The project is initiated to address the problem faced by the remote farmers in buying or getting access to agricultural product like pesticides, seeds etc.
* The project aims to build an online agricultural app that can connect the farmers to the manufacturer directly for buying the requirement.

1. **What are the current problem**

* Difficulties in sourcing fetilizers, seeds and pesticides
* Lack of technical knowledge of the farmers.
* Dependencies in intermediaries

1. **With this project how many problems could be solved**?

* Wider reach- manufactures can reach remote areas farmers
* User friendly app for easy transaction of products
* Farmers can directly order agricultural product from manufacturer.
* Faster delivery of agricultural products.

1. **What are the resources required?**

|  |
| --- |
| SENIOR JAVA DEVELOPER |
| JAVA DEVELOPER |
| TESTER |
| PROJECT MANAGER |
| NETWORK ADMIN |
| DB ADMIN |
| PROJECT COORDINATOR |
| FINANCIAL HEAD |

* Infrastructure- computers, internet, testing device.
* Technology- development platform, database
* Budget- 2 crore

1. **How much organizational change is required to adopt this technology**?

* Customer support
* Operational changes
* Training team for educating the farmers in using the app

1. **Time frame to recover ROI(return on investment)**

* An estimated of 2-3 years

1. **How to identify stakeholders?**

* Primary stakeholders- Farmers, manufactures
* Secondary stakeholders- project team, customer support team
* Key stakeholders- Mr Henry, Mr Pandu and Mr Dooku, financial team

**QUESTION 8**

**SDLC METHODOLOIES**

Software development life cycle shows the crucial steps or phases required for developing or designing any software application. A typical SDLC got 7 phases

1. **Planning**- **enterprise** analysis, project scope, project budget.
2. **Requirement analysis**- project planning by PM, business requirement- Business requirement document created
3. **Design**- done by technical or design architecture.
4. **Build-** coding and unit testing
5. **Testing**- test plan, UAT
6. **Implementation-** done by release engineers- running the code very first time in production
7. **Maintenance-** Support team

**Methodologies involved in SDLC.**

1. **Sequential**- also known as waterfall model, it follows a linear sequential and phase-wise approach to software development.

In this methodology each phase is completed before moving on to the next one.

The budget and scope are predetermined and no changes are accepted in middle of the development.

1. **Iterative**- It involves repeating a series of phases or activities in a cyclical manner, with the goal of continually refining and improving the software product.

Here, software is developed in increments, with each iteration adding new functionality or features, requirements can change between iteration, and development process can adapt to these changes.

1. **Evolutionary**- it is a combination of incremental and iterative methodologies.

Incremental methodologies first implement a few basic features and deliver to customer. Then build the next part and deliver it again to and repeat this step until the desired system is fully released.

It is better suited for larger project, risk analysis is better. Support changing environment.

1. **Agile**- Agile methodologies believe in emphasized teamwork, accountability and iterative progress toward well-defined goals, responding to change over following a plan.

Customer satisfaction and continuous delivery of valuable software. Individuals and interaction over process and tools.

**QUESTION-9**

**Waterfall,RUP,Spiral and scrum models**

1. **Waterfall** **Model**- It is the most common and classic of life cycle models, also referred to as a linear-sequential life cycle model. It is used when project requirements are small, project time is less and where business requirements are stable and no changes are coming. In this model, each phase must be completed in its entirely before the next phage can begin. At the end of each phase, a review takes place to determine if the project is on the right path and whether or not to continue or discard the project.

**Stages of waterfall model.**

1. Requirement Gathering.
2. Requirement Analysis
3. Deign
4. Development-coding
5. Testing
6. Deployment
7. Maimtenance
8. **RUP (Rational Unified Process)** – It is an iterative software development process framework. It 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. Within each iteration, the tasks are categorized into nine disciplines**.**

**Six engineering discipline.**

Business Modeling

Requirements

Analysis and Desig

Implementation

Test

Deployment

1. **Three supporting discipline**

Configuration and change mamagement

Project management

Environment

**It has four projects Life cycle Phase**

**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. **Spiral-** The spiral model gives more emphases placed on risk analysis. It has four phases- planning, Risk analysis, engineering and evaluation. A software project repeatedly passes through these phases in iterations called spiral in this model. The baseline spiral, starting in the planning phase, requirements are gathered and risk is assessed. Each subsequent spirals builds on the baseline spiral. 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. 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. In the spiral model, the angular component represents progress, and the radius of the spiral represents cost. Such model involves high amount of risk analysis. Good for large and mission-critical projects. Software is produced early in the software life cycle.
2. **Scrum-** It can be implemented either at the beginning of the projects or when you sense that project is falling behind schedule. This model exercises full admin power.

Here, we identify and priotize our requirements or user stories and go for a complete design development called as a sprint, which usually takes 2-4 weeks. Scrum product development method consists of a scrum team which compromises of product owner, scrum master and developers.

Product owner defines user stories like **AS a <user> I Want to< what is the purpose> so that <what you gain by the story.**

**Scrum Master**- facilitates daily stand-ups and monitors the performance within the sprint. Team will raise all their issues to scrum master.

**Development team**- Works in sprint to deliver features and functionality.

**Sprint review**- This happens at the end of the sprint where team will demo the complete 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.

1. **What went well in the sprint?**
2. **What did not go well?**
3. **What are the required areas of improvements in next sprint**?

**Scrum Artifacts**

**Product log**- A priotize list of all requirements and user stories, own by the product owner

**Sprint backlog**- a subset of product backlog chosen for the current sprint, own by the development team.

**Increment**- the sum of all completed product backlog items during a sprint.

**Release** **plan**- A release plan is a set of working software delivered to the business customer resulting from a set of iterations. During, release planning, teams will review a product backlog to organize user stories into the specific release and iterations that deliver a functions product to the business customers.

**Question 10- Waterfall Vs V- model.**

|  |  |
| --- | --- |
| WATERFALL MODEL | V- MODEL |
| Easy to understand and easy to use | Emphasize planning for verification and validation of the product in early stages of product development |
| Provides structure to inexperienced staff | Each deliverables must be testable |
| Milestones are well understood | Best fit for high reliability systems |
| Sets requirements stability | Phases are processed and completed one at a time |
| Works well when quality is more important than cost or schedule | Works well for small project when requirements are well understood |
| All requirements must be known upfront | Adjusting scope during the life cycle can kill a project |
| Deliverables created for each phase are considered-inhibits flexibility | No working software is produced until late during the life cycle |
| Can give a false impression of progress | Poor model for complex and object-oriented projects |
| Doesn’t reflect problem solving nature of software  Development- iteration of phases | Poor model where requirements are at a moderate to high of changing |
|  |  |

**Question 11**-

As a BA, I would go for the V- model for this online agriculture application building. The reasons are listed below:

1. Well defined requirements- The stakeholders have provided a clear and well defined requirements and v models are best suited for projects where requirements are clear and unlikely to change.
2. Error detections- V models integrate testing at every stage, ensuring that errors are detected early stage.
3. Limited budget- the project have strict budget of 2 cr and timeline of 18 months. The V- model minimizes rework through early validation, saving both time and money.

**Question-12**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Timeline** | **Resources Involved** | **Activities** |
| **RG** | Week 1-10 | PM, BA | Requirement gathering from stakeholders |
| **RA** | Week 10-23 | BA | Analyze and finalize requirements |
| **Design** | Week 23-30 | PM, Java Developers, BA | Create system architecture and detailed design |
| **D1 & T1** | Week 30-38 | Java Developers, Testers | Develop module 1 and perform module testing |
| **D2 & T2** | Week 38-47 | Java Developers, Testers | Develop module 2 and perform module testing |
| **D3 & T3** | Week 47- 62 | Java Developers, Testers | Develop module 3 and perform module testing |
| **D4 & T4** | Week 62-70 | Java Developers, Testers | Develop module 4 and perform module testing |
| **UAT** | Week 70-78 | PM, BA, Testers | Conduct User Acceptance Testing with farmers |

|  |  |  |  |
| --- | --- | --- | --- |
| TASK | START WEEK | END WEEK | DURATION |
| RG | 1 | 6 | 6 |
| RA | 7 | 12 | 6 |
| DESIGN | 13 | 22 | 10 |
| D1 | 23 | 32 | 10 |
| T1 | 33 | 37 | 5 |
| D2 | 38 | 47 | 10 |
| T2 | 48 | 52 | 5 |
| D3 | 53 | 62 | 10 |
| T3 | 63 | 67 | 5 |
| D4 | 68 | 77 | 10 |
| T4 | 78 | 82 | 5 |
| UAT | 82 | 88 | 7 |

**Question-13**

**Difference between fix bid and billing projects**

Fix Bid Project- A **Fixed Bid Project** (or Fixed Price Project) is where the scope, timeline, and cost are agreed upon at the beginning of the project. The client pays a fixed price for the entire project, regardless of the time or resources required.

A **Billing Project** (or Time & Material Project) is where the client pays based on the actual time and resources spent on the project. Costs are calculated based on hourly rates, materials used, and additional resources deployed.

**Questiom-14**

Timesheet- Requirement gathering phase.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requirement gathering phase** | | | | | |
| SL NO | TASK | ACTIONABLE ITEM | START TIME | END TIME | DURATION |
| 1 | IDENTIFY THE STAKEHOLDER - MEETING | MEETING TO LIST DOWN THE STAKEHOLDES | 10:00 AM | 11:OO AM | 1 HOUR |
| 2 | CLIENT INTERACTION | A ZOOM CALL TO UPDATE THE CLIENT | 11:00 AM | 1:00 PM | 2 HOUR |
| 3 | FINETUNING THE INPUTS FOR BRD DOCUMENTS | SME DISCUSSION- IN PERSON CALL | 2:00 PM | 3:00 PM | 1 HOUR |
| 4 | REQUIREMENT SORTING | WORKING ON THE TEMPLATE | 3:30 PM | 4:30 PM | 1 HOUR |
| 5 | TEAM MEETING | DISCUSSION ON THE INPUTS | 4:30 PM | 7:00 PM | 2.5 |
| TOTAL= 7.5 HRS |

**Timesheet for development phase.**

|  |
| --- |
| DEVELOPMENT PHASE |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | SL NO | TASK | ACTIONABLE ITEM | START TIME | END TIME | DURATION | | 1 | Development Support | Providing clarifications on business logic | 9:30 AM | 11:00 AM | 1.5 Hours | | 2 | Technical Design Walkthrough | Collaborating with developers on technical design | 11:30 AM | 1:00 PM | 1.5 Hours | | 3 | API Mapping | Mapping APIs to functional requirements | 2:00 PM | 3:30 PM | 1.5 Hours | | 4 | Sprint Review | Attending sprint review to ensure progress aligns | 4:00 PM | 5:30 PM | 1.5 Hours | |  |  |  |  |  |  | |

**Testing timesheet.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SL NO | TASK | ACTIONABLE ITEM | START TIME | END TIME | DURATION |
| 1 | Test Case Preparation | Writing test cases based on requirements | 10:00 AM | 12:OO AM | 2 HOURS |
| 2 | Test Case Review | Reviewing test cases with QA team | 12:00 AM | 2:00 PM | 2 HOURS |
| 3 | Functional Testing | Conducting functional tests | 3:00 PM | 6:00 PM | 3HOURS |
| 4 | Defect Triage Meeting | Discussing issues raised during testing | 6:00PM | 7:00PM | 1HOUR |
|  |  |  |  |  | TOTAL= 8HRS |

**UAT Timesheet**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SL NO | TASK | ACTIONABLE ITEM | START TIME | END TIME | DURATION |
| 1 | UAT Scenario preparation | Preparing user acceptance testing | 10:00 AM | 12:OO AM | 2 HOURS |
| 2 | UAT | Testing end-users on systrm functionality | 12:00 AM | 2:00 PM | 2 HOURS |
| 3 | UAT EXECUTION | Executing supporting end users during testing | 3:00 PM | 6:00 PM | 3HOURS |
| 4 | UAT FEEDBACK | Feedback Analysis Consolidating and documenting feedback | 6:00PM | 7:00PM | 1HOUR |

**DEPLOYMENT AND IMPLRMRNTATION TIMESHEET**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SL NO | TASK | ACTIONABLE ITEM | START TIME | END TIME | DURATION |
| 1 | Deployment Planning | Reviewing deployment strategy | 10:00 AM | 12:OO AM | 2 HOURS |
| 2 | Supporting Deployment Activities | Coordinating with technical teams during deployment | 12:00 AM | 2:00 PM | 2 HOURS |
| 3 | Post-Deployment Validation | Verifying system functionality after deployment | 3:00 PM | 6:00 PM | 3HOURS |
| 4 | End-User Support | Assisting users with queries and issues | 6:00PM | 7:00PM | 1HOUR |