**Q1. Identify Business Process Model for Online Agriculture Store – (Goal, Inputs, Resources, Outputs, Activities, Value created to the end Customer)**

Goal: Creating mobile and web applications enables farmers to purchase agricultural items online and support them through the online store. The ability to acquire information from producers regarding the product (pesticides, seeds, fertilizers, etc.). Reach the most significant number of clients at the ideal moment to boost agricultural product sales by on-time delivery to farmers.

Inputs:

* Manufacturers of raw materials or collaborations with raw material firms
* Delivery boys will bring the items that have been purchased, and the administrator will oversee the flow of orders and merchandise.
* Product specs, brands, prices, and high-quality pictures, and images.
* Farmers' or buyers' signup details.
* Delivery information, payment options, shopping cart, and checkout page.
* Product details from producers (Pesticides, Fertilizers, Seeds).
* Rules and restrictions (for agriculture products if any according to the state/central government)

Resources:

* People Resources:
* Business Stakeholders - Mr. Henry, Mr. Peter, Mr. Kevin, Mr. Ben
* Development Team- Project Manager, Java Developers, Network and Database Administrators, Testers, Business Analyst.
* Delivery Boys to deliver the product and receive cash payments.
* Product Resources:
* All the agricultural products such as seeds, pesticides and fertilizers
* Technological Resources:
* Internet connection via modem or wifi
* Availability of laptop or mobile or desktop
* Platform for Web and Mobile Applications
* payment gateways
* Cloud database for data storage
* Education programmer in the website to educate the farmers.
* Financial Resources:
* INR 2 crore budget under the CSR initiative.

Output:

* Farmers will be able to purchase seeds, fertilizer, and pesticides from any location with internet availability through this online marketplace for agricultural products.
* The app is accessible on mobile devices as well as on computers or laptops through the website.
* A smooth checkout process and an uninterrupted product ordering process.
* Payments made using safe means via cash on delivery, cards, wallets, and other payment methods.
* An order tracking and management system.
* The platform will also allow the customers to raise their concerns and make suggestions to add any required agricultural products.

Activity:

* The application collects information about products for farmers such as seeds, fertilizer, and pesticides.
* Manufacturers upload all kinds of information such as descriptions of products, pricing and when these things are available to sell.
* The platform displays agricultural products in categorized sections, allowing farmers to browse and filter based on their needs.
* Farmers can type specific crops or product names into search bar and can take a look at all sorts of details about the products specs, can review the ratings given by other farmers.
* The system provides multiple payment options (cash on delivery, online payment, net banking, credit/debit card, wallet).
* Once payment is confirmed, an email or SMS or WhatsApp notification is sent to the farmer.
* Farmers get a confirmation of their order and tracking numbers to track their shipment all right.
* Products are delivered to the farmer’s registered address within a specified timeframe.
* The farmer can use the system to request a return or exchange if the product is defected or does not match the description.
* The system starts paying the original wallet or payment method in reverse if a return is accepted.
* Farmers have the option to raise concerns, talk with customer service, and rate and review goods and manufacturers.

Value Created to the end customers: From any location, farmers may readily locate and buy agricultural items. Several payment options guarantee seamless transactions. Refund procedures, delivery updates, and order tracking all contribute to reliability. Reduces expenses for farmers by cutting off middlemen. Help to eliminate the distance between farmers and the manufacturing of agricultural products.

**Q2. 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.**

|  |  |  |
| --- | --- | --- |
|  | Positive | Negative |
| Internal | Strength   * The project budget is INR 2 crore, funded under a Corporate Social Responsibility (CSR) initiative, reducing financial risk. * The presence of experienced Java developers, testers, and IT support ensures smooth project execution. * Project adoption rises as a result of government actions supporting farming's digitalization. | Weaknesses   * It's possible that many farmers lack digital literacy, necessitating in-depth instruction or customer service. * The user experience may be impacted by inadequate internet access in rural locations. * The time of 18 months appears shorter. Delivery to remote areas may be challenging to provide in a timely and economical manner. |
| External | Opportunities   * The government may encourage adoption through programs like Agri-Tech financing and Digital India. * Anyone can place a 24-hour order from the store, not just distance farmers. * Tailored product recommendations based on crop type, location, and possible future additions of farming equipment and tools can enhance the experience. | Threats   * Competitors may include specialized agri-tech businesses and online marketplaces like Amazon and Flipkart. * Conventional purchasing practices may hinder adoption, necessitating intensive awareness campaigns. Farmers may be reluctant because they are concerned about internet fraud or unsuccessful transactions. * Transportation strikes, natural disasters, pandemic-like conditions, and government regulations governing the sale of seeds and pesticides could all have an impact on on-time delivery. |

**Q3. 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.**

We currently need to determine whether utilizing Java for this project is technically feasible given certain limitations, such as time and money:

Hardware Requirements (HW): The platform will be developed as a web and mobile application, requiring-

* For Developer & Tester Systems: High-performance laptops/desktops (Intel i7, 16GB RAM, SSD). Dev servers for code integration and testing environments.
* For application hosting: Dedicated cloud-based Java servers (Tomcat, JBoss, or Spring Boot microservices).
* Database Server: SQL Server (For truncate and load data)
* Content Delivery Network (CDN): For quick loading and lower latency in rural areas

Software & Technology Stack (SW): Since Java is the core technology, the stack should include-

* Frontend (UI/UX Development): Android (Java) web interface using React.js.
* Development of the Backend:
* Architecture using microservices (for scalability)
* OracleDB (For structuring raw data coming from SQL Server and load the data in data warehouse)
* AWS (For Cloud data warehouse storage system)
* Payment & Security: Integration of Razorpay, Paytm, and UPI APIs for payments
* Spring Security (For Java-based protection)

Availability of Trained Resources: In-House Team Members Available-

* **Senior Java Developer** – Ms. Juhi
* **Java Developers** – Mr. Teyson, Ms. Lucie, Mr. Tucker, Mr. Bravo
* **Database Admin** – Mr. John
* **Network Admin** – Mr. Mike
* **Testers** – Mr. Jason, Ms. Alekya

Budget Considerations: Considering the project has INR 1 crore in CSR funding, the cost of the technology must be kept within the allocated budget.

* Estimated Cost Breakdown (in INR):
* Software Licensing & Cloud Hosting: ₹25-30 lakhs
* Development & Testing Costs (Salary, Tools): ₹40-45 lakhs
* Security & Compliance (SSL, Encryption, Cybersecurity): ₹5-7 lakhs
* Marketing, Training & Support for Farmers: ₹15-20 lakhs
* Miscellaneous & Contingency Reserve: ₹5-10 lakhs

Time Frame for Development & Deployment: The project has an 18-month timeline. The approximate development timeframe is as follows:

|  |  |  |
| --- | --- | --- |
| Phase | Duration | Key Deliverables |
| **Requirement Gathering & Analysis** | 2 months | BRD, SRS, Wireframes |
| **System & application Design** | 2 months | UI/UX Mockups, Database Schema |
| **Backend & Frontend Development (MVP)**  Testing & Bug Fixing | 8 months | Functional Web & Mobile App (Beta)  Security Testing, Load Testing |
| Integration (Payment, Security, APIs) | 3 months | Secure & Scalable System |
| User testing & Training | 1month | UAT, Training & understanding |
| Pilot Launch | 1month | Farmer Onboarding, Support Setup |
| Final Deployment & Go-Live | 1month | Full Release |

**Q4. 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?**

|  |  |  |
| --- | --- | --- |
| **Key Areas** | **AS-IS (Current Process)** | **TO-BE (Future Process)** |
| Review System | To buy agricultural goods like seeds, fertilizer, and insecticides, farmers are currently depend on local markets and middlemen. | A large selection of seeds, fertilizer, and pesticides are available to farmers through the web or app straight from producers. |
| Implications | Farmers who use traditional practices purchase seeds, fertilizer, and insecticides from a physical agricultural store. | Farmers can order from any location with an internet connection at any time via the mobile app or website, and it will be delivered to their homes. |
| Comparison | Dependence on cash transactions, which raises the possibility of delays and risk. There is no digital system for tracking orders or payments. | A variety of online payment methods (UPI, Net Banking, Wallets) as well as real-time order tracking |
| Develop Requirements | Without quality control, farmers purchase locally and frequently end up with fake or outdated goods, no digital system to display the product details | In the web store manufacturers can upload and farmer can reviews, batch tracking, and authentication for each product. |
| Recommendation | There’s no system for delivery of goods or returning of goods that are faulty or expired in traditional practice. | On-time doorstep delivery will be available via integrated logistics and tracking as well as refund to the original payment method also available under easy return and exchange policy. |

**Q5. List down different risk factors that may be involved (BA Risks And process/Project Risks)**

Business Analysis (BA) Risks:

* The system may have gaps if stakeholders (farmers, manufacturers, etc.) fail to specify their demands precisely.
* As the project develops, new features or legal requirements might surface, which could have an impact on schedules.
* Stakeholder misalignment refers to disagreements on platform functionality between manufacturers, farmers, and company owners.
* Poor feature implementation may result from developers' and analysts' ignorance of farming methods.
* Technical language may be difficult for farmers and suppliers to understand when gathering requirements.
* Users' trust may be impacted by manufacturers providing inaccurate or incomplete product information.
* Because farmers are not comfortable with technology, they can be reluctant to use a digital platform.
* It could be difficult to enter the industry if there are currently other agricultural e-commerce platforms already avalable.

Process / Project Risks:

* Testing failures, development delays, or changes in requirements could cause the 18-month timetable to be extended.
* Dependence on other suppliers, such as payment gateways and delivery companies, could result in delays.
* To manage big transactions and users, the app needs strong server, powerful backend, and cloud hosting capabilities.
* Farmers' internet payments and personal information must be carefully safeguarded from fraud and hacking activity.
* The program must run without a hitch on any device, operating system, or internet speed.
* Rapid user growth could necessitate further infrastructure and technological investments if the platform is sustainable.
* The user experience may suffer if the software launches with issues due to inadequate testing.

**Q6. Perform stakeholder analysis (RACI Matrix) to find out the key stakeholders who can take Decisions and Who are the influencers**

The RACI Matrix, which stands for Responsible (R), Accountable (A), Consulted (C), and Informed (I), helps in determining the various stakeholders' roles in project execution and decision-making.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Me (BA) | Mr. Henry (Owner/Sponsor) | Mr. Peter,  Mr. Kevin, Mr. Ben (Stakeholders) | Mr. Karthik (Delivery Head) | Mr. Vandanam (PM) | Ms. Juhi (Senior Java Dev) | Mr. Teyson,  Ms. Lucie, Mr. Tucker, Mr. Bravo  (Java Devs) | Mr.  Mike (Network Admin) | Mr. John (DB Admin) | Mr. Jason, Ms. Alekya  (Testers) |
| Requirement Gathering | R | A | C | I | C | I | -- | -- | -- | -- |
| Analysis | R | A | C | C | C | I | I | -- | -- | -- |
| Design & Planning | C | A | C | C | R | C | C | C | C | I |
| Feasibility Study | C | I | I | A | R | C | C | C | C | I |
| Development | I | -- | -- | I | C | A | R | C | C | I |
| Database Management | I | -- | -- | I | C | C | C | -- | R | I |
| Infrastructure Setup | I | -- | -- | I | C | C | C | R | -- | I |
| Testing | I | I | -- | I | C | C | C | -- | -- | R |
| UAT | R | A | C | I | C | C | C | I | I | R |
| Deployment | I | I | -- | I | A | C | C | R | R | I |
| Project Monitoring | C | I | I | A | R | I | I | I | I | I |
| Go-Live & Support | I | A | C | C | R | C | C | C | C | C |

**Q7. Help Mr Karthik to prepare a business case document**

* *Why is this project initiated?*

The conventional supply chain is ineffective because it frequently involves several middlemen who raise prices and lower the quality of the final product. Mr. Henry saw this problem, understood the needs of farmers, and used this information to launch the initiative of building an online marketplace for agricultural products that would link farmers and manufacturers directly.

* *What are the current problems?*

Farmers currently face a number of procurement process issues. Dependency on middlemen, lack of price transparency, and restricted access to reliable sources can result in overpriced and bogus products. Furthermore, farmers in isolated locations struggle to get agricultural inputs on schedule, which delays farming operations. Since many farmers rely on cash transactions, payment systems present additional difficulties, making it hard to track and safeguard purchases.

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

By creating the online marketplace in which farmers can conveniently examine and purchase items, this initiative swiftly tackle the following problems:

* Directly linking farmers and manufacturers to cut out intermediaries.
* Making pricing transparency possible via a competitive, open platform.
* By using real-time stock updates, availability is improved.
* Simplifying logistics to ensure prompt delivery across regions.
* Providing safe transactions with digital payment solutions.
* *What are the resources required?*

For the project to be executed successfully, a mix of human and technological resources is needed. The platform will be created and maintained by a team that includes business analysts, project managers, software developers, database administrators, network engineers, and testers. The primary technology for development will be Java, which guarantees scalability and resilience. Cloud-based infrastructure will be necessary for the project in order to handle heavy traffic and data security measures to protect transactions. All required investments in infrastructure, technology, and resources will be financed by the money required to build this project.

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

Video lessons, customer service, and local awareness campaigns can help farmers get the fundamental training they need to use the online platform. The system will require manufacturers to connect their product catalogs, which can be handled by a methodical onboarding procedure. All things considered, the switch from conventional to digital procurement techniques will go well.

* *Time frame to recover ROl (return on investment)?*

The main goal of this project, which is supported under CSR, is social effect rather than quick financial gains. However, the project's goal is to recover operating expenditures in two to three years, and sustainability is a crucial component. In order to ensure long-term sustainability and growth, the platform is anticipated to acquire traction as digital platforms in agriculture continue to rise in use.

* How to identify stakeholders?

A stakeholder is an individual or group of individuals who either directly or indirectly affects the project in positive or negative way. They may be any of the following:

* Providing financial support for the project.
* Helping with duties or initiatives related to project.
* Requesting information on current project activities.
* Participating in the project's management and planning meetings.

A BA need to identify the causes behind their interest in this project. Their anticipations for the project the degree of their involvement. How often should you (BA) stay in touch or give updates? how important it is that they be satisfied with the project. The stakeholders in a project can either be primary stakeholders, business stakeholders, 3rd party stakeholders and negative stakeholders. In the current project following are the types of stakeholders:

* Primary Stakeholders who are directly impacted by the project and have a high level of interest and influence:
* Farmers (End Users)
* Manufacturers (Product Suppliers)
* Mr. Henry (Project Sponsor)
* Mr. Karthik (Delivery Head)
* Mr. Vandanam (Project Manager)
* Business Analyst (Me)
* Business Stakeholders are the key people within the organization who contribute to or are affected by the project:
* Mr. Peter, Mr. Kevin, Mr. Ben (Supporting Stakeholders from Mr. Henry's side)
* Development Team (Ms. Juhi, Mr. Teyson, Ms. Lucie, Mr. Tucker, Mr. Bravo - Java Developers)
* Mr. John (Database Administrator)
* Mr. Mike (Network Administrator)
* Mr. Jason & Ms. Alekya (Testers)
* Third-Party Stakeholders are the external entities interacting with or influencing the project:
* Payment Gateway Providers
* Logistics and Delivery Partners
* Government Agencies (Agricultural & Digitalization Initiatives)
* CSR Funding Committees
* Negative Stakeholders are those individuals or groups who may oppose or be negatively impacted by the project:
* Local Middlemen & Distributors
* Traditional Retailers & Agro Dealers
* Competitor Online Platforms

**Q8. 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.**

There are mainly four primary SDLC Methodology widely used in companies:

Sequential Methodology: Sequential methodology is a software development or project management approach in which phases and tasks are finished in a sequential, step-by-step fashion. Each phase must be fully completed before moving to the next, with no overlap or iteration between stages. This methodology emphasizes a structured progression, often used in traditional models like the Waterfall model or the V-model. For this 18-month online shopping store project, the sequential nature of the V-model aligns with these principles. Its emphasis on testing after each development phase (verification and validation) builds on the sequential methodology by adding a mirrored testing structure, ensuring defects are caught early without deviating from the linear flow. Following are some of the guidelines for sequential methodology:

* Requirement Clarity: Since adjustments in the middle of a process can be expensive and disruptive, make sure all criteria are clearly stated and agreed upon beforehand.
* Phase Completion: Before moving on to the next step, each one must be finished completely and validated (for example, by testing or reviews).
* Maintain Communication: Make sure that teams or individuals are communicating throughout processes clearly with one another.
* Milestone Reviews: At the conclusion of each phase, conduct official reviews or sign-offs to ensure that deliverables live up to expectations.
* Risk management: Early risk identification and mitigation is crucial since problems found later may require to revert back to earlier stages.
* Control Scope and Changes: To keep things running in sequential manner clearly define the project scope. If changes are required use proper change management method.

Iterative Methodology: Iterative methodology is a project management and development approach that involves breaking down a project into smaller, manageable cycles called iterations. Each iteration is a self-contained mini-project with its own goals, deliverables, and timeline. At the end of each iteration, a working version of the product or a specific set of features is produced. The core idea behind iterative methodology is to build the product incrementally, allowing for flexibility, early feedback, and adaptation to changing requirements.Instead of a linear, sequential approach where all planning is done upfront. Iterative methods embrace the idea that understanding evolves throughout the project. Popular examples include Rational Unified Process (RUP) model. Following are some of the guidelines for iterative methodology:

* Define Clear Iteration Goals: Each iteration should have specific, measurable, attainable, relevant, and time-bound (SMART) goals.
* Regular Reviews and Retrospectives: At the end of each iteration, conduct reviews to demonstrate the progress to stakeholders and gather feedback. Hold retrospect as a team to review the iteration process and identify areas for improvement.
* Maintain a Product Backlog: A prioritized list of all desired features, changes, and bug fixes should be maintained and used to plan future iterations.
* Embrace Change: Be prepared to adapt plans based on feedback and new information. Iterative methods are designed to accommodate change.
* Continuous Integration and Testing: Integrate and test the developed components frequently within and at the end of each iteration to ensure quality.
* Effective Communication and Collaboration: Foster strong communication and collaboration within the development team and with stakeholders.
* Manage Technical Debt: Address technical debt (shortcuts taken during development that may hinder future work) regularly to maintain the long-term health of the product.

Evolutionary Methodology: Evolutionary methodology is a software development approach where the system is developed incrementally through a series of evolving versions, each improving and enhancing upon the previous one. Unlike purely iterative or sequential models, evolutionary models focus on delivering a working system early and expanding or improving it through successive versions. This methodology is often used in software engineering, research, and product development where the final requirements are not fully known at the start, or where the product must evolve in response to user needs or market conditions. Following are some of the guidelines for evolutionary methodology:

* First, define a core product: List the features that must be present in the first iteration.
* Iterate Based on Feedback: Following each iteration, get feedback from users or stakeholders and utilize it to inform the subsequent development and improvement cycle.
* Maintain Flexibility: Be ready to modify the development strategy in response to input and evolving specifications.
* Regularly Assess Risks: At every step, assess technical, operational, or business risks and modify the focus of later iterations to reduce them.
* Make Sure Every Increment Is Functional: Every increment that is supplied should be a functional and testable version of the system.
* Integrate and Test Frequently: As new functions or enhancements are implemented, make sure that every adaptive step is tested and included into the current system to preserve stability.
* Document Progress: To monitor the platform's development and assist with subsequent iterations, maintain records of the decisions and modifications made during each cycle.

Agile Methodology: Agile methodology is an iterative and incremental approach to managing and completing projects. It emphasizes collaboration, flexibility, continuous improvement, and the delivery of working software frequently. Instead of following a rigid, sequential plan, agile methodologies break down projects into small cycles called sprints.Teams work collaboratively within these iterations, adapting to changing requirements and feedback from stakeholders. The core philosophy of Agile is to respond to the unpredictability of building and implementing software (and other types of projects) through incremental, iterative work processes and empirical feedback. The popular Agile models are Scrum, Kanban, and Extreme Programming (XP). The 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

The twelve principles of Agile Methodology are:

1. Satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity--the art of maximizing the amount of work not done--is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.

Following are some of the guidelines for agile methodology:

* Work in short sprints or iterations: Split the job up into brief repetitions of a set length, typically one to four weeks.
* Maintain a Product Backlog: An ordered list of all the features, specifications, improvements, and corrections that the product needs.
* Hold Daily Stand-up call: Conduct brief daily stand-up meetings to help the team coordinate and plan their work.
* Conduct Iteration Planning: The team plans the work to be done at the start of each iteration.
* Review Working Software Frequently: Show stakeholders the finished product at the conclusion of each iteration and solicit their input.
* Conduct Iteration Retrospectives: Examine the previous iteration on a regular basis to pinpoint areas where the procedure needs to be improved.
* Manage Scope Incrementally: Throughout the project, the scope is adjusted in response to input and knowledge gained.

**Q9. They discussed models in SDLC like waterfall RUP Spiral and Scrum . You put forth your understanding on these models.**

Waterfall Model: The waterfall methodology is a traditional life cycle model which is a linear, sequential software development approach, suitable for small projects where requirements are fully understood and no change is needed once the requirement is gathered. The model emphasizes that the project should follow logical progression steps throughout the software development life cycle. In this model, each phase must be completed before moving to the next phase. This model is easy to implement and also the number of resources required is minimal. The date for the completion and cost are easier to determine in the early stages. Although the risk factors are high and not suitable for big and complex projects. As the model does not accept the change requirements and testing is done in later stages it becomes difficult to go back to the previous phases or identify risk in early stages. The steps in waterfall model are:

* *Requirements:* In the requirement stage, all possible requirements are gathered and documented in a requirement specification document.
* *System Design:* The requirements are analysed in this stage. This system design helps in specifying hardware and software to develop the project.
* *Implementation:* Once the design is complete the development team starts the coding by developing small programs called units, that are tested in the next phase and integrated there
* *Testing:* In this stage all units that are developed are integrated in this stage and integrated together after testing of each unit.
* *Deployment:* Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market

Rational Unified Process (RUP) model: Rational Unified Process (RUP) is an iterative model and incremental model, based on building blocks or content elements that explain the content that needs to be developed, the skills needed, and the detailed process for achieving particular development objectives. It supports systematic development while adjusting to changing requirements since it is use-case focused, architecture-centric, and risk-focused. At the end of each iteration, a working version of the software is available. In this framework, components could be adjusted and different stages of the cycle can be repeated until the software is in the state in which it should be required to work. Four Phases of the RUP are:

* *Inception:* Understanding between the team and the client over what will be built.
* *Elaboration:* Understanding among team members over the architecture and design required to produce the desired system behaviour
* *Construction:* In this process of implementing a fully functional system iteratively.
* *Transition:* Delivery, fixing errors, and tweaking in order to ensure client approval constitute the transition.

In this model, the design flaws can be found in the very early stages, enabling finding issues in the early stage of development and taking corrective measures on a limited budget. Although there is a cost for each change in requirement, and the end of the project is unknown which is risky and can be extended for longer period of time and can exceed the budget.

Spiral Model: The spiral model is more of a risk-driven model and is a combination of both sequential and iterative models. The model has four phases the software passes through these four phases repeatedly in an iteration called a spiral. This model is useful for projects with high risks such as research projects or defence projects. The model considers risks that go unnoticed by most of the other models. The model provides an early indication of existing risks and reduces the cost. The four phases of the Spiral model are:

* *Objective, determination, and identification of the alternative solutions:* Starts with gathering the requirements including system requirements, subsystem requirements, and unit requirements, all are taken in this phase. The understanding between the client and analyst is also done in this phase through continuous communication.
* *Identify and resolve risk:* This phase includes identifying estimate risks including technical feasibility and management risks. Once the risks are identified risk mitigation strategy is planned and finalized.
* *Develop, verify next level product:* In this phase, software is developed based on the requirements gathered. Identified features are developed and verified through testing. The next level of product is done using activities like prototyping.
* *Review and plan for the next phase:* In this phase, the customer evaluates the work and if there is any change needed he asks for modification. Constant preventative maintenance is performed to prevent major failures.

The main drawback of this model is, implementing this model in small project is not possible, similar to the iterative model, is complicated and very costly as it is based on risk analysis, for which high-risk experts are required. The end of the project depends on the risk analysis phase.

Scrum: Scrum is a subset of Agile methodology widely used framework. Scrum breaks the project into smaller parts. Scrum processes enable organizations to adjust smoothly to rapidly changing requirements. The Scrum is divided into three categories Roles, Artifacts, and Time Boxes called sprints. A sprint lasts between 2 weeks to 4 weeks. In each sprint, the developer team delivers the part of the project after testing. The product is a finished product, a version of the final product that, can be delivered to the client on request, with the least amount of work. In this model, each sprint has an objective to be fulfilled by the team and delivered to the client based on the priority set by the client. A scrum process helps the organization in the:

* Improve the deliverable quality.
* Improve the ability of the team to handle changes.
* Spend less time producing estimates and deliver greater results
* Have greater authority over project timeline and status.
* After each sprint, the scrum has events called sprint meetings.
* *Sprint planning:* This takes place at the start of every sprint when the team chooses what they will produce. In most cases, the client prioritizes the requirements and the team decides which task to bring out from the Product backlog to the Sprint backlog.
* *Daily Scrum meeting:* This occurs daily by the end of the day, and the team answers three specific questions

1) What did you do today?

2) what will you do tomorrow?

3) Are there any challenges you faced while completing the user story?

* *Sprint Review Meeting:* This meeting occurs after each sprint. In this meeting, all developers, stakeholders, product owner, scrum masters, third-party reviewers, SMES, and anybody interested in Scrum can participate. In this meeting, they will see how many user stories the team completed, how many are pending, what challenges they faced, and what progress the team made.
* *Sprint Retrospective Meeting:* This meeting is conducted after the sprint meeting where all the developers participate. This is a self-review session to discuss what went well and what did not go well why did not go well and how to address the challenges they face and come up with precautions not to repeat the same mistake.

**Q10. Write down the differences between waterfall model and V model.**

|  |  |
| --- | --- |
| Waterfall model | V model |
| Waterfall model is a linear and sequential methodology | V model is a sequential model with testing integrated at each stage |
| Phase progressing in a straight line is the foundation of this framework. | Every stage of development (left) in this methodology has a testing phase (right). |
| There is no way to go back to a previous phase till the project is finished. | Backtracking is not restricted here |
| This model is more of a development focus, testing stage starts is only after implementation is done | This is dual focus on verification and validation, each phase tested at its own level hence no separate testing stage |
| This is a rigid model, change is not possible at the mid process | Comparatively more adaptable due to early test planning but still sequential in process |
| Errors gets detected late at the testing stage only | Errors gets detected earlier as each phase is associated with testing |
| Heavy documentation emphasis upfront as chances of defects are high | Heavy documentation plus detailed testing plan for each phase |
| More defects can be found in the testing stage as it comes at the end | Less number or no defects can be detected as each phase is verified and validate at the same time. |
| The cost and complexity is low as only one phase of development is operational. | As testing phase and development phase are going simultaneously cost is higher compare to waterfall model |
| Suitable for more simple stable projects | Suitable for complex and quality critical project. |

**Q11. As a BA, state your reason for choosing one model for this project**

As a BA after analysing the requirements, time, and budget, I will choose the V model for this project. The reason for choosing this model is that it is easier to understand, and each phase gets tested immediately after development. The timeframe of the project is 18 months only so iterative, spiral, or agile models are not feasible for this project. Even though the waterfall model is sequential the testing is done only after the implementation phase so there is a big chance of getting several defects during the testing stage and requiring to redo some of the development parts again which may cost an extra amount and the launch may get delayed. All these issues can be avoided using the V model. The requirements for this project are very clear an online shopping store where farmers can order seeds, pesticides, and fertilizer for now. The requirements are very well understandable the only downside is any defect that can occur after development is complete. To avoid this downside V model is the perfect fit for this project within the time and budget for the project.

**Q12. The Committee of Mr. Henry, Mr Pandu, and Mr Dooku discussed with Mr Karthik and finalised 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.**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | May-26  May-25  Sep-26  Jul-26  Jun-26  Apr-26  Feb-26  Jan-26  Dec-25  Nov-25  Oct-25  Sep-25  Aug-25  Jul-25  Jun-25  Apr-25  Aug-26  Mar-26 |
| Requirement Gathering (RG) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Requirement Analysis (RA) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| System Design |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Development Phase 1 (D1) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing Phase 1 (T1) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Development Phase 2 (D2) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing  Phase 2 (T2) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Development Phase 3 (D3) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing  Phase 3 (T3) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Development Phase 4 (D4) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing  Phase 4 (T4) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| User Acceptance Testing (UAT) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**Q13. Explain the difference between Fixed Bid and Billing projects**

|  |  |
| --- | --- |
| Fixed Bid Project | Billing Project |
| A fixed bid project is a project in which the time and the scopes are defined and usually has a deadline associated with it. | Billing project which is also known as hourly based project or time and materials based on actual time spend and resources used |
| In this process one need to change control process, any new change comes in it has to be estimated, then approved by the client and then added on the addition bid. | In this process communication with client needs to be clearly define what are you charging for. Involve the client in the process by having meetings and taking feedbacks on the finished features |
| Fixed price paid in periodic payments, such as 30% up advance and 40% after completion | Changeable costs that are determined by materials and effort and are billed on an hourly, daily, or monthly basis. |
| It gives less attention to the quality over the product and the fact that it fulfils the vision and requirements | It gives more attention to the quality since there is continuous communication between client and team and gets feedback on every delivery. |
| Projects (like this online store) with clear, consistent specifications. | Projects that need constant cooperation or have ambiguous, changing requirements. |

**Q14. ➢ Design Timesheet of a BA**

**➢ Development Timesheet of a BA**

**➢ Testing Timesheet of a BA**

**➢ UAT Timesheet of a BA**

**➢ Deployment n Implementation Timesheet of a BA**

**Design Timesheet:**

|  |  |  |
| --- | --- | --- |
| Time | Hours | Descriptions |
| 10:00 - 11:30 | 1 hr 30 mins | Requirement validation and refinement |
| 11:30 - 13:30 | 2 hrs | Creating process flow diagrams and wireframes |
| 14:30 - 16:00 | 1 hr 30 mins | Preparing use case and user stories |
| 16:00 - 17:30 | 1 hr 30 mins | Coordinating with design and UI team |
| 17:30 - 19:00 | 1 hr 30 mins | Documentation and feedbacks |

**Development Timesheet:**

|  |  |  |
| --- | --- | --- |
| Time | Hours | Descriptions |
| 10:00 - 10:45 | 45 mins | Sprint stand-up & developers’ synchronization |
| 10:45 - 12:15 | 1 hr 30 mins | clearing the development doubts & improving user stories |
| 12:15 - 13:30 | 1 hr 15 mins | Evaluating the technical congruency |
| 13:30 - 16:00 | 1 hr 30 mins | Validating intermediate builds & feedback sessions |
| 16:00 - 17:30 | 1 hr 30 mins | Updating RTM & tracking sprint progress |
| 17:30 - 19:00 | 1 hr 30 mins | Coordinating with QA for upcoming testing phases |

**Testing Timesheet:**

|  |  |  |
| --- | --- | --- |
| Time | Hours | Descriptions |
| 10:00 - 10:45 | 45 mins | Evaluating testing plan with QA team |
| 10:45 - 12:15 | 1 hr 30 mins | Applying parameters to test case traceability |
| 12:15 - 13:30 | 1 hr 15 mins | Discussion on fault assessment and issue discussion |
| 14:30 - 15:45 | 1 hr 15 mins | Handling contradiction in requirements |
| 15:45 - 17:15 | 1 hr 30 mins | Monitoring over final validation & regression testing |
| 17:15 - 19:00 | 1 hr 45 mins | Test result documentation & client interaction |

**UAT Timesheet:**

|  |  |  |
| --- | --- | --- |
| Time | Hours | Descriptions |
| 10:00 - 11:30 | 1 hr 30 mins | Setting up a checklist and planning for stakeholder UAT |
| 11:30 - 12:30 | 1 hr | Training with the user group |
| 12:30 - 13:30 | 1 hr | Meeting notes & feedback documenting |
| 14:30 - 15:45 | 1 hr 15 mins | Evaluating change requests |
| 15:45 - 17:45 | 2 hrs | Matching UAT conclusions to business objectives |
| 17:45 - 19:00 | 1 hr 15 mins | Getting the final sign-off paperwork ready |

**Deployment & Implementation Timesheet:**

|  |  |  |
| --- | --- | --- |
| Time | Hours | Descriptions |
| 10:00 - 11:15 | 1 hr 15 mins | Verification of pre-deployment criteria |
| 11:15 - 12:45 | 1 hr 30 mins | Coordinating with infrastructure/network teams |
| 12:45 - 13:30 | 45 mins | Completing the deployment documentation |
| 14:30 - 15:30 | 1 hr | Monitoring of the go-live and first production assistance |
| 15:30 - 17:15 | 1 hr 45 mins | Notifying stakeholders about the deployment status |
| 17:15 - 18:15 | 1 hr | Recording closing activities and final comments |