**CapstoneProject1–Part-1**

**Answer 1- BPM(Business Process Model)**

**Goal-** To bridge the gap between farmers and the product suppliers. Simplifying product discovery for farmers and streamlining and enhancing the buying, selling, and managing of agriculture products and services through digital platforms.

**Input-** Farmers details(name, location, and contact information), Farming type(crop farming, livestock, etc), product needs or interest, categories(seeds, fertilizers, pesticides, equipment, machinery, livestock feeds, etc). payment type(online banking, mobile wallets, credit/debit cards, cash on delivery).

**Resources-** A smartphone, tablet, or computer with internet access. Required to install application on customer's device and also on supplier's device. Database for storing data, user information, and transactions.

**Output-** Delivering of required products at doorstep. Transparent pricing comparisons across brands and suppliers. Inventory management insights to optimize stock and avoid shortages. Reducing dependency on middlemen and ensuring fair pieces.

**Activities-** User search for agriculture products like seeds, fertilizers, pesticides, tools, and machinery based on crop type, or region. Filter by price, brand, certification, or availability. View product description, usage, instruction, reviews, and certifications—real time updates on shipping and delivery.

**Values-** Easy discovery of seeds, fertilizers, pesticides, machinery, and other agricultural products. Access to a variety of agriculture products in one platform. Customers can browse and purchase agriculture products anytime, without needing to visit physical stores.

**Answer 2- SWOT**

**Strength Weakness**

1. Easy access to farmers for 1. Limited internet access. Many rural

Wide range of Agricultural farmers may lack stable internet.

Products. 2. Farmers may hesitate to trust online

1. Ability to serve in rural areas platforms for high value purchases.

And remote areas where 3. Delivering products to remote areas

Physical stores may not be can be costly and time consuming.

Available. 4. Lack of human interactions. farmers

1. Orders can be placed any time may prefer face to face consultation

24/7 availability will be there with expert or salespeople.

1. Collection of data for market 5. Difficulty in ensuring that product

Trends, customer behavior, delivered meet the promised standards

And product demand to 6. High initial cost building and

Improve decisions promoting the platform needs a lot

1. Operating online reduce many money at the start.

Expences associated with

Physical store.

**Opportunities Threats**

1. Many governments promote 1. Increase in competition with local

Digital agriculture solution and and global competitors can dominate

May offer funding. the market.

1. Market expansions in related 2. Technology risk like cybersecurity

Areas like crop insurance or threats or technical glitches can

Agriculture loan damage user trust.

1. Collaborating with universities, 3. Farmers may prefer traditional

Cooperatives, or government purchasing methods and resists

Can boost trust and reach. Adopting online platforms.

1. Expanding services to global 4. Economic instability. fluctuation

Market for niche agriculture in agriculture commodity prices or

Products global economic can reduce purchasing

1. Offering training on modern 5. Following local rules for agriculture

Farming techniques. And online sales can be difficult.

**Answer 3 - Feasibility Study**

1. **Hardware requirements**

**1.1 Development Phase**

**Development Machines**- High-performance PCs or laptops with at least Intel i7 or AMD equivalent, RAM 16 GB or more, storage 512GB SSD.

**Server Requirements-**

Application server (Apache Tomcat, wildFly, or GlassFish).

Database Server (PostgreSQL or MySQL hosting on cloud or local server hardware).

Cloud services (AWS EC2, Azure, or Google Cloud for scaling during Development).

* 1. **Deployment Phase**

**Production Environment**

Cloud-based deployment- AWS, Google Cloud, or Azure.

Load balancer and auto-scaling for high-traffic handling.

Backup system for disaster recovery.

**User Devices**

Compatibility with low-end smartphones for rural users with minimal specifications.

1. **Software Requirements**

**2.1 Development Tools**

**Java Frameworks-** Sprint Boot for rest APIs and backend, Hibernate (ORM).

**Frontend Technologies-**  use JSP( java server pages) or integrated with frontend frameworks like Angular/React for a modern UI.

**Database Management-** PostgreSQL or MySQL for relational data.

**Version control-** Git(GitHub, GitLab, or Bitbucket).

* 1. **Testing and Deployment**

**Testing Tools-** JUnit, selenium (for UI testing), Postman (for API testing).

**CI/CD-** Jenkins, GitHub actions, or GitLab CI for continues integration.

* 1. **Scalability Consideration-** Java multi-threading and enterprise capabilities make It suitable for handling high-traffic and concurrent users
1. **Trained Resources**

**3.1 Skill Required**

**Java Developers-** Expertise in Java and frameworks like Spring Boot and Hibernate.

**Frontend Developers-** Familiarity with integrating Java backend with frontend technologies like React or Angular.

**Database Administrators-** Expertise in SQL databases and should have knowledge of schema design for agriculture-related data.

**Cloud Engineers-** skills in developing java applications to cloud environment (AWS, Azure, Google, cloud).

**Testers-** knowledge of Java-based testing tools like Junit.

* 1. **Training Requirements**

Training developers on the specifics of the agriculture domain and a workshop for integrating the app with payment gateways and logistic APIs.

1. **Budget**

**4.1 Development Cost**

**Personnel-** java developers-$5000-$8000/month per developer.

Frontend Developers and Testers-$4000-$7000/month.

**Infrastructure-** Cloud hosting-$500-$1500/month, depending on scale.

Licence fees for open-source tools are $ 1000-$ 5000 for enterprise versions.

**Miscellaneous-** Training and onboarding-$5000-$10000.

* 1. **Estimate Total Budget**

**Initial Development (6 months)-** $75000-$120000.

**Annual maintenance and Scaling-** $20000-$50000.

1. **Timeframe**
	1. **Development Phases**

**Requirement Analysis-** 1-2 months.

**Prototyping-** 1 month.

**Core Development-** 4-6 months.

**Testing and Deployment-** 1-2 months.

* 1. **Total Timeframe**

It takes approximately 9-12 months for a fully functional application.

**Answer- 4 GAP Analysis**

**AS-IS (Existing Process)**

1. **Manual Process**
* Farmers rely on traditional markets, phone calls, or middlemen
* Payments are cash-based, leading to a lack of transparency.
* No centralized platform for listing products or accessing tools.
1. **Limited Digital Presence**
* Farmers and vendors rely on scattered online platforms like WhatsApp group making it unorganized and inefficient for transaction or communication
* No formal application to direct buyer-seller interaction.
1. **Lack of user support**
* No region-specific language support.
* Minimal assistance in resolving issues or answering queries.
1. **Inefficiencies**
* High transaction cost due to intermediaries.
* Users get market prices and trends too late making it hard to make timely decisions.

**TO-BE (Future Process)**

1. **Centralised Digital Platform**
* Unified online marketplace for agriculture products.
* Comprehensive product listing (tools, seeds, fertilizers, etc).
1. **Enhanced Accessibility**
* This platform will support multiple languages, making it easier for rural and regional users to use.
* User-friendly application for easy navigation.
1. **Integrated Payment and Delivery**
* Secure payment gateway for rural and regional users.
* Automated order tracking with real-time updates.
1. **Value-Added Features**
* Ai recommendations for product purchases based on crop type.
* Live weather updates, and market price comparisons.
1. **Transparent and Efficient Process**
* Direct buyers and sellers communication to cut intermediaries.

**Answer 5- Risk Analysis**

**Business Analyst Risk**

1. **Requirements Risk**
* Unclear Requirements – Stakeholders may not provide well-defined or clear requirements.
* Changing Requirements – frequent changes from stakeholders can disrupt project timelines and scope.
* Incomplete Requirements – overlooking key aspects of user needs.
1. **Stakeholders Risks**
* Conflicting Stakeholders Needs – different stakeholders may have conflicting priorities or expectations.
* Low stakeholder engagement – key stakeholders may be unavailable or disengaged leading to delays.
* Miscommunication – misunderstanding between BA and stakeholders due to technical or domain specifications.
1. **Domain Knowledge Risk**
* Lack of agriculture industry knowledge – The BA may not fully understand agriculture practices, processes, or markets.
* Limited user insight - Insufficient understanding of the end-user's challenges, farmers or rural sellers.
1. **Project Delivery Risk**
* Missed deadlines – Delays in gathering or refining requirements causing project delays.
* Dependency on external factors - Dependencies on suppliers, logistics partners, or third-party integrations delaying delivery.
* Resource constraints – Insufficient resources to meet BA responsibilities for example time and budget.
1. **Cultural and Social Risks**
* Rural technology resistance - Rural farmers may be reluctant to use a technology-driven platform.
* Digital divide – limited internet connectivity or low smartphone penetration in certain regions.
* Trust issues - concerns about online payments, product quality, or fraud.

 **Project Risk**

1. **Budget Risks**
* Cost overruns – exceeding the allocated budget due to unforeseen expenses or poor estimation.
* Underfunding - lack of sufficient funds to complete the project.
* Vendor cost – unexpected price hikes from third-party service providers e,g., hosting, logistics, or payment gateways.
1. **Resources Risks**
* Skill gaps – lack of required expertise in the project team e.g., developers, domain experts.
* Resource availability - key members unavailable when needed.
* Overburdened team – team members assigned to multiple tasks, reducing efficiency and quality.
1. **Technology Risks**
* Technology failures – issues with the platform infrastructure, such as crashes or incompatibility.
* Obsolete technology – choosing outdated tools or platforms that may limit scalability or features.
* Integration challenges – difficulties integrating with third-party systems, e.g., payment gateways and logistics providers.
1. **Compliance Risks**
* Regulatory delays – approval delays due to non-compliance with local law.
* Legal issues – mismanagement of intellectual property, licensing, or data privacy.
* Environment regulations – non-compliance with agriculture laws specific to certain regions.
1. **Change Management Risks**
* Resistance to change – users or team members resisting new workflows or systems.
* Training gaps – inadequate training on platform for internal teams or end users.
* Unclear transition plans – poor planning for transitioning from developers to operations.

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

|  |  |  |  |
| --- | --- | --- | --- |
| **RACI** | **NAME** | **DESIGNATION** | **DETAILS** |
| **RESPONSIBLE** | Mr.Dooku | Project Coordinator | Email - dooku12@gmail.comPh.no - 6297856425 |
| Mr.Peter | Farmer | Email – peter123@gmail.comPh.no- 9263547823 |
| Mr.Kevin | Farmer | Email id – kevin345@gmail.comPh.no - 9291493550 |
| Mr.Ben | Farmer | Email – ben10@gmail.comPh.no - 6235897452 |
| Mr. Yash | Business Analyst | Email – yashraj12@gmail.comPh.no - 8341264980 |
| Ms. Juhi | Senior Developer | Email – juhi34@gmail.comPh.no - 8569712356 |
| Mr. John | DB Admin | Email – john67@gmail.comPh.no – 6523410389 |
| Mr. Mike | Network Admin | Email – mike89@gmail.comPh.no - 7356412596 |
| **ACCOUNTABLE** | Mr. Henry | Project sponsor | Email – henry23@gmail.comPh.no - 62035987452 |
| Mr. Pandu  | Financial Head | Email – pandu90@gmail.comPh.no - 8956412375 |
| Mr. Vandana | Project manager | Email – vandana78@gmail.comPh.no - 9203654856 |
| **CONSULTED** | Mr.Dooku | Project Coordinator | Email - dooku12@gmail.comPh.no - 6297856425 |
| Mr. Henry | Project sponsor | Email – henry23@gmail.comPh.no - 62035987452 |
| Mr.Karthik | Delivery Head  | Email – karthik24@gmail.comPh.no - 7325648922 |
| Mr. Yash | Business Analyst | Email – yashraj12@gmail.comPh.no - 8341264980 |
| Mr.Peter | Farmer | Email – peter123@gmail.comPh.no- 9263547823 |
| Mr.Kevin | Farmer | Email id – kevin345@gmail.comPh.no - 9291493550 |
| Mr.Ben | Farmer | Email – ben10@gmail.comPh.no - 6235897452 |
| **INFORMED** | Mr. Henry | Project sponsor | Email – henry23@gmail.comPh.no - 62035987452 |
| Mr. Pandu  | Financial Head | Email – pandu90@gmail.comPh.no - 8956412375 |
| Mr. Vandana | Project manager | Email – vandana78@gmail.comPh.no - 9203654856 |
| Mr.Dooku | Project Coordinator | Email - dooku12@gmail.comPh.no - 6297856425 |

**Answer 7 – Business Case Document**

1. **Project initiation**

The online agriculture product store project aims to create a web and mobile platform to connect farmers in remote areas with agricultural manufacturers for purchasing fertilizers, seeds, and pesticides. This initiative, spearheaded by Mr. Henry and funded by SOONY, seeks to address critical challenges faced by farmers, such as limited access to essential farming products.

The platform will enable seamless communication between farmers and manufacturers, facilitate online purchasing, and provide door-to-door delivery of agricultural products. The project will enhance farming productivity and support sustainable agriculture in rural communities.

1. **Current Problems**
* Farmers in remote areas struggle to access essential agriculture products such as fertilizers, seeds, and pesticides due to a lack of nearby suppliers and poor transportation infrastructure.
* Middlemen increase the cost of products, reducing farmer's profitability.
* Farmers have no platform to compare prices from multiple suppliers.
* Manufacturers often find it challenging to reach farmers in remote areas.
* Delivering products in remote areas is expensive and farmers may face delays in receiving products.
* Farmers in rural areas may lack experience in using online platforms or mobile applications.
* Many remote villages have limited or unreliable internet access, making online transactions challenging.
1. **Project Objective**
* Provide a user-friendly online platform for farmers to purchase fertilizers, seeds, and pesticides.
* Enable direct communication between farmers and manufacturers to ensure cost transparency.
* Overcome logistics challenges for farmers in remote areas by offering delivery services.
* Support manufacturers by providing a direct-to-consumer channel.
1. **Resources Required**
* Project manager
* Business analyst
* Senior developer
* Developer
* Tester
* Database administrator
* Network administrator
* Ui designer
* Support and maintenance
* Trainer
* Developer tools and software
* Infrastructure
1. **Stakeholders**
* Mr. Henry (Sponsor) – drives the vision and funds the project
* Mr. Pandu (financial head) – manages the budget and financial aspects
* Mr. Dooku (Project coordinator) – ensure smooth project execution.
* Famers – End users of the platform.
* Manufacturers – suppliers pf fertilizers, seeds, and pesticides.
* APTIT Solution Team – Responsible for project development and deployment.
1. **Project Timeline**
* Phase 1- Requirement Gathering (1-2 months)
* Phase2- Design (3-4 months)
* Phase 3- Development (5-12 months)
* Phase 4- Testing (13-15 months)
* Phase 5- Deployment (16-17 months)

**Answer 8 – SDLC Methodologies**

1. **Sequential (Waterfall Model)**

The waterfall model is a sequential approach where each phase is completed before moving to the next phase.

**Phases**

1. **Requirements –** Gather all project requirements upfront.
2. **Design –** Create detailed technical specifications.
3. **Implementation –** Develop the software based on the design.
4. **Testing –** Verify the software against requirements.
5. **Deployment –** Release the software to users.
6. **Maintenance –** Address the issues post-deployment.

**Key Features**

* **Predictable –** Easy to understand and manage due to its structured nature.
* **Document Driven –** Heavily relies on documentation at every stage.

**Advantages**

* Clear deliverables at the end of each phase.
* Easy to manage for projects with well-defined requirements.
* Best suited for projects with minimal changes expected.

**Disadvantages**

* No flexibility for changes after the design phase
* Late discovery of issues, as testing happens only after implementation.
* not ideal for projects with evolving requirements.
1. **Iterative (Rational Unified process)**

The rational unified process is an iterative and incremental methodology developed by IBM.

**Phases**

1. **Inception –** Define project scope and key requirements.
2. **Elaboration –** Refine requirements and create an architectural foundation.
3. **Construction –** Develop and build the software iteratively**.**
4. **Transition –** Deliver the product and provide user training and support.

**Key Features**

* Focus on risk management.
* Divide the project into manageable iterations.
* Emphasizes on defining a stable architecture early.

**Advantages**

* Iterative approach reduces risk by addressing issues early.
* Flexible to incorporate changes during development.
* Encourage reusability of components and modular development.

**Disadvantages**

* Requires significant documentation.
* Can be complex for small or less experienced teams.
* High initial effort.
1. **Evolutionary (Spiral Model)**

The spiral model combines elements of both iterative and waterfall approaches and is highly focused on risk management

**Phases**

1. **Planning –** Identify objectives, alternatives, and constraints.
2. **Risk Analysis –** Assess risk and create mitigation strategies.
3. **Engineering –** Develop and test prototypes incrementally.
4. **Evaluation –** Gather user feedback and plan the next iteration.

**Key Features**

* Cyclical nature allows for continuous refinement of the product.
* Risk is assessed and addressed at every iteration.

**Advantages**

* Handles changes due to iterative feedback.
* Risk management is a core focus.
* Suitable for projects with undefined or evolving requirements.

**Disadvantages**

* Requires expert risk analysis, which can be costly.
* Can lead to scope creep if not properly managed.
* Not suitable for small or simple projects.

**4. Scrum (Agile Methodology)**

Scrum is a framework under the agile methodology that emphasizes iterative development, collaboration, and flexibility.

**Key Concepts**

1. **Sprints –** Time-boxed iterations usually 2-4 weeks.
2. **Scrum roles**
* **Product owner –** defines the backlog and priorities.
* **Scrum master –** ensure adherence to scrum practices.
* **Development team** - built the product.

**3. Artifacts**

* **Product backlog –** list of prioritized requirements.
* **Sprint backlog –** task to complete in the sprint.
* **Increment –** the potentially shippable product at the end of each sprint.

**4.** **Ceremonies**

* **Daily stand-ups –** short meetings to discuss progress and backers.
* **Sprint planning –** plan what to accomplish in the sprint.
* **Sprint review –** demonstrate complete work to stakeholders.
* **Sprint retrospective –** reflect on what went well and areas to improve.

**Advantages**

* High flexibility and adaptability to change.
* Continuous delivery of working product.
* Improved collaboration and communication among teams.

**Disadvantages**

* Requires disciplined teams familiar with agile principles.
* Can be less predictable in terms of time and cost.
* Scoop creep can occur without proper backlog management.

**Answer 9 – Choosing The Right SDLC Model for Online Agriculture Product Application.**

**Waterfall Model**

The waterfall model follows a linear, sequential process where each phase is completed before moving to the next.

**Phases**

1. Requirement Analysis

2. System Design

3. implementation

4. Testing

5. Deployment

6. Maintenance

**Best Use Case**

* Small projects with well-defined and stable requirements.
* Low-risk project where changes are unlikely.
* Government or financial system that requires strict Documentation.

**V-Model**

A structured, testing-driven version of the waterfall where each development phase has a corresponding testing phase.

**Phases**

1. Requirement Analysis

2. System Design

3. High-Level Design

4. Low-Level Design

5. Implementation

**Best Use case**

* Project where high reliability is critical (e.g., healthcare, banking, aerospace).
* System requires strong validation and verification.
* Large-scale projects with strict testing and regulatory requirements.

**Final verdict – V-model is better for high-quality, error-free system**

V-model is a better choice in most cases because of its built-in testing process, which ensures fewer defects at later stages.

**Answer 10 – Difference Between The Waterfall Model And the V-model**

|  |  |  |
| --- | --- | --- |
| Criteria | Waterfall Model  | V-Model |
| Development approach | Liner and Sequential | Sequential with parallel verification and validation |
| Testing phase  | Testing is performed after the development phase  | Testing is integrated into each phase alongside the development |
| Focus | Focuses mainly on development | Focuses equally on development and testing |
| Error Detection | Errors are detected late, during the testing phase | Errors are detected early, during each phase |
| Risk Handling | Higher risk due to late testing | Low risk as issues are identified early |
| Flexibility to changes | Difficult to accommodate changes once development starts | Rigid structure also hard to handle changes mid-project |
| Project size | Suitable for small to medium projects | Suitable for large, complex, and high-risk projects |
| Cost of fixing defects | High as bugs are found late | Lower, since bugs are caught early in the development cycle |
| Maintenance  | Maintenance can be challenging due to late defect discovery | Easier maintenance as issues are minimized before deployment |
| Documentation  | Requires detailed documentation  | Requires even more extensive documentation for the testing phase  |
| Design and testing relationship | Design and testing are separate phases | Each design phase has a corresponding testing phase  |
| Feedbacks loops  | Minimal feedback until the testing phase  | Regular feedback due to continuous verification |
| Time consumption | Shorter for simple projects but risky for complex ones  | Longer overall due to parallel testing but more reliable |
| Quality assurance  | Quality is ensured towards the end  | Quality is ensured throughout the development lifecycle  |
| User involvement | Limited user involvement after requirements gathering | Involves user during validation and acceptance testing |
| Model representation  | Liner downward flow step-by-step | v-shaped structure representing development and testing symmetry |
| Dependency  | Each phase depends on the completion of the previous phase | Testing phase corresponds directly to the development phase |
| Suitable for critical system | Not ideal for safety-critical application | Ideal for systems where reliability and safety are critical |
| Example of use  | Simple web applications, small software tools  | Aerospace, defense systems, medical software, financial application |

**Answer 11 – Reasons for choosing one model for this project**

For the development of an online agriculture product store, I recommend using v-model.

**Reasons for choosing the V-model**

**1. Early detections of issues –** the v-model ensures continues testing alongside development, reducing the risk of critical issues post-development.

**2. High focus on quality –** the v-model emphasizes rigorous quality checks at every stage, ensuring a secure application.

**3. Clear requirement validation –** the v-model allows continuous validation against requirements, reducing the chances of requirement mismatches.

**4. Structured and disciplined approach –** a structured framework is crucial for smooth coordination. The v-model provides clear roles and responsibilities at each phase.

**5. Low risk in long-term projects –** the v-model reduces long-term project risk by addressing issues early.

The v-model is more suited for this project due to its emphasis on continuous testing, quality assurance, and risk mitigation. This will ensure that farmers and suppliers receive a reliable, secure, and efficient online platform.

**Answer 12 – Gantt chart**

**Answer 13 – Difference Between Fix Bid and Billing Projects**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **Fixed Bid Project** | **Billing Project** |
| **Definition** | A project with a pre-agreed, fixed price for the entire scope, regardless of the time or resources used. | A project where billing is based on actual hours worked and resources used. |
| **Scope Flexibility** | The scope is clearly defined and fixed changes can lead to renegotiation. | The scope is flexible and can evolve accommodating changes easily |
| **Best Suited For** | A project with well-defined requirements and minimal changes is expected. | Projects with evolving requirements, like software development or R&D projects. |
| **Payments Terms** | Payments are typically milestone-based or upon project completion. | Payment is periodic (weekly/monthly) based on hours/resources used. |
| **Project Management** | Requires detailed planning and precise execution to stay within budget and timeline. | Agile and adaptable, allowing continuous adjustments based on project needs. |
| **Risk** | Higher risk to vendors, if the project overruns, the vendors bear the cost | Higher risk for the client, if the project extends, the client pays more.  |

**Answer 14 – Timesheet of BA in various stages of SDLC**

* **Design Timesheet of a BA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Tasks** | **Actionable Items** | **Start Time** | **End Time** | **Duration** |
| **1.** | Requirement gathering with stakeholders | Met with Peter, Kevin, and Ben. | 09:00 AM | 11:00 AM | 2 hrs |
| **2.** | Drafting Requirement document (BRD) | Finalizing key function needs | 11:30 AM | 1:00 PM | 1.5 hrs |
| **3.** | Team sync-up meeting | Discussed v-model approach | 2:00 PM | 3:00 PM | 1 hr |
| **4.** | Feasibility Study | Analyzing tech options | 3:30 PM | 5:00 PM | 1.5 hrs |

* **Development Timesheet of a BA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Tasks** | **Actionable Item** | **Start Time** | **End Time**  | **Duration** |
| **1.** | Conducted stakeholders interviews | Awaiting inputs from SMEs | 9:00 AM | 11:00 AM | 2 hrs |
| **2.** | Drafted business requirement document (BRD) | Need feedback from the project manager | 11:30 AM | 1:00 PM | 1.5 hrs |
| **3.** | Collaborated with the UI team on wireframes | Approved initial wireframes | 2:00 PM | 3:00 PM | 1 hr |
| **4.** | Reviewed test cases | Suggested revisions for UAT scenarios | 3:30 PM | 5:00 PM | 1.5 hrs |

* **Testing Timesheet of a BA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Tasks** | **Actionable Item** | **Start Time** | **End Time** | **Duration** |
| **1.** | Reviewed UAT test for completeness | The test case align with the requirement | 9:00 AM | 11:00 AM | 2 hrs |
| **2.** | Conduct UAT for product search functionality | Reported issues to the QA team | 11:30 AM | 1:00 PM | 1.5 hrs |
| **3.** | Analysed defects logged in JIRA | Coordinated with the development team | 2:00 PM | 3:00 PM | 1 hr |
| **4.** | Validated defect fixes and retested features | Follow-up with developers needes | 3:30 PM | 5:00 PM | 1.5 hrs |

* **UAT Timesheet of a BA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Tasks** | **Actionable Item** | **Start Time** | **End Time** | **Duration** |
| **1.** | Scheduled UAT sessions with stakeholders | UAT schedule shared with the team  | 09:00 AM | 10:00 AM | 1 hr |
| **2.** | Tested product ordering workflow  | Issued logged in JIRA | 10:15 AM | 12:15 PM | 2 hrs |
| **3.** | Documented UAT findings and reported to QA | Sent defect summary to QA team | 01:00 PM | 02:00 PM | 1 hr |
| **4.** | Reviewed feedback from testers and stakeholders | Coordinating fixes with the development team  | 02:15 PM | 03:45 PM | 1.5 hrs |
| **5.** | Retested after fixes and closed resolved issues | Escalated to development team | 04:00 PM | 05:00 PM | 1 hr |

* **Deployment and Implementation Timesheet of a BA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Tasks** | **Actionable Item** | **Start Time** | **End Time** | **Duration** |
| **1.** | Final review of the deployment checklist | Checklist sent to the project team | 09:00 AM | 10:00 AM | 1 hr |
| **2.** | Coordinated with developers and stakeholders for the rollout | Resolved with network admin  | 10:15 AM | 12:00 PM | 1.75 hrs |
| **3.** | Supported UAT sign-off and validated deployment | Issued escalated to the development team  | 12:30 PM | 02:00 PM | 1.5 hrs |
| **4.** | Conducted user training sections and created manuals  | Training material shared with users  | 02:30 PM | 04:00 PM | 1.5 hrs |
| **5.** | Gather feedback from end-users and stakeholders  | Scheduling follow-up meetings | 04:15 PM | 05:15 PM | 1 hr |