**Capstone 2 |Scrum Project Name: Scrum Foods (Foods Delivery Applications)**

**Q 1 – Write Agile Manifesto**

**Answer**

**Agile Manifesto**

**4 Core 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

**12 Principles of Agile**

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

**Q 2. Write minimum 40 User stories and their Acceptance Criteria along with their BV and CP**

|  |
| --- |
| User Story No 1 |
| Tasks 02  Priority High |
| AS A CUSTOMER  I WANT TO REGISTER AND LOGIN  SO THAT I CAN ACCESS THE APPLICATION |
| BV 500  CP 02 |
| Acceptance Criteria   1. The user must provide valid credentials (email/phone and password). 2. Upon successful registration, a confirmation email/SMS is sent. 3. Login redirects users to their personalized homepage. |

|  |
| --- |
| User Story No 2 |
| Tasks 03  Priority Medium |
| AS A CUSTOMER  I WANT TO SEARCH AND VIEW RESTAURANTS  SO THAT I CAN CHOOSE WHERE TO ORDER FOOD |
| BV 500  CP 05 |
| Acceptance Criteria   1. Users can filter restaurants by cuisine, price, and location. 2. Restaurant details must include ratings, menu preview, and operating hours. 3. Search results are displayed with restaurant names and thumbnails. |

|  |
| --- |
| User Story No 3 |
| Tasks 02  Priority High |
| AS A CUSTOMER  I WANT TO PLACE AN ORDER  SO THAT I CAN RECEIVE MY FOOD |
| BV 1000  CP 05 |
| Acceptance Criteria   1. Users must be able to add items to their cart. 2. Order placement should be confirmed with an order ID and ETA. |

|  |
| --- |
| User Story No 4 |
| Tasks 02  Priority Medium |
| AS A CUSTOMER  I WANT TO TRACK MY ORDER ON A MAP  SO THAT I CAN MONITOR DELIVERY IN REAL TIME |
| BV 100  CP 08 |
| Acceptance Criteria   1. The map updates dynamically to show the delivery location. 2. An ETA must be displayed along with order status ("On the Way," etc.). |

|  |
| --- |
| User Story No 5 |
| Tasks 02  Priority Low |
| AS A CUSTOMER  I WANT TO PROVIDE FEEDBACK  SO THAT I CAN RATE MY EXPERIENCE |
| BV 50  CP 02 |
| Acceptance Criteria   1. A feedback form must include rating stars and a text box for comments. 2. Feedback must be submitted successfully with a confirmation message. |

|  |
| --- |
| User Story No 6 |
| Tasks 03  Priority Medium |
| AS A DELIVERY BOY  I WANT TO VIEW MY ASSIGNED ORDERS  SO THAT I CAN PLAN MY DELIVERIES |
| BV 100  CP 03 |
| Acceptance Criteria   1. Assigned orders are displayed in a list with order IDs and addresses. 2. Delivery details include pickup and drop-off points. |

|  |
| --- |
| User Story No 7 |
| Tasks 02  Priority High |
| AS A DELIVERY BOY  I WANT TO UPDATE ORDER STATUS  SO THAT CUSTOMERS CAN TRACK THEIR ORDERS IN REAL TIME |
| BV 500  CP 05 |
| Acceptance Criteria   1. Status updates must include "Picked Up," "On the Way," and "Delivered." 2. The updated status must reflect instantly in the customer’s app. |

|  |
| --- |
| User Story No 8 |
| Tasks 03  Priority High |
| AS A RESTAURANT  I WANT TO VIEW AND MANAGE NEW ORDERS  SO THAT I CAN PREPARE FOOD ON TIME |
| BV 1000  CP 05 |
| Acceptance Criteria   1. New orders are displayed with customer details and delivery time. 2. Restaurants can update order preparation status. |

|  |
| --- |
| User Story No 9 |
| Tasks 02  Priority Medium |
| AS AN ADMIN  I WANT TO MANAGE CUSTOMER FEEDBACK  SO THAT ISSUES ARE ADDRESSED PROMPTLY |
| BV 100  CP 08 |
| Acceptance Criteria   1. Feedback must be categorized based on keywords (e.g., delivery, quality). 2. Admins can mark feedback as resolved after taking action. |

|  |
| --- |
| User Story No 10 |
| Tasks 02  Priority High |
| AS A BUSINESS OWNER  I WANT TO VIEW SALES REPORTS  SO THAT I CAN ANALYZE BUSINESS PERFORMANCE |
| BV 1000  CP 05 |
| Acceptance Criteria   1. Sales reports must be downloadable in PDF or Excel format. 2. Reports should include revenue, orders completed, and customer feedback summary. |

|  |
| --- |
| User Story No 11 |
| Tasks 03  Priority High |
| AS A CUSTOMER  I WANT TO BROWSE A RESTAURANT MENU  SO THAT I CAN SELECT ITEMS TO ORDER |
| BV 500  CP 05 |
| Acceptance Criteria   1. The menu must display categories (e.g., appetizers, mains, desserts). 2. Each item should include a name, description, price, and availability. 3. Customers can add items to their cart from the menu. |

|  |
| --- |
| User Story No 12 |
| Tasks 02  Priority Low |
| AS A CUSTOMER  I WANT TO CANCEL MY ORDER  SO THAT I CAN MAKE CHANGES IF NEEDED |
| BV 10  CP 03 |
| Acceptance Criteria   1. Cancellation must be allowed before order preparation starts. 2. A cancellation confirmation should be displayed, and customers must receive a refund (if applicable). |

|  |
| --- |
| User Story No 13 |
| Tasks 03  Priority High |
| AS A CUSTOMER  I WANT TO MAKE A PAYMENT  SO THAT MY ORDER IS CONFIRMED |
| BV 1000  CP 05 |
| Acceptance Criteria   1. Payment options must include credit/debit cards, UPI, and wallets. 2. A payment confirmation screen must display the order total and payment method. 3. On successful payment, customers must receive an order confirmation message. |

|  |
| --- |
| User Story No 14 |
| Tasks 02  Priority Medium |
| AS A DELIVERY BOY  I WANT TO MARK ORDERS AS PICKED UP  SO THAT CUSTOMERS KNOW THE ORDER IS EN ROUTE |
| BV 100  CP 02 |
| Acceptance Criteria   1. The delivery boy app must display a button to mark the order as "Picked Up." 2. The status update must reflect instantly in the customer's app. |

|  |
| --- |
| User Story No 15 |
| Tasks 03  Priority High |
| AS A RESTAURANT  I WANT TO VERIFY THE DELIVERY BOY  SO THAT ONLY AUTHORIZED PERSONNEL PICK UP ORDERS |
| BV 500  CP 05 |
| Acceptance Criteria   1. Restaurants must see the delivery boy's details (name, photo, ID). 2. Restaurants can report unauthorized pickup attempts. |

|  |
| --- |
| User Story No 16 |
| Tasks 02  Priority Medium |
| AS A RESTAURANT  I WANT TO VIEW FEEDBACK FROM CUSTOMERS  SO THAT I CAN IMPROVE MY SERVICE |
| BV 100  CP 03 |
| Acceptance Criteria   1. Feedback must be displayed with the order details (e.g., date, order ID). 2. Restaurants can sort feedback by ratings (highest to lowest). |

|  |
| --- |
| User Story No 17 |
| Tasks 02  Priority Low |
| AS A REGIONAL ADMIN  I WANT TO TRACK DELIVERY BOYS’ LOCATIONS  SO THAT I CAN MONITOR THEIR PERFORMANCE |
| BV 20  CP 08 |
| Acceptance Criteria   1. The admin panel must display live location tracking for all active delivery boys. 2. The panel must show the delivery boy's current order and ETA. |

|  |
| --- |
| User Story No 18 |
| Tasks 03  Priority Medium |
| AS A REGIONAL ADMIN  I WANT TO MANAGE CUSTOMER REFUNDS  SO THAT ISSUES CAN BE RESOLVED QUICKLY |
| BV 100  CP 05 |
| Acceptance Criteria   1. Refund requests must display customer details and order ID. 2. The admin must approve or reject refunds with appropriate reasoning. 3. Approved refunds must trigger an automated payment process. |

|  |
| --- |
| User Story No 19 |
| Tasks 03  Priority Low |
| AS AN ADMIN  I WANT TO APPROVE OR REJECT NEW RESTAURANT REQUESTS  SO THAT ONLY VERIFIED RESTAURANTS JOIN THE PLATFORM |
| BV 10  CP 05 |
| Acceptance Criteria   1. Requests must include restaurant name, license details, and contact information. 2. Admins can approve or reject requests with comments. |

|  |
| --- |
| User Story No 20 |
| Tasks 02  Priority Medium |
| AS A BUSINESS OWNER  I WANT TO UPDATE PAYMENTS FOR RESTAURANTS  SO THAT THEY RECEIVE THEIR EARNINGS PROMPTLY |
| BV 100  CP 08 |
| Acceptance Criteria   1. The system must display pending payments with restaurant details. 2. Payments must be processed securely and receipts generated. |

|  |
| --- |
| User Story No 21 |
| Tasks 02  Priority Medium |
| AS A CUSTOMER  I WANT TO LOGOUT SECURELY  SO THAT MY ACCOUNT REMAINS SAFE |
| BV 20  CP 02 |
| Acceptance Criteria   1. Users must be logged out of all sessions upon clicking the logout button. 2. A confirmation message must be displayed once the user logs out. |

|  |
| --- |
| User Story No 22 |
| Tasks 03  Priority Low |
| AS A DELIVERY BOY  I WANT TO VIEW MY EARNINGS  SO THAT I CAN TRACK MY INCOME |
| BV 50  CP 05 |
| Acceptance Criteria   1. Earnings must be displayed with a breakdown by date and order. 2. Filters should allow viewing weekly, monthly, or custom date ranges. 3. The total earnings should be calculated automatically. |

|  |
| --- |
| User Story No 23 |
| Tasks 02  Priority Medium |
| AS A DELIVERY BOY  I WANT TO RAISE ISSUES WITH MY ORDERS  SO THAT PROBLEMS CAN BE RESOLVED QUICKLY |
| BV 100  CP 03 |
| Acceptance Criteria   1. Delivery boys can select an issue type (e.g., incorrect address, unreachable customer). 2. Issues must be recorded and flagged for admin review. |

|  |
| --- |
| User Story No 24 |
| Tasks 03  Priority High |
| AS A CUSTOMER  I WANT TO RECEIVE ORDER NOTIFICATIONS  SO THAT I AM INFORMED ABOUT THE ORDER STATUS |
| BV 500  CP 05 |
| Acceptance Criteria   1. Notifications must be sent for key updates (e.g., order confirmed, food prepared, out for delivery). 2. Notifications must include the order ID and status. |

|  |
| --- |
| User Story No 25 |
| Tasks 03  Priority High |
| AS AN ADMIN  I WANT TO RESOLVE RAISED ISSUES  SO THAT OPERATIONS RUN SMOOTHLY |
| BV 1000  CP 08 |
| Acceptance Criteria   1. Issues must be categorized by type (e.g., customer, delivery boy, restaurant). 2. Admins can view issue details, take action, and mark issues as resolved. |

|  |
| --- |
| User Story No 26 |
| Tasks 02  Priority Medium |
| AS A REGIONAL ADMIN  I WANT TO MANAGE RESTAURANT ISSUES  SO THAT RESTAURANTS CAN PERFORM BETTER |
| BV 100  CP 05 |
| Acceptance Criteria   1. Restaurant issues must include details like order IDs and customer feedback. 2. Regional admins must be able to mark the issues as resolved or escalate them. |

|  |
| --- |
| User Story No 27 |
| Tasks 02  Priority Medium |
| AS A CUSTOMER  I WANT TO APPLY PROMO CODES  SO THAT I CAN GET DISCOUNTS ON MY ORDERS |
| BV 500  CP 05 |
| Acceptance Criteria   1. Customers must be able to enter promo codes at checkout. 2. Discounts must be applied, and the final order total must update accordingly. |

|  |
| --- |
| User Story No 28 |
| Tasks 03  Priority High |
| AS A BUSINESS OWNER  I WANT TO VIEW DETAILED REPORTS  SO THAT I CAN ANALYZE REGIONAL SALES AND REVENUE |
| BV 500  CP 08 |
| Acceptance Criteria   1. Reports must show revenue, orders, and refunds for each region. 2. Data must be filterable by date, region, and order type. 3. Reports must be downloadable in Excel format. |

|  |
| --- |
| User Story No 29 |
| Tasks 02  Priority Low |
| AS A CUSTOMER  I WANT TO SAVE MY FAVORITE RESTAURANTS  SO THAT I CAN ACCESS THEM QUICKLY IN THE FUTURE |
| BV 10  CP 03 |
| Acceptance Criteria   1. Customers must be able to mark a restaurant as a favourite. 2. A "Favourites" list must display all marked restaurants. |

|  |
| --- |
| User Story No 30 |
| Tasks 03  Priority Low |
| AS A CUSTOMER  I WANT TO SELECT A DELIVERY SLOT  SO THAT I CAN RECEIVE ORDERS AT MY PREFERRED TIME |
| BV 50  CP 05 |
| Acceptance Criteria   1. Customers must be able to choose a time slot during checkout. 2. A confirmation message must include the chosen slot. 3. Delivery availability must be validated for the selected time slot. |

|  |
| --- |
| User Story No 31 |
| Tasks 03  Priority Medium |
| AS A RESTAURANT  I WANT TO RAISE ISSUES WITH MY ORDERS  SO THAT I CAN SEEK RESOLUTION FROM THE ADMIN |
| BV 100  CP 05 |
| Acceptance Criteria   1. Restaurants must be able to select an issue type (e.g., payment delay, incorrect order details). 2. Issues must be recorded and flagged for admin resolution. 3. Notifications must confirm the issue has been submitted. |

|  |
| --- |
| User Story No 32 |
| Tasks 02  Priority Medium |
| AS A CUSTOMER  I WANT TO VIEW MY ORDER HISTORY  SO THAT I CAN TRACK MY PREVIOUS PURCHASES |
| BV 100  CP 03 |
| Acceptance Criteria   1. Customers must see a list of previous orders with details like date, restaurant name, and total cost. 2. Users can filter the history by date or restaurant. |

|  |
| --- |
| User Story No 33 |
| Tasks 02  Priority Medium |
| AS A DELIVERY BOY  I WANT TO RECEIVE OPTIMIZED DELIVERY ROUTES  SO THAT I CAN DELIVER ORDERS QUICKLY AND EFFICIENTLY |
| BV 100  CP 08 |
| Acceptance Criteria   1. Delivery boys must receive a real-time optimized route based on their location and delivery address. 2. Routes must update dynamically if there are delays or blockages. |

|  |
| --- |
| User Story No 34 |
| Tasks 03  Priority Medium |
| AS A CUSTOMER  I WANT TO RECEIVE FOOD RECOMMENDATIONS  SO THAT I CAN DISCOVER NEW MEALS |
| BV 100  CP 08 |
| Acceptance Criteria   1. Recommendations must appear based on user preferences, past orders, and popular items. 2. Customers can choose to add recommended items directly to their cart. |

|  |
| --- |
| User Story No 35 |
| Tasks 02  Priority Medium |
| AS A REGIONAL ADMIN  I WANT TO VIEW REGIONAL DELIVERY PERFORMANCE  SO THAT I CAN IMPROVE DELIVERY OPERATIONS |
| BV 100  CP 05 |
| Acceptance Criteria   1. Performance metrics must include average delivery time, delivery success rate, and complaints. 2. Admins can filter performance by date and delivery boy. |

|  |
| --- |
| User Story No 36 |
| Tasks 02  Priority Medium |
| AS AN ADMIN  I WANT TO REVIEW PAYMENT TRANSACTIONS  SO THAT I CAN ENSURE ACCURACY IN PAYOUTS |
| BV 100  CP 05 |
| Acceptance Criteria   1. Payment transactions must include details such as order ID, payment method, and amount. 2. Admins must be able to filter and export payment reports. |

|  |
| --- |
| User Story No 37 |
| Tasks 03  Priority Medium |
| AS A CUSTOMER  I WANT TO CONTACT CUSTOMER SUPPORT  SO THAT I CAN RESOLVE ISSUES WITH MY ORDERS |
| BV 100  CP 05 |
| Acceptance Criteria   1. Customers must have options to chat, call, or email customer support. 2. Each contact method must generate a ticket ID for reference. 3. Customers must receive a notification when their issue is resolved. |

|  |
| --- |
| User Story No 38 |
| Tasks 02  Priority Medium |
| AS A RESTAURANT  I WANT TO UPDATE ITEM AVAILABILITY  SO THAT CUSTOMERS ONLY SEE ITEMS THEY CAN ORDER |
| BV 100  CP 03 |
| Acceptance Criteria   1. Restaurants must have a toggle to mark menu items as "Available" or "Unavailable." 2. Changes must reflect instantly in the app for customers. |

|  |
| --- |
| User Story No 39 |
| Tasks 02  Priority Low |
| AS A DELIVERY BOY  I WANT TO LOGOUT SECURELY  SO THAT MY DATA IS NOT COMPROMISED |
| BV 20  CP 02 |
| Acceptance Criteria   1. All active sessions must be cleared upon logout. 2. A confirmation message must be displayed after logout. |

|  |
| --- |
| User Story No 40 |
| Tasks 03  Priority High |
| AS A BUSINESS OWNER  I WANT TO SET REVENUE TARGETS FOR EACH REGION  SO THAT I CAN TRACK PERFORMANCE EFFECTIVELY |
| BV 1000  CP 08 |
| Acceptance Criteria   1. Targets must be set by region and displayed on a dashboard. 2. A progress bar must show real-time progress against the target. 3. Owners must receive notifications when targets are met or exceeded. |

**Q3. What is Epic and write 2 Epics**

**Answer**

An Epic is a large, high-level user story or a collection of related user stories that represents a significant chunk of functionality or a broader business need. Epics are typically too big to be completed in a single sprint and are broken down into Themes and smaller, actionable User Stories for execution.

**Epic 1: Ratings and Reviews**

**Description:** Enable customers to share their experiences by rating and reviewing restaurants and delivery services, ensuring transparency and continuous improvement in service quality. Ratings and reviews will also help other customers make informed decisions.

**User Stories**

* As a customer, I want to rate my recent order so that I can provide feedback on the food and delivery service.
* As a customer, I want to write a review about the restaurant so that I can share detailed feedback with other users.
* As a restaurant, I want to view customer reviews so that I can understand and improve the quality of my service.
* As an admin, I want to monitor customer reviews so that I can address any flagged concerns or inappropriate content.
* As a customer, I want to view the average restaurant rating so that I can choose a reliable place to order from.

**Acceptance Criteria**

* Customers must see a rating option (e.g., 1 to 5 stars) after completing an order.
* A confirmation message must appear once the rating is submitted.
* Customers must have a text box for entering their review after providing a rating.
* Reviews must be stored and associated with the order ID and restaurant.
* Restaurants must see all customer reviews with associated ratings, date, and order ID.
* Reviews must be sorted by most recent by default, with a filter for ratings.
* Admins must see flagged reviews in a separate dashboard.
* Admins must be able to take actions such as approving, hiding, or deleting flagged reviews.
* Restaurant listings must display their average star rating (calculated from all submitted ratings).
* The average rating must update dynamically when new ratings are submitted.

**Epic 2: Seamless Multi-Device Support**

**Description:** Provide users with the ability to seamlessly use the Scrum Foods platform across multiple devices, ensuring consistent experiences and data synchronization for better usability and flexibility.

**User Stories**

* As a customer, I want my account to synchronize across devices so that I can access the same information and orders on all my devices.
* As a delivery boy, I want my assigned orders to appear on any device I log in from so that I can continue deliveries without interruptions.
* As a restaurant, I want to manage orders from both desktop and mobile devices so that I can operate seamlessly in different scenarios.
* As an admin, I want a unified dashboard accessible from all devices so that I can monitor platform operations anytime and anywhere.

**Acceptance Criteria**

* Customer data (e.g., order history, cart, favourites) must update in real-time when accessed from another device.
* Notifications and alerts must sync across all logged-in devices.
* Delivery order data must automatically sync when the delivery boy logs in from a new device.
* Active delivery status must remain consistent across devices.
* Order management features must be fully functional on both desktop and mobile views.
* Any updates made on one device must reflect immediately on the other.
* The admin dashboard must retain all features and real-time data updates across devices.
* Role-based access settings must remain consistent regardless of the device being used.

**Q4. What is the difference between BV and CP**

**Answer**

1. Business Value

* Definition: Measures the impact or benefit that a feature, user story, or project delivers to the organization or end-users.
* Purpose: Helps prioritize features based on their potential return on investment (ROI), customer satisfaction, or strategic alignment.
* Measurement: Often subjective and assigned by Product Owners, stakeholders, or business teams based on expected benefits.
* Scale: Can be measured on a relative scale (e.g., 1-10) or using frameworks like WSJF (Weighted Shortest Job First).
* Example: A feature that increases customer retention may have high business value.

2. Complexity Points

* Definition: Measures the difficulty, effort, and risk involved in implementing a feature or user story.
* Purpose: Helps development teams estimate the work required to complete a task.
* Measurement: Typically estimated using Story Points (e.g., Fibonacci sequence: 1, 2, 3, 5, 8, etc.), factoring in uncertainty, risk, and required effort.
* Scale: Higher values indicate more complex and time-consuming work.
* Example: A feature that requires integrating multiple third-party APIs may have high complexity points.

**Key Differences**

| Aspect | Business Value | Complexity Points |
| --- | --- | --- |
| Measures | Benefit/Impact | Effort/Difficulty |
| Assigned By | Product Owner, Stakeholders | Development Team |
| Purpose | Prioritization | Estimation of effort |
| Scale Used | Relative impact (e.g., 1-10, WSJF) | Story Points (Fibonacci) |
| Example | Feature that boosts revenue | Feature requiring complex API integration |

**Q5 Question 5 –Explain about Sprint**

**Answer**

**SPRINT**

In software development, a sprint is a fundamental concept in Agile methodologies, particularly Scrum. It refers to a short, fixed timeframe – typically between one to four weeks – during which a development team focuses on completing a specific set of tasks.

**Features of Sprint**

* **Time-boxed:** Sprints have a predetermined length that remains consistent throughout the entire project. This fixed duration ensures focus and promotes timely delivery.
* **Iterative:** Multiple sprints make up a software development project. Each sprint builds upon the previous one, with the final product incrementally taking shape.
* **Goal-oriented:** Each sprint has a specific goal, which is a set of functionalities or features targeted for completion within the sprint timeframe.

**Components of a Sprint**

* **Time Frame:** A sprint typically has a fixed duration, often ranging from 1 to 4weeks. The duration is consistent across all sprints to provide a predictable cadence for development and planning.
* **Goals and Objectives:** At the beginning of each sprint, the development team, along with stakeholders, selects a set of user stories, features, or tasks to work on during that sprint. These items are collectively referred to as the sprint backlog.
* **Planning:** During sprint planning, the development team breaks down the selected items from the product backlog into smaller tasks and estimates the effort required for each task. The team commits to completing these tasks within the sprint duration.
* **Daily Stand-ups:** Throughout the sprint, the team holds daily stand-up meetings (also known as daily scrums) to discuss progress, obstacles, and plans. Each team member shares what they've accomplished, what they're working on, and any challenges they're facing. These meetings foster communication and collaboration.
* **Development:** The development team works on the tasks identified in the sprint backlog. They collaborate closely, often using techniques like pair programming and frequent code reviews to ensure high-quality work.
* **Continuous Integration:** Developers integrate their code changes into the main codebase regularly, ensuring that the software remains functional and stable throughout the sprint.
* **Testing:** Testing is an integral part of a sprint. Automated tests are run to validate code changes, and manual testing may be conducted to ensure the quality of the software.
* **Review and Demo:** At the end of the sprint, the development team conducts a sprint review and demo. They showcase the completed work to stakeholders, gathering feedback and validation. This helps ensure that the delivered features align with expectations.
* **Retrospective:** Following the review and demo, the team holds a sprint retrospective. They reflect on what went well during the sprint, what could be improved, and actions to take in the next sprint. The retrospective encourages continuous improvement.
* **Incremental Development:** Each sprint results in a potentially shippable product increment, meaning that at the end of each sprint, a new version of the software is available with additional features or improvements.
* **Adaptability:** Agile methodologies emphasize adaptability and the ability to respond to changing requirements. If new priorities or insights emerge, adjustments can be made in subsequent sprints.

Sprints allow development teams to iteratively deliver value to customers and stakeholders in a controlled and predictable manner. By breaking down the work into manageable chunks and continuously seeking feedback, Agile teams can enhance collaboration, reduce risk, and improve the overall quality of the software being developed.

**Question 6- Explain Product Backlog and Sprint Backlog**

**Answer**

The **Product Backlog** is a dynamic, prioritized list of all the features, user stories, enhancements, bug fixes, and other work items that need to be addressed over the course of a project. It represents the entire scope of the product's development and is managed by the Product Owner. The Product Backlog is continually refined and updated based on feedback, changing requirements, and new insights.

The **Sprint Backlog** is a subset of the Product Backlog that represents the work that the development team commits to completing during a specific time period called a "Sprint." A Sprint is a fixed-duration iteration, usually lasting two to four weeks, in which the team works on the set of items from the product backlog. The sprint backlog is created during the sprint planning meeting, where the development team selects the set of items to work on based on their capacity and priority set by the product owner

|  |  |  |
| --- | --- | --- |
| Feature | Product Backlog | Sprint Backlog |
| Scope | Entire product development | A single sprint (short iteration) |
| Content | Features, user stories, bug fixes, improvements (high-level) | Selected items from product backlog, broken down into tasks (detailed) |
| Goal | Overall product vision and goals | Specific goals for the current sprint |
| Ownership | Product Owner | Development Team (with input from Product Owner) |
| Priority | Prioritized based on product vision and value | Prioritized based on sprint goal and effort |
| Change | Dynamic, can change throughout the project | Fixed for the duration of the sprint (minor adjustments possible) |
| Level of Detail | High-level overview | Detailed breakdown of tasks and user stories |
| Example | "Develop a new login system" | "Create user interface mockups for login page," "Implement backend logic for login functionality" |

**Question 7- What is Impediment Log? Write 2 Impediment Logs.**

**Answer**

An **Impediment Log** in Agile is a tool used to track and address obstacles that slow down a team's progress. It acts like a record of "blockers" that prevent the team from completing work on time and achieving their goals.

**Key points about Impediment Logs**

**Purpose:** Identify and remove roadblocks that hinder the team's performance.

**What it captures:** Any challenges or issues that slow down the team, such as missing information, dependency on external approvals, or unclear requirements.

**Benefits:** Improves team transparency, promotes proactive problem-solving, and helps maintain team velocity within a sprint.

Impediment Logs are often reviewed during Agile ceremonies like daily stand-up meetings or sprint retrospectives. The Scrum Master, who is responsible for facilitating the Agile process, typically owns the Impediment Log and works to resolve the issues listed.

|  |  |
| --- | --- |
| Field | Description |
| Log ID | 1 |
| Date | 19/03/2025 |
| Impediment | Delivery partner shortage |
| Impact | \* Increased delivery times due to fewer drivers available to pick up orders. \* Potential order cancellations if drivers cannot be found within a reasonable timeframe. \* Negative impact on customer satisfaction due to delayed deliveries. |
| Priority | High |
| Action Taken | \* Increase recruitment efforts (e.g., adjust incentives, revise job descriptions, utilize targeted advertising). \* Partner with staffing agencies specializing in delivery personnel. \* Explore offering bonuses or incentives to existing delivery partners for picking up additional shifts. \* Investigate outsourcing deliveries to a third-party logistics provider (3PL) as a temporary solution. |
| Assigned to | \* Recruitment Team (for increased recruitment efforts). \* Operations Team (for exploring partnerships & 3PL options). |
| Next Steps | \* Monitor the effectiveness of implemented solutions. \* Track key metrics like delivery times and customer satisfaction. \* Evaluate the long-term viability of each solution based on cost and effectiveness. \* Conduct a cost-benefit analysis of potential 3PL partnerships. |
| Status | Open |

**2 Impediment Logs**

|  |  |
| --- | --- |
| Field | Description |
| Log ID | 2 |
| Date | 20/03/2025 |
| Impediment | Discrepancies between restaurant menus on the app and actual restaurant offerings. |
| Impact | \* Frustrated customers who receive incorrect or unavailable items. \* Increased order cancellations and refund requests. \* Negative impact on restaurant ratings and customer satisfaction. \* Potential loss of revenue for restaurants due to inaccurate menu pricing. |
| Priority | High |
| Action Taken | \* Implement a system for restaurants to update menus in real-time or on a regular schedule. \* Develop an automatic notification system to alert customers of menu changes that might affect their order. \* Increase communication with restaurants to ensure menu accuracy. \* Explore integrating with restaurant Point-of-Sale (POS) systems for automatic menu updates. |
| Assigned to | \* Product Team (to develop update system and notifications). \* Operations Team (to improve communication with restaurants). \* Integration Team (to explore POS system integration). |
| Next Steps | \* Pilot test the updated menu system with a small group of restaurants. \* Gather feedback from both customers and restaurants on the new system. \* Refine the system based on feedback before wider rollout. \* Continuously monitor menu accuracy and customer satisfaction. |
| Status | Open |

**Question 8- Explain Velocity of the team.**

**Answer**

**Velocity** is a metric used to measure the amount of work a development team can accomplish within a specific Sprint in Agile Scrum.

**Explanation of Working:** Velocity is calculated by adding up the completed user stories within a Sprint. Velocity only considers all tasks that were completed and delivered and not any tasks that are in progress or incomplete. The only story points included in calculating Velocity are those user stories marked and accepted as ‘Done’

**Example:** If a scrum team completed tasks total of 80 story points in a single Sprint, then the Velocity for that Sprint is 80.

**Steps in Calculating Velocity**

1. **Define the Unit of Measurement:** Velocity is generally calculated in Story Points. Story Points can be measured by different types such as T-Shirt Sizing, Fibonacci Sequence, and so on. Identify the right type and use it to measure story points.
2. **Start a Sprint**: Velocity is calculated based on a Sprint and Sprint duration. So, a Sprint is the starting point for Velocity with the Story Points assigned to user stories in a Sprint
3. **Work on User Stories**: The development team works on user stories assigned to a Sprint and as the user stories are completed the Story Points associated with each user story contribute towards the total Velocity.
4. **End of Sprint**: At the end of the Sprint the user stories completed in this Sprint are verified for ‘Done’ status and the Story Points are summed up for the completed items.
5. **Calculate Velocity**: Velocity is calculated based on the total number of story points completed during the Sprint. This total is considered the team’s Velocity for the specific Sprint.

**Uses of Velocity**

1. Velocity helps teams in forecasting the continuous improvement that can be accomplished in future Sprints.
2. Velocity also assists in planning the upcoming Sprint and setting realistic goals based on the story points accomplished in the previous Sprints.
3. Velocity provides great insight into an Agile Scrum team’s work capacity and ability to complete the amount of work in future Sprints, which aids in better planning.

**Average velocity**

Average velocity in Agile development refers to the average amount of work, measured in story points, that a team completes during a series of sprints or iterations. It's a key metric used for planning and estimating future work.

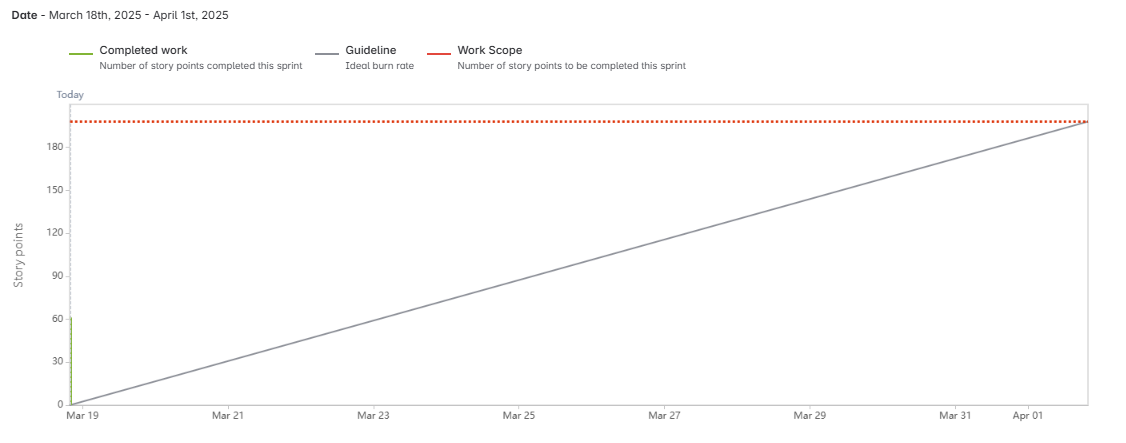
**How to calculate average velocity**

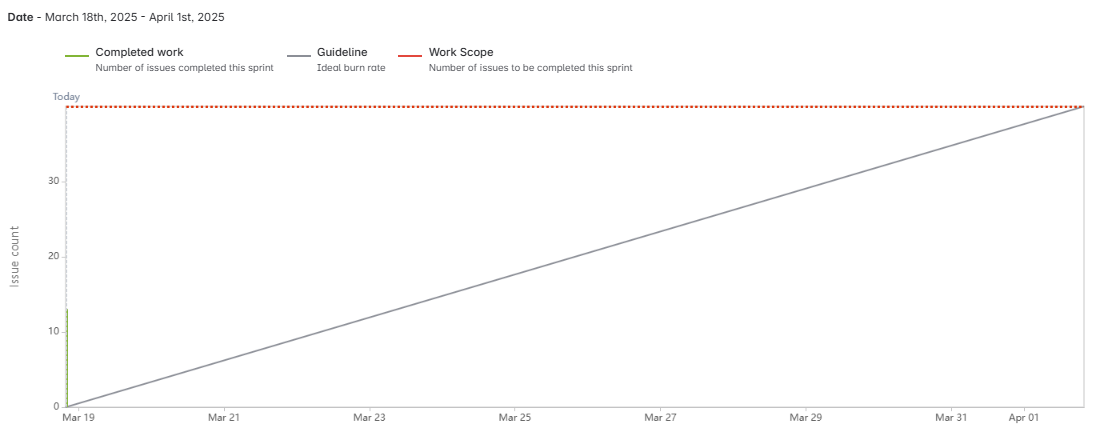
* **Select a Time Frame:** Choose a specific number of past sprints or iterations for which you want to calculate the average velocity. For example, you might choose the last 5 sprints.
* **Sum Completed Story Points:** Add up the total story points completed by the team in each of the selected sprints. This will give you the total completed work for the chosen time frame.
* **Calculate Average:** Divide the total completed story points by the number of sprints or iterations you selected. This will give you the average velocity for that period.

**Formula: Average Velocity = Total Completed Story Points / Number of Sprints**

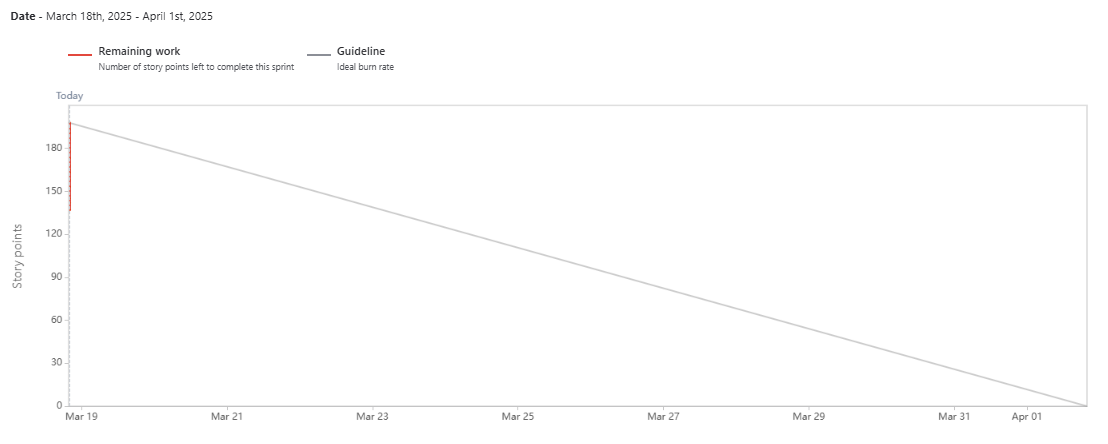
**9. Draw Sprint Burn Charts and Product Burn Down Charts**

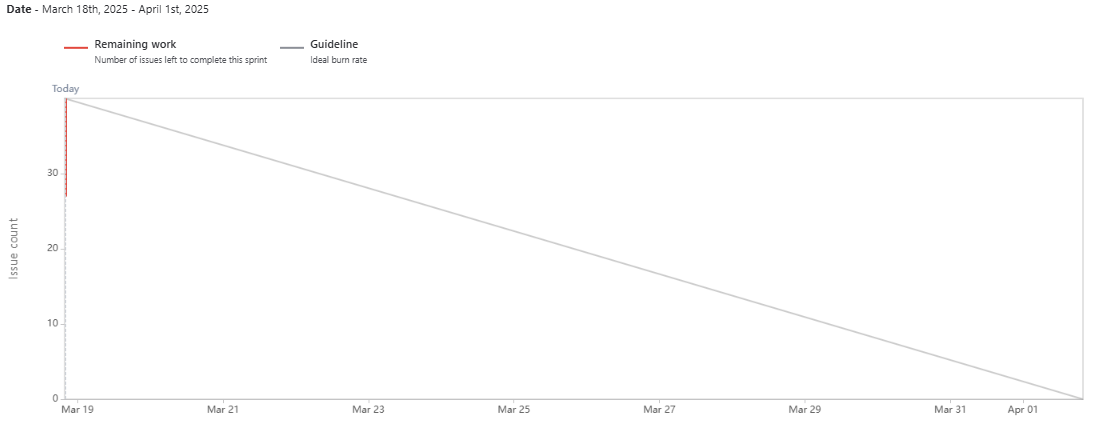
**Sprint Burnup Reports**

****

****

**Sprint burndown Report**

****

****

**Question 10- Explain Product Grooming**

**Answer**

**Product Backlog Refinement Meeting /Grooming** in Agile Scrum is a key agile practice and one of the scrum ceremonies for the Scrum team to keep the Product Backlog items refined, up-to-date, prioritized, and ready for the upcoming sprint planning. It is an ongoing activity for adding, refining, estimating, removing, re-ordering, splitting, or merging product backlog items. The key participants of this meeting are the product owner, the Scrum master, and the development team.

**List of items to be completed before meeting**

1. **Select target backlog items:** Before the meeting, select the backlog items that need refinement and are likely to be included in the next sprint or any high-priority items that need to be worked on, in the upcoming sprints.
2. **Review the selected backlog items:** The selected backlog items list should be shared with the Product Team to review and gain a clear understanding of their goal and to identify known issues or questions for discussion during the meeting.
3. **Communicate clear Agenda:** Before the scheduled refinement meeting, communicate the agenda and goals of the backlog refinement meeting to all team members in advance to understand the purpose of the meeting and come prepared.
4. **Define acceptance criteria and add relevant information:** Each backlog item planned to be discussed in the refinement meeting should have all the required information and acceptance criteria defined by the Product Owner to be reviewed by all team members before the meeting.
5. **Participants of Backlog Refinement Meeting:** Key participants of the refinement meeting include the product owner, Scrum master, and development team. If necessary, invite stakeholders or subject-matter experts who can contribute to the refinement process. Generally, the Scrum Master facilitates the meeting and the Product Owner runs the meeting.
6. **Duration and frequency:** Product backlog refinement is ongoing and a scheduled weekly or bi-weekly meeting. The duration is 2 hours or less, based on the duration of the sprint and backlog items reviewed.

**Expected outcome of the meeting**

* The top-ranked backlog items are refined and ready for consideration during the upcoming sprint planning meeting.
* The discussed backlog items should be clear, understood, estimated and prioritized.
* All dependencies are identified and discussed for each backlog item and re-ordered based on this.
* Acceptance criteria are defined and accepted by the team.
* All queries are discussed and answered.
* Any outdated or irrelevant backlog items are removed.

Any previous pending items are discussed and completed.

**Question 11- Roles of Scrum Master and Product Owner**

**Answer**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Scrum Master** | **Product Owner** |
| **Responsibility** | Facilitate Scrum process, ensure team adheres to Scrum practices | Manage product backlog, represent stakeholders, define and prioritize features |
| **Focus** | Removing impediments, improving collaboration | Maximizing product value, meeting business objectives |
| **Decision-Making Authority** | Limited, guides team in self-organizing | Makes critical product decisions (what gets built) |
| **Skills** | Agile expertise, coaching, facilitation | Business acumen, product vision, stakeholder management |
| **Interaction with Team** | Facilitates Scrum ceremonies (meetings), removes roadblocks | Provides product backlog items, gathers feedback |
| **Interaction with Stakeholders** | Keeps stakeholders informed of progress | Represents stakeholders, gathers requirements |
| **Meetings** | Facilitates all Scrum ceremonies (Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective) | Primarily attends Sprint Review and provides product direction |
| **Work Style** | Servant leader, facilitator, coach | Visionary leader, negotiator, prioritization expert |
| **Accountability** | Ensures the Scrum process is followed effectively | Owns the product backlog and its prioritization |

**Question 12- Meetings conducted in Scrum Project**

**Answer**

**Sprint planning meeting**

Before your team begins a Scrum sprint, you need to know where you’re going. This is where the sprint planning meeting comes in. A sprint planning meeting should be one of the longest Scrum meetings you hold—plan on two hours of planning for each week of your sprint. (A two-week sprint, for example, requires roughly a four-hour planning meeting.) While this may seem like a lot, remember that you only need to hold one sprint planning meeting per sprint—right at the start.

The purpose of a sprint planning meeting is simple: Establish what you and your Scrum team want to accomplish this sprint and evaluate the bandwidth you have available. From there, you can plan the sprint, assign tasks, and set deadlines. Make sure each team member understands the ins and outs of the tasks they are assigned. You’ll want to invite the product owner to this meeting so they can clear up any ambiguities and help establish expectations.

**Daily standup meeting**

As the most frequently held Agile Scrum meetings, daily standup meetings are the bread and butter of Scrum sprints. They’re short, to the point, and, as the name suggests, held each day—they’re typically the first meeting of the work day. By the end of a standup meeting, each team member should have answered two questions: What did I accomplish yesterday? And what am I going to accomplish today? Standup meetings are also a time for team members to bring up any roadblocks they are facing.

Though daily standup meetings only take between fifteen and thirty minutes, they are an effective way to keep each team member up-to-speed, on task, and openly communicating with others. Because they are held so frequently, standup meetings also allow teams to address problems as they arise, keeping the sprint moving on schedule.

**Sprint review meeting**

Sprint review meetings are held at the end of each sprint. This meeting is an opportunity for you and your team to demonstrate what you’ve accomplished to the product owner and other stakeholders outside of your team.

Your goal in a sprint review meeting is to gather feedback. As you demonstrate new product features and functionality, allow the product owner and other stakeholders to respond to and evaluate your work. Agile methodology relies on open and frequent conversations: As you and your team document, respond to, and act on feedback, remember that these conversations help create a better product.

Certain feedback points may require additional work on the product—add them to your backlog and consider including them in the next sprint. This is a matter of priority: While you should implement the feedback eventually, if other tasks are more pressing you can save it for a sprint down the road.

**Sprint retrospective meeting**

Just like review meetings, a sprint retrospective meeting is held at the end of each sprint. Whereas review meetings include the product owner and other stakeholders, retrospective meetings are primarily for the benefit of your Scrum team—there’s usually no need to get outside players involved.

During a sprint retrospective meeting, address these questions with your Scrum team: What went right this sprint? What went wrong? And what could we do differently next time?

These meetings don’t have to be long (usually somewhere between one and two hours), but they allow teams to constantly improve.

**Product backlog refinement**

Product backlog refinement meetings occur between sprints (usually just once per interim, but you could always schedule another if needed). If you’re anything like us, your backlog tasks are likely a bit rough around the edges. And that’s ok! This meeting is your chance to add clarifying details, establish deliverables, and prioritize the tasks in your backlog.

A thorough product backlog refinement meeting makes your life easier. Remember the oh so long sprint planning meeting you held at the beginning of the sprint? If you take the time to refine your backlog, sprint planning is a quicker and smoother process.

**Adhoc Meetings**

These meetings may be scheduled as and when needed to address specific topic or issues such as impediment log, discussing technical challenges, external challenges, or planning additional collaborative sessions.

**Question 13- Explain Sprint Size and Scrum Size**

**Answer**

**Sprint Size**

**Definition:** Sprint Size refers to the amount of work a Scrum team commits to completing within a single Sprint. It's not measured in a specific unit like hours or points, but rather in the complexity of the work items (user stories, features, etc.) chosen for the Sprint.

**Focus on Value, Not Time:** The ideal Sprint Size allows the team to deliver a potentially shippable product increment at the end of the Sprint. The focus is on delivering valuable functionality, not cramming a certain amount of work into a fixed timeframe.

**Scrum Size**

The scrum team size refers to the number of individuals who collectively works on the particular project. In Scrum, there's no single size fits all answer for team size, but there's a recommended range to ensure agility and effectiveness.

**Scrum Team Size:**

Ideal range: 3 to 9 people (excluding the Scrum Master and Product Owner). This small team size fosters: Better communication and collaboration; Faster decision-making due to fewer people involved; Increased agility as the team can adapt to changes quickly.

**A Scrum Team consists of three core roles:**

**Developers:** This is a self-organizing group with the skills necessary to deliver the product increments (working functionalities). The number of developers can vary depending on project complexity, but typically falls between 3 and 7.

**Product Owner:** This person represents the stakeholders and owns the product backlog (a prioritized list of features). They are responsible for- Defining the product vision and roadmap; Prioritizing backlog items; Making key product decisions.

**Scrum Master:** This individual acts as a coach and facilitator for the Scrum team. Their primary focus is to- Ensure the team understands and follows Scrum practices; Remove impediments hindering the team's progress; Facilitate Scrum ceremonies (meetings) like Sprint Planning and Daily Scrums.

**Question 14- Explain DOR and DOD**

**Answer**

**DOR (Definition of Ready)**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Definition of Done** | **Definition of Ready** |
| **Timing** | The definition of done applies at the end of the current sprint. | The definition of ready is defined and applies when a new sprint begins. |
| **Application** | The definition of done is criteria set for completion, ensuring that work completed meets quality standards and is potentially ready for release. | The definition of ready applies to criteria set for product backlog items or user stories before they can be accepted to be ready and included in the upcoming sprint. |
| **Focus** | The definition of done covers broader aspects of user stories being worked on and completed. | The definition of ready is focused on the backlog items and whether they meet all criteria to be considered for sprint. |
| **Purpose** | The definition of done is mainly focused on the internal aspects of work itself done during a sprint. | The definition of ready includes the backlog items, external factors such as dependencies, and resource availability. |
| **Scrum Component** | The definition of done is mandatory and part of scrum | The definition of ready is optional and is not part of the scrum guide. |
| **Criteria** | The definition of done refers to the work itself such as testing, documentation, and acceptance criteria defined. | DoR refers more to external items and activities like stakeholders, dependencies, and technical requirements. |
| **Outcome** | DoD helps the team to evaluate the completed work for quality and completeness and deliver value to customers. | DoR helps the scrum team to plan the work and quality standards. |

**Question 15- Explain Prioritization Techniques and MVP**

**Answer**

**Prioritization Techniques**

Agile development thrives on flexibility and prioritizing the right tasks at the right time. To achieve this, there are various techniques used to rank and order the backlog of features and user stories.

**MoSCoW Model:** This prioritization method categorizes features based on their necessity.

**Must-Have:** Features absolutely essential for the product's core functionality.

**Should-Have:** Important features that enhance the product's value.

**Could-Have:** Desirable features that would be nice to have but aren't critical.

**Won't-Have (for Now):** Features deferred for later or altogether excluded.

**Kano Model:** This technique focuses on user satisfaction and categorizes features based on how their presence or absence affects user delight:

**Excitement (Delighters):** Features that exceed user expectations and lead to high satisfaction.

**Performance (Satisfiers):** Features that meet user expectations and avoid dissatisfaction.

**Basic Needs (Must-Haves):** Features users consider essential; their absence leads to dissatisfaction.

**Indifferent:** Features users don't care much about; presence or absence has little impact.

**RICE: Reach, Impact, Confidence, Effort:** This scoring method assigns a numerical value to each backlog item based on four factors:

**Reach:** Number of users affected by the feature.

**Impact:** Level of positive impact the feature has on users.

**Confidence:** Certainty about the effort and impact estimations.

**Effort:** Relative amount of development time and resources needed.

**Cost of Delay:** This technique prioritizes features based on the potential business cost of waiting to implement them. Features that could lead to lost revenue or missed opportunities are prioritized higher.

**100-Dollar Test:** This quick prioritization method involves imagining you have $100 to spend on features. You would allocate this imaginary budget to features based on their perceived value and importance.

**Stack Ranking:** This is a simpler method where backlog items are compared to each other and ranked in order of importance, with the most important feature at the top.

**Priority Poker:** This interactive technique uses playing cards with numbers or relative values (e.g., T-shirt sizes - Small, Medium, Large) to estimate the effort or relative value of each backlog item. Team members discuss and vote on the appropriate card for each item, facilitating group consensus on prioritization.

**MVP (Minimum Viable Product)**

A Minimum Viable Product (MVP) is a development strategy in product management where a new product or service is created with the minimum features necessary to satisfy early adopters and gather feedback for future development. The concept is closely associated with the lean startup methodology and is aimed at quickly bringing a product to market to test its viability and gather valuable insights.

**Purpose of MVP**

* **Market Validation:** An MVP allows businesses to test their product ideas in the real market environment. By releasing a basic version of the product and observing how users interact with it, businesses can validate whether there is genuine demand for the product.
* **Reduced Time to Market:** Developing a full-featured product can be time-consuming. An MVP enables a quicker time to market by focusing on essential features, allowing the business to establish a presence and start gaining user feedback sooner.
* **Cost Efficiency:** Building a complete product with extensive features requires significant resources. Developing an MVP is a cost-effective approach, as it minimizes initial development costs and reduces the risk of investing heavily in a product that may not succeed.
* **Feedback Collection:** The release of an MVP provides an opportunity to collect valuable feedback from early users. This feedback is crucial for understanding user preferences, identifying potential issues, and making informed decisions for future development.
* **Iterative Development:** An iterative development process is encouraged by the MVP approach. Based on user feedback and market response, the product can be improved and expanded in subsequent iterations. This iterative approach ensures that the product evolves to better meet user needs over time.

**Types of MVP**

**High-fidelity MVP:** Although a high-fidelity MVP has few features, it offers an interface that is very interesting to use.

**Low-fidelity MVP:** it has a very simple user interface.

**Single-feature MVP:** as name suggests, it has only one feature.

**Concierge MVP:** Instead of having an automated procedure like in a full-fledged software solution, a concierge MVP uses a human (or human team) to execute tasks.

**Piecemeal MVP:** An incomplete MVP connects to external services and programmes to complete certain functions that will eventually be handled by the app itself.

**Digital MVP:** It allows for rapid iteration and testing of key hypotheses with real users.

**Hybrid MVP:** combines components from several MVP kinds to develop a unique strategy that is most appropriate for the target market and the particular requirements and objectives of the product.

**Question 16- Difference between Business Analyst and Product Owner**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Product Owner** | **Business Analyst** |
| **Primary Focus** | Represents the voice of the customer and stakeholders. Ensures the product meets their needs. | Analyses business processes, identifies areas for improvement, and proposes solutions. |
| **Responsibility** | Owns the product vision, sets priorities, and makes decisions on features and functionalities. | Gathers and documents requirements, analyses data, and supports the development of solutions. |
| **Decision-Making** | Makes decisions on product features, prioritizes the product backlog, and guides the development team. | Provides insights and recommendations, but decisions are often made by project managers or stakeholders. |
| **Stakeholder Interaction** | Collaborates closely with customers, end-users, and stakeholders to understand their needs and expectations. | Acts as a bridge between business stakeholders and the development team, facilitating communication. |
| **Role in Agile** | Integral part of Agile methodologies, especially in Scrum, where they play a central role in the development process. | Commonly found in Agile environments, providing crucial input for project success but not always a central role like the Product Owner. |
| **Project Lifecycle** | Focused on the entire product lifecycle, from ideation to release and ongoing improvements. | Primarily involved in the analysis phase, providing insights and documentation to support project development. |
| **Collaboration** | Collaborates closely with the development team, Scrum Master, and stakeholders in daily activities. | Collaborates with various stakeholders, including project managers, developers, and end-users, to gather and analyse information. |
| **Metrics** | Measures success based on the value delivered to customers and the achievement of product goals. | Success is often measured by the effectiveness of the proposed solutions and improvements to business processes. |

**Question-17 - Prepare a sample Resume of 3yrs exp Product Owner**

**ADITI GUPTA**

**Product Owner**

A results-driven and customer-focused Product Owner with 3 years of experience managing cross-functional Agile teams to deliver high-quality digital products. Adept at driving product strategy, prioritizing backlogs, and collaborating with stakeholders to achieve business goals. Proven expertise in translating customer needs into impactful features that maximize business value while ensuring seamless user experiences.

**Contact**

Pune, India

+91 98903 91274

aditi.g3110@gmail.com

**Skills**

Hard Skills:

* Release Planning
* Stakeholder Management
* Resource Forecasting
* Requirements Gathering
* Scrum
* User Acceptance Testing

Techniques:

* Product Management
* Agile Methodologies
* User Stories

Tools and Software:

* Jira
* Conﬂuence
* Kanban
* Power BI
* SQL
* Visio
* Balsamiq

Languages:

* English (Native)
* Hindi

Education:

* PGDM, BIMM Pune, 2014-2016

**Professional Experience**

Product Owner

XYZ Solutions Pvt Ltd, Pune (Feb/2023 - Present)

* Owned and managed the product backlog for Foodie catering to diverse user segments within the Food Industry domain.
* Conducted user research (interviews, surveys) and market analysis to identify emerging trends and inform product strategy.
* Prioritized and refined backlog items, ensuring alignment with business objectives, user needs, and technical feasibility.
* Collaborated with cross-functional teams (developers, designers, QA) to define user stories, acceptance criteria, and technical specifications.
* Facilitated all Scrum ceremonies (Sprint Planning, Daily Scrums, Sprint Reviews, Retrospectives) to maintain transparency and drive continuous improvement.
* Monitored key performance indicators (KPIs) and analysed product usage data to measure success and identify areas for optimization.

Associate Product Owner

ABC Solutions Pvt Ltd, Mumbai (Jan/2022 – Jan/2023)

* Successfully launched Shipkart, a new E-Commerce platform that generated $3 Mil revenue in its first quarter.
* Defined and documented product vision and roadmap aligned with company goals.
* Conducted user interviews and A/B testing to validate product concepts and features.
* Worked with marketing and sales teams to develop product messaging and go-to-market strategies.