# Prep 2 Case 1

**Q1. write Agile Manifesto
A.**  Agile Manifesto :

* These are the values and principles to be considered while working in Agile.
* It is nothing but a document that outlines 4 basic values and 12 principles
* It is not only for Scrum but also for all the frameworks that come under Agile like Scrum, XP, Kanban etc.

**4 Values of the Agile Manifesto:**

1. **Individuals and interactions over processes and tools**
Emphasizes the importance of communication and collaboration between people, rather than being overly reliant on tools or processes.
2. **Working software over comprehensive documentation**
Focuses on delivering functional software rather than spending too much time on detailed documentation that may not add immediate value.
3. **Customer collaboration over contract negotiation**
Encourages close cooperation with the customer throughout the project to ensure the software meets their needs, rather than just sticking rigidly to the contract.
4. **Responding to change over following a plan**
Agile values flexibility and the ability to adjust as needs and circumstances change, rather than rigidly adhering to an initial plan.

**12 Principles of Agile Development:**

1. Customer satisfaction 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 for 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 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 behavior accordingly.

**Q2. Write minimum 40 User stories and their Acceptance Criteria along with their BV and CP**

**A.

BV (Business Value):** This is not the cost of Development or the complexity of the feature. Business Value is how important is this feature (user Story) to the Business. This is estimated by Scrum Currency Notes. We provide Rs 1000. Rs 500, Rs 100, Rs 50, Rs 20 and Rs 10 Denominations. These estimations are done by the Stakeholders (Clients). If different values are selected by the stakeholders, then discussions will happen, and they agree to one BV value to that user story. **CP (Complexity Points):** CP is also known as Story Points (SP). CP is the effort required by the Scrum Developers to develop this feature (user story) using technology. Efforts include time taken to solve the complexity and write the code. CP is estimated by the Scrum Developers by using Poker cards. We provide pokers with values “?”, 1, 2, 3, 5, 8, 13, 20, 40, 100 and BIG. If the entire Project development takes 200 points, then this user story coding effort will be… how many points? … Thinking in this way, Scrum Developers will give CP to the User story. ). If different values are selected by the Scrum Developers, then discussions will happen, and they agree to one CP value to that user story.

|  |  |  |
| --- | --- | --- |
| User Story No.1 | Tasks: 2 | Priority: Highest  |
| As a delivery boy, user wants to register in Scrum foods so that one can deliver the orders |
| BV: 500 | CP: 02 |
| Acceptance Criteria: Registration Screen: Text boxes for username, Password, Govt ID, Mobile No, Email, Address, Phone no.Click on Register ButtonSend the Success notification to user.  |

|  |  |  |
| --- | --- | --- |
| User Story No: 2 | Tasks: 2 | Priority: Highest |
| As a restaurant owner, user want to view the orders so that they can view the list of orders  |
| BV: 500 | CP: 02 |
| Acceptance Criteria:View Order: Display list of orders in the tabular form  |

|  |  |  |
| --- | --- | --- |
| User Story No: 3 | Tasks: 2 | Priority: Highest |
| As a customer, user wants to add the address so that the order item will get to their place. |
| BV: 500  | CP: 02 |
| Acceptance Criteria: Text box to enter.Business rules: Within the radius of 5km |

|  |  |  |
| --- | --- | --- |
| User Story No: 4 | Tasks: 2 | Priority: Highest |
| As a customer, user wants to select the payment mode so that user can make the payment of their choice |
| BV: 500  | CP: 02 |
| Acceptance Criteria: Displays the modes, radio buttons to select the payment modes, payments buttons.Rule: Should select only one payment method. |

|  |  |  |
| --- | --- | --- |
| User Story No: 5 | Tasks: 1 | Priority: Highest |
| As a admin, user wants to view the restaurants so that they can approve their registration |
| BV: 500  | CP: 02 |
| Acceptance Criteria: Register in the platform with the details  |

|  |  |  |
| --- | --- | --- |
| User Story No: 6 | Tasks: 1 | Priority: Low |
| As a customer, user wants to view the price so that they can order the food |
| BV: 50 | CP: 01 |
| Acceptance Criteria: Display price in list of the menu items  |

|  |  |  |
| --- | --- | --- |
| User Story No: 7 | Tasks: 2 | Priority: Low |
| As a customer, user wants to view the contact number of delivery boy so that they can contact deliver boy for the status |
| BV: 50 | CP: 01 |
| Acceptance Criteria: - Display delivery boy mobile number- Display delivery boy name in tracking field- Display delivery boy picture |

|  |  |  |
| --- | --- | --- |
| User Story No: 8 | Tasks: 2 | Priority: Medium |
| As a restaurant owner, user wants to provide time slots so that customer can check opening and closing hours |
| BV: 100 | CP: 02 |
| Acceptance Criteria: - Click on restaurant dashboard- Add from time to time- Click on Submit- Display updated successfully  |

|  |  |  |
| --- | --- | --- |
| User Story No: 9 | Tasks: 2 | Priority: High |
| As a business owner, user wants to view restaurant revenue report. |
| BV: 200 | CP: 03 |
| Acceptance Criteria: Select reportsSelect revenue reports selectTo and From data select regionGenerate and download report  |

|  |  |  |
| --- | --- | --- |
| User Story No: 10 | Tasks: 3 | Priority: High |
| As a Regional Admin, User wants to manage regional restaurants, so that they can track the performance  |
| BV: 200 | CP: 03 |
| Acceptance Criteria: Click on performance of restaurants, select from date to date and click on generate reports which includes restaurant id, name, revenue and click on download report |

|  |  |  |
| --- | --- | --- |
| User Story No: 11 | Tasks: 2 | Priority: Medium |
| As a admin, user wants to see the regional revenue reports so that they can view the regional performance. |
| BV: 100 | CP: 03 |
| Acceptance Criteria: Select regional dropdown and view performance of each region in tabular form and download in form of excel  |

|  |  |  |
| --- | --- | --- |
| User Story No: 12 | Tasks: 2 | Priority: High |
| As a customer, user wants to chat with regional admin so that they can request for a refund  |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Fill in all mandatory fields, text box fields and display the order id. Provide a text box for description and the submit button and generate the issue id. Display the success message. |

|  |  |  |
| --- | --- | --- |
| User Story No: 13 | Tasks: 2 | Priority: High |
| As a hungry user, user wants to browse nearby restaurants so that they can order the food. |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Each restaurant displays its name, cuisine type and rating. The list can be sorted by distance or rating  |

|  |  |  |
| --- | --- | --- |
| User Story No: 14 | Tasks: 2 | Priority: High |
| As a customer, user wants to browse different restaurants and menus so that they can find a place to order food. |
| BV: 200 | CP: 02 |
| Acceptance Criteria: The menu includes dishes, prices and descriptions and show restaurant is open or close |
| User Story No: 15 | Tasks: 1 | Priority: High |
| As a customer, user wants to browse specific dishes and cuisines so that they can place the order. |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Display relevant restaurant and dishes matching the query |

|  |  |  |
| --- | --- | --- |
| User Story No: 16 | Tasks: 1 | Priority: High |
| As a customer, user wants to filter restaurants  |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Filter restaurant by cuisine type and dietary options (vegan, veg, non-veg, egg) |

|  |  |  |
| --- | --- | --- |
| User Story No: 17 | Tasks: 2 | Priority: High |
| As a customer, user wants to track order so that they know the time of delivery  |
| BV: 200 | CP: 02 |
| Acceptance Criteria: App shows real time update and order status Display estimated delivery time  |

|  |  |  |
| --- | --- | --- |
| User Story No: 18 | Tasks: 1 | Priority: High |
| As a user, user wants to rate and review the restaurant  |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Able to see other user reviews so that it helps to make dining decisions |

|  |  |  |
| --- | --- | --- |
| User Story No: 19 | Tasks: 1 | Priority: High |
| As a customer, user wants to save favorite restaurants and dishes  |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Access user list of favorites easily for future orders  |

|  |  |  |
| --- | --- | --- |
| User Story No: 20 | Tasks: 1 | Priority: High |
| As a customer, user wants to paste order history so that they can order again |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Should display the details such as order items, total cost and order date  |

|  |  |  |
| --- | --- | --- |
| User Story No: 21 | Tasks: 3 | Priority: High |
| As a customer, user wants to receive notifications so that they can receive updates |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Notifications for order confirmation, dispatch and delivery  |

|  |  |  |
| --- | --- | --- |
| User Story No: 22 | Tasks: 1 | Priority: Medium  |
| As a customer, user wants to contact customer support for the issues  |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Customer support section with contact information  |

|  |  |  |
| --- | --- | --- |
| User Story No: 23 | Tasks: 1 | Priority: High |
| As a restaurant owner, user wants to receive and manage orders so that they can update the order status  |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Manage order status and notify the restaurants about incoming orders  |

|  |  |  |
| --- | --- | --- |
| User Story No: 24 | Tasks: 2 | Priority: High |
| As a restaurant owner, user wants to access the customer reviews so that they can view and respond to customer reviews  |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Owners can address the feedback and improve their services  |

|  |  |  |
| --- | --- | --- |
| User Story No: 25 | Tasks: 1 | Priority: Medium |
| As a customer, user wants to view the promocodes and discounts so that they can order at lower price |
| BV: 100 | CP: 04 |
| Acceptance Criteria: Active promocodes  |

|  |  |  |
| --- | --- | --- |
| User Story No: 26 | Tasks: 1 | Priority: Medium |
| As a Delivery boy, user wants to view the order so that they can accept the orders |
| BV: 200 | CP: 04 |
| Acceptance Criteria: Order visibility, Real time updates, Order details, order filtering and sorting, order map view, order navigation, order completion and confirmation.  |

|  |  |  |
| --- | --- | --- |
| User Story No: 27 | Tasks: 5 | Priority: High |
| As a delivery boy user want to login so that they can accept the order |
| BV: 200 | CP: 04 |
| Acceptance Criteria: User Authentication, Error Handling, Password security, multi-factor authentication, compatibility and usability |

|  |  |  |
| --- | --- | --- |
| User Story No: 28 | Tasks: 5 | Priority: Medium |
| As a delivery boy, user want to view the feedback so that they can know the customer feedback |
| BV: 200 | CP: 04 |
| Acceptance Criteria: Access to feedback system, feedback visibility, feedback sorting and filtering, response mechanism and user support |

|  |  |  |
| --- | --- | --- |
| User Story No: 29 | Tasks: 5 | Priority: Medium |
| As a admin, user wants to view feedback so that know about customer feedback  |
| BV: 200 | CP: 04 |
| Acceptance Criteria: Access to feedback system, feedback visibility, feedback sorting and filtering, response mechanism and user support |

|  |  |  |
| --- | --- | --- |
| User Story No: 30 | Tasks: 5 | Priority: Medium |
| As a restaurant owner, user wants to view feedback so that know about customer feedback  |
| BV: 200 | CP: 04 |
| Acceptance Criteria: Access to feedback system, feedback visibility, feedback sorting and filtering, response mechanism and user support |

|  |  |  |
| --- | --- | --- |
| User Story No: 31 | Tasks: 3 | Priority: High |
| As a admin, user wants to know the issues so that they can resolve them |
| BV: 100 | CP: 03 |
| Acceptance Criteria: Display issue section, sorting and filtering of issues list, editing and modifying the issue |

|  |  |  |
| --- | --- | --- |
| User Story No: 32 | Tasks: 3 | Priority: High |
| As a regional admin, user wants to know the issue so that they can resolve them |
| BV: 200 | CP: 04 |
| Acceptance Criteria: Display issue section, sorting and filtering of issues list, editing and modifying the issue |

|  |  |  |
| --- | --- | --- |
| User Story No: 33 | Tasks: 6 | Priority: High |
| As a restaurant owner, user wants to view revenue generated so that they can view. |
| BV: 200 | CP: 04 |
| Acceptance Criteria: Select reports, select revenue reports and select to and from date and generate the report and can also download the report |

|  |  |  |
| --- | --- | --- |
| User Story No: 34 | Tasks: 2 | Priority: High |
| As a restaurant owner, user wants to know and verify the delivery boy details. |
| BV: 200 | CP: 04 |
| Acceptance Criteria: ID proof, punctuality and reliability |

|  |  |  |
| --- | --- | --- |
| User Story No: 35 | Tasks: 2 | Priority: Low |
| As a customer, user wants to view the contact number of delivery boy so that they can contact delivery boy for status. |
| BV: 50 | CP: 01 |
| Acceptance Criteria: Display delivery boy mobile number, display delivery boy name in tracking field and display delivery boy picture. |

|  |  |  |
| --- | --- | --- |
| User Story No: 36 | Tasks: 2 | Priority: Medium |
| As a restaurant owner, user want to provide the time slots so that customer can check opening and closing hours |
| BV: 100 | CP: 02 |
| Acceptance Criteria: Click on restaurant dashboard, add from time to time and click on submit. Display will be updated successfully  |

|  |  |  |
| --- | --- | --- |
| User Story No: 37 | Tasks: 3 | Priority: High |
| As a User, user wants to receive notifications about the updates |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Notification for order confirmation, notification for dispatch and notification for delivery  |

|  |  |  |
| --- | --- | --- |
| User Story No: 38 | Tasks: 1 | Priority: Medium |
| As a customer, user wants to contact customer support so that they can submit queries or issues. |
| BV: 200 | CP: 02 |
| Acceptance Criteria: Customer support section with contact information.  |

|  |  |  |
| --- | --- | --- |
| User Story No: 39 | Tasks: 4 | Priority: Medium |
| As a customer, user wants to view the order so that they can cancel it.  |
| BV: 100 | CP: 03 |
| Acceptance Criteria: Order status, method of cancellation refund, policy and time frame  |

|  |  |  |
| --- | --- | --- |
| User Story No: 40 | Tasks: 4 | Priority: High |
| As a regional admin, user wants to track the delivery so that they can view the status of the delivery. |
| BV: 100 | CP: 03 |
| Acceptance Criteria: Real time tracking security and data privacy, user friendly interface.  |

**Q3. What is epic? Write 2 epics
A.
 Epic:** An epic describes a big business requirement or a broad feature set that needs to be broken down into smaller, more manageable pieces. Since epics are too large, they are typically decomposed into smaller, more specific “**User stories”**. Each story can then be worked on in individual iterations or sprints. Epics help in understanding and prioritizing business objectives at a high level before diving into the detailed tasks. This ensures that larger goals are visible and aligned with the business's strategic priorities.

**Epic 1: Ratings and reviews:**User wants to view ratings and reviews for restaurants on scrum food so that they can make informed decisions about where to order food form.

User wants to provide ratings and reviews for restaurants on scrum food so that they can share experiences with other users and contribute to the community.

**Acceptance Criteria:**

* Users can view average ratings and reviews for watch restaurants on the restaurants details page.
* Users can read detailed reviews and comments left by other customers.
* Users can sort and filter reviews based on criteria such as rating and relevance
* Users can rate the restaurants and leave a review after placing the order.
* Users can edit or delete their own reviews within a specified timeframe.
* Reviews are displayed in a way that provide helpful insights to other users
* The rating and review system maintains the integrity and authenticity of user feedback

**Epic 2: Real-Time order tracking for food delivery app:**

The real time order tracking epic aims to provide users with a seamless and transparent experience by allowing them to track the status and location of their food orders in real-time. This feature enhances customer satisfaction reduces support inquiries and improves overall user engagement.

User stories: As a customer, user want to see live status of their order.

* Display current status of the order such as ‘order confirmed’, ‘preparing order’, ‘out for delivery’ and ‘delivered’
* Provide real-time updates as the order progresses through various stages.

As a customer, user wants to track the location of their order

* Integrate GPS services to show the delivery partners real-time location on map.
* Allow customers to view the estimated time of arrival (ETA) based on delivery partners location.

As a customer, user wants to receive notifications for order updates

* Send push notifications or SMS updates to inform customers about order confirmation, preparation and delivery status changes.
* Provide delivery partner details, including name, contact information and a profile picture

As a customer, I want to contact delivery partner directly.

* Enable in-app chat or call functionality to allow customers to communicate with the assigned delivery partner.
* Ensure privacy by using masked phone numbers or secure messaging channels.

As a customer, I want to view the delivery route.

* Display the delivery route on the map, showing the path the delivery partner will take to reach the destination.
* Allow customers to track the progress of the delivery in real-time along the route.

As a customer, I want to provide feedback on the delivery experience.

* Allow customers to rate the delivery partner and overall delivery experience after the order is delivered.
* Implement a feedback system with written comments to gather valuable insights.

As a customer, I want to see estimated delivery time adjustments.

* Account for real time traffic conditions and other factors that may effect the delivery time.
* Update the estimated delivery time accordingly and inform the customer promptly

As a customer, I want to have seamless tracking experience across platforms.

* Ensure real time order tracking feature is available and consistent on all supported platforms.

As a admin, I want to monitor order tracking performance.

* Provide analytics and reporting on order tracking metrics, such as average delivery time and customer satisfaction ratings.
* Use data to identify areas for improvement and optimize the delivery process.

**Acceptance criteria:**

**Real-time order updates:**

* The app should provide real-time updates on the status of the user’s order, such as ‘Order received’, ‘preparing’, ’out of delivery’ and ‘delivered’.

**Order location tracking:**

* The app should display the live location of the delivery driver while enroute to the user’s address.
* The map should update at regular intervals to reflect the driver’s movement accurately.

**Estimated Delivery Time:**

* The app should provide an accurate estimated time of delivery (ETA) based on driver’s current location, distance to the delivery address, and traffic conditions.

**Delivery Notifications:**

* Users should receive push notifications or in-app alerts for significant order updates, such as when the order is dispatched for delivery or when it is near the delivery address.

**Map zoom and interactions:**

* Users should be able to zoom in and out on the map to view the delivery driver’s route more closely.
* The map should support standard interactions, such as planning and rotating, to improve the user experience.

**Delivery Status History:**

* Users should have access to the delivery status history, allowing them to see the timeline of their order from placement to delivery completion.

**Accuracy and reliability:**

* The real-time tracking information should be accurate and reliable, providing users with the most up-to-date data available.
* The system should handle location updates efficiently, minimizing delays or inaccuracies.

**Privacy and Security:**

* The real time tracking feature should adhere to data privacy regulations and ensure that user location data is handled securely and used only for order tracking purpose.

**Compatibility:**

* The real-time order tracking should work smoothly across various platforms, including IOS and Android devices as well as web browsers.

**Opt-out options:**

* Users should have the option to disable real-time order tracking if they prefer not to share their location information.

**User Education:**

* Provide clear instructions or tooltips to educate users on how to use the real-time order tracking feature effectively.

**Support for Multiple Orders:**

* If a user places multiple orders, the app should allow them to track each order individually with its own status and location updates.

By meeting these acceptance criteria, the food delivery app can successfully implement real-time order tracking, providing users with a transparent and convenient way to monitor their orders from the moment they are placed until they are delivered to their doorstep.

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

**A.**

What is Business Value (BV)?

**Definition:** Business Value (BV) refers to the importance or benefit that a particular feature, user story, or task brings to the business or organization. It focuses on how much the feature contributes to achieving business goals such as increasing revenue, improving customer satisfaction, reducing costs, or gaining a competitive advantage.

**Techniques Used to Measure BV:**

* Scoring: Business value is often scored on a scale (e.g., 1 to 5, 1 to 10) based on its perceived impact. Stakeholders or product owners typically assign these scores.
* Prioritization Frameworks: Frameworks like MoSCoW (Must have, Should have, Could have, Won't have) or Weighted Shortest Job First (WSJF) can help assess the relative business value of features.
* Return on Investment (ROI): BV can be measured by estimating the potential return on investment a feature might generate for the business.
* Customer Feedback: User or customer feedback can help assess the perceived value a feature will bring.

**Example of Business Value (BV):**

* Feature: Implementing a new payment gateway.
	+ BV: High, because it will directly enable the business to accept payments online, increase revenue, and improve user experience.
* Feature: Adding a small aesthetic change to the website layout.
	+ BV: Low, as it provides limited direct benefit to the business or customer.

**What is Complexity Points (CP)?**

**Definition:** Complexity Points (CP) are a measure of the difficulty or effort required to complete a task, feature, or user story. CP considers technical complexity, uncertainty, and the amount of work needed to implement a particular feature or task.

**Techniques Used to Measure CP:**

* Story Points: A common technique for estimating complexity, where teams assign points (often based on Fibonacci numbers) to represent the relative difficulty of tasks.
* T-shirt Sizing: A simple method where tasks are categorized as Small, Medium, Large, or Extra Large based on their complexity.
* Ideal Days or Hours: Estimating how many ideal days or hours it would take to complete a task without interruptions or other distractions.
* Expert Judgment: In some cases, the complexity is assessed based on the experience and expertise of team members or technical leads.

**Example of Complexity Points (CP):**

* Feature: Integrating with a third-party payment provider.
	+ CP: High, because it involves complex technical integration, security concerns, testing, and possibly debugging external APIs.
* Feature: Updating the font style across the website.
	+ CP: Low, because it's a simple task requiring minimal technical effort and resources.

**Summary:**

* Business Value (BV) is focused on the impact and importance of a task or feature to the business. It helps prioritize features that provide the most value.
* Complexity Points (CP) measure the effort and difficulty required to implement a task. It helps teams estimate the technical resources and time needed for tasks.

**Q5. Explain about Sprint**

**A.** A Sprint is a fundamental concept in Agile frameworks, particularly in Scrum, where it refers to a time-boxed iteration during which a specific set of tasks or user stories are completed. Sprints typically last 2 to 4 weeks, depending on the project needs and team preferences. During this time, the Scrum team works collaboratively to develop a potentially shippable product increment.

Key Characteristics of a Sprint:

1. Time-boxed: A Sprint has a fixed duration, usually 1 to 4 weeks, to ensure consistent progress and timely delivery.
2. Goal-Oriented: Each Sprint is aimed at delivering a specific set of features or tasks that contribute to the overall project goal.
3. Commitment: At the start of the Sprint, the team commits to completing a set of prioritized user stories or tasks. These are defined in the Sprint Backlog.
4. Iterative and Incremental: Sprints allow for incremental development, meaning that after each Sprint, there is a tangible product increment. This can be reviewed and tested to gather feedback.
5. Collaboration and Transparency: Sprints promote teamwork and ensure that all stakeholders have visibility into progress through daily stand-ups and Sprint reviews.

Phases of a Sprint:

1. Sprint Planning: The team, along with the product owner, defines the tasks that need to be completed during the Sprint and ensures that everyone understands the Sprint goal.
2. Daily Stand-up: Every day, the team meets briefly to discuss progress, identify blockers, and make sure tasks are on track.
3. Sprint Review: At the end of the Sprint, the team presents the completed work to stakeholders for feedback and discussion.
4. Sprint Retrospective: After the review, the team reflects on what went well, what could be improved, and how to enhance processes in the next Sprint.

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.

**Q6. Explain Product backlog and sprint back log**

**A. Product Backlog**: The Product Backlog is a dynamic, prioritized list of all the features, functionalities, enhancements, and bug fixes that need to be developed for a product. It serves as the single source of work for the development team, created and maintained by the Product Owner.

Key Characteristics of Product Backlog:

1. Prioritized: The items in the Product Backlog are ordered by priority, with the most valuable or urgent tasks at the top. The Product Owner decides the priority based on business value, customer needs, and strategic goals.
2. Living Document: The Product Backlog is continually refined and updated throughout the project. As the product evolves, new items may be added, and existing items may be modified, removed, or reprioritized.
3. High-Level Items: The Product Backlog includes broad, high-level items, often written as epics or user stories, which are later broken down into smaller tasks (user stories) during Sprint Planning.
4. Represents the Product’s Scope: It provides an overview of the product's goals and helps the team understand what needs to be done to deliver the final product.

Example: For an e-commerce website, the Product Backlog may include user stories like:

* “As a customer, I want to register an account.”
* “As a user, I want to add items to my shopping cart.”
* “As an admin, I want to generate sales reports.”

**Sprint Backlog:** The Sprint Backlog is a subset of the Product Backlog that includes all the tasks and user stories the team commits to completing during a specific Sprint. It is a short-term plan created during Sprint Planning to guide the development work within the Sprint.

Key Characteristics of Sprint Backlog:

1. Commitment for the Sprint: The Sprint Backlog contains the tasks that the development team will work on during the Sprint. It is derived from the Product Backlog but only includes items that the team believes can be completed within the Sprint timeframe (usually 1 to 4 weeks).
2. Detailed Tasks: The user stories from the Product Backlog are broken down into smaller, actionable tasks in the Sprint Backlog. These tasks are detailed enough to be worked on by the team during the Sprint.
3. Owned by the Development Team: The Sprint Backlog is owned and managed by the development team, not the Product Owner. They are responsible for ensuring the Sprint Backlog is achievable and for adjusting it during the Sprint if necessary.
4. Evolving: The Sprint Backlog can evolve throughout the Sprint. For instance, if a task turns out to be more complex than expected, the team may update the backlog to reflect new findings or reassign tasks.

Example: For a Sprint focused on implementing the user registration feature, the Sprint Backlog may include:

* “Design the user registration page (5 hours)”
* “Develop front-end registration form (8 hours)”
* “Implement back-end validation (10 hours)”
* “Test user registration functionality (4 hours)”

**Differences Between Product Backlog and Sprint Backlog:**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **Product Backlog** | **Sprint Backlog** |
| **Definition** | A prioritized list of all features and tasks for the product. | A list of tasks selected from the Product Backlog for a specific Sprint. |
| **Focus** | High-level, long-term product requirements and improvements. | Short-term, focused tasks for the current Sprint. |
| **Ownership** | Managed and prioritized by the Product Owner. | Owned and managed by the development team. |
| **Level of Detail** | Contains high-level user stories or epics. | Contains detailed tasks that can be completed during the Sprint. |
| **Duration** | Long-term, evolves throughout the product lifecycle. | Short-term, specific to the current Sprint (usually 1-4 weeks). |

 **Q7. What is impediments log? write 2 impediments**

**A.** An Impediments Log is a tool used in Agile methodologies (especially Scrum) to track and manage obstacles or issues that hinder the progress of the team’s work. These obstacles, known as impediments, can prevent the team from achieving their goals during a Sprint or from working at their full potential. The Impediments Log helps ensure that these issues are identified, recorded, and addressed promptly to maintain smooth workflow and productivity.

Key Features of an Impediments Log:

1. Tracking Issues: It keeps a record of any issues that arise and impact the team's ability to deliver work as planned.
2. Visibility: It provides visibility to the Scrum Master, team members, and stakeholders about the challenges being faced.
3. Resolution Process: The log also often includes a resolution or action plan to address and resolve the impediments.
4. Continuous Improvement: The log is used to improve processes by identifying recurring issues that need to be resolved for future sprints.

Examples of Two impediments:
- Delivery partner shortage in a specific region
- Technical issue causing intermittent order processing failure

Delivery partner shortage in a specific region:

|  |  |
| --- | --- |
| Login ID | 1 |
| Description | Delivery Partner storage in specific region |
| Impact | Delays in order deliveries and increase customer dissatisfaction |
| Priority  | High (Due to impact on customer’s experience) |
| Assigned to | Operations team and HR team |
| Status  | Open |
| Action taken | Operations team is actively recruiting new delivery partners in the region. The HR team is working on fast-tracking the on-boarding process  |
| Resolution | Delivery partner recruitment efforts are ongoing and the HR team is streamlining the onboarding process to expedite new hires. Regular updates are being provided in team meetings |

Technical issue causing intermittent order processing failure

|  |  |
| --- | --- |
| Login ID | 2 |
| Description | Technical issue causing intermittent order processing failures  |
| Impact | Delays in order processing and potential revenue loss |
| Priority  | High (due to impact on revenue and customer experience)  |
| Assigned to | Tech team and QA team |
| Status  | IN progress |
| Action taken | Tech team has identified the root cause and is working on a fix. The QA team is conducting extensive testing to ensure issue is resolved. |
| Resolution | The tech team has impediment a fix and conducted through testing. The issue has been resolved and orders are now processing smoothly. |

**Q8. Explain Velocity of the Team**

**A.** Velocity in Agile is a measure of the amount of work a team can complete in a Sprint, usually represented in story points, but it can also be measured in hours or other units. It is calculated by adding up the story points of all the user stories that were fully completed during a Sprint. Velocity is used by Scrum teams to track their progress and plan for future work.

**Key Characteristics of Velocity:**

1. Measurement Unit: Velocity is commonly measured in story points, which represent the complexity and effort required to complete a user story. Some teams also use hours, but story points are generally preferred because they focus on relative effort and complexity rather than time.
2. Based on Completed Work: Only completed user stories (those meeting the definition of "done") are included when calculating velocity. Any work that is not finished, or does not meet the definition of done, is excluded from the calculation.
3. Dynamic and Evolving: Velocity may change over time, especially as the team matures, learns new skills, or adjusts to new challenges. The key point is that it should be used as a guideline for future planning and not as a strict, fixed metric.

How is Velocity Calculated?

To calculate the velocity for a Sprint:

1. Add up the story points for all the user stories that were fully completed in the Sprint.

Formula: Velocity=Sum of Story Points of Completed User Stories

For example, if a team completes three user stories in a Sprint:

* User Story 1: 5 story points
* User Story 2: 8 story points
* User Story 3: 3 story points

The velocity for that Sprint would be:

5+8+3=16 story points5 + 8 + 3 = 16 story points

Thus, the velocity for this Sprint is 16 story points.

Average Velocity: Average Velocity is the mean of the velocity values of multiple Sprints. By calculating the average velocity, teams can better understand their consistent performance over time and gain more reliable estimates for future Sprints.

To calculate the average velocity:

1. Add up the velocity of several Sprints.
2. Divide by the number of Sprints to get the average.

Formula:

Average Velocity= Sum of Velocity of Multiple Sprints​/ Number of Sprints

Example: If a team has completed the following velocities in 3 Sprints:

* Sprint 1: 16 story points
* Sprint 2: 18 story points
* Sprint 3: 14 story points

Then, the average velocity would be:

Average Velocity=16+18+14​/3=48/3​=16 story points

Use of Velocity for Sprint Planning:Velocity is a powerful tool for planning future Sprints. Here’s how it can be used:

1. Predicting the Amount of Work for Future Sprints: By looking at the team’s average velocity, you can estimate how many story points the team is likely to complete in the next Sprint. For example, if the average velocity is 16 story points, you might plan to take on around 16 story points worth of work in the next Sprint (depending on the availability and complexity of the tasks).
2. Capacity Adjustments: Velocity helps in adjusting the capacity based on holidays, team member availability, or other factors that could affect the team's ability to work in the next Sprint. For example, if a team member is on vacation, the team may decide to reduce the number of story points they commit to in the next Sprint.
3. Workload Balancing: Using velocity, teams can balance their workload to avoid overcommitting or under committing in a Sprint. If the team completes around 16 story points on average, and a new Sprint has 18 story points planned, they can assess whether this is achievable based on their previous velocity and adjust accordingly.
4. Identifying Trends: By tracking velocity over multiple Sprints, teams can spot trends such as whether their velocity is increasing, decreasing, or remaining stable. A decreasing velocity could indicate issues such as technical debt, unclear requirements, or other impediments that need to be addressed.

**Q9. Draw Sprint Burn Charts n Product Burn Down Charts**

**A.**

**Sprint Burn Down Chart:**

* Represents work remaining within a sprint.
* X-axis: Days in the Sprint, Y-axis: Remaining Work (Story Points or Hours).
* Ideal line shows a steady decline; actual progress may vary.



**Product Burn Down Chart**

* Represents work remaining for the entire product/release.
* X-axis: Sprints, Y-axis: Total Work Remaining (Story Points or Features).
* Shows progress across multiple sprints toward project completion.



**Q10. Explain about Product Grooming**

**A.** Product Grooming (or Product Backlog Grooming) is a key activity in Agile methodologies, particularly in Scrum, that involves refining and prioritizing the items in the Product Backlog. This process ensures that the backlog remains up-to-date, well-organized, and ready for the upcoming Sprint Planning sessions. The goal is to make sure that the backlog contains a clear and prioritized list of work that is well understood by the team.

Product Grooming is typically conducted throughout the project lifecycle, often in regular sessions, and is a collaborative effort involving the Product Owner, Scrum Master, and the development team.

Key Activities in Product Grooming:

1. Prioritizing Backlog Items: The Product Owner is responsible for prioritizing items based on business value, stakeholder needs, and project goals. This ensures that the most important tasks are worked on first, and resources are allocated to maximize value.
2. Clarifying and Detailing User Stories: User stories may need additional details, acceptance criteria, or clarifications to ensure they are ready for implementation. The team collaborates with the Product Owner to break down high-level user stories into more specific and actionable tasks.
3. Estimation: The team may estimate the effort required to complete user stories during grooming sessions. This often involves assigning story points or using other estimation techniques (e.g., T-shirt sizing). The aim is to ensure that the team can gauge how much work each item will require.
4. Removing or Re-prioritizing Items: Items that are no longer relevant or have lower priority may be removed from the Product Backlog. Items can also be re-prioritized to reflect changes in the business environment or project direction.
5. Splitting Large User Stories (Epics): Large user stories (called Epics) that are too big to be completed in a single Sprint are broken down into smaller, more manageable user stories. This helps in creating bite-sized chunks of work that are achievable within a Sprint.
6. Reviewing and Updating Dependencies: During grooming, dependencies between user stories or tasks are identified and managed. This helps avoid blockers during the Sprint and ensures the team can work efficiently.

**Q11. Explain the roles of Scrum Master and Product Owner**

**A.**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **Scrum Master** | **Product Owner** |
|

|  |
| --- |
| **Primary Focus** |

|  |
| --- |
|  |

 |

|  |
| --- |
| Ensuring Scrum process runs smoothly |

|  |
| --- |
|  |

 |

|  |
| --- |
| Maximizing product value |

|  |
| --- |
|  |

 |
|

|  |
| --- |
| **Key Responsibility** |

|  |
| --- |
|  |

 | Facilitating Scrum events, coaching team, removing impediments | Managing backlog, defining priorities, and aligning with stakeholders |
|

|  |
| --- |
| **Role in the Team** |

|  |
| --- |
|  |

 | Servant-leader, mentor, and facilitator | Business representative and decision-maker |
|

|  |
| --- |
| **Backlog Management** |

|  |
| --- |
|  |

 | Supports the team in understanding backlog items | Owns and prioritizes the product backlog |
|

|  |
| --- |
| **Stakeholder Interaction** |

|  |
| --- |
|  |

 | Works mainly with the development team | Acts as a bridge between business and the Scrum team |
|

|  |
| --- |
| **Decision Making** |

|  |
| --- |
|  |

 | Guides the team but does not make product decisions | Makes product-related decisions and sets priorities |
|

|  |
| --- |
| **Success Criteria** |

|  |
| --- |
|  |

 | Team productivity, adherence to Agile principles | Delivery of high-value features and business outcomes |
|

|  |
| --- |
| **Collaboration with Team** |

|  |
| --- |
|  |

 |

|  |
| --- |
| Helps the team self-organize and improve processes |

|  |
| --- |
|  |

 |

|  |
| --- |
| Provides clarity on requirements and priorities |

|  |
| --- |
|  |

 |
|

|  |
| --- |
| **Focus on Value** |

|  |
| --- |
|  |

 |

|  |
| --- |
| Ensures that Scrum practices bring efficiency and transparency. |

|  |
| --- |
|  |

 | Ensures the team works on the highest-value items. |

**Q12. Explain all Meetings Conducted in Scrum Project**

**A. Sprint Planning:** This meeting happens at the beginning of a sprint, where the Product Owner presents backlog items, and the Development Team selects tasks to complete. The team sets a Sprint Goal and plans the work needed for successful delivery.

**Daily Scrum (Stand-up):** A 15-minute daily meeting where team members discuss progress, upcoming tasks, and any roadblocks. It ensures transparency, quick decision-making, and alignment among team members.

**Sprint Review:** Conducted at the end of the sprint, where the team demonstrates completed work to stakeholders. The focus is on gathering feedback, ensuring alignment with business goals, and discussing potential improvements.

**Sprint Retrospective:** A reflection meeting held after the Sprint Review, where the Scrum Team discusses what went well, what didn’t, and identifies actions for continuous improvement in the next sprint.

**Backlog Refinement (Grooming):** A regular session where the Product Owner and the team refine the backlog by clarifying, updating, and prioritizing user stories. This ensures that upcoming sprints have well-defined and actionable tasks.

**Product Backlog Refinement (Grooming)**: A continuous process where the Product Owner and the Development Team review and refine backlog items. It includes clarifying requirements, breaking down large user stories, and prioritizing work to ensure the backlog is well-prepared for upcoming sprints.

**Release Planning**: A strategic meeting where the Scrum Team and stakeholders plan for upcoming releases. It involves defining release goals, setting timelines, and ensuring that features align with business objectives. This helps in managing dependencies and delivering value incrementally.

**Adhoc Meetings**: These are unscheduled discussions that occur as needed to address urgent issues, dependencies, or decision-making that cannot wait for the next planned meeting. They help resolve blockers, align with stakeholders, or discuss technical challenges quickly.

**Q13. Explain Sprint Size and Scrum Size**

**A. Sprint Size:** The size of a sprint refers to its duration, which is typically between 1 to 4 weeks. The length is determined based on factors like project complexity, stakeholder expectations, and the team’s ability to deliver value within the timeframe. Shorter sprints (1–2 weeks) allow for faster feedback, while longer sprints (3–4 weeks) may be suitable for complex projects.

**Scrum Team Size:** A Scrum team is typically small and cross-functional, with an ideal size of 5 to 9 members. It includes a Product Owner, Scrum Master, and Development Team. Teams smaller than 3 may lack diverse skills, while teams larger than 9 may face collaboration challenges. The goal is to maintain efficiency, effective communication, and self-organization within the team.

**Q14. Explain DOR and DOD**

**A. Definition of Ready (DoR):** This ensures that a user story or backlog item is clear, well-defined, and ready for development. A backlog item is considered "ready" when it meets criteria such as clear acceptance criteria, detailed requirements, dependencies resolved, and estimated effort. This prevents blockers and ensures smooth development.

**Definition of Done (DoD):** This ensures that a task, user story, or increment is fully completed and meets quality standards. A work item is considered "done" when it is coded, tested, reviewed, integrated, and meets all acceptance criteria. The DoD ensures that only high-quality, shippable products are delivered at the end of a sprint.

**Q15. Explain Prioritization Techniques and MVP**

**A. Prioritization Techniques:** These are methods used to decide which backlog items should be worked on first based on business value, urgency, and impact. Common techniques include:

* **MoSCoW Method:** Categorizes tasks as Must-Have, Should-Have, Could-Have, and Won’t-Have to prioritize effectively.
* **Kano Model:** Focuses on customer satisfaction by classifying features into Basic, Performance, and Delightful categories.
* **Value vs. Effort Matrix:** Helps teams prioritize high-value, low-effort tasks for maximum impact.
* **WSJF (Weighted Shortest Job First):** Used in SAFe Agile, it prioritizes based on cost of delay and job duration.

**Minimum Viable Product (MVP):** The MVP is the smallest version of a product that delivers core functionality while requiring minimal effort and development. It helps validate ideas, gather user feedback, and iterate based on real-world usage before full-scale development. The goal is to reduce risks, save time, and maximize learning while ensuring the product meets essential user needs.

**Q16. Difference between Business Analyst n Product Owner**

**A.**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **Business Analyst** | **Product Owner** |
|

|  |
| --- |
|  |

**Primary Focus** | Gather and analyze business requirements | Defines and prioritizes product backlog |
| **Role in Agile** | Acts as a bridge between business and IT team | Represents stakeholders and ensures value delivery |
| **Backlog Ownership** | Supports backlog refinement and requirement gathering | Owns and manages the product backlog |
| **Decision-making** | Provides recommendation based on analysis | Makes financial decisions on features and priorities |
| **Stakeholder Interaction** | Works closely with business teams to understand needs  | Directly collaborates with stakeholders and customers |
| **Documentation** | Prepares BRDs, user stories and process flows | Defines acceptance criteria and backlog items  |
| **Focus on Value** | Ensures business needs are well-documented | Ensures the team delivers high-value features |
| **Success criteria** | Clear requirements, smooth communication | Delivering a successful product aligned with goals |

**Q17. Prepare a sample Resume of 3yrs exp Product Owner**

**A. Product Owner Resume**

**Name:** Srikar **Phone:** +91 9999999999 **Email:** yourname@gmail.com

**Professional Summary**

Results-driven Product Owner with 3 years of experience in Agile environments, specializing in backlog management, stakeholder collaboration, and product roadmap execution. Proven ability to bridge business and technology teams to deliver customer-centric products that drive business value. Adept at prioritizing requirements, defining user stories, and ensuring high-quality product releases.

**Skills & Competencies**

* Product Backlog Management & Prioritization
* Agile & Scrum Methodologies
* User Story Writing & Acceptance Criteria
* Roadmap Planning & Release Management
* Stakeholder Communication & Requirement Gathering
* Data-Driven Decision Making
* Competitive Market Analysis
* Jira, Confluence, and Other Agile Tools

**Work Experience**

**Product Owner**

**ABC Company| Hyderabad | [01/2023 – Present]**

* Collaborate with cross-functional teams (developers, designers, QA, and business stakeholders) to define and deliver product features.
* Own and maintain the product backlog, ensuring prioritization aligns with business objectives.
* Define and refine user stories, acceptance criteria, and sprint goals to ensure smooth development cycles.
* Conduct market research and customer feedback analysis to make data-driven decisions for product enhancements.
* Lead sprint planning, backlog grooming, and review meetings to ensure clear communication between business and development teams.
* Work closely with UX/UI teams to improve user experience and product usability.
* Track key performance indicators (KPIs) and use insights to iterate and improve product performance.

**Business Analyst**

**DEF Company| Hyderabad | [02/2020 – 07/2022]**

* Acted as a bridge between business stakeholders and the development team to gather and document requirements.
* Assisted in product backlog refinement, ensuring clear and well-defined user stories.
* Supported the Product Owner in defining roadmaps, prioritizing features, and conducting competitive analysis.
* Facilitated Sprint Review and Retrospective meetings to capture feedback and improve processes.
* Created detailed process flows, wireframes, and functional specifications to guide development teams.

**Education**

Masters | HFG University | 2020