**Food Delivery Application**

**Prepared by: Venkatesh Dhanana Phone: 8247813228**

**Question 1:** write Agile Manifesto.

The Agile Manifesto consists of four core values and twelve principles that guide agile software development.

**Four Core Values of Agile Manifesto:**

1. Individuals and interactions over processes and tools.
2. Working software over comprehensive documentation.
3. Customer collaboration over contract negotiation.
4. Responding to change over following a plan.

**Twelve Agile Principles:**

1. **Customer satisfaction** is the highest priority through early and continuous delivery of valuable software.
2. **Welcome changing requirements**, even late in development, to give the customer a competitive advantage.
3. **Deliver working software frequently**, from a couple of weeks to a couple of months, with a preference for shorter timescales.
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 within a development team is **face-to-face conversation**.
7. **Working software** is the primary measure of progress.
8. Agile processes **promote sustainable development**. Developers, sponsors and users should 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. Regularly reflect on how to become **more effective**, then tune and adjust behaviour accordingly.

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

**1. Customer Module.**

|  |  |  |
| --- | --- | --- |
| User Story No: 1 | Tasks: 2 | Priority: High |
| As a customer  I want to register on Scrum Foods  So that I can place food orders. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA Registration page should have fields for name, email, phone, and password. The system should send a confirmation email/SMS after successful registration. Invalid inputs should display appropriate error messages. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 2 | Tasks: 2 | Priority: High |
| As a customer  I want to log in to my account  So that I can access my profile and orders. | | |
| BV: 1000 | | CP: 5 |
| ACCEPTANCE CRITERIA Login should require email/phone and password. The system should validate credentials and provide feedback for incorrect login attempts. Forgot password functionality should be available. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 3 | Tasks: 2 | Priority: High |
| As a customer  I want to search for restaurants based on location  So that I can view nearby food options. | | |
| BV: 500 | | CP: 8 |
| ACCEPTANCE CRITERIA Search should allow filtering by location, cuisine, food and rating. Restaurants should be displayed with basic details (name, rating, delivery time). | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 4 | Tasks: 2 | Priority: Medium |
| As a customer  I want to view the restaurant menu  So that I can select items to order. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA Menus should display item name, description, price and availability. Customers should be able to add items to the cart. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 5 | Tasks: 2 | Priority: High |
| As a customer  I want to place an order  So that I can receive food delivery. | | |
| BV: 1000 | | CP: 13 |
| ACCEPTANCE CRITERIA Customers should be able to select delivery location and payment method. Order confirmation should be provided after successful payment. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 6 | Tasks: 2 | Priority: Medium |
| As a customer  I want to track my order in real-time.  So that I know the delivery status. | | |
| BV: 500 | | CP: 13 |
| ACCEPTANCE CRITERIA Order tracking should show statuses (Order Received, Preparing, Out for Delivery, Delivered). A map view should show the delivery boy’s location. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 7 | Tasks: 2 | Priority: Medium |
| As a customer  I want to cancel my order before food preparation starts. | | |
| BV: 500 | | CP: 8 |
| ACCEPTANCE CRITERIA Cancellation should be allowed within a specific timeframe. Refund should be processed if applicable. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 8 | Tasks: 2 | Priority: Low |
| As a customer I want to rate and review restaurants.  So that I can share my experience. | | |
| BV: 500 | | CP: 5 |
| ACCEPTANCE CRITERIA Customers should be able to submit ratings (1-5 stars) and text reviews. Reviews should be displayed for other customers. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 9 | Tasks: 2 | Priority: Medium |
| As a customer  I want to receive notifications about my order status. | | |
| BV: 500 | | CP: 5 |
| ACCEPTANCE CRITERIA Push notifications should be sent at each status update. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 10 | Tasks: 2 | Priority: Low |
| As a customer  I want to save my favourite restaurants.  So that I can order easily. | | |
| BV: 500 | | CP: 3 |
| ACCEPTANCE CRITERIA Customers should be able to add/remove restaurants from favourites. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 11 | Tasks: 2 | Priority: High |
| As a customer  I want to Add the Address.  So that I can get the order to my address. | | |
| BV: 500 | | CP: 2 |
| ACCEPTANCE CRITERIA Text Box to enter address.  Business Rules: Within the radius of 5 km | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 12 | Tasks: 2 | Priority: High |
| As a customer  I want to select the payment mode.  So that I can make payment of my choice. | | |
| BV: 500 | | CP: 3 |
| ACCEPTANCE CRITERIA Display payment modes, radio buttons to select payment modes, payments button.  Business Rule: Can select only one payment mode | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 13 | Tasks: 2 | Priority: Low |
| As a customer  I want to view the price.  So that I can order the food. | | |
| BV: 50 | | CP: 5 |
| ACCEPTANCE CRITERIA Display price in the list of menu items. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 14 | Tasks: 2 | Priority: Low |
| As a customer  I want to view the contact number of delivery boy.  So that I can contact delivery boy for the status. | | |
| BV: 50 | | CP: 1 |
| ACCEPTANCE CRITERIA Display delivery boy mobile number.  Display delivery boy name in tracking field.  Display delivery boy picture. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 15 | Tasks: 2 | Priority: High |
| As a customer  I want to chat with Reg. Admin.  So that I can request for refund. | | |
| BV: 200 | | CP: 2 |
| ACCEPTANCE CRITERIA Text Box Fields should be available.  Display Order Id, Text Box for description  Submit Button should be available  Generate Issue Id and Display Successful message. | | |

**2. Delivery Boy Module**

|  |  |  |
| --- | --- | --- |
| User Story No: 16 | Tasks: 2 | Priority: High |
| As a Delivery Boy  I Want To Register In Scrum Foods  So That I Can Deliver Order | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA Registration form should include Name, Email, Phone, Vehicle Type, License Number and ID verification. System should send a confirmation email/SMS upon successful registration. Admin should approve/reject the registration request. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 17 | Tasks: 2 | Priority: High |
| As a Delivery Boy  I Want To Login In Scrum Foods  So that I can access my assigned deliveries. | | |
| BV: 1000 | | CP: 5 |
| ACCEPTANCE CRITERIA Login should require email/phone and password. The system should validate credentials and provide feedback for incorrect login attempts. Forgot password functionality should be available. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 18 | Tasks: 2 | Priority: High |
| As a delivery boy  I want to view available delivery orders  So that I can choose an order to deliver. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA A list of available orders should be displayed with pickup and delivery location details. Orders should show estimated delivery time, distance and restaurant details. Orders that are already assigned should not be available. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 19 | Tasks: 2 | Priority: Medium |
| As a delivery boy  I want to accept or reject delivery orders  So that I can manage my workload. | | |
| BV: 500 | | CP: 8 |
| ACCEPTANCE CRITERIA A delivery boy can accept or reject an order within a given time limit. Once accepted, the order should be marked as “In Progress” and assigned to the delivery boy. If rejected, the order should be reassigned to another available delivery boy. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 20 | Tasks: 2 | Priority: Medium |
| As a delivery boy  I want to view my assigned orders  So that I can manage my deliveries efficiently. | | |
| BV: 500 | | CP: 5 |
| ACCEPTANCE CRITERIA The list should display current and completed deliveries separately. Each order should show restaurant details, customer details, and order status. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 21 | Tasks: 2 | Priority: Low |
| As a delivery boy  I want to confirm the order pickup  So that the system can update the status. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA The delivery boy must verify the order ID with the restaurant before pickup. The system should update the order status to "Out for Delivery" after pickup confirmation. The delivery boy should receive customer details after pickup confirmation. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 22 | Tasks: 2 | Priority: Medium |
| As a delivery boy  I want to use navigation to reach the customer’s location. | | |
| BV: 500 | | CP: 8 |
| ACCEPTANCE CRITERIA A "Navigate" button should open Google Maps with the customer’s address. The system should provide step-by-step navigation. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 23 | Tasks: 2 | Priority: High |
| As a delivery boy  I want to update the order status in real time  So that customers and restaurants are informed. | | |
| BV: 500 | | CP: 5 |
| ACCEPTANCE CRITERIA The system should allow the delivery boy to update the order status (Picked Up, Out for Delivery, Delivered). Customers should receive push notifications for each status update. The system should log timestamp details for each update. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 24 | Tasks: 2 | Priority: Low |
| As a delivery boy  I want to complete the delivery and collect cash payment if applicable. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA The delivery boy should confirm order delivery with a digital signature or OTP verification. If the payment method is COD, the delivery boy should enter the received amount. The system should confirm order completion and update earnings accordingly. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 25 | Tasks: 2 | Priority: Low |
| As a delivery boy  I want to report an issue related to a delivery  So that it can be resolved. | | |
| BV: 500 | | CP: 8 |
| ACCEPTANCE CRITERIA The system should allow the delivery boy to submit an issue related to delivery, restaurant, or customer. The issue should be categorized and assigned to the admin for resolution. The delivery boy should receive updates on the issue resolution status. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 26 | Tasks: 2 | Priority: Low |
| As a delivery boy  I want to log out of my account securely. | | |
| BV: 100 | | CP: 3 |
| ACCEPTANCE CRITERIA The system should end the session and log out the user. The user should be redirected to the login page | | |

**3. Restaurant Module**

|  |  |  |
| --- | --- | --- |
| User Story No: 27 | Tasks: 2 | Priority: High |
| As a restaurant owner  I want to register my restaurant on the platform so that I can receive customer orders. So that the system can update the status. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA Registration form should include restaurant name, email, phone number, address, cuisine type, license details, and payment details. System should send a confirmation email/SMS upon successful registration. Admin should approve/reject the registration request. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 28 | Tasks: 2 | Priority: High |
| As a restaurant owner  I want to View Orders So that I can view the list of orders | | |
| BV: 500 | | CP: 2 |
| ACCEPTANCE CRITERIA View Order, Display List of orders in the tabular Form. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 29 | Tasks: 2 | Priority: Medium |
| As a restaurant owner  I want to provide time slots. So that customers can check Open and Closing Hours. | | |
| BV: 100 | | CP: 2 |
| ACCEPTANCE CRITERIA Click on restaurant dashboard  Add from time to time  Click on submit and Display updated successfully | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 30 | Tasks: 2 | Priority: High |
| As a restaurant owner  I want to log in to my account  So that I can manage my restaurant profile and orders. | | |
| BV: 1000 | | CP: 5 |
| ACCEPTANCE CRITERIA Login requires email/phone and password. The system should validate credentials and provide feedback for incorrect login attempts. Forgot Password functionality should be available. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 31 | Tasks: 2 | Priority: High |
| As a restaurant owner  I want to add, update and remove menu items  So that customers can order food accurately. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA The system should allow restaurants to add new food items with name, description, price, and image. Restaurants should be able to update prices and availability status. Customers should not see unavailable items. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 32 | Tasks: 2 | Priority: High |
| As a restaurant owner I want to accept or reject incoming orders  So that I can manage my kitchen workload. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA Orders should have an "Accept" or "Reject" option. If accepted, the system should notify the customer and delivery boy. If rejected, the customer should be notified immediately with an alternative suggestion. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 33 | Tasks: 2 | Priority: Medium |
| As a restaurant owner  I want to verify and assign a delivery boy to an order  So that the food reaches the customer. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA System should display available delivery boys. The assigned delivery boy should receive a notification. Once assigned, the order status should update to "Out for Delivery." | | |

**4. Admin & Regional Admin Module**

|  |  |  |
| --- | --- | --- |
| User Story No: 34 | Tasks: 2 | Priority: High |
| As an Admin  I want to approve, reject or suspend restaurant accounts  So that only verified restaurants operate on the platform | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA Restaurants should submit required documents for verification. The system should allow admins to approve or reject applications. Approved restaurants should be listed in the system. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 35 | Tasks: 2 | Priority: High |
| As an admin  I want to add, update and remove regional admins  So that they can manage their respective regions. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA The system should allow adding, updating and deactivating regional admins. Regional admins should be assigned specific regions. Admin should be able to reset passwords for regional admins. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 36 | Tasks: 2 | Priority: High |
| As an admin  I want to approve, reject or suspend delivery boys  So that only verified riders can deliver food. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA Delivery boys should submit necessary identification documents. The system should allow approval or rejection of new applicants. The admin should be able to deactivate a delivery boy if necessary. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 37 | Tasks: 2 | Priority: High |
| As an admin  I want to resolve customer issues  So that user satisfaction is maintained. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA Customers should be able to raise complaints via the app. The system should categorize complaints by issue type. Admin should be able to respond and resolve complaints. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 38 | Tasks: 2 | Priority: High |
| As an Admin  I want to process refund requests  So that customers receive reimbursements for valid claims. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA Customers should be able to request a refund. The admin should review the request and approve/reject it. The system should notify the customer and update the refund status. | | |

**5. Business Owner Module**

|  |  |  |
| --- | --- | --- |
| User Story No: 39 | Tasks: 2 | Priority: High |
| As a business owner  I want to securely log in  So that I can access business insights and manage high-level operations. | | |
| BV: 1000 | | CP: 5 |
| ACCEPTANCE CRITERIA The system should allow login using an email and password. Two-factor authentication should be enabled. Failed login attempts should be logged. | | |

|  |  |  |
| --- | --- | --- |
| User Story No: 40 | Tasks: 2 | Priority: High |
| As a business owner  I want to view key performance metrics  So that I can analyse the success of the platform. | | |
| BV: 1000 | | CP: 8 |
| ACCEPTANCE CRITERIA The system should display revenue trends, user growth, and order volume. Reports should be filterable by date range, region, and restaurant category. The system should generate downloadable CSV or PDF reports. | | |

**Question 3:** What is Epic? Write 2 Epics?

An **Epic** is a large body of work that can be broken down into smaller, manageable user stories.

**Epic 1: Order Management System for Scrum Foods**

**Description:** This epic focuses on the end-to-end process of ordering food, from restaurant selection to order completion and feedback.

**User Stories under this Epic:**

* **Customer Registration & Login –** Users should be able to create accounts and log in securely.
* **Search & View Restaurants** – Customers can search and filter restaurants based on cuisine, ratings, food and location.
* **View Restaurant Menu** – Customers can browse available food items with descriptions, images and prices.
* **Place an Order** – Customers should be able to add items to a cart and place an order.
* **Payment Processing** – Customers should be able to pay via credit/debit cards, UPI and COD.
* **Real-time Order Tracking** – Customers should track order status and estimated delivery time.
* **Order Cancellation & Refunds** – Customers should be able to cancel an order and get eligible refunds.
* **Delivery Completion & Feedback** – Customers should rate the restaurant and delivery experience.

**Epic 2:** **Delivery Partner Management System**

**Description:** This epic focuses on the workflow of delivery boys, from accepting orders to completing deliveries and receiving earnings.

**User Stories under this Epic:**

* **Delivery Partner Registration & Login –** Delivery boys should sign up and log in securely.
* **View Available Orders** – Delivery partners should see nearby available orders for delivery.
* **Accept & Pick Up Orders** – Delivery partners should confirm order pick-up from restaurants.
* **Navigate to Customer Location** – The system should provide real-time navigation assistance.
* **Update Delivery Status** – Delivery partners should mark orders as picked up, out for delivery or Delivered.
* **COD Handling** – If applicable, delivery partners should collect COD payments.
* **Report Issues & Raise Disputes** – Delivery partners should be able to report issues such as delays, incorrect orders or customer absence.
* **View Earnings & Payment History** – Delivery partners should see their earnings and transaction history.

**Question 4:** What is the difference between BV and CP?

|  |  |
| --- | --- |
| **Business Value (BV)** | **Complexity Points (CP)** |
| BV represents the importance of a user story to the business and stakeholders. | CP represents the effort and complexity required by developers to implement a user story. |
| Estimated by Stakeholders/Clients based on business priorities. | Estimated by Scrum Developers using Planning Poker. |
| Measured in currency denominations (₹1000, ₹500, ₹100, ₹50, ₹20 & ₹10). | Measured in Story Points (SP) (1, 2, 3, 5, 8, 13, etc.). |
| Focuses on business impact and customer needs. | Focuses on development effort, including coding complexity and problem-solving. |
| "Enable real-time order tracking" may have BV ₹1000 because customers expect it. | The same feature may have CP 8 because it requires GPS integration and backend development. |
| High BV does not always mean high CP. | High CP does not always mean high BV. |
| Helps prioritize user stories that deliver the most value. | Helps estimate sprint capacity and plan development effort. |

**Question 5:** Explain about Sprint.

A **Sprint** is a time-boxed iteration in Scrum during which the Scrum Team works on a set of prioritized user stories to create a potentially shippable product increment.

**Key Characteristics of a Sprint:**

1. Fixed Duration:  
Sprints typically last 1 to 4 weeks (commonly 2 weeks).  
The duration remains consistent throughout the project.

2. Sprint Planning:  
Before a sprint begins, the Scrum Team selects user stories from the Product Backlog based on priority and effort.  
The team commits to delivering these stories within the sprint.

3. Daily Scrum (Stand-up Meeting):  
A 15-minute daily meeting where developers discuss:  
What was done yesterday?  
What will be done today?  
Any blockers or challenges?

4. Development & Testing:  
The team designs, develops, tests and integrates features incrementally.

5. Sprint Review:  
At the end of the sprint, the Scrum Team demos the completed work to stakeholders.  
Feedback is collected to improve future sprints.

6. Sprint Retrospective:  
The team reflects on what went well, what went wrong, and how to improve in the next sprint.

**Sprint Workflow:**

* Sprint Planning
* Work Execution (Daily Scrums, Development, Testing, Bug Fixing)
* Sprint Review (Demo to Stakeholders)
* Sprint Retrospective (Team Improvement Discussion)

**Sprint Scenario Example for Scrum Foods:**

**Sprint Duration:** 2 Weeks

**Sprint Goal:** Implement User Registration & Login

|  |  |
| --- | --- |
| **Day** | **Activity** |
| Day 1 | Sprint Planning (Select user stories) |
| Day 2-10 | Development + Daily Scrum Meetings |
| Day 11-12 | Testing & Bug Fixing |
| Day 13 | Sprint Review (Demo to Client) |
| Day 14 | Sprint Retrospective (Lessons Learned) |

**Question 6:** Explain Product backlog and sprint backlog?

In Scrum, both the **Product Backlog** and **Sprint Backlog** are essential for managing and prioritizing work.

**1. Product Backlog**

The Product Backlog is a list of all features, enhancements, bug fixes and technical tasks required for the product. It is continuously refined and prioritized by the Product Owner based on business needs.

Key Characteristics:

* Owned by the Product Owner
* Contains all user stories, features and requirements for the product
* Items are prioritized based on business value (BV)
* It is dynamic and evolves as new requirements emerge
* Contains epics, user stories and tasks

Example for Scrum Foods:

|  |  |  |
| --- | --- | --- |
| Product Backlog Item | Priority | BV (Business Value ₹) |
| User Registration & Login | High | ₹1000 |
| Search Restaurants | High | ₹1000 |
| Real-time Order Tracking | High | ₹1000 |
| Payment Integration | Medium | ₹500 |
| Customer Feedback System | Medium | ₹500 |

**2. Sprint Backlog**

The Sprint Backlog is a subset of the Product Backlog that contains only the items the Scrum Team commits to completing in a sprint. It is created during Sprint Planning and is owned by the Development Team.

Key Characteristics:

* Owned by the Development Team
* Contains user stories selected for the current sprint
* Includes tasks needed to complete each user story
* Fixed during the sprint (cannot add new items unless agreed by the team)
* Helps track daily progress

Example Sprint Backlog for Scrum Foods (Sprint 1 - 2 Weeks):

|  |  |  |
| --- | --- | --- |
| Sprint Backlog Item | Status | CP (Complexity Points) |
| User Registration UI | To Do | 5 |
| User Login Functionality | In Progress | 8 |
| Password Reset Feature | To Do | 5 |
| Database Setup for Users | In Progress | 8 |

**Question 7:** What is impediments log? Write 2 impediments.

An **Impediments Log** is a **record of obstacles, blockers or challenges** that prevent the Scrum Team from making progress in a sprint. The **Scrum Master** is responsible for tracking and resolving these impediments to ensure smooth workflow.

**Key Characteristics of an Impediments Log:**

* Helps in identifying and removing blockers affecting sprint progress.
* Maintained by the Scrum Master.
* Includes issue details, impact, priority and resolution status.
* Discussed in Daily Stand-ups and Sprint Retrospective meetings.

**Example of Two Impediments for Scrum Foods:**

**Impediment 1:** API Integration Delay

**Description**: The third-party payment gateway API is not responding properly, causing a delay in implementing online payments.

**Impact**: Customers cannot complete payments, blocking the entire order processing flow.

**Priority**: High

**Status**: Scrum Master coordinating with API provider for resolution.

**Resolution Plan**: Request support from the payment provider and implement a temporary alternative like COD.

**Impediment** **2**: Environment Setup Issues

**Description**: Some developers are facing issues setting up the local development environment due to missing dependencies.

**Impact**: Developers cannot start working on assigned user stories, delaying sprint progress.

**Priority**: Medium

**Status**: Scrum Master working with DevOps to resolve setup issues.

**Resolution Plan**: Create a step-by-step installation guide and conduct a setup session for affected developers.

**Question 8:** Explain Velocity of the Team

Velocity is a key agile metric that measures the amount of work completed by the scrum team in a sprint. It is calculated as the **sum of the Complexity Points (CP) of all completed user stories** in a sprint.

It is used to predict future sprint capacity.  
It can improve sprint planning by setting realistic goals.  
It can measuring team performance over multiple sprints.

**Velocity Calculation:**

Velocity is calculated at the end of each sprint using below formula.

Velocity = ∑ (Complexity Points of Completed User Stories)

For example, if the Scrum Team completed the following user stories in Sprint 1:

User Story Complexity Points (CP) Status

User Registration 5 Completed

Login Functionality 8 Completed

Search Restaurants 13 Completed

Payment Gateway Integration 8 Not Completed

Velocity Calculation for Sprint 1: Velocity = 5+8+13 = 26 CP (The incomplete story’s points **are not counted**)

If Team Velocity for multiple sprints are

Sprint 1 – 26 CP, Sprint 2 – 30 CP, Sprint 3 – 28 CP and Sprint 4 – 32 CP

Then Average Velocity Calculation is, Average Velocity= (26+30+28+32)/4 = 29CP

This means the Scrum Team can commit to around 29 CP worth of work in future sprints.

**Question 9:** Draw Sprint Burn Charts n Product Burn Down Charts

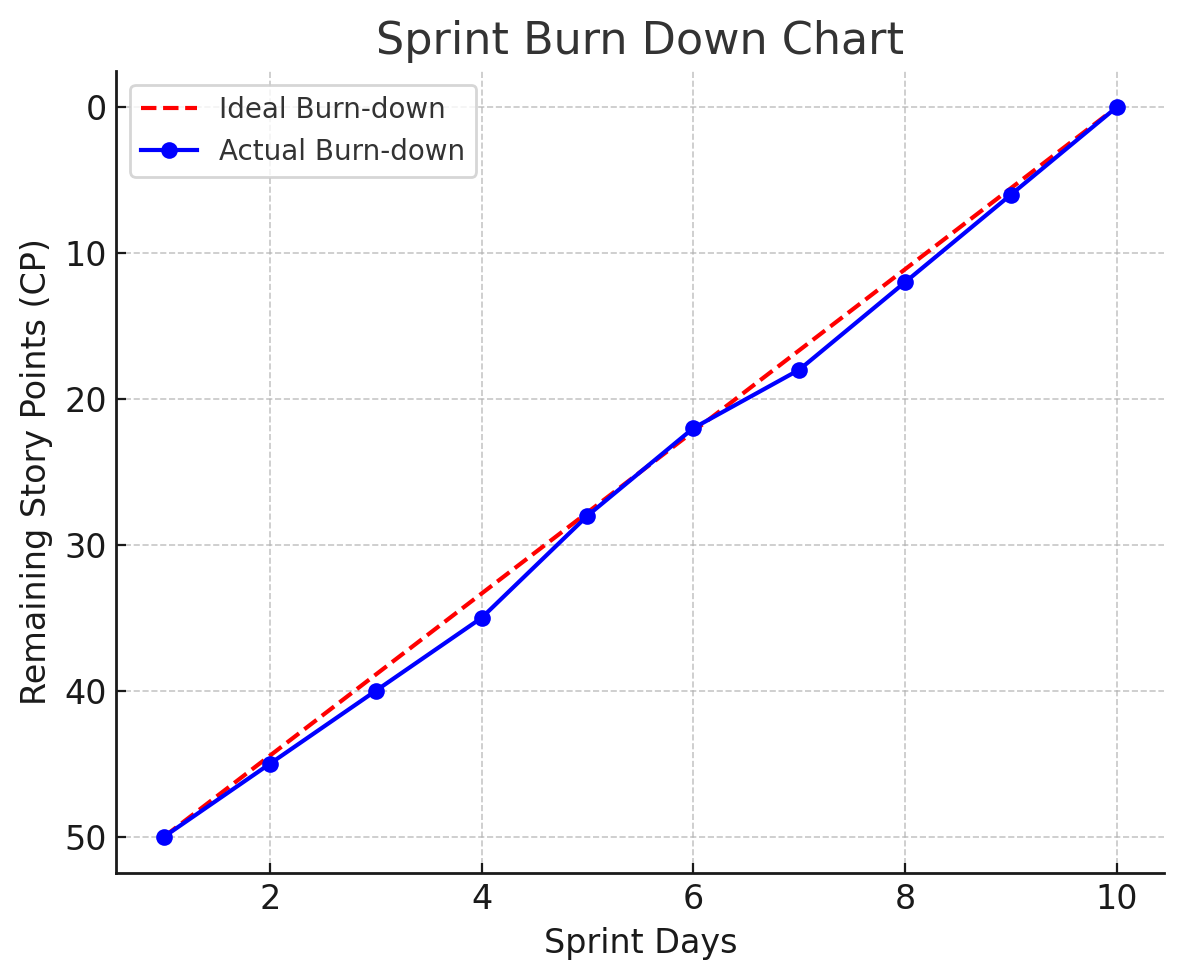
These charts visually track progress and help in understanding how work is being completed during a sprint and across the project.

**Sprint Burn down Chart**

* Tracks remaining work within a sprint (usually 1-4 weeks).
* X-axis: Sprint Days
* Y-axis: Remaining Story Points (CP)
* Ideal burn-down line vs. actual progress line.

**Product Burn down Chart**

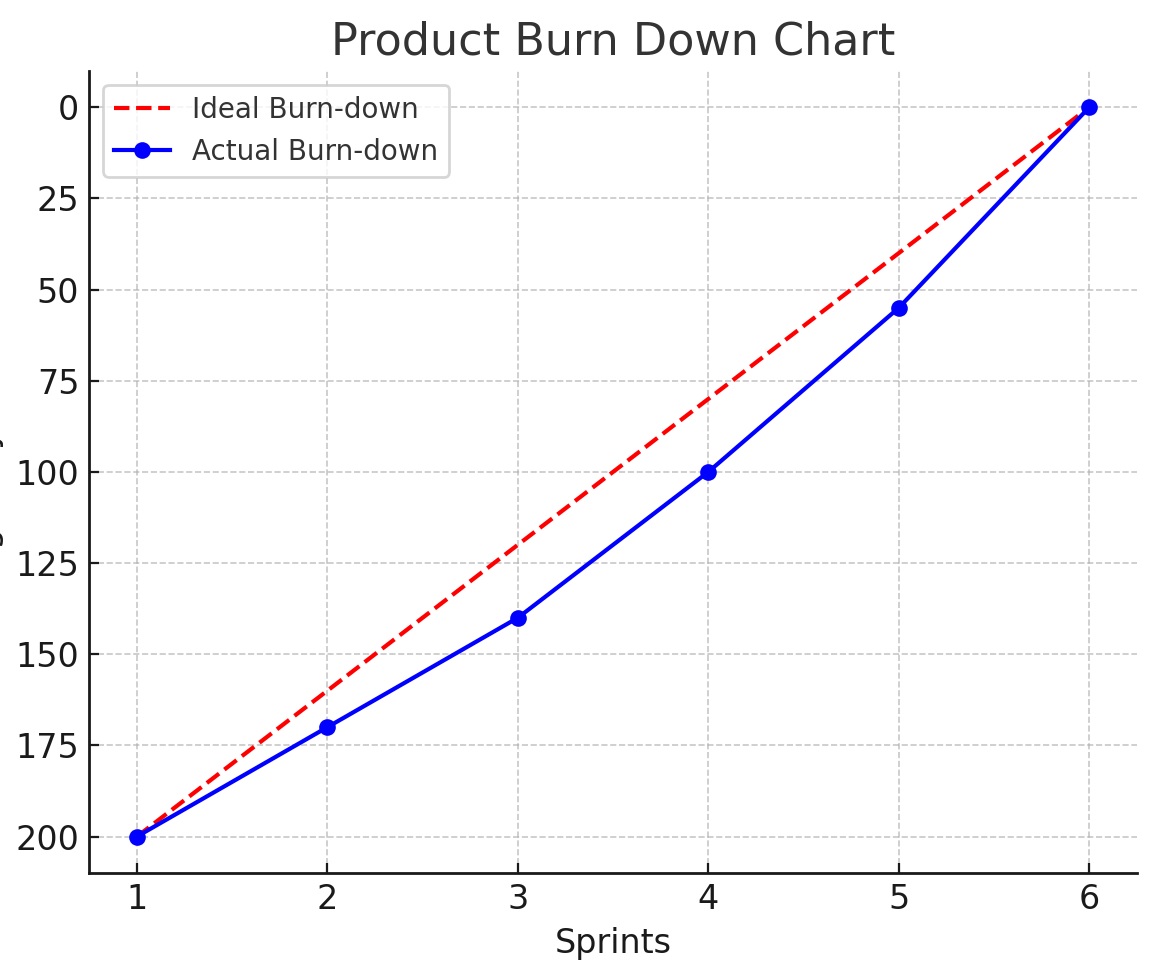
* Tracks remaining work across multiple sprints until the product is complete.
* X-axis: Sprints
* Y-axis: Total Remaining Story Points (CP)
* Helps forecast when the project will be completed.



Shows how work is progressing within a sprint.

The red dashed line represents ideal burn-down (smooth progress).

The blue line represents actual progress, showing variations.



Tracks remaining work across multiple sprints.

Helps to know when the project will be completed.

If actual progress follows the ideal line, the project is on track.

**Question 10:** Explain about Product Grooming

**Product grooming** also known as **Backlog Grooming (Refinement),** is an ongoing process in Agile where the **Scrum Team reviews, updates and prioritizes items in the Product Backlog.** The goal is to ensure that backlog items are **well-defined, estimated and ready** for upcoming sprints.

**Benefits:**

Improved Sprint Planning: A well-groomed backlog makes sprint planning faster and more efficient.

Reduced Risk: Clear requirements and accurate estimates reduce the risk of scope creep and delays.

Increased Team Focus: A prioritized and well-understood backlog helps the team stay focused on delivering value.

Better Communication: Grooming sessions facilitate communication and collaboration between the Product Owner and the development team.

**Who involved in the Product Grooming.**

The Product Owner, Scrum Master, Development team members and sometimes other stakeholders participate in grooming sessions.

**Key Activities in Product Grooming**

Review User Stories: Ensure each backlog item is well-defined with clear acceptance criteria

Prioritize Backlog Items: The Product Owner ranks items based on Business Value (BV)

Estimate Complexity: Developers assign Complexity Points (CP) to user stories

Break Down Large Items: Split epics into smaller, manageable user stories

Clarify Requirements: Resolve ambiguities by discussing with stakeholders

Remove/Deprioritize Items: Eliminate outdated or unnecessary backlog items

**Frequency**:

Grooming sessions are typically held regularly, often before each sprint planning meeting or as needed to maintain backlog health.

**Question 11:** Explain the roles of Scrum Master and Product Owner

In Scrum, Scrum Master and Product Owner (PO) play distinct but complementary roles in ensuring the success of a project.

**Product Owner (PO)** – The "Visionary"

The Product Owner is responsible for defining the product vision, prioritizing work and ensuring the team delivers maximum business value.

**Key Responsibilities:**

* Defines Product Vision & Roadmap – Aligns the product with business goals.
* Manages the Product Backlog – Prioritizes and refines backlog items.
* Writes User Stories & Acceptance Criteria – Ensures clarity for the team.
* Works with Stakeholders – Collects and translates business requirements.
* Ensures Business Value (BV) Estimation – Helps stakeholders assign BV.
* Collaborates with the Scrum Team – Clarifies doubts and refines scope.
* Accepts or Rejects Completed Work – Ensures features meet expectations.

**PO in Scrum Foods Project**

As the Product Owner, you are responsible for defining and prioritizing features such as real-time tracking, 24/7 availability and a smooth ordering experience for customers.

**Scrum Master** – The "Facilitator"

The Scrum Master is responsible for coaching, guiding and removing obstacles that blocks the Scrum Team’s progress.

**Key Responsibilities:**

* Facilitates Scrum Events – Daily Stand-ups, Sprint Planning, Sprint Review and Retrospective.
* Removes Impediments – Identifies and resolves blockers for the team.
* Ensures Agile Best Practices – Coaches the team in Scrum principles.
* Protects the Team – Shields the team from external interruptions.
* Improves Team Efficiency – Helps optimize workflow and collaboration.
* Encourages Self-Organization – Supports the team in making decisions.

**Scrum Master in Scrum Foods Project**

**Satya Rathnakar**, as the Scrum Master, ensures the Scrum Team stays focused on delivering high-quality features without distractions. He helps resolve blockers like API integration issues, deployment failures or unclear user stories.

**Difference between Scrum Master & Product Owner**

|  |  |  |
| --- | --- | --- |
| Aspect | Product Owner (PO) | Scrum Master |
| Focus | Business Value & Product Vision | Team Efficiency & Agile Process |
| Key Responsibility | Defines features & prioritizes work | Facilitates Scrum & removes blockers |
| Backlog Management | Owns & refines the Product Backlog | Helps team understand backlog items |
| Decision Making | Decides what features get built | Ensures team works efficiently |
| Collaboration With | Customers, Stakeholders, Scrum Team | Scrum Team, Product Owner, Organization |

**Question 12:** Explain all Meetings Conducted in Scrum Project

Scrum follows a structured framework with **five key meetings** to ensure smooth project execution. These meetings help in **planning, tracking progress, reviewing work and improving efficiency.**

**1. Sprint Planning Meeting**

* Sprint Planning Meeting is used to define what work will be completed in the next sprint.
* Product Owner, Scrum Master and Developers will attend in this meeting at the start of each Sprint (1-4 weeks)

**Key Activities:**

* Product Owner presents prioritized backlog items (user stories).
* Scrum Team selects stories for the sprint (based on capacity & velocity).
* Developers assign Complexity Points (CP) to user stories.
* Define the Sprint Goal (overall objective for the sprint).

Outcome: A Sprint Backlog with a list of user stories to be completed.

**2. Daily Stand-up (Daily Scrum)**

* Daily Stand-up call/meeting is held to provide a quick status update and discuss blockers.
* Scrum Master, Developers and Product Owner (optional) will attend the meeting every day, usually 15 minutes

**Format:** Each developer answers 3 questions:

* What did I do yesterday?
* What will I do today?
* Do I have any blockers?

**3. Sprint Review Meeting**

* Sprint Review Meeting is used to showcase the completed work to stakeholders.
* Scrum Team, Stakeholders, Product Owner and Scrum Master will attend the meeting at the end of each Sprint.

**Key Activities:**

* Developer’s present completed user stories (working software).
* Stakeholders give feedback (accept/reject the work).
* Product Owner decides if the Definition of Done (DoD) is met.

**4. Sprint Retrospective**

* Sprint Retrospective is used to analyse the last sprint and find areas for improvement.
* Scrum Master, Developers and Product Owner will attend the meeting after Sprint Review, before the next Sprint Planning

**Key Questions Discussed:**

* What went well? (Positive aspects)
* What didn’t go well? (Challenges faced)
* What can be improved? (Actionable steps for improvement)

**5. Product Grooming (Refinement Meeting)**

* It is used to refine and update the Product Backlog.
* Product Owner, Scrum Master and Developers will attend the meeting regularly (before Sprint Planning)

**Key Activities:**

* Prioritizing backlog items based on Business Value (BV).
* Estimating Complexity Points (CP) for new user stories.
* Breaking down large epics into smaller user stories.
* Removing unnecessary backlog items.

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

Agile teams need to manage their work efficiently, and two key aspects that help in this are **Sprint Size** and **Scrum Size**. These concepts determine the duration and capacity of the team to deliver work effectively.

**What is Sprint Size?**

Sprint Size refers to the total amount of work (in Story Points) that the Scrum Team commits to completing in a single sprint. It depends on:

* Sprint Duration (usually 1-4 weeks)
* Team Capacity (availability of developers)
* Historical Velocity (average Story Points completed per sprint)

**Example of Sprint Size Calculation:**

If the Scrum Team has 8 developers, each working full-time, and the team has completed 40 Story Points on average in past sprints, then, the ideal Sprint Size would be 40 Story Points for the next sprint.

**What is Scrum Size?**

Scrum Size refers to the total scope of work (Product Backlog size) for the entire Scrum project. It represents the total effort needed to complete the entire product.

**Example of Scrum Size Calculation:**

Suppose the total estimated Story Points for all backlog items = 400

If the team’s velocity per sprint = 40 Story Points

Then, the estimated number of sprints required = 400 / 40 = 10 sprints

**Question 14:** Explain DOR and DOD

**DOR (Definition of Ready)** and **DOD (Definition of Done)** are essential criteria in Scrum that ensure work items are **well-prepared before development starts** and **fully completed before being accepted**.

**1. Definition of Ready (DOR) – Before Work Starts**

DOR ensures that a User Story is clear, well-defined and ready for development. It prevents the team from working on vague or incomplete requirements.

DOR is used before a User Story is pulled into a Sprint (during Sprint Planning).

**Common DOR Criteria:**

* User Story has a clear description and Acceptance Criteria
* User Story is small enough to be completed within one Sprint
* Business Value (BV) and Complexity Points (CP) are assigned
* Dependencies are identified and resolved
* UI/UX designs or API documentation (if needed) are available
* No external approvals are pending

**Example of DOR in Scrum Foods:**

A User Story: "As a customer, I want to track my food delivery in real-time."

**DOR Checklist:**

* GPS Tracking API is selected
* UI/UX mockups for the tracking screen are available
* API documentation is shared with the developers
* User Story is estimated (BV: ₹500, CP: 8)
* Dependencies with the database team are resolved

**2. Definition of Done (DOD) – After Work is completed**

DOD ensures that a User Story or feature is fully completed before being accepted. It prevents unfinished or buggy code from being released.

DOD is used before marking a User Story as "Done" and before releasing a Sprint Increment

**Common DOD Criteria:**

* Code is written, tested and peer-reviewed
* Feature passes Unit Tests & Integration Tests
* Code is deployed in a staging/test environment
* No major defects (e.g., high/critical bugs are fixed)
* UI/UX is verified against the designs
* Product Owner accepts the story based on Acceptance Criteria

**Example of DOD in Scrum Foods:**

A User Story: "As a customer, I want to track my food delivery in real-time."

DOD Checklist:

* GPS Tracking API is successfully integrated
* UI displays live tracking updates
* Feature passes all functional tests
* No high/medium priority bugs found in testing
* Product Owner approves the feature after review

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

**Prioritization Techniques**

In Agile Scrum, teams must prioritize user stories and features to ensure that the most valuable work is done first. Prioritization helps deliver maximum business value within limited time and resources.

Here are some commonly used prioritization techniques:

**1. MoSCoW Method**

It is a simple method that categorizes tasks into four priority levels:

* M – Must Have: Critical features without which the product fails
* S – Should Have: Important but not essential in the first release
* C – Could Have: Nice-to-have features if time allows
* W – Won't Have: Features for future sprints/releases

**Example in Scrum Foods:**

|  |  |
| --- | --- |
| Feature | Category |
| Customer Login | Must Have |
| Order Tracking | Should Have |
| Food Recommendations | Could Have |
| AR-based Virtual Restaurant View | Won't Have |

**2. Business Value (BV) vs. Complexity Points (CP)**

This technique balances Business Value (BV) against Complexity Points (CP) to prioritize features.

* High BV, Low CP → Do first (quick wins)
* High BV, High CP → Plan carefully
* Low BV, Low CP → Consider if time allows
* Low BV, High CP → Avoid or last priority

**Example:**

|  |  |  |  |
| --- | --- | --- | --- |
| User Story | BV (₹1000, ₹500, ₹100, etc.) | CP (1, 2, 3, 5, etc.) | Priority |
| Login Feature | ₹1000 | 3 | High |
| AI-based Food Suggestion | ₹500 | 13 | Medium |
| Augmented Reality View | ₹100 | 20 | Low |

**What is MVP (Minimum Viable Product)?**

MVP is the smallest version of a product that delivers value to customers and can be launched quickly.

* Includes only essential features needed for initial users
* Helps collect customer feedback early
* Reduces time to market
* Avoids unnecessary complexity

**Example of MVP for Scrum Foods:**

Phase 1 (MVP Release)

* User Registration & Login
* Restaurant Listing & Menu
* Online Ordering & Payment
* Order Tracking

Phase 2 (Post-MVP Enhancements)

* AI-based Food Suggestions
* Subscription Plans
* AR-based Restaurant Experience

MVP is used for below reasons

* Launches the product quickly
* Focuses on core customer needs
* Saves time & resources

**Question 16:** – Difference between Business Analyst n Product Owner

Both **Business Analysts (BAs)** and **Product Owners (POs)** play critical roles in Agile and Scrum teams, but their responsibilities, focus areas and decision-making powers differ significantly.

|  |  |  |
| --- | --- | --- |
| Aspect | Business Analyst (BA) | Product Owner (PO) |
| Primary Role | Analyses business needs, gathers requirements and translates them into clear documentation. | Owns the product vision, defines the backlog and ensures business value delivery. |
| Focus Area | Business processes, requirement analysis, stakeholder communication. | Product backlog, prioritization, customer needs, maximizing product value. |
| Decision-Making Authority | Advises on requirements but doesn’t have final authority. | Owns product decisions and prioritization. |
| Interaction with Stakeholders | Works with business users, customers and technical teams to gather and analyse requirements. | Works closely with customers, leadership and the Scrum Team to define and refine the product backlog. |
| Backlog Management | Helps refine user stories and ensures detailed documentation. | Owns and prioritizes the backlog based on business value. |
| Technical Involvement | Works with developers but doesn’t prioritize development tasks. | Engages in backlog grooming, sprint planning and defines development priorities. |
| Customer Involvement | Indirect – collects feedback and requirements. | Direct – makes decisions based on customer feedback. |
| Agile Framework | Can work in various methodologies (Scrum, Kanban, Waterfall, etc.). | Works specifically within Scrum as a key role. |
| Sprint Involvement | Supports the team by clarifying requirements. | Owns Sprint Planning, Backlog Grooming and Acceptance of user stories. |
| Example Responsibility in Scrum Foods | Documents user requirements like “Customers should be able to track orders”. | Decides “Real-time tracking should be implemented before AI food recommendations”. |

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

Venkatesh D

xyz@gmail.com | xxx xxx 7964

PROFESSIONAL SUMMARY

Results-driven Product Owner with 3 years of experience in agile environments, specializing in product backlog management, user story creation, and cross-functional team collaboration. Passionate about delivering high-value software solutions that enhance customer experience and meet business goals. Strong expertise in Scrum, Agile methodologies, stakeholder management and backlog prioritization.

SKILLS & EXPERTISE

* Agile & Scrum Methodologies
* Product Backlog Management
* User Stories & Acceptance Criteria
* Stakeholder Engagement & Communication
* Sprint Planning & Prioritization
* Requirement Gathering & Analysis

WORK EXPERIENCE

#1. Product Owner

[Company Name] – [Location] | [Month/Year] – Present

Managed and prioritized product backlog to align with business goals, ensuring high-value feature delivery.

Collaborated with Scrum teams (developers, QA, UX) to refine and execute user stories.

Defined clear Acceptance Criteria and worked closely with the team to ensure successful feature implementation.

Led Sprint Planning, Backlog Grooming, and Sprint Reviews, driving continuous product improvement.

Acted as the primary liaison between stakeholders and the development team, ensuring business needs were met.

Conducted competitive analysis to identify trends and opportunities for product enhancements.

Measured team velocity and adjusted backlog priorities for optimal sprint performance.

#2. Business Analyst (Prior Role, if applicable)

[Previous Company Name] – [Location] | [Month/Year] – [Month/Year]

Gathered and documented business requirements from stakeholders and translated them into user stories.

Worked closely with Product Owners and Scrum teams to define and prioritize features.

Conducted market research and user feedback analysis to improve product features.

Assisted in Sprint Planning and Backlog Grooming sessions.

EDUCATION

[Degree] – [University Name] – [Year]

TOOLS & TECHNOLOGIES

JIRA | Azure DevOps

Google Analytics | Tableau | Power BI (Optional)

SQL

[Your Name]