COEPD- BUSINESS ANALYSIS COURSE

CAPSTONE PROJECT- 2

Scrum Project Implementation

Introduction:

Scrum Project Name: Scrum Foods (Foods Delivery Applications)

Scrum Project Description: Scrum Foods provides fast, reliable online food delivery application targeting customers of all age groups offering 24/7 service along with tracking of the delivery real time.

Question 1: Write the Agile Manifesto.

The Agile Manifesto is the foundational document that defines the Agile Perspective in business analysis, emphasizing a flexible mindset embodied in a set of values and principles.

It articulates the four **foundational values** and twelve **reinforcing principles** that define the Agile mindset. Within the BABOK V3's **Agile Perspective**, these components serve as the ethical and operational guide for Business Analysts operating in adaptive, iterative, and high-change environments.

The Four Values:

These values represent the preference for one set of items on the left over the items on the right:

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.

The Twelve Principles:

The principles guide the implementation of the values:

1. Our highest priority is to satisfy the customer through **early and continuous delivery** of valuable software.
2. Welcome **changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage.

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

Question 2: What is a User Story? Define Acceptance Criteria? (write 40 user stories along with acceptance criteria, BV and CV.)

User Stories are a technique used to articulate stakeholder needs by focusing on the **value** delivered to an end-user or customer. These stories serve as incremental units of business value that are used to define system requirements, prioritize development work, and guide the iterative delivery process in an Agile approach. The format, *As a <Role>, I want to <Goal>, so that <Benefit>*, ensures that each requirement is traceable to a specific stakeholder, action, and resulting business outcome.

Acceptance Criteria define the specific conditions that must be met for a solution to be considered complete and ready for stakeholder acceptance. They serve as the **testable rules or checkpoints** that validate whether a user story or requirement has been satisfied.

ID	Role	Goal	Benefit	Full User Story	Acceptance Criteria	BV (1-500)	CP (1-20)
Customer (C) - Ordering & Experience							

C-1	Customer	browse local restaurants by my location	I can quickly find available food options.	As a Customer, I want to browse local restaurants by my location, so that I can quickly find available food options.	- The app displays a list of restaurants within a user-defined radius. - Each restaurant entry shows its name, cuisine, and rating.	400	8
C-2	Customer	search for specific dishes or ingredients	I can find exactly what I am craving.	As a Customer, I want to search for specific dishes or ingredients, so that I can find exactly what I am craving.	- The search function returns a list of matching menu items. - Search results include the restaurant name and location.	300	8
C-3	Customer	view high-qualit y photos of food items	I can better visualize my potential order.	As a Customer, I want to view high-quality photos of food items, so that I can better visualize my potential order.	- Each menu item has a high-resolution image associated with it. - Tapping the image allows a full-screen view.	100	5
C-4	Customer	add custom instructions to my order	My food is prepared exactly to my liking.	As a Customer, I want to add custom instructions to my order, so that my food is prepared exactly to my liking.	- A text box is available in the shopping cart for special instructions. - The instructions are transmitted to the restaurant with the order.	200	3
C-5	Customer	save my favorite restaurants in a list	I can easily reorder from them later.	As a Customer, I want to save my favorite restaurants in a list, so that I can easily reorder from them later.	- A "favorite" button is available on each restaurant page. - A dedicated section in the user profile shows all favorite restaurants.	200	5
C-6	Customer	filter restaurants by price range and rating	I can quickly narrow down my choices.	As a Customer, I want to filter restaurants by price range and rating, so that I	- Filters for price (e.g., $, $$, $$$) and a 1-5 star rating are available. - The displayed list updates instantly as	300	8

				can quickly narrow down my choices.	filters are applied.		
C-7	Customer	view the estimated delivery time before placing an order	I can manage my expectations for food arrival.	As a Customer, I want to view the estimated delivery time before placing an order, so that I can manage my expectations for food arrival.	- The estimated time is prominently displayed on the checkout screen. - The estimate is based on the restaurant's average preparation time and delivery distance.	400	13
C-8	Customer	track the real-time location of the delivery boy on a map	I can know exactly where my order is.	As a Customer, I want to track the real-time location of the delivery boy on a map, so that I can know exactly where my order is.	- The order status page shows a live map with the delivery boy's icon. - The icon updates every 30 seconds.	500	21
C-9	Customer	re-order a past meal with a single click	I can save time on future orders.	As a Customer, I want to re-order a past meal with a single click, so that I can save time on future orders.	- A "Reorder" button is available on each item in the past orders list. - Clicking the button adds the previous order items to the new cart.	200	5
C-10	Customer	rate and review the entire delivery experience	I can contribute to the platform's service quality.	As a Customer, I want to rate and review the entire delivery experience, so that I can contribute to the platform's service quality.	- A 5-star rating and comment box appear after the order is delivered. - The user can rate both the restaurant and the delivery boy.	300	5
C-11	Customer	pay using various methods (e.g., credit card, cash, e-wallet)	I have flexible payment options.	As a Customer, I want to pay using various methods (e.g., credit card, cash, e-wallet), so that I have flexible payment	- Options for "Credit/Debit Card," "Cash on Delivery," and "E-wallet" are available at checkout. - Selecting a payment method changes the final checkout screen.	500	13

				options.			
C-12	Customer	receive SMS confirmation upon successful order placement	I have immediate proof that my order went through.	As a Customer, I want to receive SMS confirmation upon successful order placement, so that I have immediate proof that my order went through.	- An SMS is sent to the user's registered mobile number within 1 minute of a successful order. - The SMS contains the order number and total amount.	200	5
C-13	Customer	easily contact customer support from the order screen	I can get immediate help with any issues.	As a Customer, I want to easily contact customer support from the order screen, so that I can get immediate help with any issues.	- A "Contact Support" button is visible on the order status screen. - The button initiates a live chat with a support agent.	300	13
C-14	Customer	schedule a delivery for a future time or date	I can plan my meals ahead of time.	As a Customer, I want to schedule a delivery for a future time or date, so that I can plan my meals ahead of time.	- A "Schedule" option is available at checkout. - The user can select a delivery date and time in the next 7 days.	200	8
C-15	Customer	view the total order price, including all fees and taxes	I know the exact final cost upfront.	As a Customer, I want to view the total order price, including all fees and taxes, so that I know the exact final cost upfront.	- A clear price breakdown (subtotal, delivery fee, tax) is shown on the checkout page. - The total price matches the amount charged to the payment method.	400	5
C-16	Customer	receive a loyalty reward or discount notification	I am encouraged to use Scrum Foods	As a Customer, I want to receive a loyalty reward or discount notification, so	- The app sends a push notification about a new discount code. - The discount is automatically applied to	200	3

			repeatedly.	that I am encouraged to use Scrum Foods repeatedly.	the cart upon redemption.		
C-17	Customer	receive a notification when my restaurant starts preparing my food	I know the order process has begun.	As a Customer, I want to receive a notification when my restaurant starts preparing my food, so that I know the order process has begun.	- A push notification is sent to the user when the restaurant updates the order status to "In Preparation".	100	3
C-18	Customer	edit my delivery address before the order is confirmed	I can correct any last-minute mistakes.	As a Customer, I want to edit my delivery address before the order is confirmed, so that I can correct any last-minute mistakes.	- An "Edit Address" button is available on the cart screen. - The user can change the address only before the order is paid and confirmed.	200	5
Restaurant /Vendor (R) - Management & Fulfillment							
R-19	Restaurant	register my business details and bank account	I can legally operate on the platform and receive payments.	As a Restaurant, I want to register my business details and bank account, so that I can legally operate on the platform and receive payments.	- The registration form includes fields for business name, address, tax ID, and bank details. - The platform validates the bank account details before approval.	400	8

R-20	Restaurant	manage my menu's availability in real-time	I can prevent orders for out-of-stock items.	As a Restaurant, I want to manage my menu's availability in real-time, so that I can prevent orders for out-of-stock items.	- A toggle switch is available for each menu item to mark it as "available" or "unavailable." - Unavailable items are greyed out or hidden from the customer's view.	300	8
R-21	Restaurant	receive instant, audible alerts for new orders	I can start food preparation immediately.	As a Restaurant, I want to receive instant, audible alerts for new orders, so that I can start food preparation immediately.	- A loud, distinct sound plays on the restaurant's tablet when a new order arrives. - The alert continues until the order is opened.	400	5
R-22	Restaurant	view the customer's special instructions clearly	I can fulfill the order accurately.	As a Restaurant, I want to view the customer's special instructions clearly, so that I can fulfill the order accurately.	- Special instructions are displayed in a separate, bolded section on the order ticket. - The instructions are printed on the physical order receipt.	200	3
R-23	Restaurant	accept or reject an order within a set time limit	I can manage my kitchen capacity efficiently.	As a Restaurant, I want to accept or reject an order within a set time limit, so that I can manage my kitchen capacity efficiently.	- The restaurant has 5 minutes to either accept or reject a new order. - If rejected, an automated message is sent to the customer.	400	8
R-24	Restaurant	manually set the preparation time for complex orders	the customer receives an accurate estimated delivery time.	As a Restaurant, I want to manually set the preparation time for complex orders, so that the customer receives an accurate estimated	- The restaurant can choose from a list of predefined prep times (e.g., 15 mins, 30 mins) for a new order.	200	5

				delivery time.			
R-25	Restaurant	update the order status to "Ready for Pickup"	a delivery boy can be efficiently dispatched and notified.	As a Restaurant, I want to update the order status to "Ready for Pickup," so that a delivery boy can be efficiently dispatched and notified.	- A "Ready" button is available on the order screen. - Clicking the button triggers a notification to the assigned delivery boy.	400	8
R-26	Restaurant	view a summary dashboard of my daily sales and total orders	I can monitor my business performance on the platform.	As a Restaurant, I want to view a summary dashboard of my daily sales and total orders, so that I can monitor my business performance on the platform.	- The dashboard shows total revenue and number of orders for the current day. - The data can be filtered by week or month.	400	13
R-27	Restaurant	temporarily close my shop online for non-business hours	customers aren't disappointed by placing orders I can't fulfill.	As a Restaurant, I want to temporarily close my shop online for non-business hours, so that customers aren't disappointed by placing orders I can't fulfill.	- A toggle switch to "Go Offline" is available on the main screen. - When offline, the restaurant page is not visible to customers.	200	2
R-28	Restaurant	review the delivery boy's profile and rating	I can ensure my food is handled by a reliable person.	As a Restaurant, I want to review the delivery boy's profile and rating, so that I can ensure my food is handled by a reliable person.	- The assigned delivery boy's profile and rating are visible on the order screen.	100	3

R-29	Restaurant	upload new promotional banners or deals for my menu	I can attract more customers to my page.	As a Restaurant, I want to upload new promotional banners or deals for my menu, so that I can attract more customers to my page.	- An admin panel allows uploading a banner image and linking it to a specific menu item. - The banner is visible on the restaurant's main page for customers.	300	8
R-30	Restaurant	see feedback and ratings specifically for my food and service	I can identify areas for improvement in my operation.	As a Restaurant, I want to see feedback and ratings specifically for my food and service, so that I can identify areas for improvement in my operation.	- The restaurant dashboard has a "Reviews" section with customer ratings and comments. - The restaurant can reply to customer reviews publicly.	200	5
Delivery Boy (DB) - Logistics & Income							
DB-31	Delivery Boy	register my personal and vehicle details with verification	I can be approved to start accepting jobs.	As a Delivery Boy, I want to register my personal and vehicle details with verification, so that I can be approved to start accepting jobs.	- The registration form includes fields for personal ID, vehicle type, license plate, and bank details. - The system verifies the submitted documents before activating the account.	400	13
DB-32	Delivery Boy	toggle my availability status (online/offline)	I can control when I receive delivery job offers.	As a Delivery Boy, I want to toggle my availability status (online/offline), so that I can control when I receive delivery job offers.	- A toggle is available on the home screen to switch between "Online" and "Offline" status. - Only "Online" drivers receive job offers.	300	3

DB-33	Delivery Boy	receive a clear notification for a new job with route and pay details	I can quickly decide whether to accept the delivery.	As a Delivery Boy, I want to receive a clear notification for a new job with route and pay details, so that I can quickly decide whether to accept the delivery.	- A push notification is sent with the restaurant and customer address, and estimated earnings. - The driver has 60 seconds to accept the job before it is passed to another driver.	500	8
DB-34	Delivery Boy	access GPS navigation for the pick-up and drop-off locations	I can efficiently complete the delivery route.	As a Delivery Boy, I want to access GPS navigation for the pick-up and drop-off locations, so that I can efficiently complete the delivery route.	- A "Navigate" button on the job screen opens Google Maps or Waze with the destination pre-filled.	500	5
DB-35	Delivery Boy	update the order status to "Picked Up"	the customer's tracking is updated accurately.	As a Delivery Boy, I want to update the order status to "Picked Up," so that the customer's tracking is updated accurately.	- The "Picked Up" button is only enabled after the driver is within 100 meters of the restaurant. - The customer's map tracking begins upon this update.	400	5
DB-36	Delivery Boy	contact the customer via in-app call or message	I can resolve any issues with finding the address.	As a Delivery Boy, I want to contact the customer via in-app call or message, so that I can resolve any issues with finding the address.	- The "Contact Customer" button is available on the job screen. - The button initiates a call or a chat within the app without revealing the user's personal number.	300	8

DB-37	Delivery Boy	confirm delivery by receiving a digital customer signature/code	The order is officially completed and verified.	As a Delivery Boy, I want to confirm delivery by receiving a digital customer signature/code, so that the order is officially completed and verified.	- The app prompts the delivery boy to enter a code provided by the customer at drop-off. - The code must match the system-generated code for the order to be marked as "Delivered."	400	13
DB-38	Delivery Boy	view my past delivery history and total earnings	I can track my productivity and income.	As a Delivery Boy, I want to view my past delivery history and total earnings, so that I can track my productivity and income.	- A dashboard shows a list of all completed deliveries. - Each entry lists the date, order number, and earnings.	300	5
DB-39	Delivery Boy	receive my earnings into my registered bank account automatically	I have reliable access to my income.	As a Delivery Boy, I want to receive my earnings into my registered bank account automatically, so that I have reliable access to my income.	- Payments are deposited weekly into the delivery boy's bank account. - The system sends a notification upon successful deposit.	500	21
DB-40	Delivery Boy	rate the restaurant's handover process	I can provide feedback to improve my efficiency at pickup.	As a Delivery Boy, I want to rate the restaurant's handover process, so that I can provide feedback to improve my efficiency at pickup.	- A rating option (1-5 stars) is provided after the "Picked Up" status is updated. - The rating is submitted to the system without affecting personal metrics.	100	3

The scores are justified by looking at **risk and necessity** (for BV) versus **technical difficulty** (for CP).

- **High BV** (like 400-500) goes to features that are **core to the business model** and customer trust, such as being able to order, pay, track, and manage money (DB-39, C-8, C-11). Without them, the app is useless.
- **Low BV** (like 100-200) goes to features that are "nice-to-have," like rating the *handover* process (DB-40) or seeing *high-quality photos* (C-3).
- **High CP** (like 15-20) goes to features that require **external integration or real-time data**, such as payment processing (C-11), live GPS tracking (C-8), or complex verification and automatic payments (DB-39, DB-31).
- **Low CP** (like 2-5) goes to features that are **simple data display or text input**, like closing the shop online (R-27), adding custom instructions (C-4), or viewing a simple dashboard (R-27).

We prioritize the stories with the best **BV/CP ratio** (high BV, low CP) to deliver the most value efficiently.

Business Value (BV), within the context of BABOK V3's **Value** Core Concept, is the net benefit derived from a solution or an initiative. It represents the importance, worth, or usefulness of something to a stakeholder within an enterprise. In an Agile context, BV is a key factor for **prioritization** of requirements and is measured by metrics such as increased revenue, reduced costs, improved customer satisfaction, or mitigated risk.

Complexity Points (CP), in an Agile context, are a measure of the **difficulty** of implementing a requirement or user story. Unlike effort-based sizing, complexity points are an abstract measure that considers multiple factors, including:

- The technical difficulty of the task.
- The level of uncertainty or risk involved.
- The number of integrations or dependencies on other systems.
- The team's prior experience with a similar task.

CP is an input into the prioritization process, often considered alongside Business Value, to ensure a balanced product backlog.

Question 3: What is an Epic? Write 2 epics.

An **Epic** is a large, complex, and often open-ended body of work (a requirement, feature, or user story) that encompasses multiple smaller, related requirements or User Stories. Epics are typically broken down (or **decomposed**) into smaller, manageable components—often called **Features** or **User Stories**—which are then estimated and delivered incrementally over several iterations or Sprints.

The typical decomposition hierarchy flowing down from the highest strategic level is:

1. **Themes/Initiatives (Highest Level):** Broad, strategic business goals (e.g., "Expand Market Share in Asia").
2. **Epics:** Large features derived from a theme (e.g., "Implement Multi-Lingual Support").
3. **Features:** Mid-level functions derived from an Epic (e.g., "Allow user to select preferred language").
4. **User Stories:** Smallest unit of value, deliverable within a single iteration (e.g., "As a user, I want a language dropdown, so that I can switch from English to Spanish.").

Epics serve as a crucial tool in **Requirements Life Cycle Management** for ensuring traceability and alignment between strategic intent and detailed implementation tasks.

Epic 1: Real-Time Order Visibility and Tracking

Component	**Description**
Epic Title	Real-Time Order Visibility and Tracking System
Role	Primary Customer, but also Restaurant and Delivery Boy
Goal	To implement a fully integrated, real-time tracking and status notification system.
Benefit	To fulfill the core promise of reliable service and provide high customer confidence, significantly reducing support inquiries.
Decompositio n Scope	Encompasses all requirements related to geolocation, status updates (C-8, DB-35, C-17), and estimated time calculations (C-7, DB-34).
High-Level Story	As a **Scrum Foods Customer**, I want **live, continuous updates on my order's status and location**, so that **I have complete confidence in the delivery process and can minimize waiting time.**

Epic 2: Secure and Seamless Multi-Role Transaction Management

Component	Description
Epic Title	Secure and Seamless Multi-Role Transaction Management
Role	Primary Customer, Restaurant, and Delivery Boy
Goal	To integrate secure, diverse payment methods, manage commission structures, and ensure timely, accurate financial payouts.
Benefit	To establish financial stability and trust across all platform users (customers, vendors, and drivers), enabling rapid scaling of the platform.
Decomposition Scope	Encompasses all requirements related to payments (C-11), registration and verification (R-19, DB-31), reporting (R-26, DB-38), and automated payouts (DB-39).
High-Level Story	As a **Scrum Foods Platform**, I want **robust and secure financial processes for ordering, receiving payments, and automatic payouts**, so that **I can maintain cash flow, ensure compliance, and guarantee reliable income for all partners.**

Question 4: Differences between Business Value (BV) and Complexity Points (CP)

Business Value (BV)	**Complexity Points (CP)**
Business value is the net benefit or worth derived by a stakeholder or the enterprise from the successful implementation of a requirement.	Complexity points are an abstract measure of the difficulty, effort, and uncertainty involved in designing and implementing a requirement.
Business-Centric: Focuses on the strategic, financial, or operational goals of the organization.	**Development-Centric:** Focuses on the technical challenges and resources required by the team.
Revenue generation, cost savings, risk mitigation, regulatory compliance, competitive advantage, or customer satisfaction.	Technical difficulty, number of dependencies (integrations), technical risk/uncertainty, and team experience.
Primarily the **Product Owner** (or **Business Analyst** acting as proxy/facilitator) in consultation with business stakeholders.	Primarily the **Development Team** (using techniques like Planning Poker or relative estimation).
To define **What** to do (the priority based on organizational need).	To estimate **How** much effort is required (the cost/effort of delivery).
Directly relates to the **Value** Core Concept and **Strategy Analysis** Knowledge Area.	Relates to the **Underlying Competencies** of Systems Thinking and Analytical Thinking for estimation.

Question 5: What is a Sprint?

A **Sprint** (or **Iteration**) is a fixed, time-boxed period during which a cross-functional team works to produce a **potentially shippable increment** of

product functionality. It is the heart of the Scrum framework and a key mechanism for delivering value incrementally, aligning with the Agile principle of "delivering working software frequently"

Time-Boxing	Sprints are of a fixed duration, typically ***two to four weeks***. Once set, the time-box cannot be extended, which enforces discipline and predictable delivery cycles.
Goal	Each Sprint has a defined Sprint Goal, which is a high-level objective that the team commits to achieving by delivering a set of features (User Stories) from the Product Backlog.
Scope	The specific Product Backlog Items selected for the Sprint form the Sprint Backlog. Once the Sprint begins, the scope is generally frozen or highly controlled to maintain focus and maximize the chance of achieving the goal.
Output	The successful outcome of a Sprint is a Product Increment—a working, tested, and potentially shippable piece of software or solution.
Events/Meetings	A Sprint is a container for all other Scrum events: Sprint Planning (to define goal and scope), Daily Scrum (to inspect progress), Sprint Review (to demonstrate the increment), and Sprint Retrospective (to inspect and adapt the process).
Business Analysis Role	The Business Analyst's role during the Sprint is focused on Elicitation and Collaboration (clarifying requirements in real-time) and Requirements Analysis and Design Definition (refining future stories and creating acceptance criteria for stories currently in the Sprint).

SPRINT PROCESS:

Sprint Event	**Timing**	**Scrum Foods Action**
1. Sprint Planning	Day 1 (4-8 hours)	**Objective**: Define the Sprint Goal. "Enable a Customer to successfully log in, find a nearby restaurant, add one item to a cart, and complete a cash-on-delivery order." **BA/PO Action**: Select the highest priority User

		Stories (e.g., C-1, C-4, C-11 [Cash only]). The team commits to the work and refines the CP estimates.
2. Sprint Execution	Day 1 to Day 10 (10 days)	**Team Action**: Developers code the features, QA verifies against Acceptance Criteria. **BA Action (Elicitation & Collaboration)**: The BA remains available to answer questions on the Acceptance Criteria for stories like C-1 and C-11, ensuring the developers build exactly what the business needs.
3. Daily Scrum (Daily Standup)	Daily (15 minutes)	**Team Action**: The Development Team coordinates work (What did I do? What will I do? What are my impediments?). **BA Role**: Not an active speaker, but listens for blockers related to requirements ambiguity and proactively clarifies or resolves dependencies on external stakeholders (e.g., getting final logo assets).
4. Sprint Review	Day 11 (2 hours)	**Objective**: Inspect the Product Increment and gather feedback. **Team Action**: Demonstrate the working features (e.g., show the live browsing and checkout process). **Stakeholder Action**: Customers/Owners validate the Business Value of the completed work against their original needs.
5. Sprint Retrospective	Day 11 (1-2 hours)	**Objective**: Inspect the process and adapt. **Team Action**: Discuss how the team worked (What went well? What could be improved?). **BA Action**: Proposes changes to collaboration methods, such as starting backlog refinement earlier to improve the quality of stories entering the next Sprint.

EXAMPLE: A Sprint is a **two-week cycle** that delivers a working product chunk. For Scrum Foods, let's say **Sprint 1** aims for **"Minimum Ordering Capability."**

1. **Planning:** We pick high-priority stories (e.g., **C-1 Browse**, **C-15 Pricing**, **C-11 Cash Payment Only**). The team commits to making those features *work*.
2. **Execution:** Developers code. The BA's job is to be the **requirements expert**, clarifying the **Acceptance Criteria** for the team *daily* so there are no delays.
3. **Review:** We demo the working app to the business stakeholders. The customer can actually browse and checkout with cash payment.
4. **Retrospective:** The team discusses how to work better next time—maybe they need clearer technical details on the payment API, which informs the **BA's refinement work** for the next Sprint.

Question 6: Explain about the Product backlog and the Sprint backlog.

The **Product Backlog** and the **Sprint Backlog** are the primary artifacts used to manage the requirements and work scope in a Scrum/Agile environment, establishing a clear line of sight between strategic direction and tactical execution.

1. Product Backlog (Strategic/Initiative Horizon)

The Product Backlog is the **single, definitive, and ordered list** of everything that is known to be needed in the product. It is a living artifact that constantly evolves and is never truly "complete."

Product Owner	The **Product Owner** is singularly accountable for the Product Backlog, including its content, ordering, and availability. The Business Analyst often supports the PO through Elicitation and Collaboration and Requirements Analysis (refinement).
Contents	It contains all requirements—expressed as User Stories, Epics, features, functions, non-functional requirements (NFRs), and defects (bugs).

Ordering (Prioritization)	Items are ordered primarily by Business Value, cost/effort (CP), risk, and dependencies. This ordering directly supports the BABOK task of Prioritize Requirements (5.3).
Refinement	This is the ongoing activity of detailing, estimating, and ordering backlog items. The top items (those needed soon) are refined down from Epics into ready-to-work User Stories, fulfilling the need for continuous Requirements Analysis and Design Definition.

2. Sprint Backlog (Delivery Horizon)

The Sprint Backlog is the subset of the Product Backlog items **selected for delivery** during the current Sprint, plus the **plan** for delivering them.

Owner	The Development Team owns the **Sprint Backlog**. Only the team can add, remove, or adjust the work once the Sprint begins.
Contents	1. User Stories: The highest-priority items pulled from the Product Backlog. 2. Tasks: The decomposition of those User Stories into smaller, technical, executable pieces of work (e.g., "Write database query," "Design UI screen").
Sprint Goal	The Sprint Backlog must be aligned with and support the single Sprint Goal that the team commits to achieving.
"Definition of Done" (DoD)	The entire Sprint Backlog's completion is validated against the Definition of Done—a formal description of the quality standards, testing, and approval necessary for a feature to be considered fully integrated and "shippable." The DoD is critical for the BABOK task of Define Solution Scope (7.4) and Solution Evaluation.

Question 7 : What is impediments log? write 2 impediments.

An **Impediment Log** (or Obstacle Log/Blocker List) is a dynamic, documented record of all issues, problems, or obstacles that are preventing the Development Team or a specific stakeholder from performing their work efficiently or achieving the Sprint Goal. It is a critical tool for risk management and performance assessment in an Agile context.

Purpose	To centralize, track, and manage the resolution of barriers that hinder delivery. Tracking impediments contributes directly to the Performance Assessment task in BABOK, as unresolved impediments directly impact time and scope.
Owner	Typically the Scrum Master ***is responsible for facilitating the removal of impediments***, while the Business Analyst ***is often responsible for logging, clarifying, and helping resolve impediments related to requirements, scope ambiguity, or stakeholder access.***
Source	Impediments are ***primarily raised during the Daily Scrum meeting*** when a team member reports a blocker to their progress.
Resolution	Resolution often requires action from outside the Development Team (e.g., senior management, infrastructure, or the Product Owner/BA for clarity).
Required Information	Each entry must contain: a clear description of the obstacle, the team member/work item affected, the date logged, and the required action/resolution status.

Impediment ID	**Impediment Description**	**Affected User Story/Area**	**Resolution Required By**
I-001	The team cannot proceed with the Real-Time Order Tracking (C-8) story because the API credentials and documentation for the third-party GPS/Mapping service have not yet been	C-8: Track real-time location.	Head of IT

	provided by the Head of IT.		
I-002	The Restaurant Registration (R-19) form is blocked because the Legal team has not finalized the mandatory "Terms and Conditions" text regarding vendor liability, which must be displayed on the sign-up page.	R-19: Register business details.	Legal Department

Question 8: Explain Velocity of the Team.

Velocity of the Team is a measure of the average amount of work (expressed in **Complexity Points/Story Points**) that a Development Team can reliably complete and deliver as a **"Done"** product increment during a single Sprint.

Velocity is a historical metric, used for empirical planning and forecasting, and it is a key input for the Business Analyst and Product Owner when performing the following tasks:

Measurement Unit	Velocity is always ***measured in the team's chosen unit of effort***, most commonly Complexity Points (CP) or Story Points (SP). It represents the sum of the complexity points of all User Stories that ***met the Definition of Done*** (DoD) in a Sprint.
Purpose (Forecasting)	***Velocity enables the Product Owner to accurately predict how many Sprints it will take to deliver a large set of features*** (e.g., an Epic or a Release). This supports the Plan Business Analysis Approach and Plan Stakeholder Engagement tasks by providing realistic timelines.
Stability	Velocity ***should be calculated as an average over the last three to five Sprints*** to smooth out anomalies (like holidays or unexpected impediments). A stable velocity indicates a predictable team, which is a goal of Performance Assessment.
Team-Specific	Velocity is ***internal and relative only to the specific team that generated it***. It is not a measure of performance for comparison between different teams, as estimation scales and team definitions vary.
Input for Planning	During Sprint Planning, the team looks at its current stable velocity to determine the maximum amount of work (in Complexity Points) they should commit to pulling from the Product Backlog for the upcoming Sprint, ensuring a sustainable pace.

Question 9: – Draw Sprint Burndown Charts & Product Burndown Charts.

Burn Charts are graphical tools used to visually track the progress of work over time against a defined scope. They are a core mechanism for transparency and inspecting progress, allowing the team and stakeholders to quickly assess if the project or Sprint is on track to meet its goals.

1. Sprint Burndown Chart (Tactical/Delivery Horizon)

The Sprint Burndown Chart is a **tactical tracking tool** used primarily by the Development Team and Scrum Master to monitor the progress of the current Sprint.

Purpose: To track the **remaining work** (in hours or complexity points) within the Sprint Backlog against the time remaining in the Sprint.

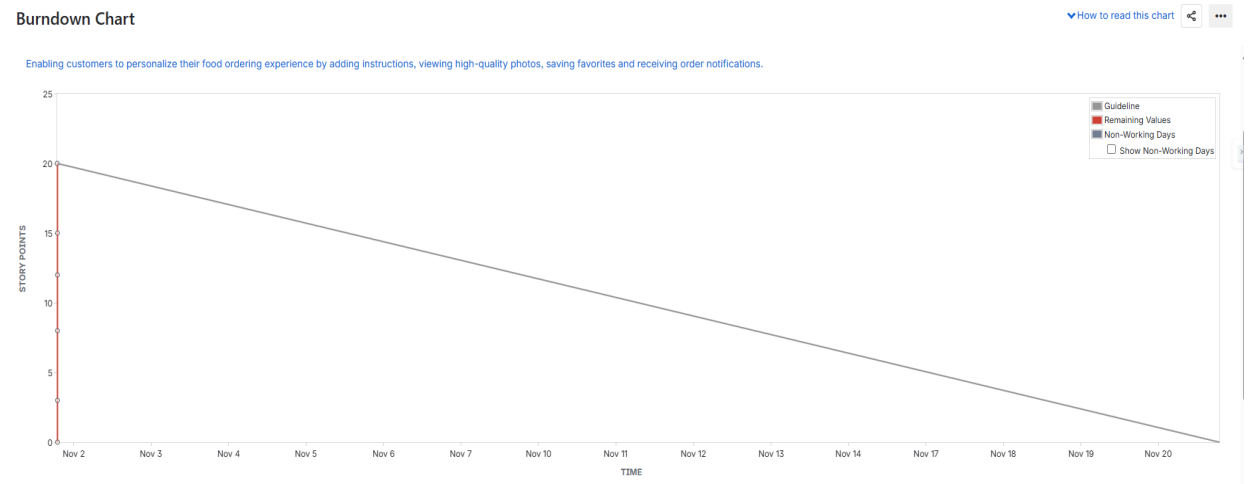
Metrics: **Y-Axis:** Remaining Work (e.g., Task Hours or Complexity Points).

X-Axis: Days of the Sprint.

Lines: **Ideal Line:** A straight diagonal line showing the perfect rate of work completion needed to finish on time.

Actual Line: The daily plot of the remaining work.

Interpretation: If the **Actual Line** is consistently *above* the **Ideal Line**, the team is behind schedule and risks failing the Sprint Goal. If it's *below*, the team is ahead. This provides a clear, quantitative input for the daily **Daily Scrum** and the **Performance Assessment** task.



The sprint burndown chart (excluding the non-working days) presents a continuous timeline of progress throughout the 3-week sprint. Starting with 20 story points, the team maintained a steady completion rate, reducing the total work to 0 by the end of the sprint. The smooth linear trend indicates ***consistent daily progress*** and a successful sprint execution without any interruption or scope changes. (***this sprint burndown chart is generated for example purposes only, progress may differ in actual real life projects.***)

				Story Points		
Date	Issue	Event Type	Event Detail	Inc.	Dec.	Remaining
01/Nov/25 6:54 PM	SF-14 SF-19 SF-5 SF-6 SF-7	Sprint start		5 3 5 3 4		20
01/Nov/25 6:59 PM	SF-5	Burndown	Issue completed		5	15
	SF-6	Burndown	Issue completed		3	12
	SF-7	Burndown	Issue completed		4	8
	SF-14	Burndown	Issue completed		5	3
	SF-19	Burndown	Issue completed		3	0

2. Product Burndown Chart (Strategic/Initiative Horizon)

The Product Burndown Chart is a **strategic forecasting tool** used primarily by the Product Owner and stakeholders to track progress toward the overall product or release goal.

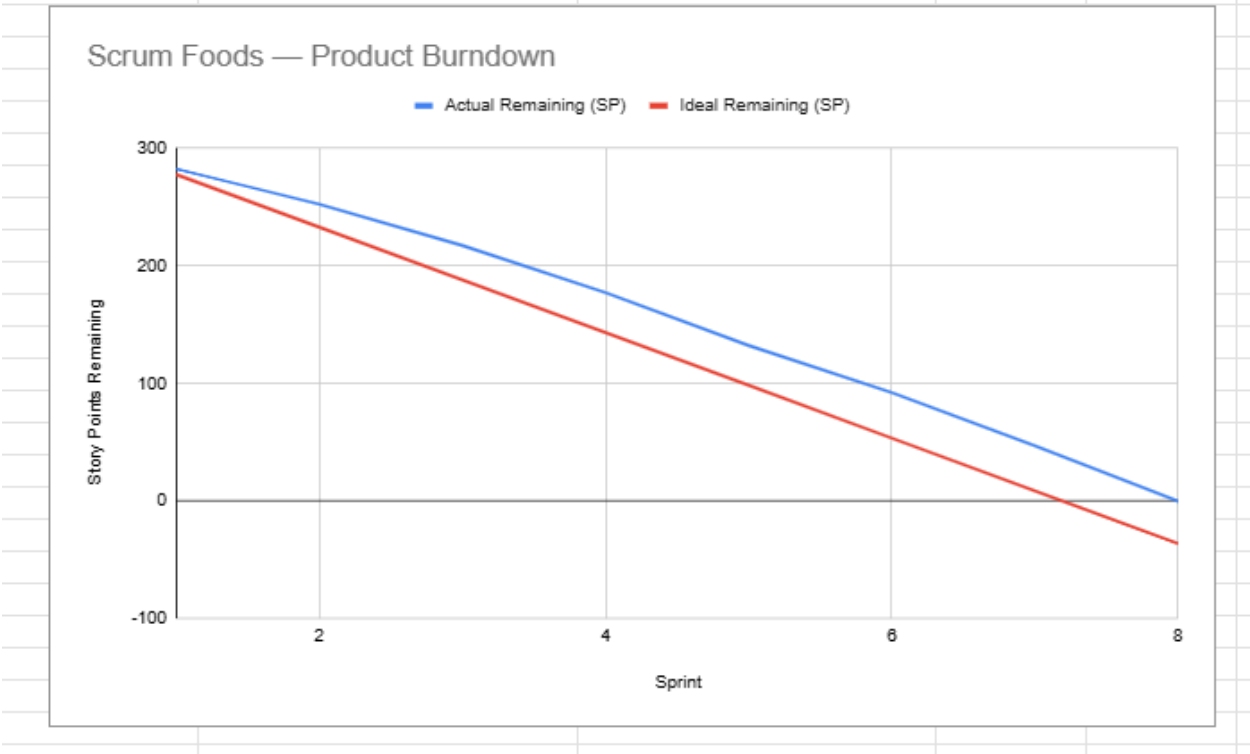
Purpose: To track the **remaining total work** (in Complexity Points) in the Product Backlog against the number of Sprints remaining in the project or release timeline.

Metrics: **Y-Axis:** Remaining total work in the Product Backlog (Complexity Points).

X-Axis: Sprints completed or time elapsed.

Interpretation: The slope of the chart indicates the average **Team Velocity**. It helps the BA/PO determine the **release date** (where the actual line hits zero) or the remaining scope that can be delivered by a fixed date. It is essential for managing stakeholder expectations and supporting **Strategy Analysis** and **Requirements Life Cycle Management**.

Scope Creep: Any sudden upward spike in the *Remaining Work* line indicates that new requirements (scope creep) have been added to the Product Backlog during the period, necessitating scope or schedule adjustments.



(chart created manually on MS EXCEL)

X-axis: Sprint numbers (1 → 8)
Y-axis: Story Points Remaining
Blue line: Actual Remaining Work
Red line: Ideal (Planned) Remaining Work

What This Chart Is Saying:

Sprint 1 – We start with around 280 SP

This is our total backlog for the Scrum Foods project.

From Sprint 1 → 3

- The blue line (actual) is slightly above the red line (ideal).
- This means our team **didn't complete as many story points as planned** during the first sprints.
- This is very normal — early sprints often involve setup, learning, and refining estimates.

Sprint 4 → 6

- The gap between actual and ideal narrows.
- This means the team's **velocity improved**.
- They are now completing work faster and catching up to plan.

Sprint 8

- The blue line hits **zero story points remaining**.
- This means **the team completed the entire product backlog by the final sprint** exactly as needed.
- The red line goes slightly below zero (which happens in ideal straight-line calculations) — and that's okay.

The chart shows our team started a little slow, gained momentum, and successfully finished all work by Sprint 8.

Question 10: What is product grooming?

The term **Product Backlog Grooming** (or **Backlog Refinement**) refers to the ongoing activity of detailing, estimating, and ordering items in the Product Backlog. It is not a formal Scrum Event but an **essential, continuous activity** that consumes up to 10% of the Development Team's capacity in a given Sprint.

Backlog Refinement directly relates to the BABOK V3 **Requirements Life Cycle Management** Knowledge Area (specifically **Maintain Requirements**) and the **Requirements Analysis and Design Definition** Knowledge Area. It is the process that ensures requirements are constantly ready for development.

The goal is to achieve a state of "Ready" for the top items on the backlog, meaning they are:

- **D-Detailed Appropriately:** They contain enough information, including clear **Acceptance Criteria**, for the team to confidently start work.
- **E-Estimated:** They have a size estimate (in Story Points or Complexity Points) agreed upon by the Development Team.
- **E-Emergent:** The backlog is constantly updated to reflect new learning, feedback, and market changes.

- **P-Prioritized:** The items are ordered based on value, risk, and dependencies.

The acronym **DEEP** is often used to describe the characteristics of a healthy Product Backlog.

Key Activities Performed During Grooming:

Activity	Performed By	Purpose
Decomposition *(or splitting the epic or userstories into sub tasks)*	BA, Product Owner (PO), Team	Breaking down large items (Epics) into smaller, manageable User Stories that can fit into a single Sprint.
Adding Detail	BA, Team	Eliciting and documenting specific Acceptance Criteria for each User Story, ensuring clarity and testability.
Re-Estimation	Development Team	Reviewing existing estimates and assigning new Complexity Points (CP) to newly detailed or decomposed stories.
Ordering	Product Owner, BA	Re-prioritizing items based on the latest estimates, newly identified dependencies, and updated business value.
Junk Removal	PO, BA	Eliminating items that are no longer relevant or valuable to the project.

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

The roles of the Scrum Master and Product Owner are essential for the effective application of the **Agile Perspective** in business analysis, providing leadership and accountability across the project.

1. Product Owner (PO)

The **Product Owner** is singularly responsible for maximizing the value of the product resulting from the work of the Development Team. They are the **voice of the customer** and the **single source of truth** for the business requirements.

Focus Area	Key Responsibilities	BABOK V3 Alignment
Vision & Strategy	Defining and clearly communicating the Product Goal and the overall product vision.	Strategy Analysis
Backlog Management	Owning, ordering, and maintaining the Product Backlog (the master list of all requirements).	Requirements Life Cycle Management
Value Maximization	Making final decisions on prioritization (Business Value) and scope to ensure the highest value is delivered first.	Value Core Concept
Acceptance	Formally accepting or rejecting the completed work (the Sprint Increment) at the Sprint Review.	Solution Evaluation

2. Scrum Master (SM)

The **Scrum Master** is a **servant-leader** for the Scrum Team, responsible for promoting and supporting Scrum as defined in the Scrum Guide. Their primary focus is on the *process* and removing obstacles.

Focus Area	Key Responsibilities	BABOK V3 Alignment
Process Coaching	Ensuring the team adheres to Scrum rules, roles, and artifacts (like Sprints, Daily Scrum, etc.).	Elicitation and Collaboration
Impediment Removal	Facilitating the removal of impediments (blockers) that slow the Development Team down (e.g., getting a key server access, resolving cross-team dependencies).	Business Analysis Planning and Monitoring
Facilitation	Leading and ensuring the effectiveness of key Scrum events (Sprint Planning,	Underlying Competency: Facilitation and Conflict Resolution

	Retrospective, Review).	
Protecting the Team	Shielding the Development Team from external distractions, scope changes, or interference during a Sprint.	Requirements Life Cycle Management

Question 12: Explain all Meetings Conducted in Scrum Project.

The meetings, or **Events**, in a Scrum project are collectively known as the **Sprint Cycle**. They are **time-boxed** (fixed duration) to ensure they are focused and productive. These events provide structure and regularity, enabling the team to **Inspect and Adapt** continuously, which is fundamental to the Agile Perspective in BABOK V3.

1. Sprint Planning

- **When:** Beginning of the Sprint (e.g., 8 hours for a 4-week Sprint).
- **Attendees:** Product Owner (PO), Scrum Master (SM), Development Team (DT).
- **Purpose:** The entire Scrum Team collaborates to define the **Sprint Goal** (the *why*) and select the Product Backlog Items (the *what*) that will be delivered in the Sprint. The team also defines the plan for delivering the items (the *how*), creating the **Sprint Backlog**.

2. Daily Scrum (Daily Standup)

- **When:** Every day during the Sprint, at the same time and location (e.g., 15 minutes).
- **Attendees:** Development Team (primarily), SM, PO (often attends but is not required to speak).
- **Purpose:** To **Inspect** progress toward the Sprint Goal and adapt the **Sprint Backlog** plan for the next 24 hours. This is a synchronization meeting, typically structured around three questions (or similar format):
    1. What did I do yesterday that helped the Development Team meet the Sprint Goal?
    2. What will I do today to help the Development Team meet the Sprint Goal?
    3. Do I see any **impediments** that prevent me or the Development Team from meeting the Sprint Goal?

3. Sprint Review

- **When:** End of the Sprint (e.g., 4 hours for a 4-week Sprint).
- **Attendees:** Scrum Team, Stakeholders (Customer, Sponsor, Subject Matter Experts).
- **Purpose:** To **Inspect** the **Product Increment** and **Adapt** the Product Backlog. The team demonstrates the work that is **"Done"** (meets the Definition of Done). Stakeholders provide feedback, and based on this, the **Product Backlog** is often re-prioritized and re-estimated for future Sprints. This is the forum for the PO to get formal acceptance.

4. Sprint Retrospective

- **When:** After the Sprint Review and before the next Sprint Planning (e.g., 3 hours for a 4-week Sprint).
- **Attendees:** Scrum Master and Development Team (internal team meeting).
- **Purpose:** To **Inspect** the *process* itself (not the product) and create a plan for improvements in the next Sprint. Key questions include:
    1. What went well during the Sprint?
    2. What challenges did we face?
    3. What specific, actionable changes can we make to our process (tools, communication, estimation) in the next Sprint?

5. Backlog Refinement (Not a formal Scrum Event)

- **When:** Ongoing activity throughout the Sprint (usually 10% of the Development Team's time).
- **Attendees:** Product Owner, Business Analyst, and Development Team members.
- **Purpose:** To ensure the Product Backlog items targeted for future Sprints are **"Ready"** (clear, estimated, and small enough). This continuous work is vital for efficient Sprint Planning and is where the Business Analyst spends much of their time.

Business Analyst’s Role:

- Clarifies business requirements and acceptance criteria.
- Ensures user stories are well-structured (INVEST) and testable.
- Helps translate business needs into actionable items.
- Assists in defining the Sprint Goal based on business priorities.

Question 13: Explain Sprint Size and Scrum Size.

1. Sprint Size (Time-Box)

Sprint Size refers to the **duration** of a single Sprint or iteration.

- **Definition:** It is the fixed, time-boxed period during which the Development Team works to complete a committed set of work.
- **Measurement:** Always measured in **time**, typically in **weeks**. Common sizes are **two weeks** (most popular) or **four weeks**.
- **Key Characteristic:** Once the Sprint Size is set, it should remain **consistent** throughout the project. Consistency allows the team to establish a stable **Velocity**, which makes planning and forecasting predictable.
- **BABOK V3 Relevance:** Consistency in Sprint Size is crucial for the **Business Analysis Planning and Monitoring** Knowledge Area, enabling accurate **Performance Assessment** and reliable forecasting of the project schedule.

2. Scrum Team Size (Team Capacity)

Scrum Team Size (more accurately called the **Development Team Size**) refers to the number of people performing the work.

- **Definition:** It is the size of the cross-functional group that does the work of creating the potentially shippable increment.
- **Measurement:** Measured in the **number of people**.
- **Scrum Guide Recommendation:** The Development Team should be small enough to remain agile and large enough to complete significant work within a Sprint. The common recommendation is **3 to 9 members**, excluding the Product Owner (PO) and Scrum Master (SM).
- **Why the Size Limit?** Teams larger than 9 people typically require too much communication overhead, making the process less efficient and introducing

unnecessary complexity. Teams smaller than 3 members may lack the necessary skills to deliver a complete, quality increment.

- **BABOK V3 Relevance:** This size limit ensures the team can operate effectively, which minimizes communication overhead and supports timely **Elicitation and Collaboration** to clarify requirements without constant formal meetings.

Question 14: Explain DOR and DOD.

DOR (Definition of Ready) and **DOD (Definition of Done)** are essential quality gates that prevent problems from moving through the development pipeline.

1. Definition of Ready (DOR)

The **Definition of Ready (DOR)** is a formal agreement between the **Product Owner** (PO) and the **Development Team** that specifies the criteria a User Story must meet *before* it can be pulled into a Sprint during **Sprint Planning**.

Its primary purpose is to ensure that work entering the Sprint is clear, feasible, and ready for immediate execution, thereby preventing delays caused by ambiguity or incomplete requirements.

Characteristic	Description	BABOK V3 Alignment
Criteria	DOR often aligns with the INVEST principle (Independent, Negotiable, Valuable, Estimable, Small, Testable) or the DEEP principle discussed under Backlog Grooming.	
Responsibility	The Business Analyst (BA) and Product Owner ensure all criteria are met during Backlog Refinement.	Requirements Analysis and Design Definition
Goal	To protect the Sprint Goal by preventing unclear or impossible	Risk Management (minimizing execution risk)

	work from being committed to.	

Example DOR for "Scrum Foods"

A User Story is "Ready" only if:

1. **Acceptance Criteria** are clearly defined and approved by the PO.
2. The User Story has a **Complexity Point (CP)** estimate from the Development Team.
3. All known **dependencies** (e.g., API keys, design mockups) are resolved or available.
4. The story is small enough to be completed within **one Sprint**.

2. Definition of Done (DOD)

The **Definition of Done (DOD)** is a formal list of criteria that a potentially shippable increment of product functionality must satisfy *at the end* of a Sprint to be considered complete and ready for release.

Its primary purpose is to establish a **shared understanding of quality** and ensure that no unfinished work (e.g., incomplete testing or documentation) is passed to the customer.

Characteristic	Description	BABOK V3 Alignment
Scope	Applies to all Product Backlog Items, including User Stories, bugs, and technical tasks.	
Quality Gate	Work that meets the DOD is considered ready for the Sprint Review and potentially shippable. Work that fails to meet the DOD cannot be demonstrated and must return to the Product Backlog.	Solution Evaluation

Responsibility	The Development Team is responsible for delivering the work to meet the DOD. The Product Owner relies on the DOD for final acceptance.	Underlying Competency: Quality Assurance

Example DOD for "Scrum Foods"

An increment is "Done" only if:

1. All code has been written and passed a **peer code review**.
2. All **Acceptance Criteria** for the story are met (PO's sign-off).
3. Unit, integration, and user interface (UI) **testing** are complete and passed.
4. The feature is deployed to the **staging/test environment**.
5. All relevant **documentation** (e.g., API docs) is updated.
6. The work is **error-free** and ready for the Sprint Review.

The DOD creates **transparency** around the work performed and establishes a baseline for future increments.

Question 15: Explain Prioritization Techniques and MVP.

1. Prioritization Techniques:

Prioritization Techniques are structured methods used by Business Analysts and Product Owners to rank requirements (User Stories, features, Epics) based on their relative importance and necessity. The goal is to ensure the team is always working on the highest-value items first, maximizing Return on Investment (ROI).

Technique	Description	Focus (Criteria)	BABOK Relevance

MoSCoW	Categorizes requirements into four levels of importance for a specific release: Must have (critical), Should have (important, but not vital), Could have (nice to have), and Won't have (deferred).	Time/Necessity	Quickly managing scope and release planning.
Weighted Shortest Job First (WSJF)	Ranks features by dividing the total expected Business Value (Cost of Delay) by the Job Size (Complexity Points/Effort).	Value vs. Cost	Maximizing economic efficiency and flow.
Kano Analysis	Categorizes features based on how they satisfy customers: Basic (must be present), Performance (more is better), and Excitement (delighters).	Customer Satisfaction	Understanding emotional impact and competitive differentiation.
Budget/Time Box	A pragmatic approach where prioritization is constrained by a fixed budget or a fixed timeline. Requirements are cut until the remaining scope fits the box.	Constraint Management	Managing project constraints and scope against deadlines.

For the "Scrum Foods" project, the combination of **Business Value (BV)** and **Complexity Points (CP)** that we used earlier is a core element of the **WSJF** technique, driving prioritization decisions (High BV/Low CP⇒ High Priority).

2. Minimum Viable Product (MVP):

The **Minimum Viable Product (MVP)** is the smallest possible version of a new product or feature set that delivers core value to customers, allowing the team to collect validated learning with the least amount of effort.

Characteristic	**Description**	**BABOK Relevance**

Goal	Learning and Validation. The primary objective is not profit, but to validate key business hypotheses (e.g., "Will users pay for real-time tracking?").	Strategy Analysis
Focus	Contains only the absolute "Must Have" requirements, as identified by MoSCoW, that complete a single, full customer journey.	Requirements Analysis
Iterative Start	The MVP is the first iteration of the solution. Subsequent iterations add more features to create the MMP (Minimum Marketable Product) and eventually the full product.	Solution Evaluation

MVP Example for "Scrum Foods"

For the Scrum Foods project, the MVP might be defined as the features necessary to complete the simplest transaction:

- **Functionality:**
    1. Customers can **log in/register** (Core).
    2. Customers can **browse a fixed list** of nearby restaurants (C-1).
    3. Customers can **add an item** to the cart and **place an order** (C-12).
    4. The only available payment method is **Cash on Delivery** (C-11 simplified).
    5. The system sends a **basic confirmation alert** to the restaurant (R-21 simplified).

The key features we would **omit** from the MVP (even if they have high value) to keep it minimal are: Real-Time Tracking (C-8), Discount Coupons (C-5), Multiple Payment Gateways (C-11 full), and complex historical reporting. These features would be added in post-MVP iterations.

Question 16: Difference between Business Analyst & Product Owner.

The difference lies primarily in the **Decision Rights** and **Focus Area**. The PO defines *what* the value is and *what* to build, while the BA helps discover, elaborate, and manage the requirements needed to achieve that value.

Feature	Product Owner (PO)	Business Analyst (BA)

Accountability	Value & Vision. Accountable for maximizing the value of the product and managing the Product Backlog.	Requirements & Detail. Accountable for the quality and clarity of the requirements (User Stories) and the design of the solution.
Decision Rights	Decides on the priority (Business Value), scope, and final acceptance of the product increment.	Facilitates the discovery and recommends the best solution options; does not have final decision authority over what is built.
Focus	Strategic & External. Focuses on the Market, Customer, ROI, and the 'Why' of the solution.	Tactical & Internal. Focuses on the Details, Acceptance Criteria, Process Modeling, and the 'How' (feasibility) of the requirements.
Artifact Ownership	Product Backlog (The Master List).	Models, Documentation, Acceptance Criteria (The Details).
Key Activity	Prioritization and Stakeholder Management.	Elicitation, Analysis, and Backlog Refinement (Grooming).

Overlap and Collaboration:

In many organizations, especially smaller ones or where the PO role is part-time, the BA often *assists* or *acts as a proxy for* the PO in non-decision-making tasks. This typically includes:

- **Eliciting** detailed requirements from stakeholders.
- **Refining** large Epics into detailed User Stories.
- **Writing** Acceptance Criteria and supporting documents.
- **Modeling** the business process or data flows.

The PO uses the BA's refined requirements to confidently make the **final prioritization decisions**.

Question 17: Prepare a sample Resume of a Product Owner with 3 years of experience.

Sample Resume: Product Owner (3 Years Experience)

Contact Information

- **[Your Name]**
- **[Your Phone Number]**

- **[Your Email Address]**
- **[Your LinkedIn Profile URL]**

Professional Summary

- Goal-oriented Product Owner with **3 years of experience** defining product vision, maximizing value, and leading Scrum teams in fast-paced software development environments.
- Proven ability to translate complex business strategies into clear, actionable **Product Backlogs**.
- Expert in stakeholder management, driving feature prioritization using data (e.g., **WSJF, Business Value**), and guiding products from **MVP** to successful market launch.

Professional Experience

Product Owner, Tech Solutions Inc. (Focus: Food Delivery/Logistics)

[Start Date] – Present (3 Years)

- Owned the strategic vision and entire **Product Backlog** for a B2C food delivery platform (**Scrum Foods** project context).
- Maximized product value by achieving a **20% increase in customer engagement** through the prioritized launch of the **Real-Time Order Tracking** feature.
- Managed and groomed a backlog of over **100 User Stories** and Epics, ensuring a constant supply of **"Ready"** work for a dedicated 5-person Scrum team.
- Decreased development risk by leading **Backlog Refinement sessions**, personally writing all **Acceptance Criteria**, and defining the **Definition of Ready (DOR)**.

- Successfully defined and launched the **Minimum Viable Product (MVP)**, focusing on core ordering and cash-only transactions, leading to the initial 10% market penetration.
- Utilized **Product Burndown Charts** to effectively manage **scope creep** and provide accurate, data-driven forecasting to executive stakeholders, maintaining a predictable release schedule.

Skills & Competencies

- **Agile Expertise:** Scrum, Kanban, Daily Scrum, Sprint Planning, Retrospective, Scaling Agile.
- **Product Management:** Product Vision, Roadmap, Market Analysis, Competitive Analysis, Release Planning, MVP Definition.
- **Prioritization Techniques:** WSJF, MoSCoW, Business Value, Risk Assessment.
- **Tools:** JIRA, Confluence, Microsoft Azure DevOps (ADO), MS Excel, Tableau (for burndown analysis).
- **Requirements:** User Stories, Acceptance Criteria, Process Modeling (UML/BPMN), Definition of Done (DOD).

Education & Certifications

- **Certified Scrum Product Owner (CSPO)** or **Professional Scrum Product Owner (PSPO I)**
- **[Degree Name]** (e.g., Bachelor of Science in Business Administration) - [University Name]

Overall Roles and Responsibilities

- **Defining the Product Vision and Strategy:** Acting as the voice of the customer and setting the long-term direction for the product.
- **Maximizing Product Value:** Ensuring the Development Team is always working on the requirements that deliver the greatest return on investment (ROI).
- **Product Backlog Ownership:** Solely accountable for the content, availability, transparency, and ordering of the Product Backlog.
- **Prioritizing Requirements:** Using techniques like WSJF or MoSCoW to sequence work based on value, effort, dependencies, and risk.
- **Stakeholder Communication:** Managing expectations and collaborating with all stakeholders (customers, executives, sales, marketing) regarding product progress and scope.
- **Accepting the Increment:** Inspecting the work at the **Sprint Review** and having the authority to formally accept or reject the completed increment against the **Acceptance Criteria**.
- **Defining the MVP:** Determining the minimal set of features required to launch the first usable version of the product for validated learning.