**Online Agriculture Products Store**

**Decode the case study**

1. **Project Idea**: - To Develop an online agriculture product store to facilitate remote area farmers to buy agricultural products.
2. **Current Needs: -** Build online platform user friendly, easy accessible and should be able to accept products (seeds, fertilizers, pesticides etc. ) from manufacturers. Farmers will browse the products according to them, what they need, and request to buy and deliver to farmer’s location.
3. **Overview of the Project: -** It is an online agriculture platform, which will build from scratch where farmers can buy products from companies through online.
4. **Current Problems: -** Farmers are not able to buy fertilizers, seeds, pesticides etc due to remote area.

* **Question 1 – Write Agile Manifesto**

**Definition** :-

Agile is an iterative software development approach that delivers value to users in small increments, rather than through a single large launch. Agile teams continuously evaluate requirements and results, which enables them to implement change efficiently

**History**

The history of Agile can be traced back to the 1970s, when industries like aerospace, defense, and computer needed more flexible project management methods. These industries were creating products that were either obsolete or unneeded, or they were abandoning projects mid-cycle.

The Agile Manifesto was created in 2001 by a group of 17 software developers who wanted to document their shared beliefs about how to develop software. The manifesto was signed at a meeting in Snowbird, Utah, and included the following key values: Collaboration over documentation, Self-organization over rigid management, Working software over comprehensive documentation, Customer collaboration over contract negotiation, and Responding to change instead of following a plan

**Agile**

* Agile is light weight and can be implemented where faster delivery is required.
* Agile no documentation is required.
* Customer retention since no documentation.
* Agile is Customer focus and  Prioritizing customer needs and satisfaction

**Agile 4 values :-**

These values are derived from the Agile Manifesto, which was created in 2001 at the Snowbird ski resort in Utah. The Agile methodology is based on questioning traditional processes and focusing on what drives value for the business.

* **Individuals and interactions over processes and tools**
* **Working software over comprehensive documentation**
* **Customer collaboration over contract negotiation**
* **Responding to change over following a plan**

**Some benefits of using Agile values include:**

* **Flexibility**: Agile values promote flexibility for team members and the ability to adjust to change.
* **Regular analysis**: Regular analysis helps identify problematic areas and improve the process.
* **Customer feedback**: Listening to and acting on customer feedback helps ensure customer satisfaction and trust.

**The following 12 Principles are based on the**[**Agile Manifesto**](https://www.agilealliance.org/agile101/the-agile-manifesto/)**.**

* Our highest priority is to satisfy the customer through the early and continuous delivery of valuable software.
* Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.
* Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
* Business people and developers must work together daily throughout the project.
* Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done
* The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
* Working software is the primary measure of progress.
* Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
* Continuous attention to technical excellence and good design enhances agility
* Simplicity–the art of maximizing the amount of work not done–is essential.
* The best architectures, requirements, and designs emerge from self-organizing teams.
* At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.
* **Question 2 – User Stories- Acceptance Criteria-BV-CP**

|  |  |  |
| --- | --- | --- |
| User Story 1 | Tasks 2  | Priority : Highest |
| AS A DELIVERY BOY,I WANT TO REGISTER IN SCRUM FOODS SO THAT I CAN DELIVER ORDERS  |
| BV:500 | CP:02 |
| ACCEPTANCE CRITERIA |   |   |
| Registration Screen, Text Boxes for User Name, Password, Nation ID, Mobile No, Email, Address, Phone Number. Click on Register Button. Send Successful Notification to the user |

|  |  |  |
| --- | --- | --- |
| User Story 2 | Tasks 2  | Priority : Highest |
| AS A RESTARURANT OWNER, I WANT TO VIEW ORDERSSO THAT I CAN VIEW THE LIST OF ORDERS  |
| BV:500 | CP:02 |
| ACCEPTANCE CRITERIA |
| View Order, Display List of orders in the tabular Form |

|  |  |  |
| --- | --- | --- |
| User Story 3 | Tasks 2  | Priority : Highest |
| AS A CUSTOMER, I WANT TO ADD ADDRESSSO THAT I CAN GET THE ORDERS TO MY ADRESSE  |
| BV:500 | CP:02 |
| ACCEPTANCE CRITERIA |
| TEXT BOX TO ENTERBUSINESS RULES : RADIUS WITHIN 5 KM. |

|  |  |  |
| --- | --- | --- |
| User Story 4 | Tasks 2  | Priority : Highest |
| AS A CUSTOMER, I WANT TO SELECT THE PAYMENT MODESO THAT I CAN MAKE THE PAYMENT OF MY CHOICE  |
| BV:500 | CP:02 |
| ACCEPTANCE CRITERIA |
| Display payment modes, radio buttons to select payment modes, payments button.Business Rule. Can select only one payment mode |

|  |  |  |
| --- | --- | --- |
| User Story 5 | Tasks 2  | Priority : Highest |
| AS AN ADMINI WANT TO VIEW THE RESTARANTSSO THAT I CAN APPROVE THEIR REGISTRATION  |
| BV:500 | CP:02 |
| ACCEPTANCE CRITERIA |
| RESGISTRATION THE THE PORTAL WITH DETAILS |
| User Story 6 | Tasks 2  | Priority : Highest |
| AS A CUSTOMERI WANT TO VIEW THE PRICESO THAT I CAN ORDER THE FOOD |
| BV:50 | CP:01 |
| ACCEPTANCE CRITERIA |
| DISPLAY PRICE IN THE LIST OF MENU ITEMS |

|  |  |  |
| --- | --- | --- |
| User Story 7 | Tasks 2  | Priority : Highest |
| AS A CUSTOMERI WANT TO VIEW THE CONTACT NUMBER OF DELIVERY BOYSO THAT I CAN CONTACT THE DELIVERY BOY FOR THE STATUS  |
| BV:50 | CP:01 |
| ACCEPTANCE CRITERIA |
| 1. Display delivery boy mobile number 2. Display delivery boy name in tracking field3. Display delivery boy picture |

|  |  |  |
| --- | --- | --- |
| User Story 8 | Tasks 2  | Priority : MEDIUM |
| AS A RESTAURANT OWNER I WANT TO PROVIDE TIME SLOTS SO THAT CUSTOMER CAN CHECK OPENING AND CLOSING HOURS |
| BV:50 | CP:01 |
| ACCEPTANCE CRITERIA |
| 1. . Click on restaurant dashboard2. Add from time to time3. Click on submit4. Display updated successfully |

|  |  |  |
| --- | --- | --- |
| User Story 9 | Tasks 2  | Priority : Highest |
| AS A Business OWNERI WANT TO VIEW RESTAURANT REVENUE REPORTSO THAT I CAN VIEW THE RESTAURANT’SREVENUE |
| BV:200 | CP:03 |
| ACCEPTANCE CRITERIA |
| Select ReportsSelect Revenue ReportsSelect to and from dateSelect Region (can select all)Generate ReportDownload Report in EXCEL |

|  |  |  |
| --- | --- | --- |
| User Story 10 | Tasks 2  | Priority : Highest |
| AS A REG ADMINI WANT TO MANAGE REGIONAL RESTAURANTSSO THAT, I CAN TRACK THE PERFORMANCE OF REGIONAL RESTAURANTS |
| BV:200 | CP:03 |
| ACCEPTANCE CRITERIA |
| CLICK ON PERFORMANCE OF RESTAURANTSSELECT FROM DATE TO DATECLICK ON GENERATE REPORT WHICH INCLUDE RESTAURANTS ID, NAME, REVENUECLICK ON DOWNLOAD REPORT SHOULD BE IN EXCEL |

|  |  |  |
| --- | --- | --- |
| User Story 11 | Tasks 2  | Priority : MEDIUM |
| AS AN ADMIN I WANT TO SEE THE REGIONAL REVENUE REPORTS,SO THAT I CAN VIEW THE REGIONALPERFORMANCE |
| BV:100 | CP:03 |
| ACCEPTANCE CRITERIA |
| Select regional drop downView performance of each rest of that region in tabular form which includes rest name, revenue, generatedDownload in excel or PD |

|  |  |  |
| --- | --- | --- |
| User Story 12 | Tasks 2  | Priority : Highest |
| AS A CUSTOMERI WANT TO CHAT WITH REG ADMINSO THAT I CAN REQUEST FOR REFUND |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1)BR-ALL MANDATORY2)TEXT BOX FIELDS3)DISPLAY ORDER ID4)TEXT BOX,FOR DESCRIPTION5)SUBMIT BUTTON6)GENERATE ISSUE ID7)DISPLAY SUCCESSFUL |

|  |  |  |
| --- | --- | --- |
| User Story 13 | Tasks 2  | Priority : Highest |
| AS A HUNGRY USERI WANT TO BROWSE NEARBY RESTAURANTSSO THAT I CAN ORDER THE FOOD |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1Each restaurant entry displays its name, cuisine type, and rating 2)This list can be sorted by distance or rating |

|  |  |  |
| --- | --- | --- |
| User Story 14 | Tasks 2  | Priority : Highest |
| AS A CUSTOMERI WANT TO BROWSE DIFFERENT RESTAURANTS ANDMENUSSO THAT I CAN FIND A PLACE TO ORDER FOOD |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1)The menu includes dishes , prices and descriptions2) Show the restaurant is open or closed |

|  |  |  |
| --- | --- | --- |
| User Story 15 | Tasks 2  | Priority : Highest |
| AS A CUSTOMERI WANT TO BROWSE FOR SPECIFIC DISHES ANDCUISINESSO THAT I CAN FIND A PLACE TO ORDER FOOD |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1)App displays relevant restaurant and dishes matching the query |

|  |  |  |
| --- | --- | --- |
| User Story 16 | Tasks 1  | Priority: Highest |
| AS A CUSTOMER I WANT TO FILTER RESTAURANTSSO THAT I CAN FIND A PLACE TO ORDER FOOD |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1)Filter restaurants by cuisine type and dietary options (vegan, veg,non veg,egg)  |

|  |  |  |
| --- | --- | --- |
| User Story 17 | Tasks 1  | Priority: Highest |
| AS A CUSTOMERI WANT TO TRACK MY ORDERSO THAT I KNOW THE TIME OF DELIVERY |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1)App shows real time update on the order status 2)Display estimated delivery time  |

|  |  |  |
| --- | --- | --- |
| User Story 18 | Tasks 1  | Priority: Highest |
| AS A USERI WANT TO RATE AND REVIEW RESTAURANTSSO THAT I CAN RATE AND REVIEW THERESTAURANTS I HAVE VISITED |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
|  1)Can see reviews from other users to help me make dining decision |

|  |  |  |
| --- | --- | --- |
| User Story 19 | Tasks 1  | Priority: Highest |
| AS A USERI WANT TO VIEW PAST ORDER HISTORYSO THAT I CAN ORDER AGAIN |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
|  1)Can see the details such as order items, total cost and order date |

|  |  |  |
| --- | --- | --- |
| User Story 20 | Tasks 3  | Priority : Highest |
| AS A USERI WANT TO RECEIVE NOTIFICATIONSSO THAT I CAN RECEIVE UPDATES |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1Notifications for order confirmation2)Notification for dispatch3)Notification for delivery |

|  |  |  |
| --- | --- | --- |
| User Story 21 | Tasks 1 | Priority : Medium |
| AS A CUSTOMERI WANT TO CONTACT CUSTOMER CARESO THAT I CAN SUBMIT QUERIES OR ISSUES |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1.Customer support section with contact information |

|  |  |  |
| --- | --- | --- |
| User Story 22 | Tasks 2 | Priority : high |
| AS A RESTAURANT OWNERI WANT TO RECEIVE AND MANAGE ORDERSSO THAT I CAN UPDATE ORDERS |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1.MANAGE ORDER STATUS2.NOTIFY RESTAURANTS ABOUT INCOMING ORDERS |

|  |  |  |
| --- | --- | --- |
| User Story 23 | Tasks 2 | Priority : HIGH |
| AS A RESTAURANT OWNERI WANT TO ACCESS TO CUSTOMERS REVIEWSSO THAT I CAN VIEW AND RESPOND TO CUSTOMER REVIEWS |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1.OWNERS CAN ADDRESSE FEEDBACK2.OWNERS CAN IMPROVE THEIR SERVICES |

|  |  |  |
| --- | --- | --- |
| User Story 24 | Tasks 2 | Priority : MEDIUM |
| AS A CUSTOMERI WANT TO APPLY PROMO CODES AND DISCOUNTSSO THAT I CAN ORDER AT LOWER PRICE |
| BV: 100 | CP: 04 |
| ACCEPTANCE CRITERIA |
| 1.ACTIVE PROMOCODES |

|  |  |  |
| --- | --- | --- |
| User Story 25 | Tasks 2 | Priority : MEDIUM |
| AS A CUSTOMERI WANT TO APPLY PROMO CODES AND DISCOUNTSSO THAT I CAN ORDER AT LOWER PRICE |
| BV: 100 | CP: 04 |
| ACCEPTANCE CRITERIA |
| 1.ACTIVE PROMOCODES |

|  |  |  |
| --- | --- | --- |
| User Story 26 | Tasks 2 | Priority : HIGH |
| AS A DELIVERY BOYI WANT TO VIEW THE ORDERSSO THAT I ACCEPT THE ORDERV |
| BV: 200 | CP: 04 |
| ACCEPTANCE CRITERIA |
| 1.Order visibility2)Real-time updates3)Order details4)Order filtering and sorting5)Order map view6)Order navigation7)Order completion and confirmation |

|  |  |  |
| --- | --- | --- |
| User Story 27 | Tasks 2 | Priority : HIGH |
| AS A DELIVERY BOYI WANT TO LOGINSO THAT I CAN ACCEPT THE ORDER |
| BV: 200 | CP: 04 |
| ACCEPTANCE CRITERIA |
| 1User Authentication2)Error Handling3)Password security4)Multi-factor Authentication5)Compatibility and Usability |

|  |  |  |
| --- | --- | --- |
| User Story 28 | Tasks 5 | Priority : MEDIUM |
| AS A DELIVERY BOYI WANT TO VIEW FEEDBACKSO THAT I CAN KNOW THE CUSTOMER FEEDBACK |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1.Access to feed back system2)Feedback Visibility3)Feedback sorting and filtering4)Response Mechanism5)User Support |

|  |  |  |
| --- | --- | --- |
| User Story 29 | Tasks 5 | Priority : MEDIUM |
| AS AN ADMINI WANT TO VIEW FEEDBACKSO THAT I CAN KNOW THE CUSTOMER FEEDBACK |
| BV:200 | CP:02 |
| ACCEPTANCE CRITERIA |
| 1.Access to feed back system2)Feedback Visibility3)Feedback sorting and filtering4)Response Mechanism5)User Support |

|  |  |  |
| --- | --- | --- |
| User Story 30 | Tasks 5 | Priority : MEDIUM |
| AS AN RESTAURANT OWNERI WANT TO VIEW FEEDBACKSO THAT I CAN KNOW THE CUSTOMER FEEDBACK |
| BV:200 | CP:04 |
| ACCEPTANCE CRITERIA |
| 1.Access to feed back system2)Feedback Visibility3)Feedback sorting and filtering4)Response Mechanism5)User Support |

|  |  |  |
| --- | --- | --- |
| User Story 31 | Tasks 5 | Priority : MEDIUM |
| AS AN ADMINI WANT TO KNOW THE ISSUESO THAT I CAN RESLOVE THEM |
| BV:100 | CP:03 |
| ACCEPTANCE CRITERIA |
| 1.Display issue section2)Sorting and filtering of issues list3)Editing and modifying the issues  |

|  |  |  |
| --- | --- | --- |
| User Story 32 | Tasks 5 | Priority : HIGH |
| AS AN REGIONAL ADMINI WANT TO KNOW THE ISSUESO THAT I CAN RESLOVE THEM |
| BV:100 | CP:03 |
| ACCEPTANCE CRITERIA |
| 1.Display issue section2)Sorting and filtering of issues list3)Editing and modifying the issues  |

|  |  |  |
| --- | --- | --- |
| User Story 32 | Tasks 3 | Priority : HIGH |
| AS AN REGIONAL ADMINI WANT TO KNOW THE ISSUESO THAT I CAN RESLOVE THEM |
| BV:100 | CP:03 |
| ACCEPTANCE CRITERIA |
| 1.Display issue section2)Sorting and filtering of issues list3)Editing and modifying the issues  |

|  |  |  |
| --- | --- | --- |
| User Story 33 | Tasks 6 | Priority : HIGHEST |
| AS A RESTAURANT OWNERI WANT TO VIEW REVENUE GENERATEDSO THAT I VIEW RESTAURANTS REVENUE |
| BV:200 | CP:03 |
| ACCEPTANCE CRITERIA |
| Select Reports Select Revenue ReportsSelect to and from dateSelect Region (can select all)Generate Report Download Report in EXCEL |

|  |  |  |
| --- | --- | --- |
| User Story 34 | Tasks 2 | Priority : HIGH |
| AS A RESTAURANT OWNERI WANT TO KNOW DELIVERY BOYSO THAT I VERIFY THE DELIVERY BOY |
| BV:200 | CP:04 |
| ACCEPTANCE CRITERIA |
| 1.ID PROOF2)PUNCTUALITY3)RELIABLITY |

|  |  |  |
| --- | --- | --- |
| User Story 35 | Tasks 2 | Priority : LOW |
| AS A CUSTOMERI WANT TO VIEW THE CONTACT NUMBER OFDELIVERY BOYSO THAT I CAN CONTACT DELIVERY BOY FOR THESTATUS |
| BV:50 | CP:1 |
| ACCEPTANCE CRITERIA |
| 1.Display delivery boy mobile number 2. Display delivery boy name in tracking field3. Display delivery boy picture |

|  |  |  |
| --- | --- | --- |
| User Story 36 | Tasks 2 | Priority : MEDIUM |
| AS A RESTAURANT OWNERI WANT TO PROVIDE TIME SLOTSSO THAT CUSTOMER CAN CHECK OPENING ANDCLOSING HOURSV:  |
| BV:100 | CP: 02 |
| ACCEPTANCE CRITERIA |
| 1.Click on restaurant dashboard2. Add from time to time3. Click on submit4. Display updated successfully |

|  |  |  |
| --- | --- | --- |
| User Story 37 | Tasks 3 | Priority : HIGH |
| AS A USERI WANT TO RECEIVE NOTIFICATIONSSO THAT I CAN RECEIVE UPDATES:  |
| BV:200 | CP: 02 |
| ACCEPTANCE CRITERIA |
| 1.Notifications for order confirmation2)Notification for dispatch3)Notification for delivery |

|  |  |  |
| --- | --- | --- |
| User Story 38 | Tasks 1 | Priority : MEDIUM |
| AS A CUSTOMERI WANT TO CONTACT CUSTOMER SUPPORTSO THAT I CAN SUBMIT QUERIES OR ISSUES:  |
| BV:200 | CP: 02 |
| ACCEPTANCE CRITERIA |
| CUSTOMER SUPPORT SECTION WITH CONTACT INFORMATION |

|  |  |  |
| --- | --- | --- |
| User Story 39 | Tasks 4 | Priority : MEDIUM |
| AS A CUSTOMERI WANT TO VIEW THE ORDERSO THAT I CAN CANCEL IT |
| BV:100 | CP: 03 |
| ACCEPTANCE CRITERIA |
| Order statusMethod of cancellationRefund policyTime frame |

|  |  |  |
| --- | --- | --- |
| User Story 40 | Tasks 4 | Priority : HIGH |
| AS A REGIONAL ADMINI WANT TO TRACK THE DELIVERYSO THAT I CAN VIEW THE STATUS OF THE DELIVERY |
| BV:100 | CP: 03 |
| ACCEPTANCE CRITERIA |
| Real time trackingSecurity and data privacyUser friendly Interface |

* **Question 3– What is epic? Write 2 epics.**

In Agile project management, an epic is a large body of work that's broken down into smaller user stories or tasks

**Definition**

An epic is a large user story that can't be delivered in a single iteration, or is large enough to be split into smaller user stories

 **Purpose**

Epics are used to plan ahead and organize work over multiple sprints. They can represent a large project or major part of a product.

**Flexibility**

Epics are flexible and can be adjusted based on customer feedback and team cadence. User stories can be added or removed as needed

**Benefits**

Epics can help create structure and hierarchy in product goals. They can also help visualize and categorize processes

**Themes**

Epics can work across different teams and projects, but they're usually united under a broad banner label called a theme.

**EPIC**

1. **Rating and Reviews**

As a user,I want to view ratings and reviews for restaurants on scrum foods, so that I can make informed decisions about where to order food from.

As a user, I want to provide ratings and reviews for restaurants on scrum foods, so that Ican share my experiences with other users and contribute to the community

**Acceptance Criteria**

●Users can view average ratings and reviews for each restaurant on the restaurants details page

● Users can read detailed reviews and comments left by the other customers

● Users can sort and filter reviews based on criteria such as rating and relevance

● Users can rate the restaurants and leave a review after placing the order

● User can edit or delete their own reviews within a specified timeframe

● Reviews are displayed in a way that provides helpful insights to other users

● The rating and review system maintains the integrity and authenticity of user feedback

 2. **Real-Time Order Tracking for Food Delivery App**

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

User Stories:  **As a customer, I want to see the live status of my order.**

● Display the current status of the order, such as "order confirmed,""preparing," "out for delivery," and "delivered."

● Provide real-time updates as the order progresses through various stages. As a customer, I want to track the location of my delivery.

● Integrate GPS or location services to show the delivery partner's real-timelocation on a map.

Allow customers to view the estimated time of arrival (ETA) based on the delivery partner's location. As a customer, I want to receive notifications for order updates.

● Send push notifications or SMS updates to inform customers about order confirmation, preparation, and delivery status changes.

● Provide delivery partner details, including name, contact information, and a profile picture. As a customer, I want to contact the delivery partner directly.

● Enable in-app chat or call functionality to allow customers to communicate with the assigned delivery partner.

● Ensure privacy by using masked phone numbers or secure messagingchannels. As a customer, I want to view the delivery route.

● Display the delivery route on the map, showing the path the delivery partner will take to reach the destination.

● Allow customers to track the progress of the delivery in real-time along theroute. As a customer, I want to provide feedback on the delivery experience.

● Allow customers to rate the delivery partner and overall delivery experience after the order is delivered.

● Implement a feedback system with written comments to gather valuableinsights. As a customer, I want to see estimated delivery time adjustments.

● Account for real-time traffic conditions and other factors that may affect the delivery time.

● Update the estimated delivery time accordingly and inform the customer promptly. As a customer, I want to have a seamless tracking experience across platforms.

● Ensure the real-time order tracking feature is available and consistent on all supported platforms (e.g., mobile app, web). As an admin, I want to monitor order tracking performance.

● Provide analytics and reporting on order tracking metrics, such as average delivery time and customer satisfaction ratings.

● Use data to identify areas for improvement and optimize the delivery process.

**Acceptance Criteria**:

**Real-Time Order Updates**:

● The app should provide real-time updates on the status of the user's order, such as "Order received," "Preparing," "Out for delivery," and "Delivered."

**Order Location Tracking**:

● The app should display the live location of the delivery driver while en route to the user's address.

● The map should update at regular intervals to reflect the driver's movement accurately.

**Estimated Delivery Time**:

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

**Delivery Notifications**:

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

**Map Zoom and Interaction**:

● Users should be able to zoom in and out on the map to view the delivery driver's route more closely.

● The map should support standard interactions, such as panning and rotating, to improve the user experience.

**Delivery Status History**:

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

**Accuracy and Reliability:**

● The real-time tracking information should be accurate and reliable, providing users with the most up-to-date data available.

● The system should handle location updates efficiently, minimizing delays or inaccuracies

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

**Compatibility**:

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

**Opt-Out Option**:

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

**User Education**:

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

**Support for Multiple Orders**:

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

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

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

**In Agile, "business value**" refers to the tangible benefit a feature or task brings to the company, like increased revenue or improved customer experience, while "**complexity points"** represent the level of difficulty or effort involved in implementing that feature, considering factors like technical challenges and integration complexities; essentially, business value measures the "worth" of a task while complexity points measure how hard it will be to complete it

**BV – Business Value**

This is not the cost of Development or the complexity of the feature. Business Value is how important is this feature (user Story) to the Business. This is estimated by Scrum Currency Notes. We provide Rs 1000. Rs 500, Rs 100, Rs 50, Rs 20 and Rs 10 Denominations. These estimations are done by the Stakeholders (Clients). If different values are selected by the stakeholders, then discussions will happen, and they agree to one BV value to that user story.

Examples of Business Value considerations: Increased revenue, improved user experience, compliance with industry regulations, competitive advantage.

**Techniques used: Moscow, Currency notes technique**

**MoSCoW**: The MoSCoW technique is a prioritization and requirements management technique used in project management and software development. It helps stakeholders and project teams prioritize and categorize requirements or tasks based on their importance and urgency. The acronym "MoSCoW" stands for the four categories into which requirements or tasks are typically classified

 **Must Have**:

 These are critical requirements or tasks that are essential for the project's success. Without these, the project would likely fail to meet its Objectives.

**Should Have:**

 These are important requirements or tasks that are not absolutely critical but significantly contribute to the project's value. They are high-priority items that should be included if possible.

 **Could Have:**

 These are desirable requirements or tasks that would enhance the project but are not essential. They are often considered nice-to-have features that can be addressed if time and resources allow.

 **Won't Have** (this time):

 These are requirements or tasks that are deliberately deprioritized and will not be included in the current phase of the project. They may be considered for future iterations or versions.

The MoSCoW technique helps project teams and stakeholders make informed decisions about what should be included in a project based on the available resources, constraints, and objectives. It facilitates open communication between stakeholders,ensures that the most critical aspects are addressed first, and allows for flexibility in project scope as priorities change. Here's an example of how the MoSCoW technique might be used in a software development project:

 Must Have:

 User authentication, basic user profile management, core functionality of the application.

 **Should Have**:

 Advanced user settings, integration with third-party services,improved user interface design.

**Must Have**:

 User authentication, basic user profile management, core functionality of the application.

 Should Have:

 Advanced user settings, integration with third-party services,improved user interface design.

**Should Have**:

 Advanced user settings, integration with third-party services,improved user interface design.

**Could Have**:

 Additional language support, social media sharing features, enhanced search functionality.

 **Won't Have** (this time):

 Gamification features, integration with legacy systems,complex data visualization.By categorizing requirements or tasks using the MoSCoW technique, project teams canfocus on delivering the most critical and valuable components of a project while providing a clear framework for managing priorities and expectations.

**Key points about the difference**

**Focus:**

Business value focuses on the positive impact on the business, whereas complexity points focus on the technical challenges and effort required to develop a feature

**Measurement:**

Business value is often measured qualitatively based on factors like customer satisfaction, market share, or cost savings, while complexity points are usually assigned a numerical value using a relative scale (like story points) to reflect the development effort.

**Prioritization Tool:**

In Agile, teams often use a "value vs. complexity" matrix to prioritize tasks by considering both the potential business value and the level of complexity involved, allowing them to focus on features that deliver high value while being relatively manageable to implement

Example:

 **High Business Value, Low Complexity:**

Implementing a feature that significantly improves user experience on a popular product page but requires minimal development effort.

**High Business Value, High Complexity:**

Developing a new integration with a complex third-party system that could greatly expand customer reach but requires significant technical expertise.

**CP – Complexity Points**

CP is also known as Story Points (SP). CP is the effort required by the Scrum Developers to develop this feature (user story) using technology. Efforts include time taken to solve the complexity and write the code. CP is estimated by the Scrum Developers by using Poker cards. We provide pokers with values “?”, 1, 2, 3, 5, 8, 13, 20, 40, 100 and BIG. If the entire Project development takes 200 points, then this user story coding effort will be… how many points? … Thinking in this way, Scrum Developers will give CP to the User story. ). If different values are selected by the Scrum Developers, then discussions will happen, and they agree to one CP value to that user story

**Techniques used:** **Planning poker**

 :Planning Poker is a consensus-based technique commonly used in Agile and Scrum methodologies for estimating the effort or complexity of user stories or tasks. It help steams collaborate and arrive at a shared understanding of the work involved in a particular item. The term "CP" in your question likely stands for "Complexity Points,"which are used in Planning Poker to represent the relative effort or complexity of a task.Here's how the Planning Poker technique works:

**Preparation:**

● The team gathers to estimate the effort or complexity of user stories or tasks that need to be completed in a sprint or iteration.

**Estimation Cards:**

**Preparation:**

● The team gathers to estimate the effort or complexity of user stories or The team gathers to estimate the effort or complexity of user stories or tasks that need to be completed in a sprint or iteration.

**Estimation Cards:**

**Estimation Cards** Estimation cards are used, each containing a number representing a predefined range of Complexity Points (e.g., 0, 1, 2, 3, 5, 8, 13, 20, 40,100).The cards are often designed such that the numbers increase non-linearly, reflecting that as tasks become more complex, the uncertainty in estimates also increases.

**Facilitator:**

● A facilitator (often the Scrum Master or Agile Coach) leads the session and explains the process to the team.

**Estimation Process:**

● For each user story or task, the team discusses the requirements, assumptions, and any relevant information. Each team member selects an estimation card representing their individual view of the Complexity Points required to complete the task. The card is kept face down.

**Reveal and Discussion:**

● After everyone has selected a card, all team members reveal their chosen cards simultaneously. If there's a wide variation in estimates, team members have a discussion to share their reasoning and insights. This helps clarify assumptions and leads to a shared understanding.

Re-Estimation and Consensus:

● After the discussion, team members may revise their estimates by selecting a different card based on the insights gained. The process of discussion and re-estimation continues until a consensus is reached. The goal is to align the team's estimates.

Repeat for Each Task:

● The process is repeated for each user story or task in the backlog.

**Final Estimate:**

● The final estimate for each user story or task is often determined by the mode or median of the estimates chosen by the team members.

Planning Poker helps mitigate individual biases and provides a collaborative and transparent way to estimate work. It encourages discussions, helps identify potential challenges, and allows the team to make informed decisions about how much work can be taken on in a sprint or iteration In summary, Business Value focuses on the business impact and significance of tasks or features, while Complexity Points focus on the technical effort and complexity involved in implementing those tasks or features. Both concepts are valuable in project management and software development, as they help prioritize and plan work based on both business goals and technical constraints.

* **Question 5 –Explain about Sprint**

**SPRINT**:

In the context of software development and project management, a sprint is an a time-boxed, iterative development period during which a specific set of tasks and goals are worked on by a development team. Sprint is a core concept in Agile methodologies, such as Scrum, which emphasizes flexibility, collaboration, and delivering value to the customer in shorter cycles.

Also, Sprints are time-boxed iterations of a continuous project development cycle—short, repeatable phases that last between one and four weeks. Sprints lie at the core of Agile and Scrum methodologies, an approach that breaks down large, complex product development projects into smaller, more manageable pieces.

What is sprint Duration: 2 Weeks - Your sprint Value \_ Scrum is a sub unit of Sprint. What is scrum Duration: 1 day – Your scrum Value\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_?PBI: Product Backlog Item Task: Unit of Work done by 1 Developer in 1 Scrum WIP: Work In Progress Sprint Backlog

|  |  |  |  |
| --- | --- | --- | --- |
| **PIB** | **TASK** | **WIP** | **DONE** |
| Registration | 3 | 2 | 1 |
| Login | 2 | 2 | 0 |
| Add restaurants | 3 | 2 | 1 |
| Order foods | 3 | 0 | 3 |
| Make Payment | 3 | 3 | 0 |
| Track Delivery | 3 | 3 | 0 |
| Customer Feedback | 3 | 3 | 0 |

Here are the key characteristics and components of a sprint:

 **Time Frame:**

 A sprint typically has a fixed duration, often ranging from 1 to 4 weeks. The duration is consistent across all sprints to provide a predictable cadence for development and planning.

 **Goals and Objectives:**

 At the beginning of each sprint, the development team, along with stakeholders, selects a set of user stories, features, or tasks to work on during that sprint. These items are collectively referred to as the sprint backlog.

 **Planning:**

 During sprint planning, the development team breaks down the selected items from the product backlog into smaller tasks and estimates the effort required for each task. The team commits to completing these tasks within the sprint duration.

 **Daily Stand-ups:**

 Throughout the sprint, the team holds daily stand-up meetings(also known as daily scrums) to discuss progress, obstacles, and plans. Each team member shares what they've accomplished, what they're working on, and any challenges they're facing. These meetings foster communication and collaboration

**Development:**

 The development team works on the tasks identified in the sprint backlog. They collaborate closely, often using techniques like pair programming and frequent code reviews to ensure high-quality work.

 Continuous Integration:

 Developers integrate their code changes into the main codebase regularly, ensuring that the software remains functional and stable throughout the sprint.

 **Testing:**

 Testing is an integral part of a sprint. Automated tests are run to validate code changes, and manual testing may be conducted to ensure the quality of the software.

 Review and Demo:

 At the end of the sprint, the development team conducts sprint review and demo. They showcase the completed work to stakeholders, gathering feedback and validation. This helps ensure that the delivered features align with expectations.

 **Retrospective:**

 Following the review and demo, the team holds a sprint retrospective. They reflect on what went well during the sprint, what could be improved, and actions to take in the next sprint. The retrospective encourages continuous improvement

**Incremental Development**:

 Each sprint results in a potentially shippable product increment, meaning that at the end of each sprint, a new version of the software is available with additional features or improvements.

 **Adaptability:**

 Agile methodologies emphasize adaptability and the ability to respond to changing requirements. If new priorities or insights emerge, adjustments can be made in subsequent sprints.Sprints allow development teams to iteratively deliver value to customers and stakeholders in a controlled and predictable manner. By breaking down the work into manageable chunks and continuously seeking feedback, Agile teams can enhance collaboration, reduce risk, and improve the overall quality of the software being developed.

* **Question 6 – Explain Product backlog and sprint back log**

**Product Back Log**

The Product Backlog is a dynamic, prioritized list of all the features, user stories ,enhancements, bug fixes, and other work items that need to be addressed over the course of a project. It represents the entire scope of the product's development and is

managed by the Product Owner. The Product Backlog is continually refined and updated based on feedback, changing requirements, and new insights. The Sprint Backlog is a subset of the Product Backlog that represents the work that the development team commits to completing during a specific time period called a "Sprint." A Sprint is a fixed-duration iteration, usually lasting two to four weeks, in which the team works on a set of items from the Product Backlog.

**Sprint Back Log**

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

|  |  |
| --- | --- |
| **Product Backlog** | **Sprint backlog** |
| List of all the items that need to be completed for developing the end product | List of items to be completed in each sprint |
| Product owner collects the backlog from the customer and assigns to the team | The team collects the backlog from the PO and decides the time frame to complete during each sprint |
| Specific to the end goal | Specific to the sprint |
| Will vary based on the customer vision | Will vary based on the product vision as defined by the PO |
| The entire set of work in progress | The subset of the product backlog |
| Independent of the sprint backlog | Purely dependent on the product backlog |
| All product features listed and story points are assigned to every user story individually | For every sprint, the to-do list is the sprint backlog. The task is disintegrated from the user story to allow the team to estimate the time for completion in hours. |
| Planning poker tool is used to estimate time for each product backlog  |  Estimation is done at activity or task level |
| Product owner owns it  | Scrum team owns  |
| Until the completion of the entire project, the PO will maintain the backlogs  | Every new sprint will get new backlogs added by the team |

* **Question 7 -What is impediments log? write 2 impediments** .

**An impediment log**, also known as an issue log or obstacle log, is a document or tool used in Agile software development to track and manage obstacles, bottlenecks, or any factors that impede the progress of a project or team.

In terms of Scrum, they are “blockers” that prevent the Scrum Team from completing work, which in return impacts velocity. Anything that prohibits the team from doing work is considered an impediment. Impediments are hurdles or obstacles and these are recorded in Impediment Logs. They are “blockers” preventing the Scrum Team from completing work and should be included within impediment logs. All challenges faced by the team will be logged in the impediments log.

**A sick team member**

**Lack of system knowledge**

**Lack of management support**

**2 Impediments**:

● Delivery partner shortage in a specific region

● Technical issue causing intermittent order processing failure .

**1. Delivery partner shortage in a specific region**

|  |  |
| --- | --- |
| Login | 1 |
| Description | Delivery Partner shortage in specific region |
| Impact | Delays in order deliveries and increases customer dissatisfaction |
| Priority | High ( due to its impact and customer experience) |
| Assigned to | Operation Team and HR Team |
| Status | Open |
| Action Taken | The operation team is actively recruiting new delivery partners in the region. The HR team is working on fast tracking the onboarding process. |
| Resolution | Delivery Partner recruitment efforts are ongoing and the HR team is streamlining the onboarding process to expedite new hires. Regular updates are being provided in team meetings. |

**2.Technical issue causing intermittent order processing failure**

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

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

Velocity refers to the measure of the amount of work a development team can complete during a sprint.The calculation of velocity is performed by the development team itself, as they are responsible for estimating the effort required to complete each user story or backlog item.

**Tracking completed work:**

Tracking completed work in Agile development typically involves calculating the total story points completed by the team over a specific time frame, usually a sprint or iteration. Here's how you can calculate completed work:

**Identify Completed Stories**

 At the end of the sprint or iteration, review the user stories or tasks that were completed and accepted as done.

**Sum Story Points**

Add up the story points assigned to all the completed user stories. Exclude any story points that were not fully finished or accepted during the sprint.

**Calculate Total Completed Work**

The sum of story points completed represents the total completed work for that sprint.

This completed work can be used to calculate the team's velocity for that specific sprint,as mentioned in the previous response. It provides insights into the team's capacity and helps with future sprint planning and estimation.

**Summing story points:**

Summing story points involves adding up the numerical values assigned to individual user stories or tasks during the estimation process in Agile development. Story points are used to represent the effort, complexity, and size of a piece of work relative to other items on the backlog. Here's how you can sum story points:

 **List Completed User Stories**

Gather a list of user stories or tasks that have been completed during a specific sprint or iteration.

**Identify Story Point Values**

Each user story or task should have a story point value assigned to it during the estimation process. These values are usually relative, such as 1, 2, 3, 5, 8, 13, etc., representing increasing levels of complexity or effort.

**Add Up Story Point Values**

Sum up the story point values for all the completed user stories or tasks. For example, if you completed user stories with story point values of 3, 5, and 8, the sum would be 16.

The sum of story points provides a quantitative measure of the work completed by the team during a sprint. This sum is often used to calculate the team's velocity, which helps in future sprint planning and estimation

**Average velocity:**

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

**Select a Time Frame**

Choose a specific number of past sprints or iterations for which you want to calculate the average velocity. For example, you might choose the last 5 sprints.

**Sum Completed Story Points**

 Add up the total story points completed by the team in each of the selected sprints. This will give you the total completed work for the chosen time frame.

**Calculate Average**

Divide the total completed story points by the number of sprints or iterations you selected. This will give you the average velocity for that period.

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

**Use for Planning**

The average velocity can serve as a guideline for future sprint planning. It helps the team estimate how much work they can commit to in upcoming iterations based on their historical performance.

 Keep in mind that average velocity is a rough estimate and can fluctuate based on various factors. It's important to consider the team's capacity, any changes in team composition, and improvements in estimation accuracy over time

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

**Sprint Burn down :-**



**Product Burn down :-**





* **Question 10 – Explain about Product Grooming**

Product grooming, also known as backlog grooming or refinement, is a crucial activity in Agile development that involves preparing and refining items in the product backlog to ensure they are well-understood, prioritized, and ready for development. Let's breakdown the process step by step:

**Setting the Context**

At the beginning of the backlog grooming process, the team and relevant stakeholders come together to understand the overall goals and objectives of the project. This helps set the context for the work to be done and aligns everyone's understanding.

**Backlog Review**

The product owner and the development team review the items in the product backlog. This involves assessing the user stories, tasks, and other items to ensure they are accurate, up-to-date, and still relevant to the project's goals.

**Prioritization**

 During backlog grooming, the team collaboratively prioritizes the backlog items based on their value to the product and the needs of the users or customers. This helps ensure that the most important and valuable work is addressed first.

**Refinement and Estimation**

In this step, the backlog items are refined to provide clear and detailed descriptions. The team breaks down user stories into smaller tasks and discusses the technical requirements. Estimation involves assigning story points or other sizing metrics to each item, indicating the relative effort needed for implementation.

**Dependency Analysis**

The team examines potential dependencies between backlog items. Identifying and understanding dependencies helps in planning the order of implementation and managing potential bottlenecks.

**Acceptance Criteria**

Well-defined acceptance criteria are established for each backlog item. These criteria outline the conditions that must be met for the item to be considered complete and ready for delivery. Clear acceptance criteria help prevent misunderstandings and ensure a shared understanding of what is expected.

**Backlog Grooming Meetings**

 These are recurring meetings where the product owner and the development team come together to perform the activities mentioned above. These meetings often occur before sprint planning sessions to ensure that the upcoming sprint backlog is well-prepared.

 Backlog grooming is an iterative process that helps maintain a healthy and well-organized product backlog. It ensures that the development team always has a prioritized list of well-defined, estimated, and ready-to-develop items. This, in turn, supports the efficient planning and execution of sprints and helps the team deliver value to customers in a more predictable and effective manner

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

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Product Owner** | **Scrum Master** |
| **Nature of Work** | Collaborates with all the stakeholders and brings the vision of a product into the product backlog | Acts as a team coach and is responsible for maintaining the quality of the product |
| **Responsibilities** | Responsible for completing the project on time. Acts as an intermediary between development team and the customers | Ensures the scrum framework is followed and helps the development team create a quality product |
| **Accountability** | Responsible for project backlog and the timely completion of the product and for providing updates to the clients and stakeholders | Accountable for the quality of the entire project and for giving updates to the management about the completion of the product |
| **Reporting** | Reports to top management and clients | Reports to top management about the efficiency of the team and the quality of the product. |
| **Qualities** | Communication and the leadership skills, creativity ,critical thinking and a sharp mind are key assets for any product owner | Thorough knowledge of scrum theory and practices. Being able to lead the team but without the sense of authority |

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

**Spint Planning:**

 This meeting kicks off each sprint, which is a time-boxed iteration of work, usually spanning 2-4 weeks. During this meeting, the Scrum team, including the Product Owner, Scrum Master, and Development Team, collaborates to determine which backlog items (user stories, features, etc.) will be worked on in the upcoming sprint. The team also breaks down these items into tasks and estimates the effort required.

**Daily Stand-up (Daily Scrum):**

 Held daily during the sprint, this short meeting aims to facilitate quick and focused communication among team members. Each team member answers three key questions: What did I accomplish since the last stand-up? What will I work on until the next stand-up? Are there any obstacles or impediments in my way? This meeting helps keep everyone aligned and informed about the progress and challenges.

**Sprint Review:**

 At the end of each sprint, the team holds a review meeting to show case the work completed during the sprint to stakeholders, customers, and the Product Owner. The team demonstrates the potentially shippable product increment and gathers feedback. Based on this feedback, the Product Owner can update the backlog.

**Sprint Retrospective:**

 Also held at the end of each sprint, the retrospective is a dedicated time for the team to reflect on their processes and practices. The team discusses what went well, what could be improved, and any potential changes they'd like to make in the next sprint to enhance their efficiency and effectiveness.

**Backlog Refinement (Grooming)**

: While not officially part of the Scrum events, backlog refinement is an important ongoing activity. During these sessions, the team and the Product Owner review and refine backlog items, adding details, clarifications, and estimates to make them ready for inclusion in future sprints.

**Product Backlog Refinement**

: This meeting focuses on refining the product backlog items. The team and the Product Owner discuss and clarify requirements, priorities, and any changes needed in the backlog items. This ensures that the backlog is well-prepared for upcoming sprints.

**Release planning:**

This meeting occurs at the start of the project or major release and involves the product owner, development team, and stakeholders. It aims to discuss and plan the high level scope, timeline, and goals for the project.

**Ad hoc meetings:**

These meetings may be schedules as needed to address specific topics or issues, such as resolving impediments, discuss technical challenges, or conducting additional planning or collaboration sessions.

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

**Sprint Size**

In Scrum, a "sprint" is a time-boxed iteration during which the development team works to deliver a potentially shippable product increment. The length of a sprint is referred to as the "sprint duration" and is usually fixed throughout the project. Common sprint durations are 1 to 4 weeks. The choice of sprint duration depends on factors such as team velocity, project complexity, and business needs. A shorter sprint encourages more frequent opportunities for feedback and adaptation, while a longer sprint provides more time for development.

**Scrum Team Size**

The Scrum team size refers to the number of individuals who collectively contribute to the development of the product. A Scrum team consists of three key roles: the Product Owner, the Scrum Master, and the Development Team. The Development Team, in particular, is responsible for creating the product increment. Scrum recommends that the Development Team size be kept small, typically between 3 to 9 members, to facilitate effective communication, collaboration, and decision-making.

* **Question 14 – Explain DOR and DOD**

**Definition of Ready (DOR):**

The Definition of Ready outlines the criteria that a product backlog item (user story, feature, task, etc.) should meet before it is considered ready to be taken into a sprint for development. The DOR ensures that the item is well-defined, understood, and prepared for efficient development.

The specific criteria in the DOR can vary from team to team, but commonly include elements such as:

● **Clear description and acceptance criteria**: The item's requirements are clearly stated, and the conditions for its successful completion are well-defined.

● **Dependencies identified**: Any dependencies on external factors, teams, or resources are identified and addressed.

● **Estimable**: The team has enough information to provide a reasonable estimate of the effort required

.● **Testable**: It's possible to determine whether the item has been successfully implemented through testing.

● **Minimal ambiguity**: The item's details are clear, and any uncertainties are resolved.

**Definition of Done (DOD):**

The Definition of Done outlines the criteria that must be met for a product increment or backlog item to be considered complete and potentially shippable. The DOD ensures that the team maintains a consistent level of quality and completeness in their work.

The specific criteria in the DOD can vary based on the team's standards, the nature of the project, and the industry, but commonly include elements such as:

● **Code complete**: All development work is finished, including coding, testing, and integration.

● **Peer-reviewed**: Code has been reviewed by other team members for quality and adherence to coding standards.

● **Automated tests passed**: Automated tests (unit tests, integration tests,etc.) have been successfully executed and passed.

● **Functional requirements met**: The item meets all specified acceptance criteria and functional requirements.

● **Documentation updated**: Any necessary documentation, user guides, or technical documentation has been updated.

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

**Prioritization Techniques:**

Prioritization techniques are methods used to determine the order in which tasks,features, or items should be addressed in a project. These techniques help teams allocate resources effectively and focus on delivering the most valuable work first. Some common prioritization techniques include:

**MoSCoW:**

 This technique categorizes items into Must have, Should have, Could have, and Won't have categories. It helps clarify essential features from those that are optional or lower priority.

**Weighted Shortest Job First (WSJF):**

 WSJF assigns a priority score to each item based on factors like business value, time sensitivity, and risk. Items with higher scores are considered more important to work on.

**Kano Model:**

 This model categorizes features into Basic Needs, Performance Needs, and Delighters. It helps prioritize based on how features impact user satisfaction.

**Value vs. Effort Matrix:**

 Items are plotted on a matrix based on their potential value and effort required. This helps identify quick wins and high-value tasks.

**Relative Prioritization:**

 Teams compare items pairwise to determine which is more important. This helps create a relative ranking of items.

**Buy a Feature**:

 Stakeholders are given a budget to 'buy' features, which helps prioritize features based on how much value they see in them.

**Minimum Viable Product (MVP):**

 An MVP is the smallest version of a product that includes just enough features to provide value to early adopters and gather feedback. The MVP approach helps validate assumptions, learn from users, and iteratively build upon a product's foundation.

 It involves:

 **Core Functionality**:

 An MVP focuses on delivering the core functionalities that address the primary needs or pain points of the target users.

 **Minimal Features**:

 The MVP omits non-essential features to avoid unnecessary complexity and expedite development.

 Testing Hypotheses:

 The MVP tests assumptions and hypotheses about user behavior, market demand, and product viability.

 **Iterative Development**:

 Based on user feedback, the product is refined and expanded in subsequent iterations, gradually adding more features.

**Early Value:**

 The MVP allows the product to be released faster, gaining valuable insights and attracting early adopters.

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

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Business Analyst** | **Product Owner** |
| **Role focus** | Understand business needs, processes, and requirements. | Define, prioritize, and convey requirements for the product |
| **Requirement Gathering** | Gathers and documents detailed business requirements. | Creates user stories and defines product features |
| **Problem solving** | Identifies problems, in efficiencies, and suggests improvements | Drives the product vision,strategy, and value proposition. |
| **Communication** | Acts as a liaison between business stakeholders and development teams. | Collaborates with stakeholders, customers, and the development team |
| **Documentation** | Creates documentation of business rules, workflows and requirements | Manages the product backlog and maintains clear user stories. |
| **Scope Definition** | Helps define the scope of projects based on business needs | Defines the scope of product features and enhancements. |
| **Vision and Strategy** | Focuses on specific projector process improvements. | Has a holistic vision for the product and its strategic direction |
| **Backlog Management** | Not typically responsible for managing a product backlog | Manages and prioritizes the product backlog items. |
| **Prioritization** | Does not have a primary role in prioritizing features | Prioritizes features based on business value, user needs, and market trends |
| **Decision Making** | Provides input but not responsible for final product decisions | Makes final decisions on product features, enhancements, and priorities |
| **Iterative Development** | May or may not be involved in iterative development cycles. | Actively participates in sprint planning, reviews, and retrospectives |
| **Collaboration** | Collaborates with business stakeholders and development teams |  Collaborates closely with stakeholders, customers, and the development team. |
| **Acceptance** | Ensures business requirements are met. | Ensures user stories meet acceptance criteria and align with product vision. |
| **Leadership and Strategy** | Focuses on tactical solutions and improvements. | Focuses on strategic leadership and product direction. |
| **Continuous improvement** | Contributes to process improvements and business efficiency. | Incorporates user feedback for ongoing product enhancement |

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



