**AGILE METHOD**

Introduction:

Agile method is the combination of iterative and increment process model. Team members must have trust in one another. The distribution of skills must be appropriate to the problem. This process that encourages frequent inspection and adaptation. focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations.

Agile Software Development Lifecycle:



1. Concept - Projects are envisioned and prioritized
2. Inception - Team members are identified, funding is put in place, and initial environments and requirements are discussed
3. Iteration/Construction - The development team works to deliver working software based on iteration requirements and feedback
4. Release - QA (Quality Assurance) testing, internal and external training, documentation development, and final release of the iteration into production
5. Production - Ongoing support of the software
6. Retirement - End-of-life activities, including customer notification and migration

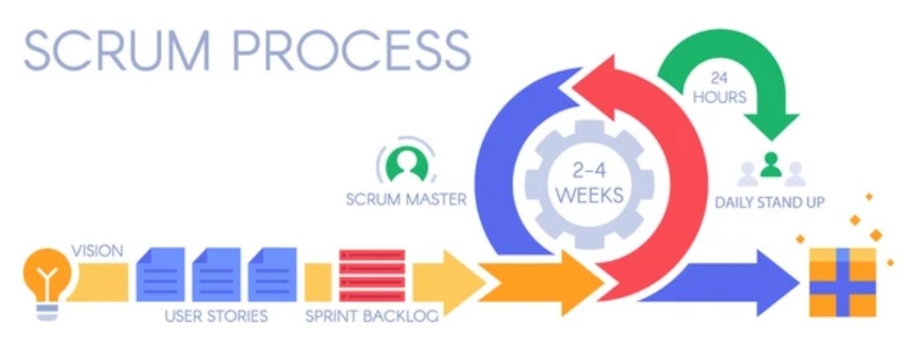


SCRUM PROCESS

SCRUM:

Scrum is an agile development methodology used in the development of Software based on an iterative and incremental processes. Scrum is adaptable, fast, flexible and effective agile framework that is designed to deliver value to the customer throughout the development of the project.

Differences: Agile is a philosophy, whereas Scrum is a type of Agile methodology. Scrum is broken down into shorter sprints and smaller deliverables.



· Scrum meeting

· A team leader called a “scrum master” leads the meeting & assess the responses from each person (Scrum team) .

· Short meetings held daily by the scrum team.

· Three key Q’s are asked & answered by all team members

· What did you do since the last team meeting? (progress)

e.g.:

1.Quantity

2. Whole process

· What obstacles are you encountering?

e.g.: cancel option

· What do you plan to accomplish by the next team meeting?

· Demos

· Deliver software increment to the customer

· Customer evaluates the functionality

Product Backlog in Agile Methodology

* The product backlog is an ordered list of everything that might be needed in the product and is the single source of requirements for any changes to be made to the product.
* The product backlog is used to:capture requests for modifying a product. This can include add new features, replacing old features, removing features and fixing issues; and ensure the delivery team is given work which maximizes the business benefit to the owner of the product.

List of features require for our problem statement.

* Availability of product
* Time of Rent----Return date
* Cost
* Address
* Payment
* Delivery
* Notifications
* Pickup

Why Product Backlog is Important?

* It is prepared so that estimates can be given to each and every feature.
* It helps in planning the road-map for the product.
* It helps in re-ranking the features so that more value can be added to the product.
* It helps in determining what to prioritize first. Team ranks the item and then builds value.

Sprint backlog

Backlog:

* A prioritized list of project requirements or features that provides business value for the customer.
* Items can be added to the backlog at anytime.
* The product manager assesses the backlog and updates priorities as required.

Sprints:

* Consists of work units that are required to achieve a requirement.
* During the sprint, the backlog items that the sprint work units addresses are frozen (changes are not allowed).
* The sprint allows team members to work in a short-term, but stable environment.

Features of the application:

1. Toy availability status
2. Rent duration and time taken to deliver the toy to the customer.
3. Calculation of cost
4. Payment gateway

Advantages

* Each person will have multiple skills
* A face-to-face conversation is the best form of communication
* Product is developed fast and frequently delivered (weeks rather than months)
* Even late changes in requirements are welcomed
* It follows the incremental approach
* People and interactions are emphasized rather than process and tools. Customers, developers and testers constantly interact with each other
* Changes in the requirements are accepted even in the later stages of the development

Limitations

* Increase in **Cost**-
* Increase in **Time**-
* More useful for **small** and fast-moving projects-
* Need members with **multiple skills**-
* Need **experienced** members-