**INVEST**

I - independent

N - Negotiable

V - Valuable

E - Estimable

S - Small

T - Testable

**1. Independent:** Independent means the user story should be self-contained and not dependent on other stories for completion.

=>An example of an independent user story is “As a user, I can register a new account so that I can access the app’s features.”

=>An example of a user story that is not independent is “As a user, I can update my account profile after registering my new account.”

The second story depends on the first one being completed to be actionable.

Of course, some logical dependencies between stories are unavoidable. However, applying INVEST encourages finding ways to separate dependencies whenever possible.

**2. Negotiable :** The negotiable criterion emphasizes that the user story is not a detailed specification set in stone. Instead, it is a starting point for an ongoing conversation between the product owner and development team to reach alignment.

**3. Valuable:** The valuable criterion emphasizes that every user story must clearly capture the value it will deliver if completed as described.

Some of the ways to make stories valuable include:

Describing how it helps users complete relevant tasks or achieve goals

Quantifying the expected business benefit

Identifying the risks or costs it will mitigate

Specifying how it aligns with the product vision and strategy

**4. Estimable:** For a story to be estimable, it must be defined clearly enough that the team can gauge the relative effort required to complete it.

**5. Small:** Small means user stories should capture the smallest possible increment of customer value.

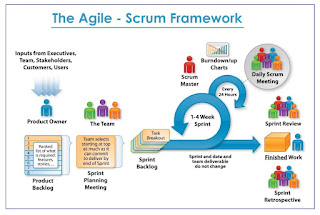
**6. Testable:** The final INVEST criterion is that a good user story must be testable, meaning that there are clear, objective criteria for determining whether the story was successfully implemented as intended.

**SCRUM**

Scrum is a framework used primarily in agile software development for managing and completing complex projects. It emphasizes teamwork, collaboration, and iterative progress toward a well-defined goal. The term "Scrum" originates from rugby, where it refers to a method of restarting play after an infringement.

In software development, Scrum operates on the principles of transparency, inspection, and adaptation.

In scrum project move forward with series of iteration that are called Sprints. Each sprint size is typically two to four weeks long. It is based on inspect and adaptive cycle. Produce product incrementally and iteratively, thus reduce risk and enhance visibility.



**1 Product Owner**

* Product Owner (PO) is client's representative, define features of product and decide release date and content
* Priorities features according to market value and be responsible for the profitability of product
* Accept or reject work items

Example :  
In a software development company, the team is working on creating a task management application aimed at improving productivity for both individuals and teams. The Product Owner (PO) in this scenario is Sarah.

**Role of the Product Owner (Sarah):**

Sarah is responsible for understanding the needs of the users and stakeholders, prioritizing features, and ensuring that the development team delivers a product that meets those needs.

**2 Scrum Master**

* Coach for scrum team , Enacting scrum values , Ensure team's productivity
* Build winning team, apply agile principles and make system effective.

EXAMPLE:

In a software development company, the team is struggling with collaboration and communication issues. The product delivery has been delayed multiple times due to misunderstandings, lack of transparency, and inefficient coordination among team members.

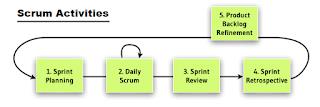
**Introduction of a Scrum Master:** The management decides to introduce a Scrum Master to help address these issues and improve team effectiveness.

ACTIONS:

1. facilitating daily standups
2. removes obstacles
3. encouraging collaboration
4. promoting transparency
5. facilitating retrospectives
6. coaching and mentoring

**3 Team**

* 5-9 Members team (Developer , Tester) , Self-organizing, High performance team
* Build winning product, Work collaboratively and share responsibilities, Cross functional team.



Scrum Activities:  
1. Sprint Planning  
2. Daily Scrum  
3. Sprint Review  
4. Sprint Retrospective  
5. Product Backlog Refinement  
  
**1 Sprint Planning:**  
       Goal: Team to plan and agree on backlog items they can complete and confirm the tasks required to support acceptance

**2 Daily Scrum:**  
Goal: Plan for the day, Inspect and Adapt daily towards reaching the sprint goal.  
Description:

* Daily development Team standup for 15 minutes in circle and talk only on three points
* What I did since last daily scrum meeting?
* What I am planning to work on today?
* Impediments (Issue/blocker) if any?
* Scrum master protect the team and facilitate for being effective.
* This give an opportunity to team to inspect and adapt daily on the sprint goal.
* Who: Scrum Team, Scrum Master, When: Daily throughout the sprint , Duration: 15 minutes maximum

**3 Sprint Review:**  
Goal: Get feedback on product development. Inspect and adapt on the product feature.  
Description:

* During this meeting team demonstrate 100% completed work.
* Scrum master facilitate the environment.
* In case of new request, Product owner (PO) note and updates the product backlog as required.
* Product owner is final decision maker on acceptance.
* Duration: 2 hours for a 2 week sprint, Who: Scrum Team, Scrum Master, PO, Stakeholders, When: Last day of sprint

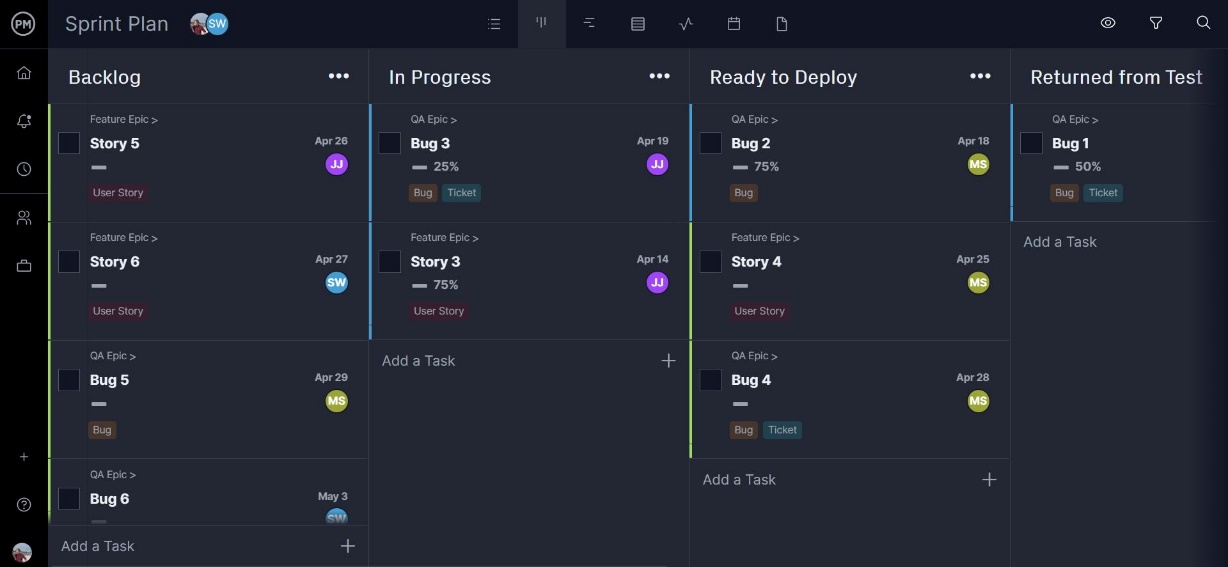
**4 Sprint Retrospective:**  
Goal: To inspect and adapt to become more effective and efficient on process, people, culture aspect.  
Description:

* Participation in the discussion to inspect and adapt as scrum team.
* Scrum master play vital role in sprint retrospective, Scrum master bring in the culture of openness, trust and respect as people discuss the improvement areas, facilitate and focus on improvement and changes that pointing fingers at others.
* This is platform to scrum master to help team resolve ineffectiveness in the systems
* Inspect and Adapt: Try everything that makes sense, reject things that didn’t work even after repeated trails. Shape your culture, process and practice.
* Duration: 2 hours for a 2 week sprint, Who: Scrum Team , When: Last day of sprint

**5 Product Backlog Refinement:**  
Goal: Keep product backlog items ready, uncertainty to certainty  
Description:

* Product owner provide clarity on each product backlog item (All uncertainty clarified into certainty )
* Product owner Update product backlog. 100% be present and involve all team members
* Team understand, carefully listen to need of product owner, understand the acceptance criteria. Help product owner to order the backlog.
* Duration: 1-3 hours depending on the team’s need. , Who: Scrum Team, Scrum master, PO, When: Continuous process, in between the sprints.

Scrum Artifacts:  
Below are Scrum Artifacts.  
1) Product Backlog,   2) Sprint Backlog, 3) Product Increment  
  
**1 Product Backlog**  
This is an ordered list of ideas for the product, which can come from the product owner, team members, or stakeholders. A description and estimate of effort complement each product backlog item.  
  
The product backlog is ordered to maximize the value delivered by the Scrum team. The development team’s work comes from the product backlog, and nowhere else. Every feature, enhancement, bug fix, documentation requirement, every bit of work the team does comes from a product backlog item.



This product backlog shows project tasks and user stories, as well as their deadline, who’s assigned to complete them, their priority level and percent complete. Managers can easily drag and drop these tasks to refine the product backlog. In addition, Project Manager also allows team members to interact in real time.

**2 Sprint Backlog**  
The sprint backlog is the list of refined product backlog items chosen for development in the current sprint, together with the team's plan for accomplishing the work. It reflects the team's forecast of what work can be completed. Once the sprint backlog is established, the development team begins work on the new product increment.  
  
**3 Product Increment**  
Every sprint produces a product increment, the most important Scrum artifact. A product Increment is the "goal line" for each sprint and, at the end of the sprint, it must:

* Be of high enough quality to be given to users
* Meet the Scrum team's current definition of done
* Be acceptable to the product owner

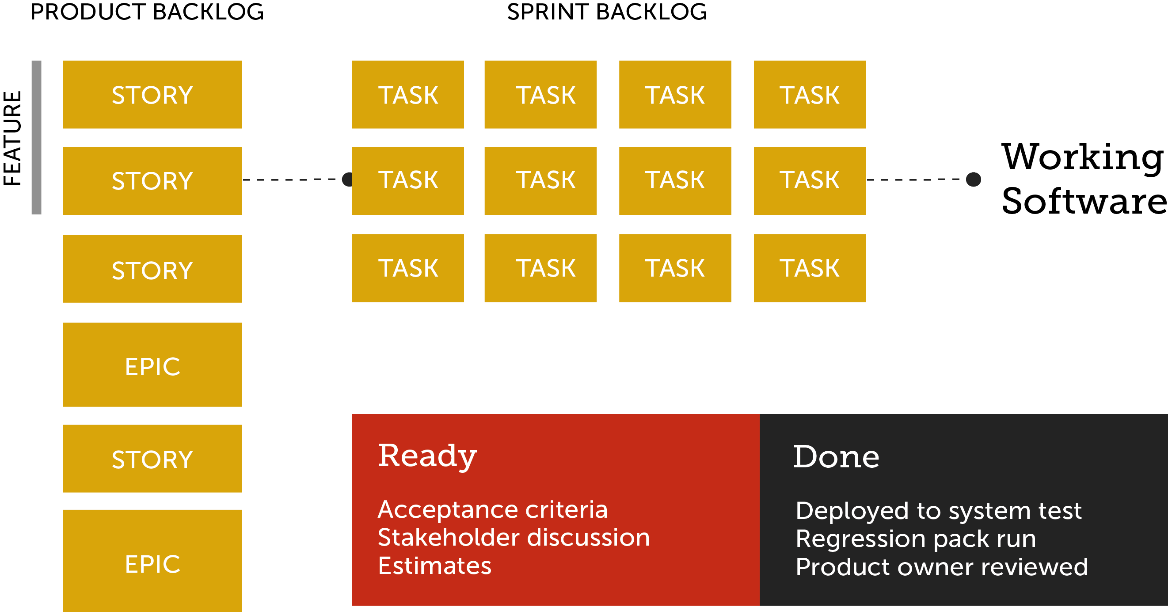
"Epic" refers to a large body of work that can be broken down into smaller, more manageable pieces called user stories. Epics are essentially high-level containers for grouping related user stories that collectively represent a significant feature, requirement, or theme within a project.

PRODUCT BACKLOG

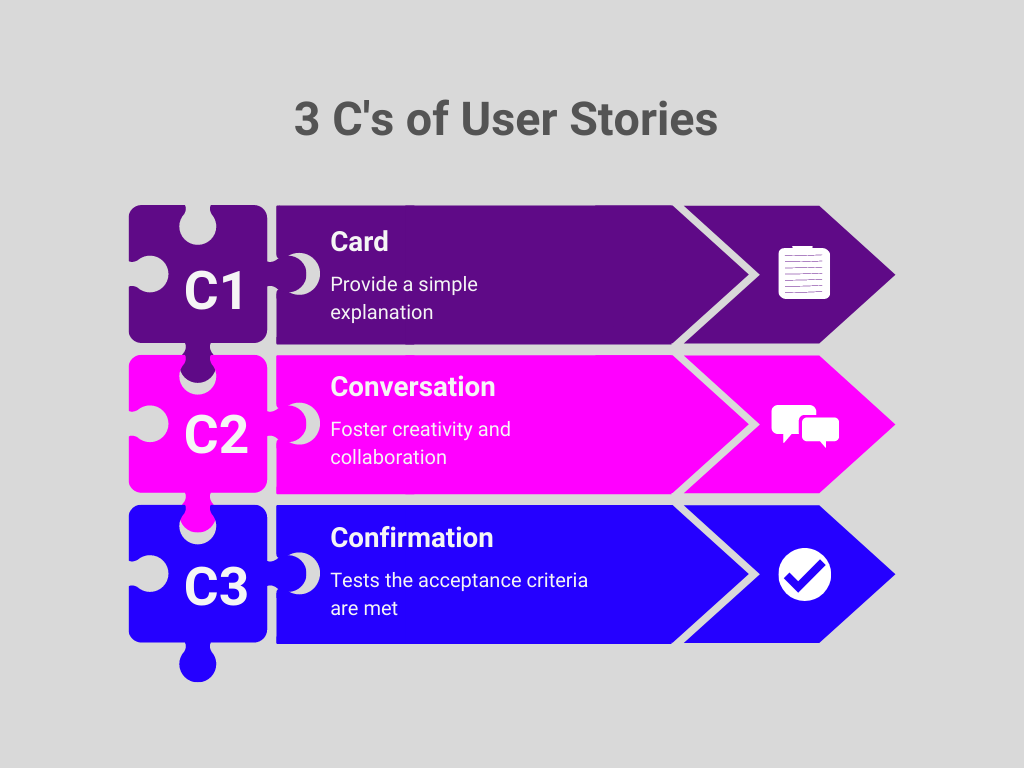
The product backlog is a to-do list for the team – a list of all of a product’s features in priority order.

In Scrum the product backlog is usually first generated by a workshop of stakeholders (including the Product Owner)– perhaps guided by a Product Vision – where all of the important features that can be thought of are noted down.

This list then feeds into the first sprint backlog. After that, as the Scrum team gains a fuller understanding of the product and the customer’s needs, the product backlog will grow and develop through successive iterations.



**3 C’s**



**Card**

Although you can take this literally, the idea behind the first C – card refers to the user story’s **optimal size that can fit on a notecard**.

The card should not contain all information about the requirement. Instead, it must have enough information to plan, identify the need, and remind the team of the overall story.

### ****Conversations****

Before placing user stories in the sprint, the product owner should **ask customers for elaboration and validation.**

These conversations are necessary as a user story may be difficult to interpret. Plus, background knowledge could be essential for implementation.

These conversations allow product owners to inform stakeholders of what’s going on. It also allows everyone to digest information accordingly.

Conversations can include email, internal chat, or any online tool for your cross-functional team.

### ****Confirmations****

The final C of a user story stands for the **acceptance criteria.** It confirms that the user story has been implemented correctly and is successfully delivered.

It would be best if you defined acceptance criteria before development begins. It helps determine when each user story is finished and working as intended.