

Module 1: Introduction to User Stories (5 Hrs)

1. What Are User Stories?

- Definition and purpose in Agile methodology.
- User stories vs. requirements vs. use cases.

2. Characteristics of Good User Stories

- INVEST principle: Independent, Negotiable, Valuable, Estimable, Small, Testable.
- The 3 Cs Framework: Card, Conversation, Confirmation.

3. Importance of User Stories in Agile

- Role in sprint planning, backlog grooming, and product delivery.
 - Stakeholder alignment and communication.
-

Module 2: Structure of a User Story (4 Hrs)

1. Basic Format:

- "As a [user role], I want [goal] so that [reason/benefit]."

2. Detailed Components:

- **User Role:** Identifying personas or end users.
- **Goal:** Capturing intent or functionality.
- **Reason/Benefit:** Explaining value for the user and business.

3. Acceptance Criteria:

- Defining "done" for the story.
 - Gherkin syntax (Given-When-Then) for clarity.
-

Module 3: Writing Effective Technical User Stories (5 Hrs)

1. Understanding Technical Requirements:

- Capturing backend and system requirements in user story format.
- Examples: API integrations, database changes, performance improvements.

2. Balancing Technical and User Needs:

- Translating technical details into user-centric language.

- Ensuring business value in technical tasks (e.g., refactoring, security updates).

3. Breaking Down Epics into Stories:

- Techniques for decomposition.
 - Examples of splitting stories for iterative development.
-

Module 4: Practical Writing Workshop (3 Hrs)

1. Hands-on Exercises:

- Write stories for common technical tasks (e.g., "As a developer, I need logging to track system errors so that I can debug issues efficiently.").
- Review and critique user stories in groups.

2. Case Studies:

- Analyze and rewrite poorly written user stories.
 - Work on stories from real-world scenarios.
-

Module 5: Tools and Best Practices (4 Hrs)

1. Tools for Writing and Managing Stories:

- Jira, Trello, Azure DevOps, or other backlog tools.

2. Best Practices:

- Collaborating with stakeholders to refine stories.
- Continuous improvement through feedback loops and retrospectives.

3. Anti-patterns to Avoid:

- Vague goals, lack of acceptance criteria, too much technical jargon, and over-detailed stories.
-

Module 6: Validation and Testing (3 Hrs)

1. Acceptance Criteria Testing:

- Mapping acceptance criteria to test cases.
- Collaborating with QA teams for story validation.

2. Closing the Feedback Loop:

- Using feedback from delivered stories to improve future writing.
-

Assessment and Certification

1. Final Project:

- Participants create a set of user stories for a mock product or feature.

2. Evaluation:

- Stories reviewed based on clarity, adherence to INVEST, and actionable criteria.

3. Certificate of Completion:

- Recognition for participants who complete the course successfully.