

Chapter 5

Title: *Agile Crash Course: Agile Project Management; Agile Delivery*

Prepared by: Anagh Sharma (2309809)

Date: 01/11/2023

1. Building projects in stages wherein the stages themselves are deliverable. Thus, iteratively various versions of a project over various stages are delivered to the consumers. All these versions are usable for the consumer.
2. Sprints: Period between consecutive stages. Thus, Agile is time boxed. Note: An Epic is a User Story that is so that it cannot be delivered in a single sprint.
3. Sprints duration: average 2 weeks in Agile.
4. User stories: Features that a developer works on during the sprints. Note: Agile inherently allows for close and daily interaction between businesspeople and developers.
5. MVP: Minimal Viable Project. The MVP is a minimal version of the project delivered in the earliest stages of the Agile project development. The MVP is thus an early and elementary working version of the project. No bells and whistles.
6. Test, plan, and analyse: At the end of each sprint.
7. Flexibility: The scope of the project may evolve as the project is being developed. Note: Regular adaptation to changing circumstances is common.
8. Documentation: Kept short and simple.
9. Roles blur: Roles like developer and tester are less rigid and engineers play various roles. Thus, roles blur, overlap and complement each other. Note: Self-organizing teams is a trait of Agile.
10. Team's Velocity: Speed of execution and delivery. Note: Sustainable development or the ability to maintain a constant pace.
11. Principle measure of progress: Working software.
12. Retrospective: Reflect for improvement.
13. User Stories: Something that specifically adds value to the customer. Everything else is either a task, subtask, or requirement.
14. Product Backlog/Master Story List:
 - Lists all the User Stories that will be part of the product.
 - Dynamic in Agile.
15. Sprint Backlog:
 - User Stories being worked on in a particular Sprint.
 - These User Stories are sourced from Product Backlog/Master Story List.
16. Story Points:
 - The level of complexity of a User Story.
 - You assign Story Points to the User Story you create.
 - For example: On a scale from 1 to 3.
 - Should be assigned instinctively.
17. Velocity:
 - Relates to the speed of execution.
 - Total number of Story Points that a team delivers in a sprint.
18. Swimlanes:
 - A visual representation of Agile Project Development that is used in tools like the Kanban/Agile Board. For example:

- Swimlanes (Y-axis): Rows representing criterion such as a User Story or even an Engineer performing tasks.
- Statuses (X-axis): Status of a story on Agile (For example: To Do, Doing, Done statuses.)

19. Release:

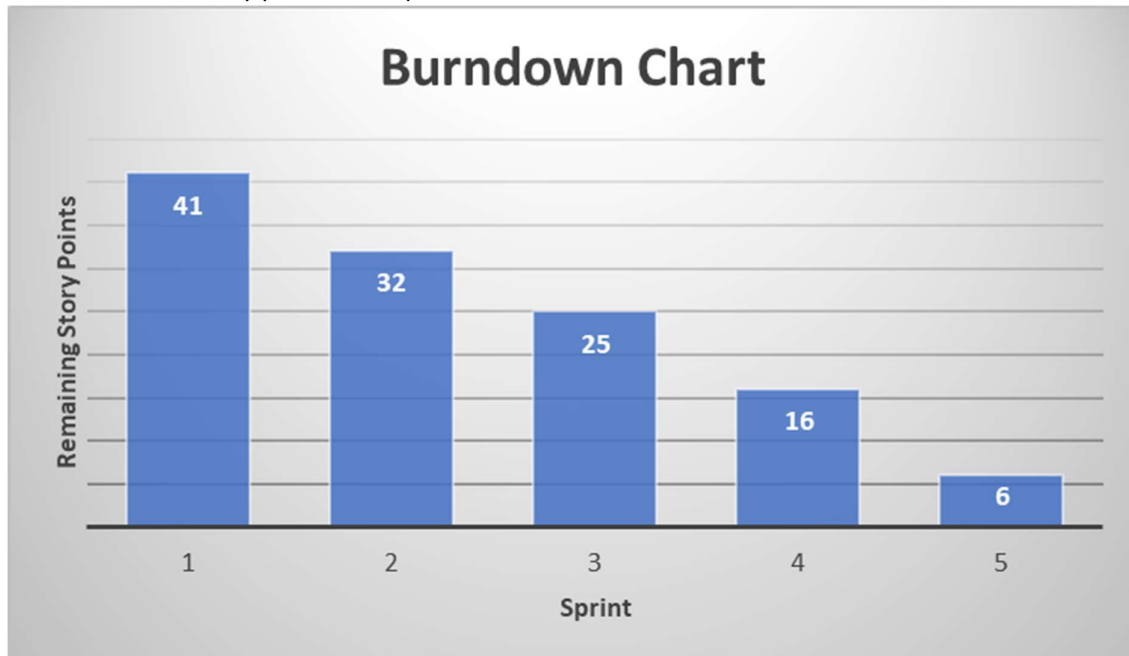
- A User Story that has been moved to a production environment.
- A release is generally made once a month. (Or after 2 sprints.)

20. Three roles in an Agile team:

- Product Owner : The end customer
- Scrum Master : The team leader.
- Delivery people : The people developing the project.

21. Tools:

- Burndown Chart: Visually represents the amount of work remaining at each stage in terms of story points. Example below.



- Agile Kanban Board:
 - Visual representation of the work being done in the Agile project development.
 - For example:
 - Swimlanes (Y-axis): Rows representing criterion such as a User Story or even an Engineer performing tasks.
 - Statuses (X-axis): Status of a story on Agile (For example: To Do, Doing, Done statuses.)
 - Example:

BACKLOG	SWIM LANE	TEAM MEMBER	TO DO	DOING	DONE
User story 28	1	Charlie	User story 7 User story 18 User Story 24	User story 2	User story 8
User story 4	2	Angie	User story 11	User story 40	User story 1 User story 5
etc					

22. Rituals

- Sprint Planning
 - Participants : Delivery Team and Scrum Master
 - Goal : Select User Stories for the next Sprint
- Daily Stand-ups
 - Participants : Delivery Team and Scrum Master
 - Goal : Stand-up meeting near the Kaban board for 15 minutes where everyone describes what they did yesterday, what they are doing today, and any impediments.
- Sprint Review/Demo Sessions
 - Participants : Product Owner, Delivery Team, and Scrum Master
 - Goal : Present to the Product Owner OR just the team at the end of a sprint the User Stories which were delivered, and which weren't.
- Retrospectives
 - Participants : Delivery Team and Scrum Master
 - Goal : Review, revise, and adjust at the end of a sprint. What went well? What didn't? What could we do differently?

23. Demonstration of tools such as Trello and Jira. Daily simplistic and straightforward.

24. Agile Kanban Board using Microsoft Planner

- Integrates well with other Microsoft tools
- Can be customized easily
- It is advised that project Tasks should be treated as User Stories. While it is true that a task may be trivial and thus theoretically cannot be called a User Story, it is still advised to treat the terms interchangeably from a developer's perspective.

25. Before starting Agile: Sprint 0, Planning

- Define team and roles
- Decide time for Agile rituals
- Ready tools such as Kanban tools
- Ensure due diligence has been done for the project
- Determine constraints of the project
- Plan timelines for sprints, tests, and releases

26. How to start

- Create User Stories
- Assign them story points
- Define priorities
- Start the first Sprint
- Continue to iterate

27. Complete Agile Process

1. Define the Vision: Start with the end goal or the main objective of the project. What are you trying to achieve? This can be documented in a vision statement or a product roadmap.

2. Create Epics: An epic is a large chunk of work that can be broken down into smaller tasks. It's more granular than the vision but not as detailed as individual tasks. For example, "User Authentication" might be an epic for a software project.

3. Develop User Stories: For each epic, create user stories. A user story is a description of a feature or requirement from an end-user perspective. It typically follows the format: “As a [type of user], I want [an action] so that [a benefit/a value]”.

4. Break Down User Stories into Tasks: A user story can often be broken down into multiple tasks. For instance, for the user story “As a user, I want to reset my password”, tasks could be:

- Design the “Forgot Password” interface.
- Develop backend logic for password reset.
- Implement email service for sending reset links.
- Test the password reset functionality.

5. Estimate Tasks: Once tasks are identified, the team can estimate the time or effort required for each task. In Agile, this could be in story points or hours, depending on the team’s preference.

6. Prioritize: Determine the order of importance for your user stories and tasks based on business needs, dependencies, risks, and other factors.

7. Plan Sprints: In Scrum (a type of Agile methodology), work is divided into sprints, which are time-boxed iterations (often 2-4 weeks). Place your prioritized tasks into these sprints based on the team’s capacity and the project timeline.

8. Review and Adjust: After each sprint, conduct a retrospective. Discuss what went well, what didn’t, and how processes can be improved. This feedback loop ensures that the project remains adaptable, and the team continues to improve.

9. Maintain a Backlog: Not all tasks or stories will fit into the immediate sprints. Keep a well-maintained and prioritized backlog of these items. Over time, some backlog items might become irrelevant or new ones might be added.

10. Continuous Communication: Foster an environment of open communication. Ensure that team members can discuss challenges, clarify doubts, and collaborate on solutions. This keeps everyone aligned and ensures that tasks are accurately defined and understood.

Remember, the goal of Agile is flexibility and adaptability. As you move through the project, it’s okay (and expected) that tasks, priorities, and even the broader scope may change. The key is to maintain communication and ensure that the team is always working towards the most valuable outcomes for the project.