

Agile - Scrum

Companies using Agile Scrum



- Scrum is **one of the Agile types**/frameworks.
- Scrum is **used the most** compared to the other Agile types
- Scrum encourages the **team to work together, learn through experiences, self-organize** while working on a problem, and **reflect** on their wins and losses to improve continuously.
- In the Scrum team, there is no overall team leader; **How a problem is tackled or who do what is decided by the team as a whole.**
- **Scrum team creates and adapts their ways of working.**
- What puts Scrum aside from other Agile frameworks are: Scrum **three roles** in the Scrum team, **Scrum events(meetings)**, and scrum **artifacts** (**Artifact** refers to **key information** needed during the development of a product.)

3

Roles



Product Owner



Development Team



Scrum Master

3

Artifacts



Product Backlog



Sprint Backlog



Product Increment

5

Meetings



Grooming



Sprint Planning



Daily Stand Up



Demo



Retro

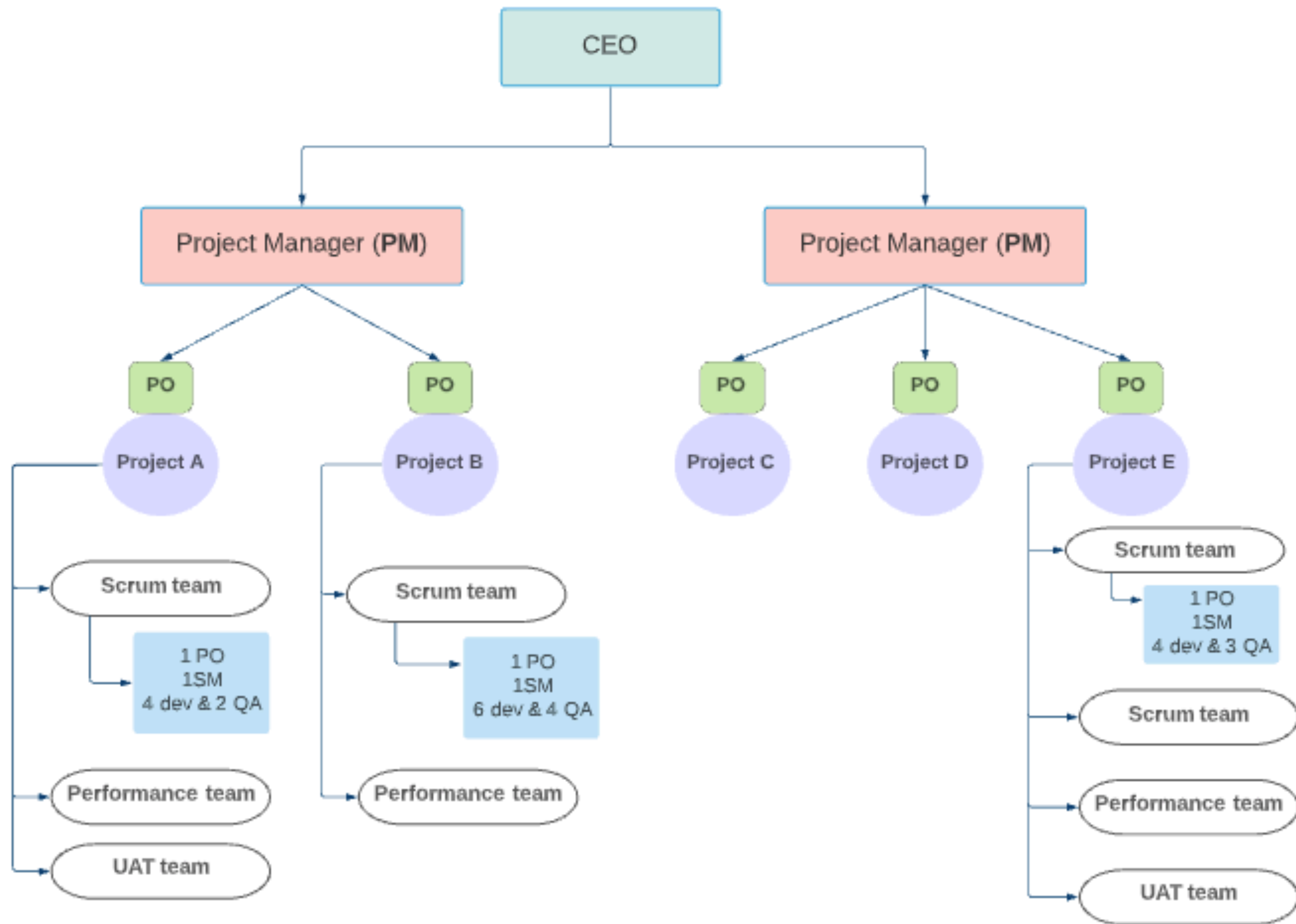
Scrum Team has 3 roles -> PO , SM , Dev-team



Team Structure

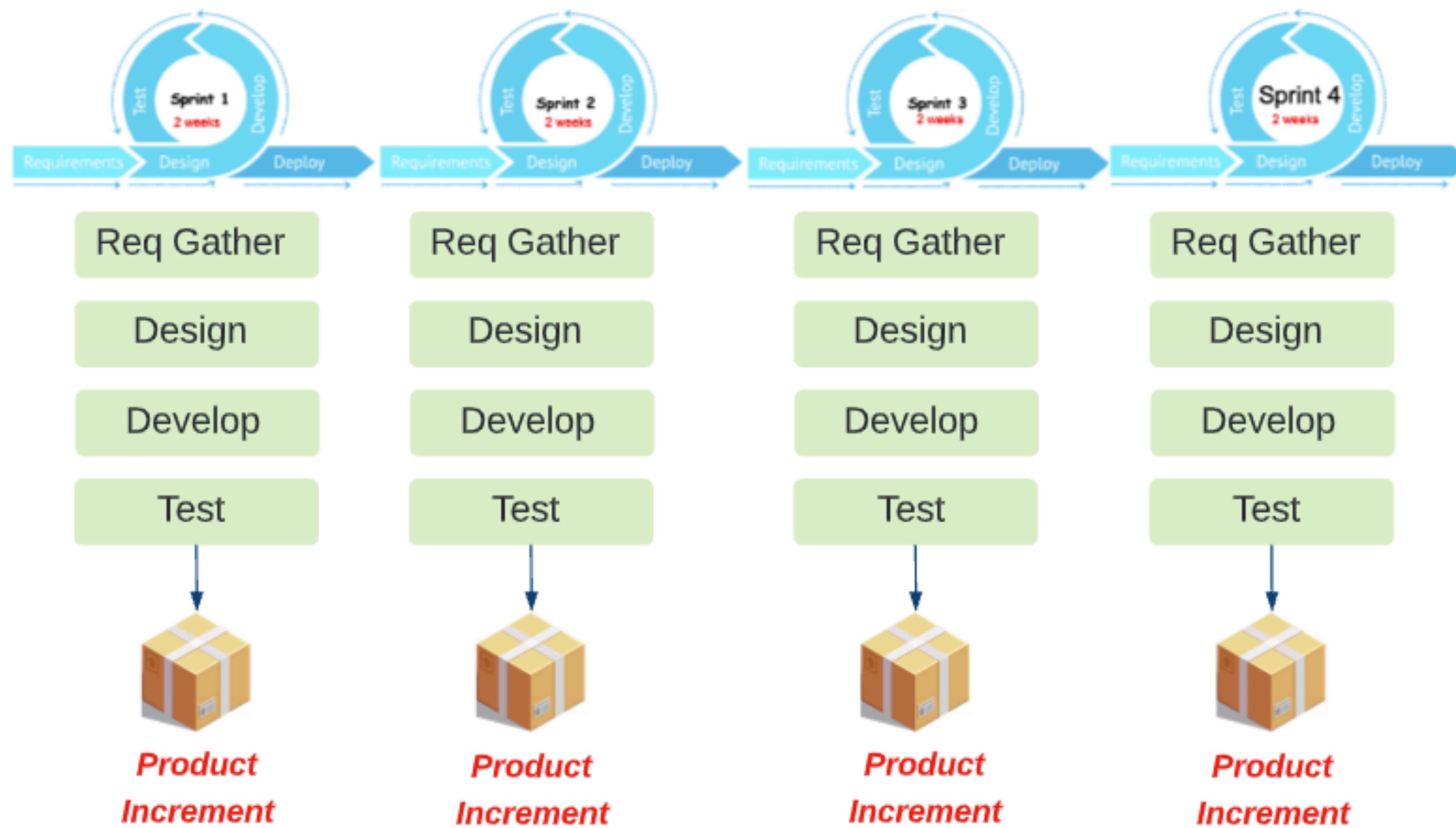
Teams

- One **PM** is **responsible several projects** in the company
- One **PO** is responsible for **one project**
- One project can have 2 - 3 teams, or even more than 10 teams
- Each **Scrum team** has 3 roles -> PO, Dev team, SM



Sprint

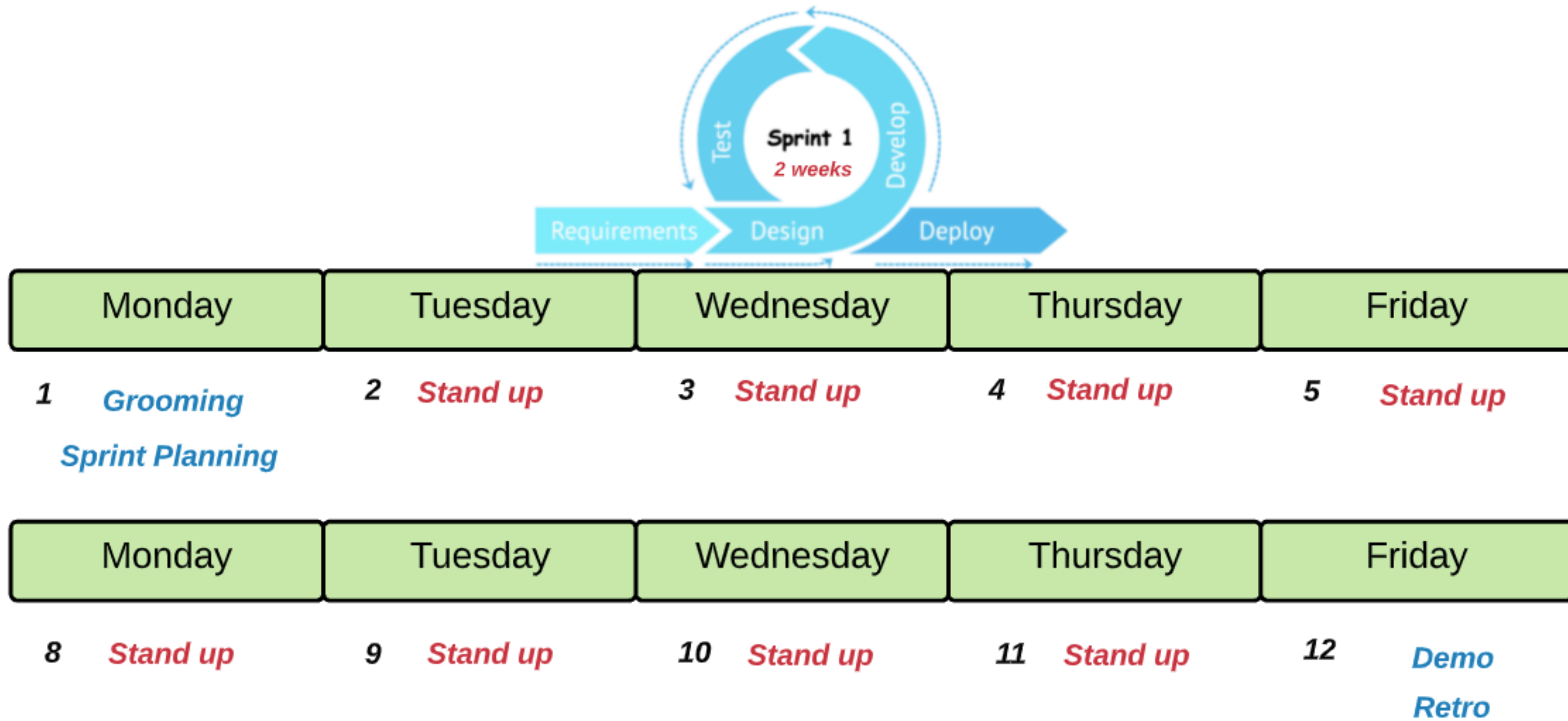
- Sprint is a **fixed time box**, also called **iteration**
- A sprint is between one week and four weeks. [2 weeks sprint cycle is the most common]
- The **team** members **decides** the **sprint length** together.
- An active Sprint cannot be extended.
- A sprint is used to accomplish **sprint goals** such as:
 - Conduct scrum meetings - grooming, sprint planning, stand up, demo and retro
 - **Specific requirements must be completed** (developed & tested)
 - **Demonstrate the completed work to the client**

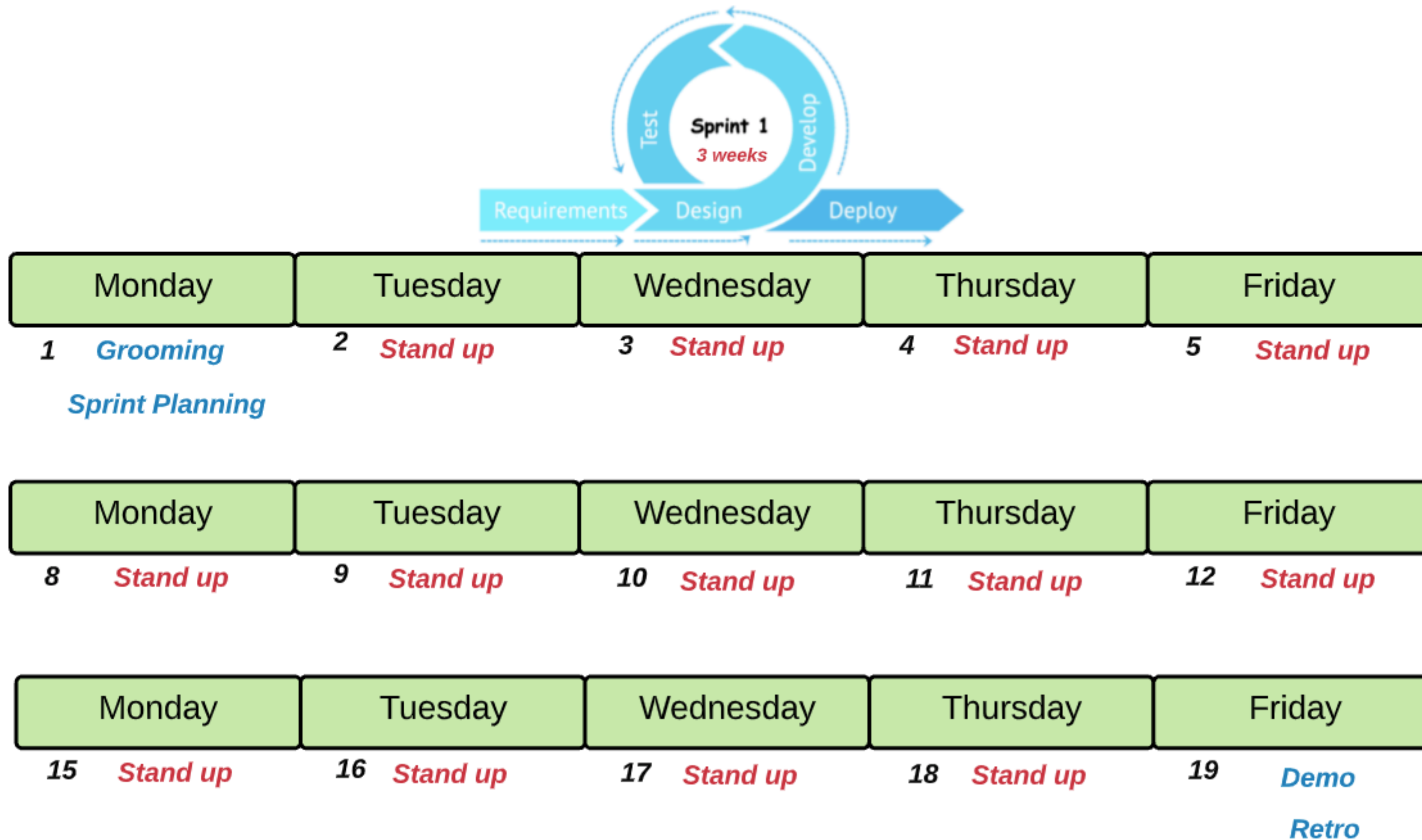


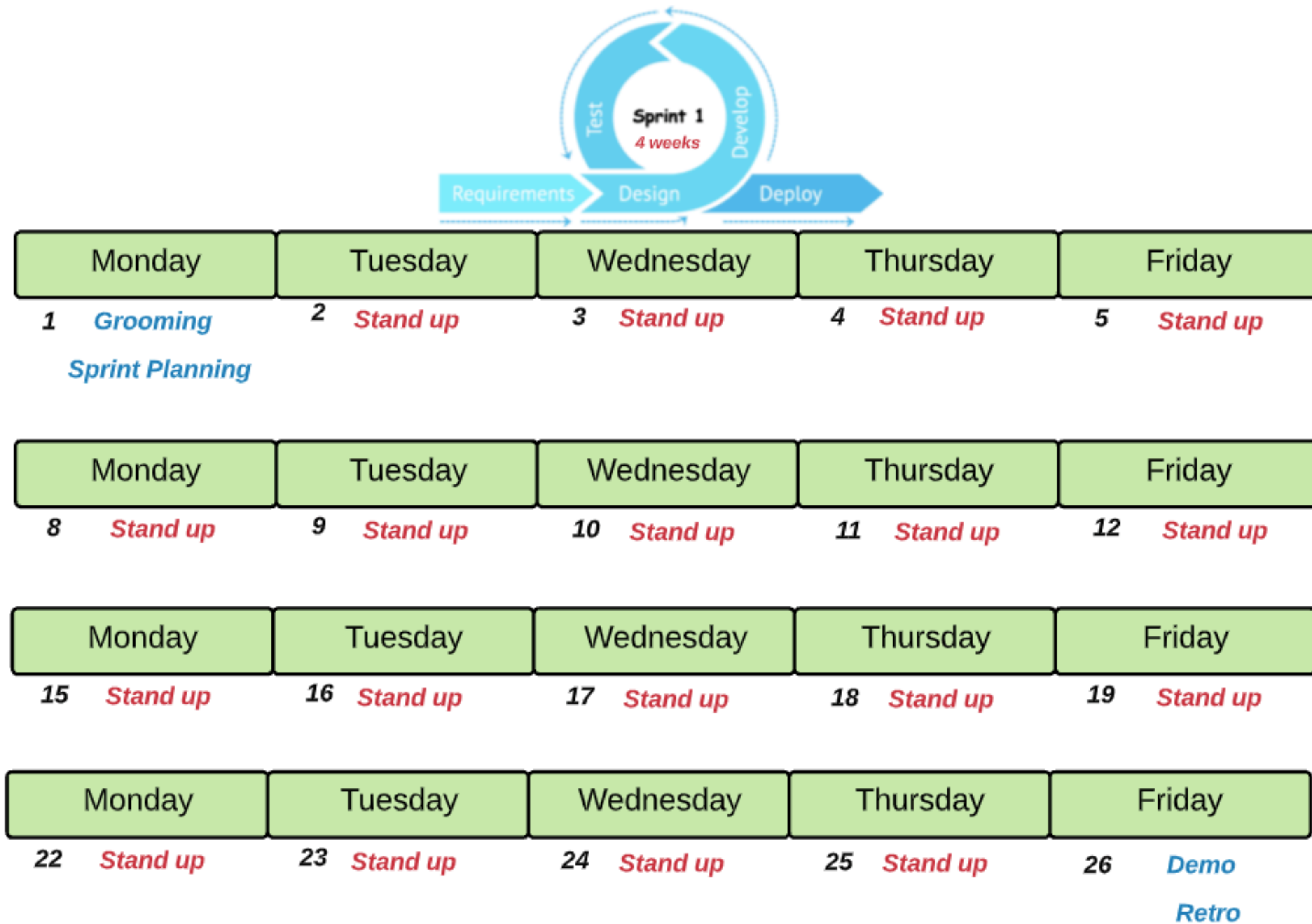
Release the app



Example of 2-month release cycle







Sprint 1

Sprint 2

Sprint 3

MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY	SATURDAY	SUNDAY
01 S1 starts	02	03	04	05	06	07
08	08	10	11	12 S1 ends	13	14
15 S2 starts	16	17	18	19	20	21
22	23	24	25	26 S2 ends	27	28
29 S3 starts	30					

User Story

- In Agile, especially in Scrum, **User Stories** are replaced the **traditional requirements** specification.
- A User Story **short and simple** description of a software **requirement**.
- **Product Owner writes the User Stories.**
- With a User Story, the software functionalities are described from the user's perspective.
- When PO writes User Stories, PO confirms each User Story meets an INVEST Criteria:
 - I – **Independent** (among other stories)
 - N – **Negotiable** (a flexible statement of intent, not a contract)
 - V – **Valuable** (providing a valuable vertical slice to the customer)
 - E – **Estimable** (small and negotiable)
 - S – **Small** (fits within an iteration)
 - T – **Testable** (can be tested and verified)

- The recommended form of expression of User Story is the 'user-voice form', as follows:

As a (user role), I want to (activity), so that (business value)

With a user story in this format, the teams are guided to understand who is using the system, what they are doing with it, and why they are doing it.

Example User Stories from the Canvas app









- As a student, I want to watch recordings so that I can review the classes.
- As a teacher, I want to upload files so that I can share class materials with students.
- As a student, I should be able to view the calendar so I know what will be the next.
- As a teacher, I want to create a quiz _ FrontEnd
- As a teacher, I want to create a quiz _ BackEnd
- As a teacher, I want to delete a quiz.
- As a teacher, I want to edit a quiz title.
- As a teacher, I want to assign a quiz to a specific student account.
- As teacher, I want to publish a quiz.
- As a teacher, I want to unpublish a quiz.
- As a student, I want to take a quiz with multiple attempt so I can get the highest grade.

Product Backlog

- **PO saves the User Stories** in an area called - **Product backlog** in a project management app [Jira, Rally]
 - Product backlog is a list of user stories and other tasks(bug, etc) need to be done
 - Product backlog owner is PO
 - Product backlog items are listed in **prioritized** order
 - Product backlog constantly change
 - There is only one Product backlog for one project

Jira - Backlog

Backlog 28 issues

-   G03-320 As a user, I want to login
-   G03-298 As a user, I should be able to create vehicles/cars.
-   G03-310 As a user, I should be able to view 3 options under my account name.
-   G03-305 As a user, I should be create Vehicle Contract

Epic


- **Epic** is a **large user story** which can be **divided into smaller user stories**.
- One Epic cannot be completed in a single sprint;
- To complete one Epic takes at least two sprints or more; it depends on a feature's complexity level.
- **PO writes the Epic** and the user stories that belongs to that Epic.

Example Epic from the real project

Epic: Calander feature

As an instructor, I should be able to create, edit, view and delete events on the program calendar calendar.

The blow are the User Stories that belongs to the Calander Epic

- 1694 As an instructor, I should be able change between weekly, monthly, annual and program view ...
- 1698 As a user, I should be able to click on the settings button on the calendar page and modif... 
- 1713 As a user, I should be able to modify First day of the week and Time zone settings on the Cale...
- 1699 As an instructor, I should be able to create regular event
- 1700 As an instructor, I should be able to create synched event
- 1710 As an instructor I should be able to add deadline events
- 1676 Automatically add graded quizzes to Calendar
- 1722 Editing a regular event

Acceptance Criteria (AC)

- AC is written by PO, or BA
- Each user story has at least one AC
- Short, focused **guidance** for the developers & testers to work **on each story** completely

Example User Stories & ACs from the real project

Agile Story

As an instructor, I should be able change **between weekly, monthly, annual** and program view of the calendar

Acceptance Criteria

1. User should be able to click on the **Week** button/option on the right side of the Calendar page and see **Weekly view of the Calendar**
8. User should be able to click on the **Program** button and see the program view of the Calendar.
9. On the program view, first and last day of the year that the program started should be shown on the Calendar page: **Jan 1, 2021**

– Dec 31, 2021



Agile story

As a student, I should be able to access the calendar page

Acceptance Criteria

1. Students should be able to see `Calendar` link on the left navigation menu
2. Students should be able to access the Calendar page by clicking on the *Calendar link*
5. Student should be able to see the following options: `Previous–Today–Next` and `Week–Month–Year–Program`



6. Students should see the *Month view* as the default view.
7. Calendar colors should be the following: Weekdays - `#ffffff`, Weekends - `#f4f4f4`, Selected day - `#e9e9e9`
8. Today's date should be shown on the Calendar page

