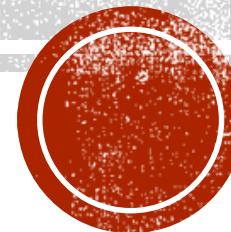


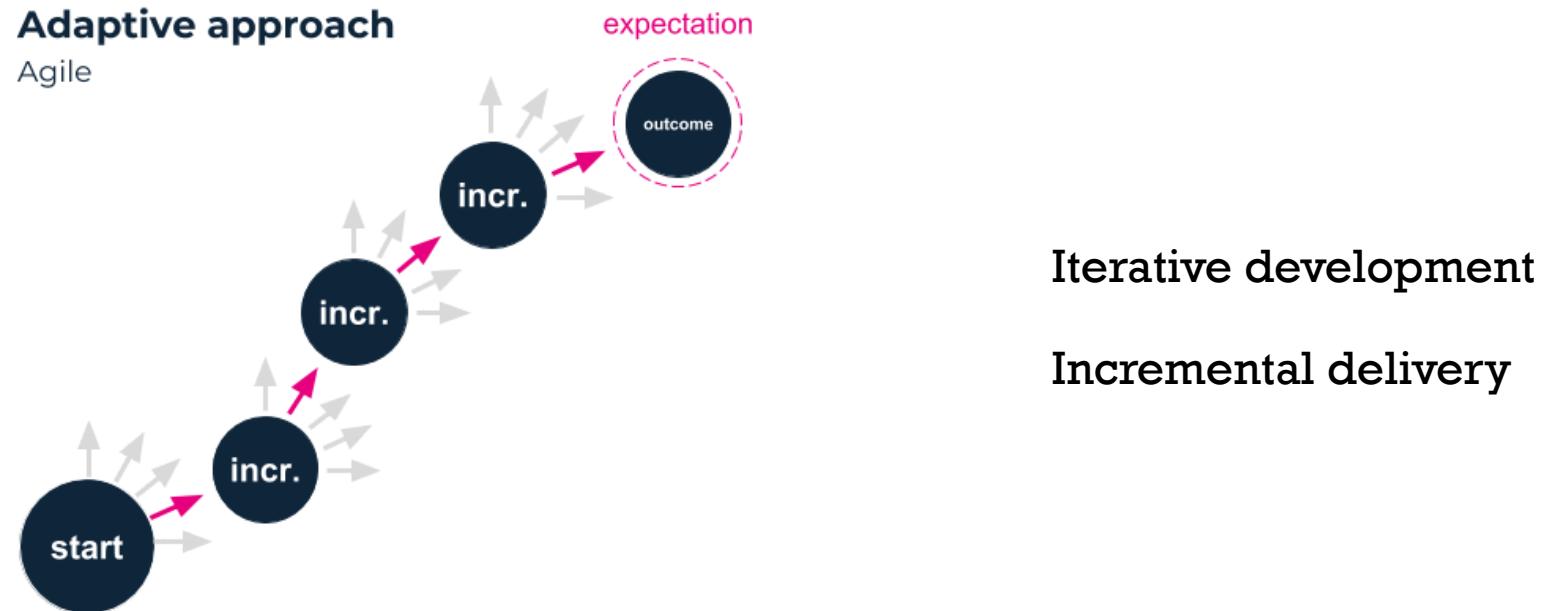
# **Software Engineering**

## **Agile Development**

**Ehsan Sharifi**

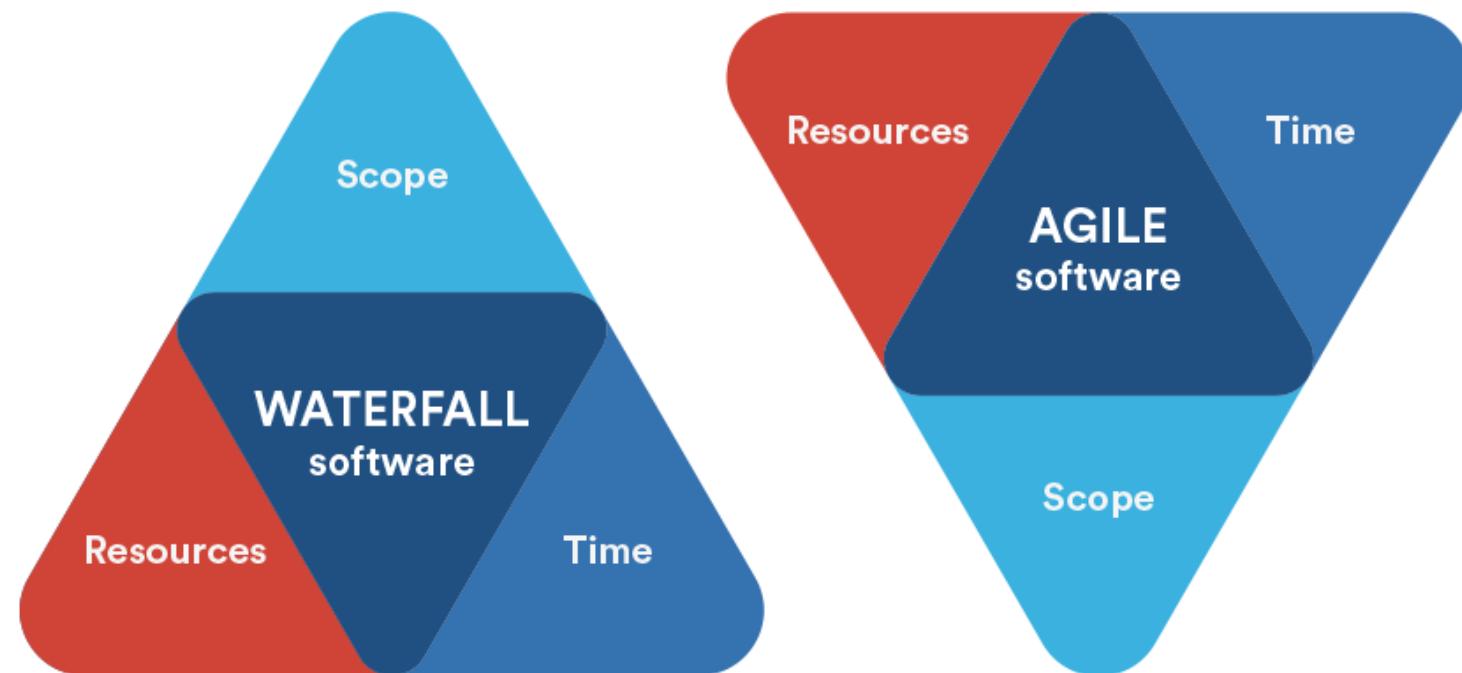


# Agile and Scrum Fundamentals



# Agile and Scrum Fundamentals

Fixed



Estimated



# Agile Manifesto

## Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- |                              |      |                             |
|------------------------------|------|-----------------------------|
| Individuals and interactions | Over | processes and tools         |
| Working software             | Over | comprehensive documentation |
| Customer collaboration       | Over | contract negotiation        |
| Responding to change         | Over | following a plan            |

That is, while there is value in the items on the right, we value the items on the left more.



# Agile Manifesto

در سال ۲۰۰۱ توسط ۱۷ نفر از توسعه دهندگان نرم افزار توافق و مدون شد و بر اساس آن ارزش‌های زیر معرفی شدند:

- اشخاص و تعاملات آنها بر پروشه ها و ابزارها ارجحیت دارند.
- یک نرم افزار قابل استفاده بر مستندات جامع و کامل ارجحیت دارد.
- همکاری و مشارکت مشتری در فرآیند توسعه بر بندهای قراردادی ارجحیت دارد.
- پذیرش تغییرات بر پیروی از یک برنامه از پیش تعریف شده ارجحیت دارد.



# The Agile Principles

## *The 12 agile principles\**

### 1 Satisfy the **customer**



### Welcome **change**



### Deliver **frequently**



### 4 Work together



### 5 Trust and **support**



### Face-to-face **conversation**



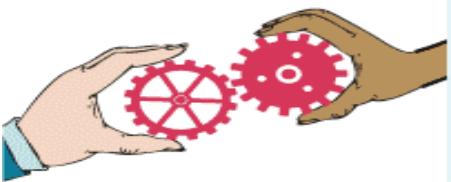
### Working **software**



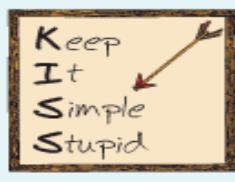
### 8 Sustainable **development**



### 9 Continuous **attention**



### 10 Maintain **simplicity**



### 11 Self-organizing **teams**



### 12 Reflect and **adjust**



Copyright © 2016 Knowledge Train Limited. www.knowledgetrain.co.uk \*Quoted from www.agilemanifesto.org



# The Agile Principles

۱. رضایت مشتری از طریق تحویل سریع نرم افزار دارای ارزش؛
۲. استقبال از تغییرات نیازمندی‌ها، حتی در اواخر توسعه؛
۳. نرم افزار قابل استفاده زود به زود تحویل مشتری می‌شود (از چند هفته تا چند ماهه با ترجیح دوره‌های کوتاه‌تر)؛
۴. همکاری نزدیک و روزانه بین افراد کسب‌وکار و تیم توسعه؛
۵. پروژه‌ها حول افراد بانگیزه، که باید به آنها اعتماد کرد، شکل می‌گیرند؛
۶. مکالمه رو در رو بهترین شکل ارتباطات است؛
۷. نرم افزار قابل استفاده مقیاس اصلی پیشرفت است؛



# The Agile Principles

۸. فرآیندهای اجایل توسعه پایدار را ارتقا می دهند. اسپانسرها، توسعه دهندگان و کاربران بایستی قادر به حفظ پایداری سرعت توسعه باشند؛
۹. توجه مستمر به برتری فنی و طراحی خوب چابکی را ارتقا می دهد؛
۱۰. سادگی - هنر به حداقل رساندن کارهای انجام نشده - ضروری است؛
۱۱. بهترین معما ری ها، نیازمندی ها و طراحی ها از دل تیمهای خودسازمانده بیرون می آیند؛
۱۲. انطباق با تغییرات توسط تیم به طور منظم.



# What's Scrum?

- A framework within which people can address complex adaptive problems
- ریشه لغوی: به شروع دوباره در بازی راگبی آمریکایی اسکرام گفته می شود



# What's Scrum?

- Scrum is:
  - Lightweight
  - Simple to understand
  - Difficult to master

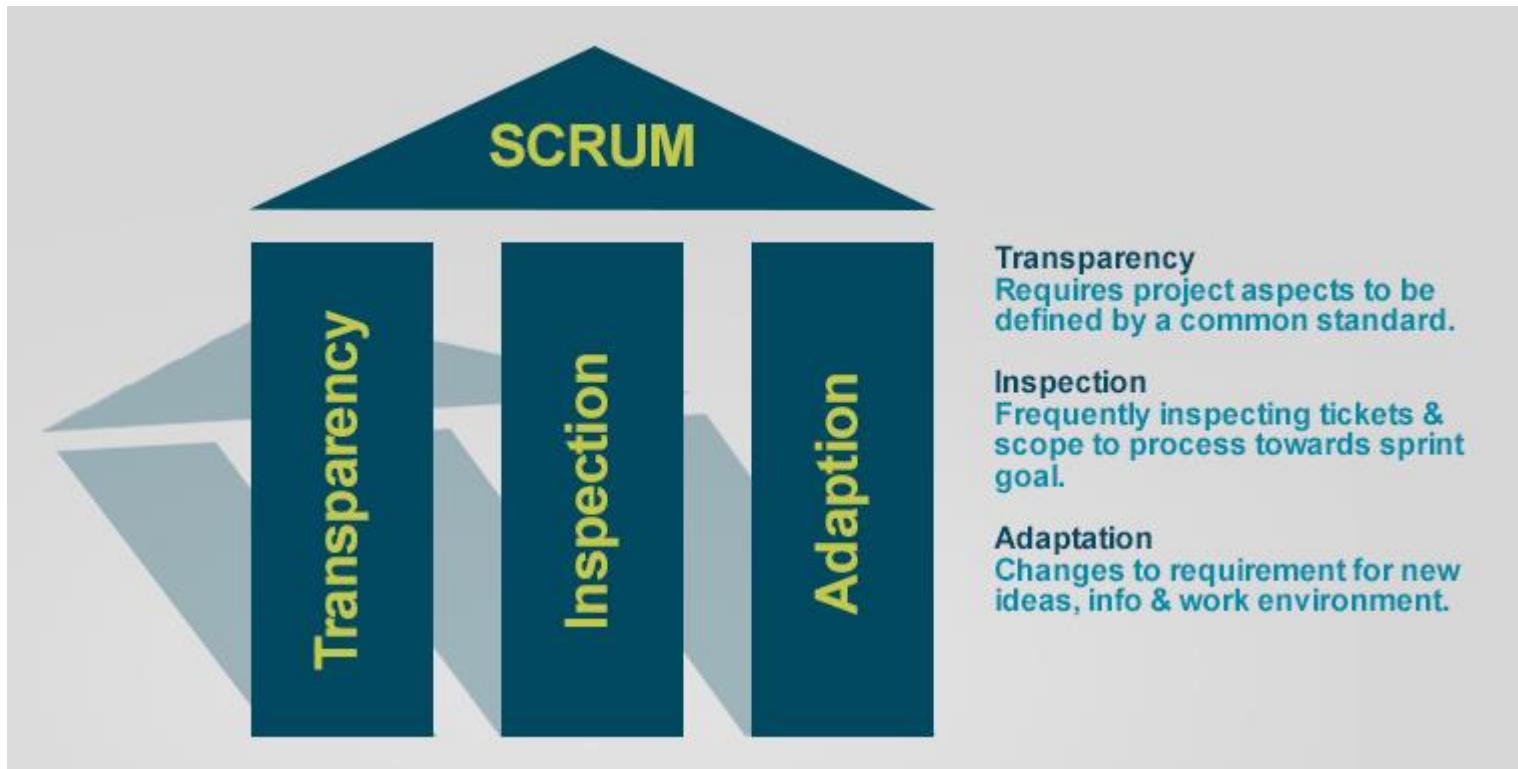


# Scrum Theory

- Scrum is founded on empirical process control theory, or empiricism.
  - تجربه گرایی فرض می کند که دانش از تجربه می آید و تصمیمات بر اساس آنچه اتفاق افتاده است اخذ می شود



# Scrum Pillars



# Scrum Pillars

## شفافیت

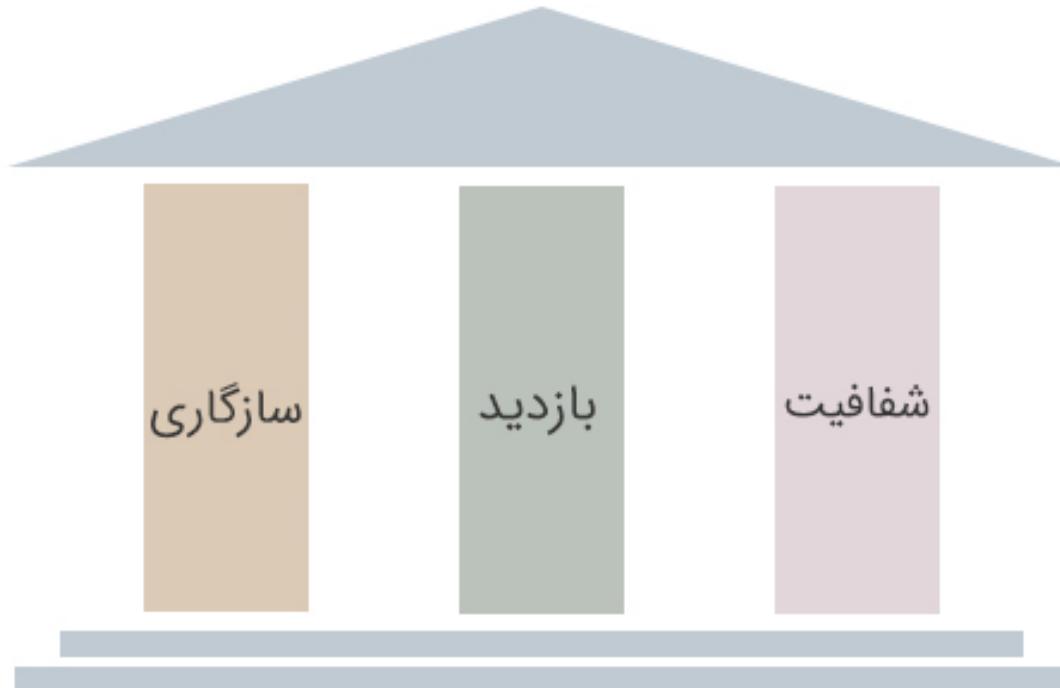
تمام جنبه‌های مختلف فرآیند که بر خروجی آن اثرگذارند، باید برای افرادی که فرآیند را کنترل می‌کنند، کاملاً مشهود و قابل دیدن باشند.

## بازدید

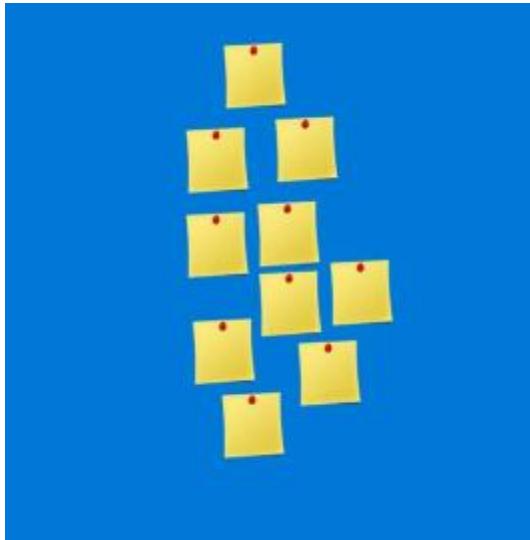
جنبه‌های مختلف فرآیند، بایست طی بازه‌های زمانی مشخص مورد بررسی قرار گیرند تا انحرافات آن قابل تشخیص باشد.

## سازگاری

تنظیم و تعدیل فرآیند در سریع‌ترین زمان ممکن به منظور جلوگیری از انحرافات بیشتر



# Starting up the Project



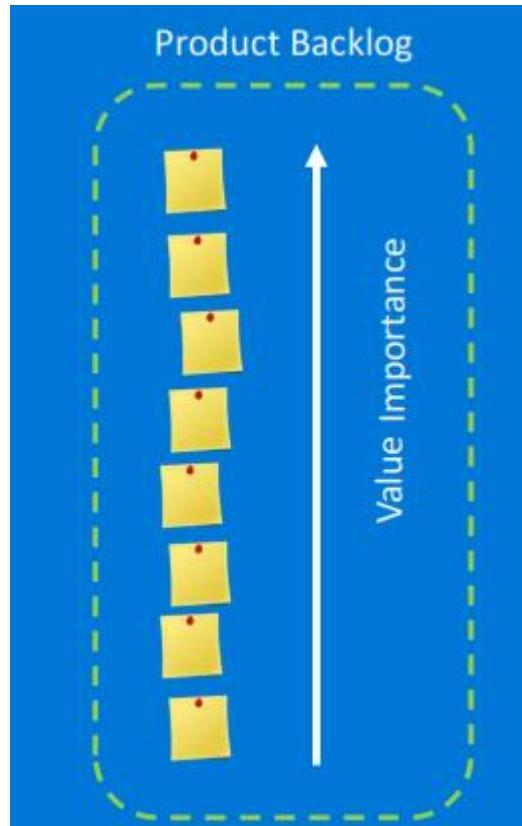
۱- کارهایی که مهمتر هستند را روی یک برگه یا استیکی نوت می نویسیم و به دیوار می چسبانیم

۲- چون از رویکرد تطبیقی استفاده می کنیم همه کارهای پروژه را مشخص نمی کنیم و فقط ایده ها و قابلیتهای اولیه که می دانیم مورد انتظار مشتری است را مدون می کنیم.

۳- کارهای مشخص شده بصورت تصادفی برای اجرا انتخاب نمی شوند بلکه بر اساس اهمیت مرتب می شوند

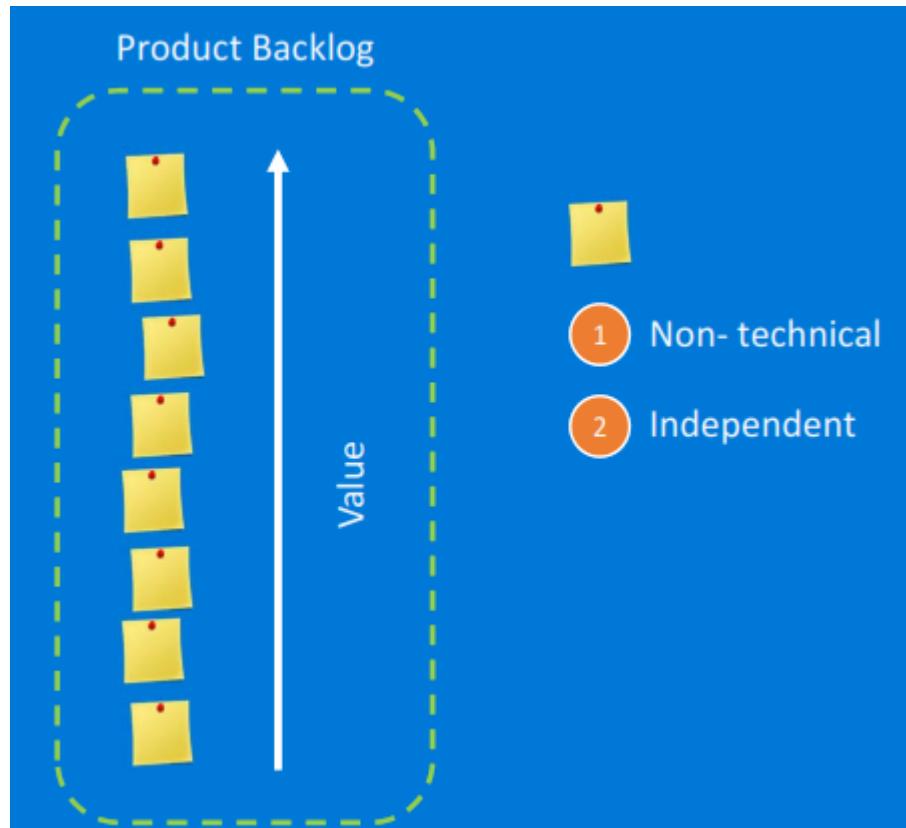


# Starting up the Project



- تولید محصول به ترتیب اهمیت فیچرهای تضمین می کند که فیچرهای با ارزش تر زودتر به دست مشتری برسد و رضایت او را جلب کند.
- از طرف دیگر تولید فیچرهای با ارزش تر بازخوردهای با ارزش تری هم ایجاد می کند که منجر به تطبیق بهتر مسیر با نیازمندیهای مشتری می گردد.
- به این ساختار لیست مرجع نیازمندیها گفته می شود. لیستی از آیتمها که بر اساس اهمیت مرتب شده اند.
- چگونه می توان اهمیت یک آیتم را مشخص نمود؟ بهترین پاسخ ارزش تجاری هر آیتم می باشد.
- فاکتورهای مختلفی در تعیین ارزش تجاری هر آیتم وجود دارند که طبق کتاب آقای مایک کوهن می توان به موارد زیر اشاره کرد:  
**Financial Value – Cost – New Knowledge – Risk**
- ساده ترین راه بر اساس میزان بازگشت سرمایه است.

# Starting up the Project



هر آیتم در لیست باید دارای دو مشخصه باشد.

1. باید غیر تکنیکال باشد تا بتوان از طریق آن براحتی با مشتری ارتباط برقرار کرد.
2. باید مستقل از هم باشند تا آزادانه بتوان آنها را در لیست سورت نمود.

چه زمانی کارهای تکنیکال باید انجام شوند؟ زمانی که به اولین فیچری که به آن نیاز دارد رسیده باشیم آن کار تکنیکال انجام می شود.

معماری نرم افزار و رابط کاربری چگونه عملی می شود؟ بصورت تدریجی و به اندازه نیاز هر فیچر انتخاب شده، ایجاد و تکمیل می شود.



# Product Backlog Item

## Product Backlog Item

1 Non- technical

2 Independent

- A feature to provide a report of the past transactions.
- Adding the possibility for the system administrator to block certain users.
- ✗ Installing and setting up the database.
- ✗ Preparing development environments.
- ✗ Preparing UML diagrams.
- ✗ Preparing the solution architecture.
- ✗ Adding a feature to send monthly newsletters.
- ✗ Prepare layout and colors for the user interface.



# Product Backlog Item

Product Backlog Item

1

Non- technical

2

Independent



As a ..., I want to ...[, so that ...].



User Story

As a customer, I want to reset my password.

As the system admin, I want to reset certain users' password.



# Product Backlog Item

Product Backlog Item

1 Non- technical

2 Independent

- ✓ A feature to provide a report of the past transactions.



As a user, I want to receive a report of my past transactions, to make sure there are no mistakes.



# Product Backlog Item

Product Backlog Item

1 Non- technical

2 Independent



Adding the possibility for the system administrator to block certain users.



As the admin, I want to block certain users, to prevent fraud.



# Product Backlog Item

Product Backlog Item



Non- technical



Independent



Non-Functional Features

Performance, Security, Maintainability, Scalability, ...

- Add them to Definition of Done
- Add as an item to Product Backlog

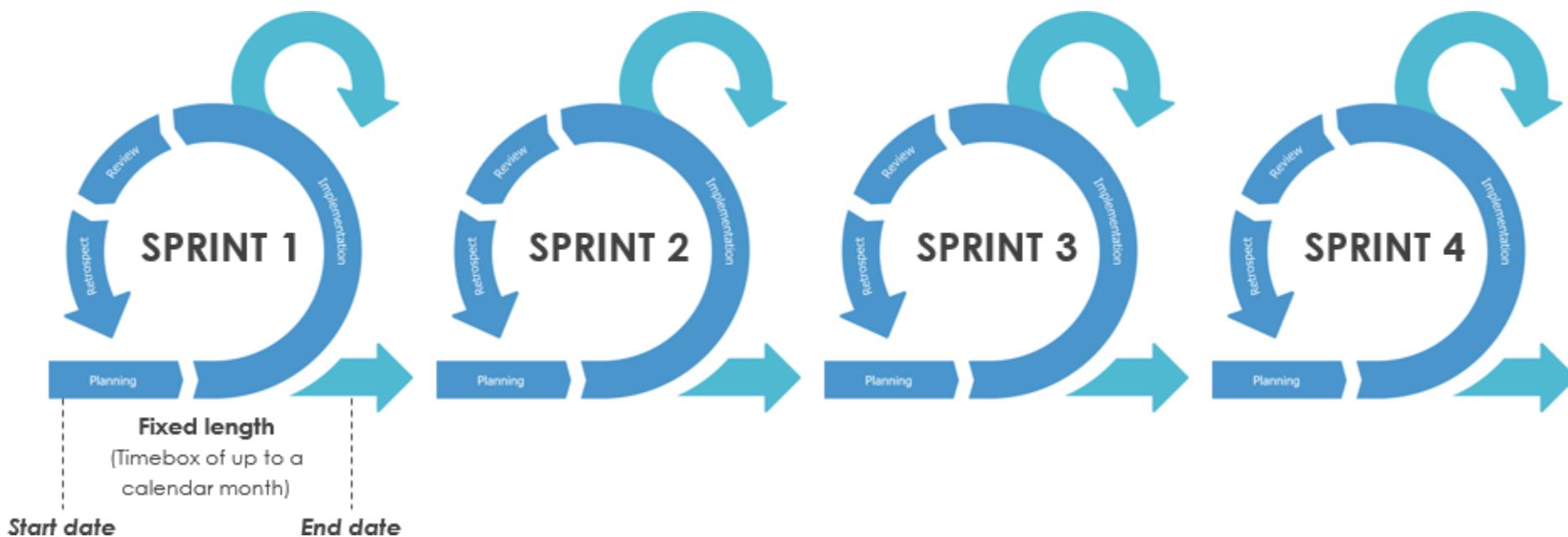
As a user, I want to receive such and such in maximum 3 seconds.



# Sprint

“Sprint” is the Scrum term for “iteration”

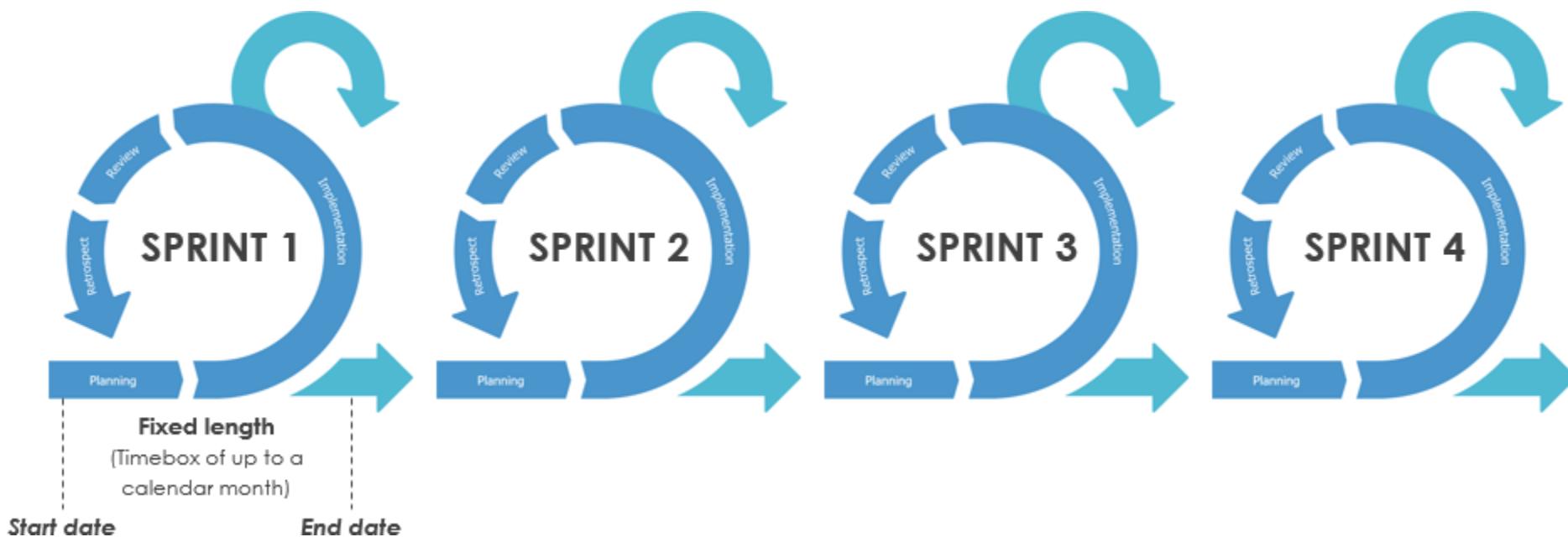
Timeboxed: A timeframe that has a maximum duration



# Sprint

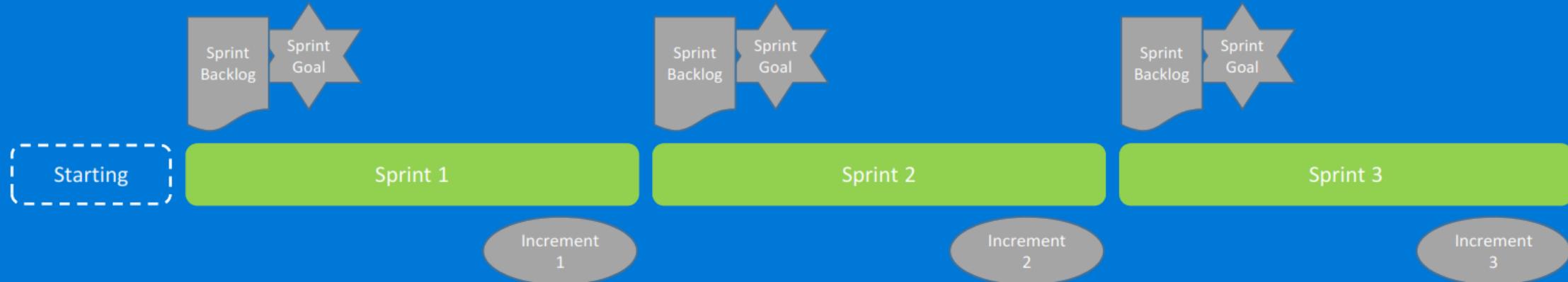
Parkinson's Law / Student Syndrome

Maximum duration of a Sprint is 1 month





## Product Backlog



Each Sprint goal is:

To create a piece of working software

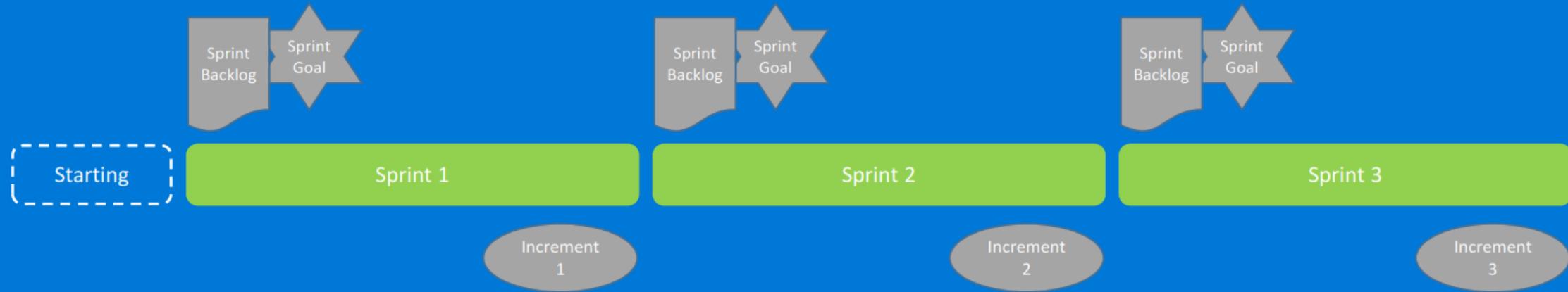
Each Sprint's output is an Increment

Each Increment is a different version of a software and includes all previous increments.





## Product Backlog



## Releasing

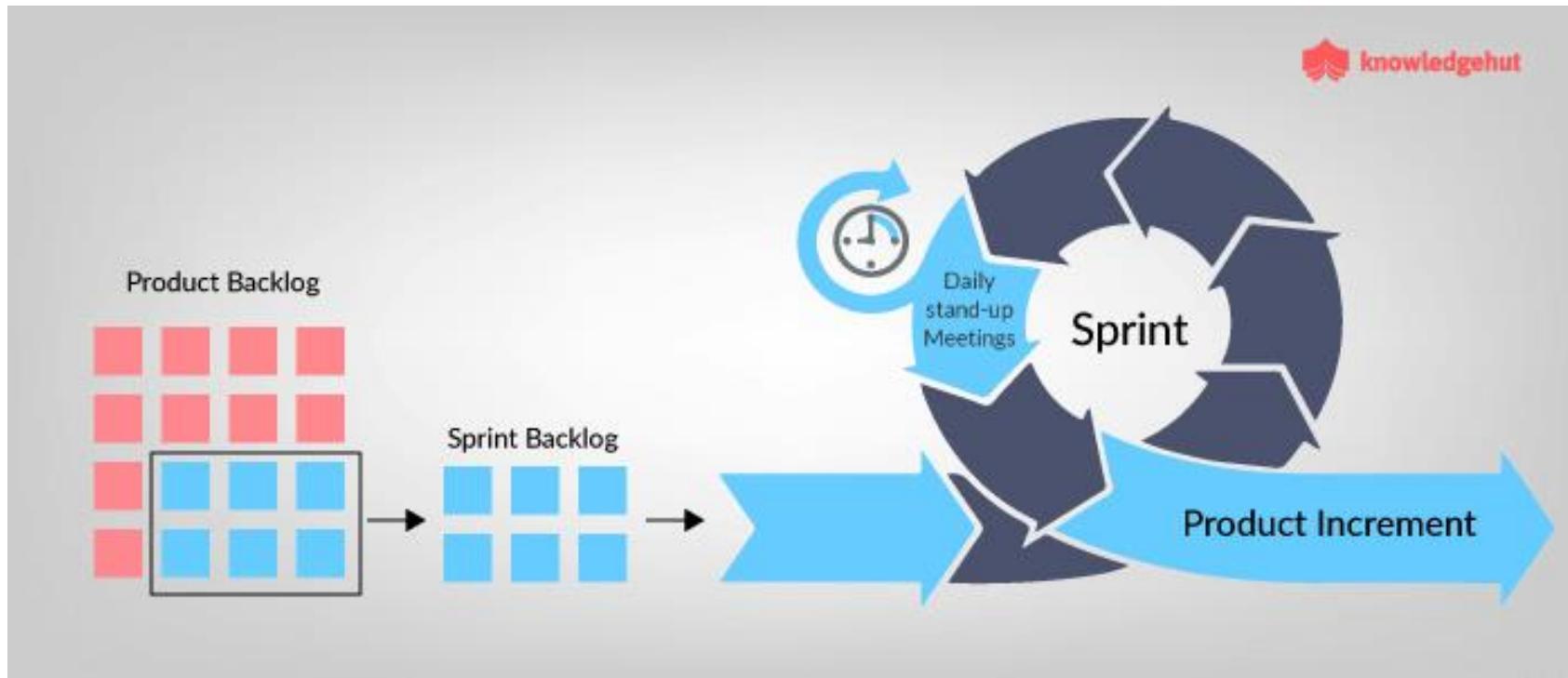
- It doesn't need to release each increment at the end of each sprint.
- But all increments should be potentially shippable and releasable.
- Usually occur every few sprints.



# **SPRINT PLANNING**

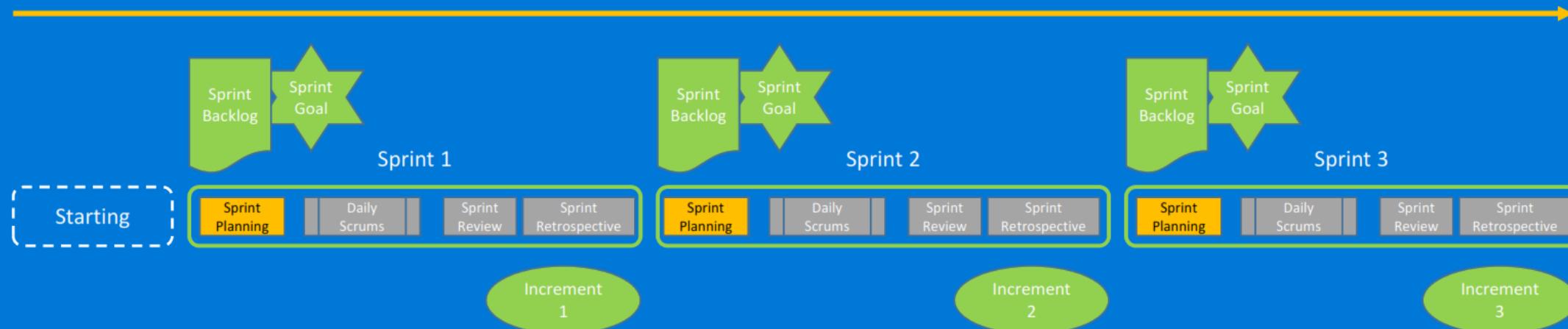


# Sprint Planning





## Product Backlog



## Sprint Planning

To create the Sprint Backlog and Sprint Goal

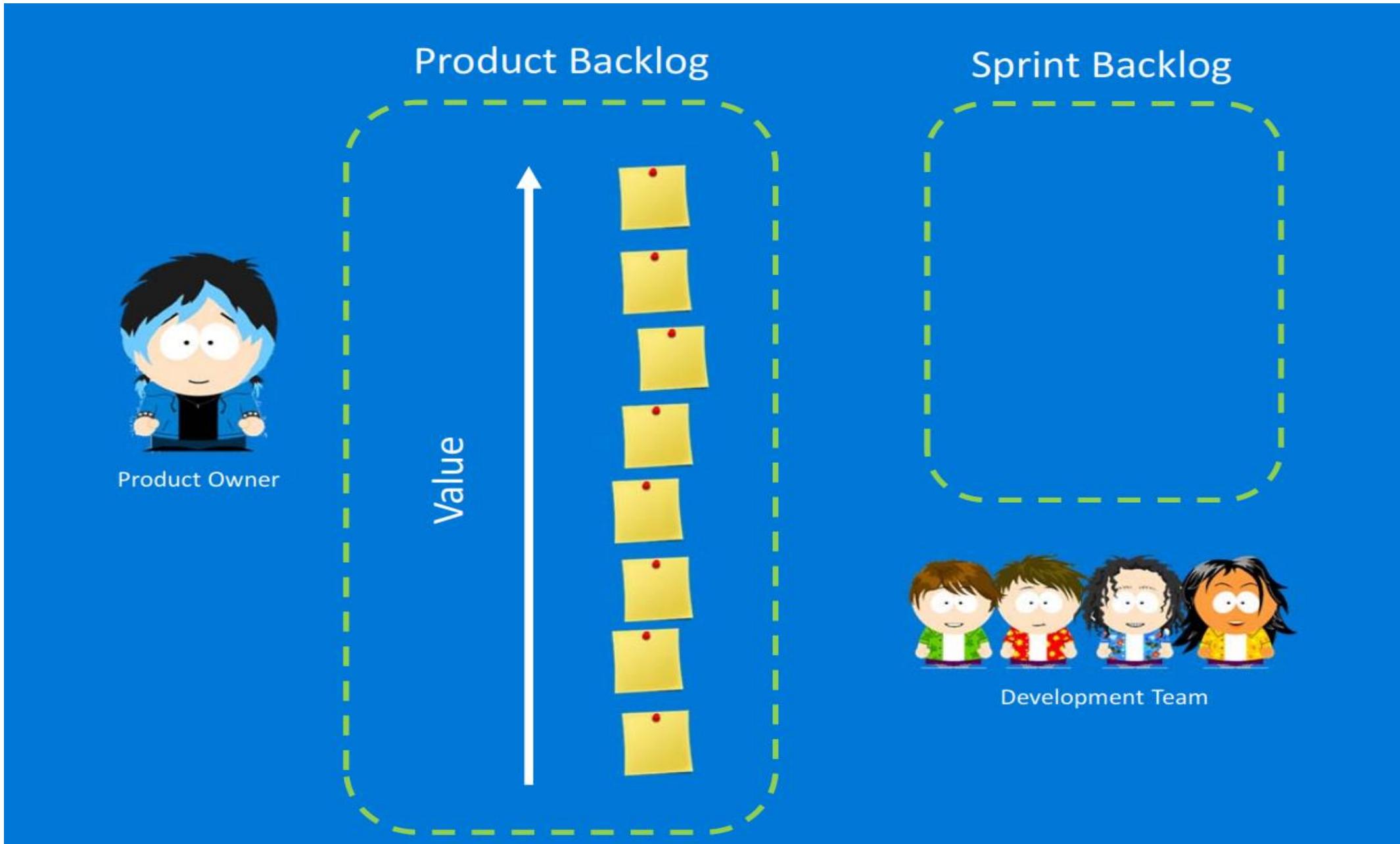
Timeboxed

8 hours in a one-month Sprint and shorter proportionally for shorter Sprints

Duration Example

2-week Sprint => 4 hours Sprint Planning duration





## Story Point

Relative effort-based unit

## Ideal Days

Pure time, not elapsed time

Sprint 1:

Sprint Backlog



20

Sprint 2:

Sprint Backlog



15

Sprint 3:

Sprint Backlog



19

Average amount of completed work during Sprints

18 units per Sprint.

It's called **Velocity**

We can say: "Our Velocity is 18 units per Sprint".



The development team owns the Sprint Backlog



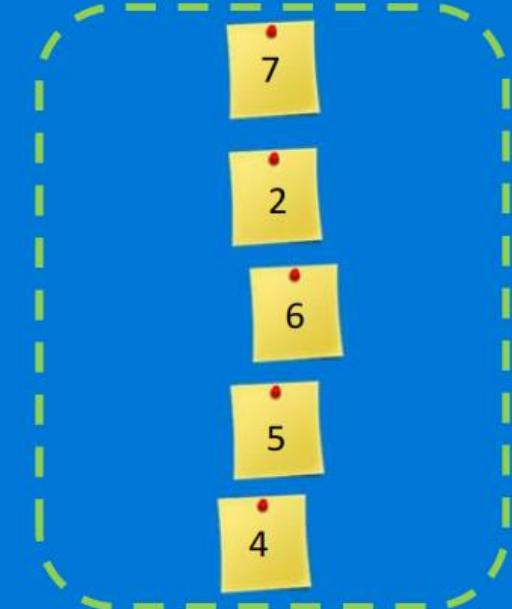
Product Owner

## Product Backlog

Value



## Sprint Backlog



Development Team



The development team owns the Sprint Backlog

Product Backlog



Sprint Backlog



چه اتفاقی می افتد اگر تیم توسعه تواند همه کارهای انتخاب شده را انجام دهد؟

>



Development Team



# The Scrum Team



Product Owner



Scrum Master



Development Team

1 person

Full-time or part-time

Business oriented

1 person

Full-time or part-time

Scrum coach and facilitator

3 to 9 people

Full-time (recommended)

Specialist



# The Scrum Team



Product Owner



Scrum Master



Development Team

If you want to have more developers than 9, you should have more teams and we call it  
**Scaled Scrum**

Question: How many Product Owners, Scrum Masters and Development Teams do we need in multiple teams (Scaled Scrum)?





Product Owner



Scrum Master



Development Team



Product Owner



Scrum Master



Development Team



Product Owner

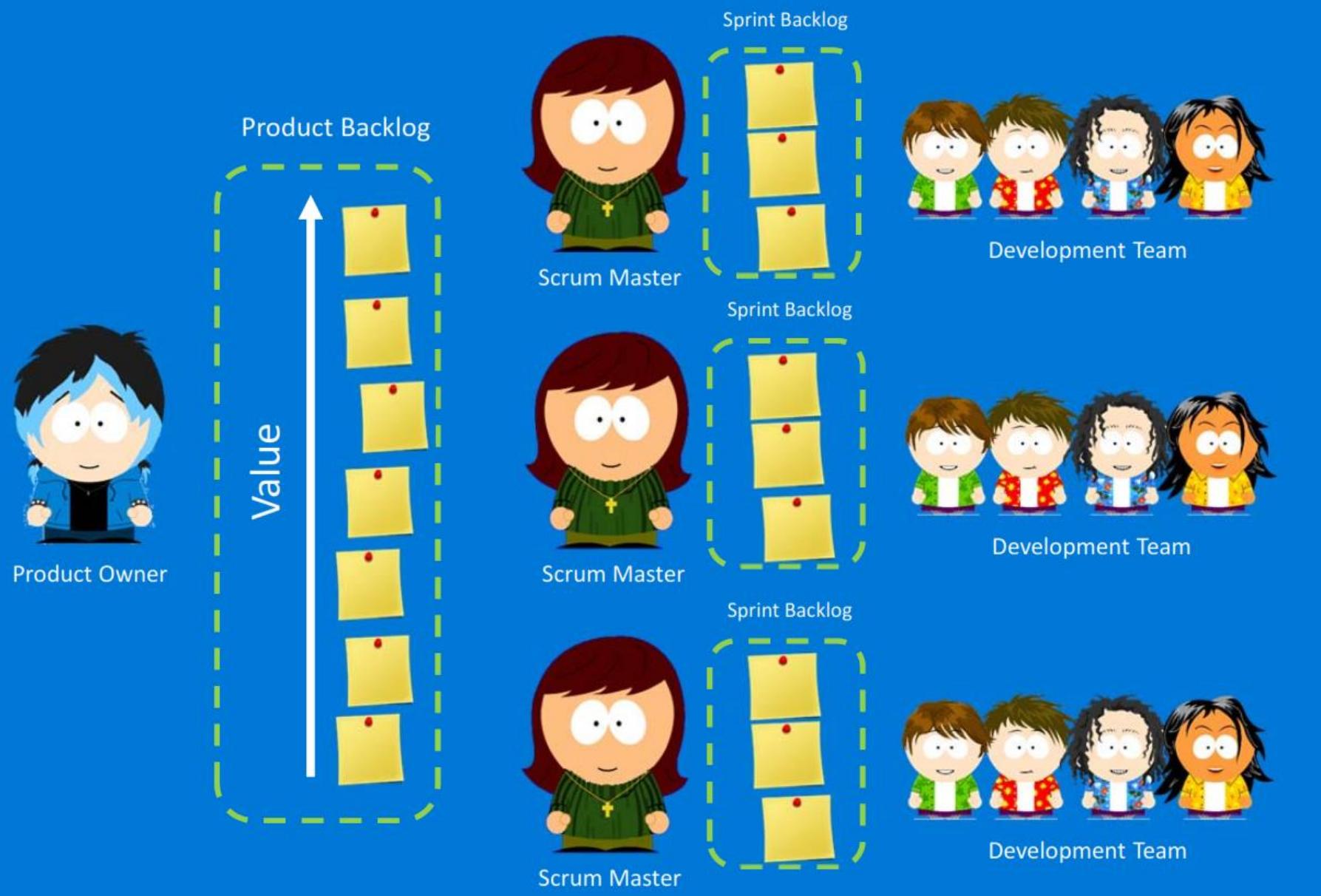


Scrum Master

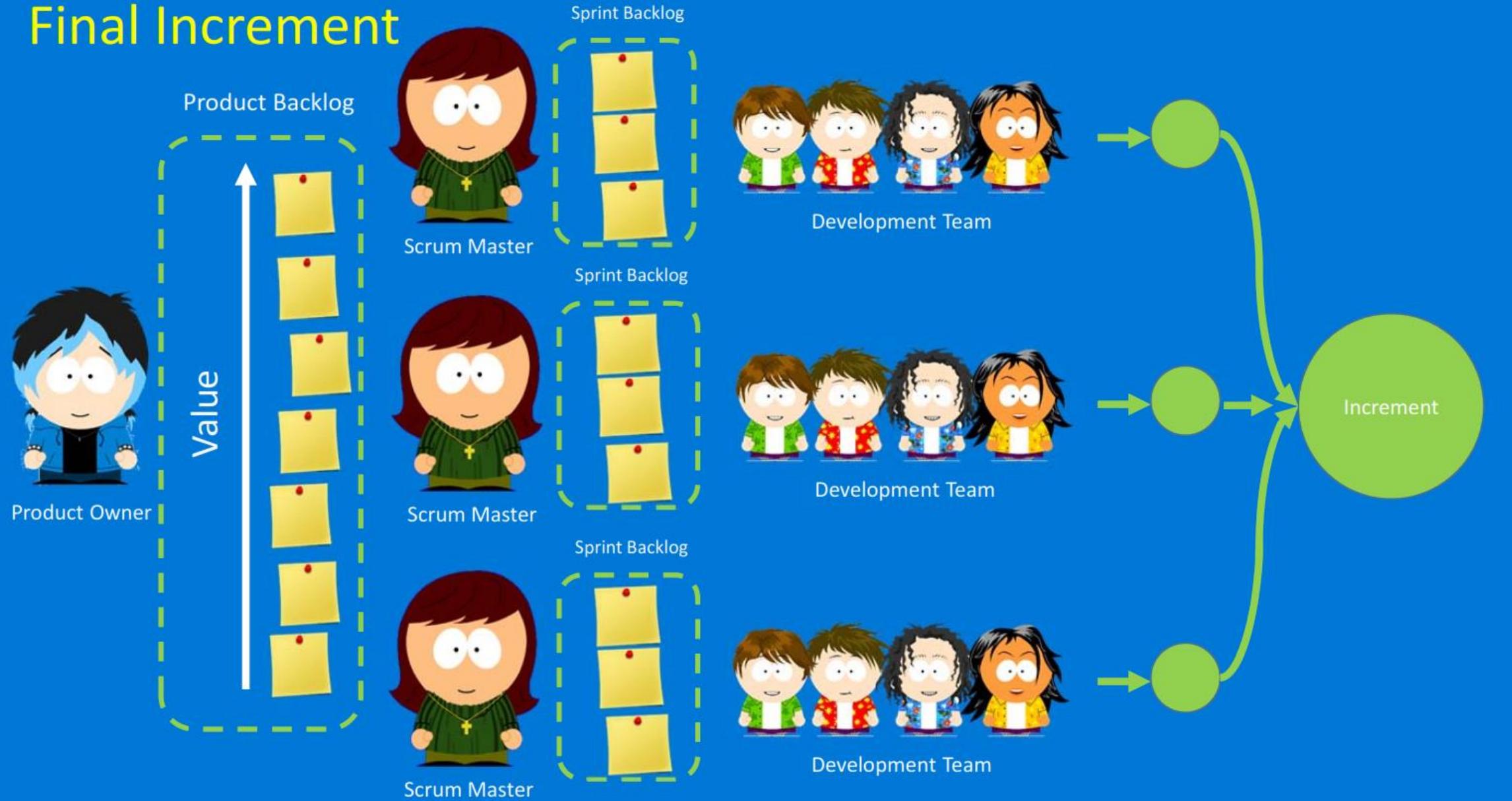


Development Team





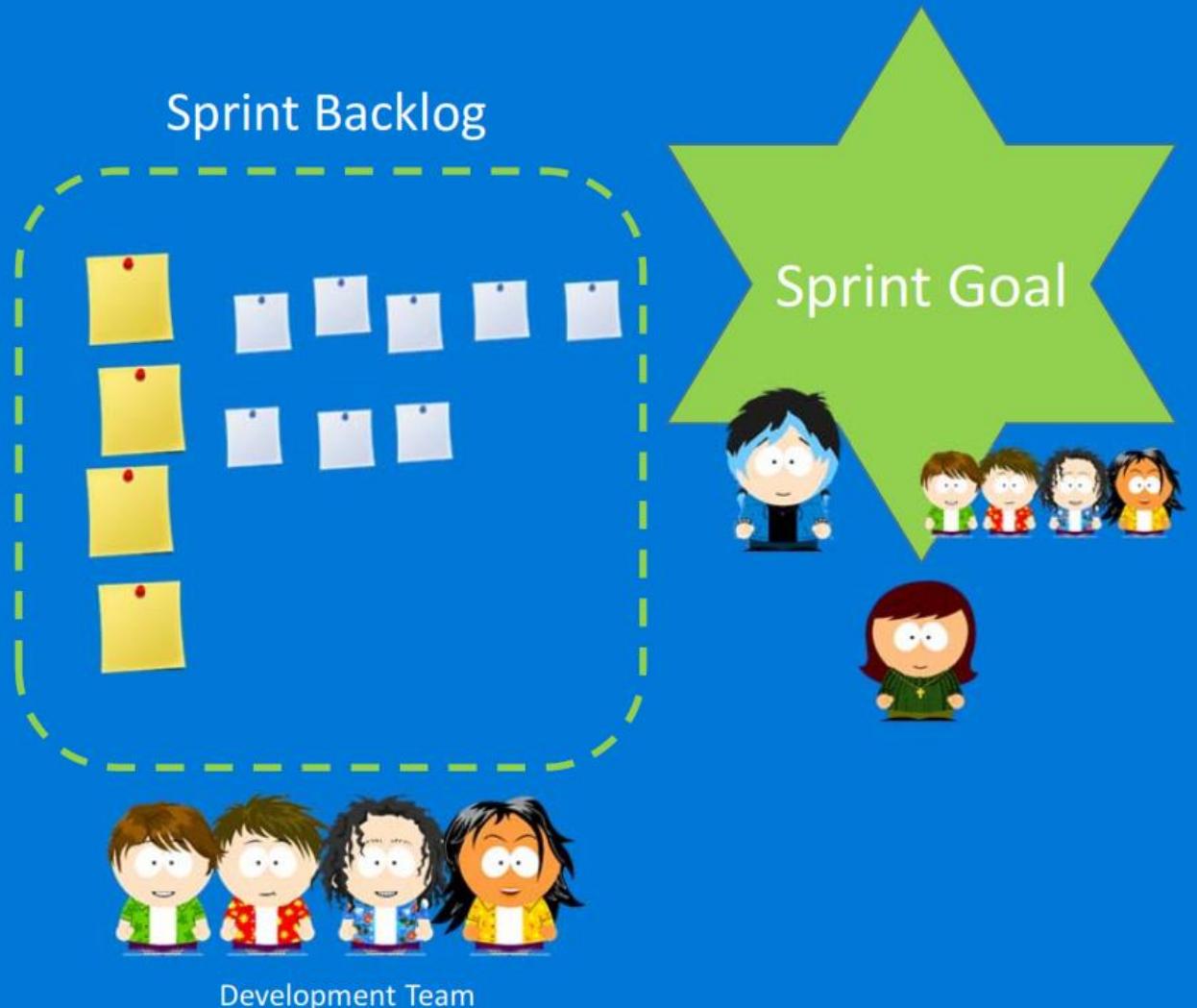
# Final Increment



# Sprint Backlog Elements

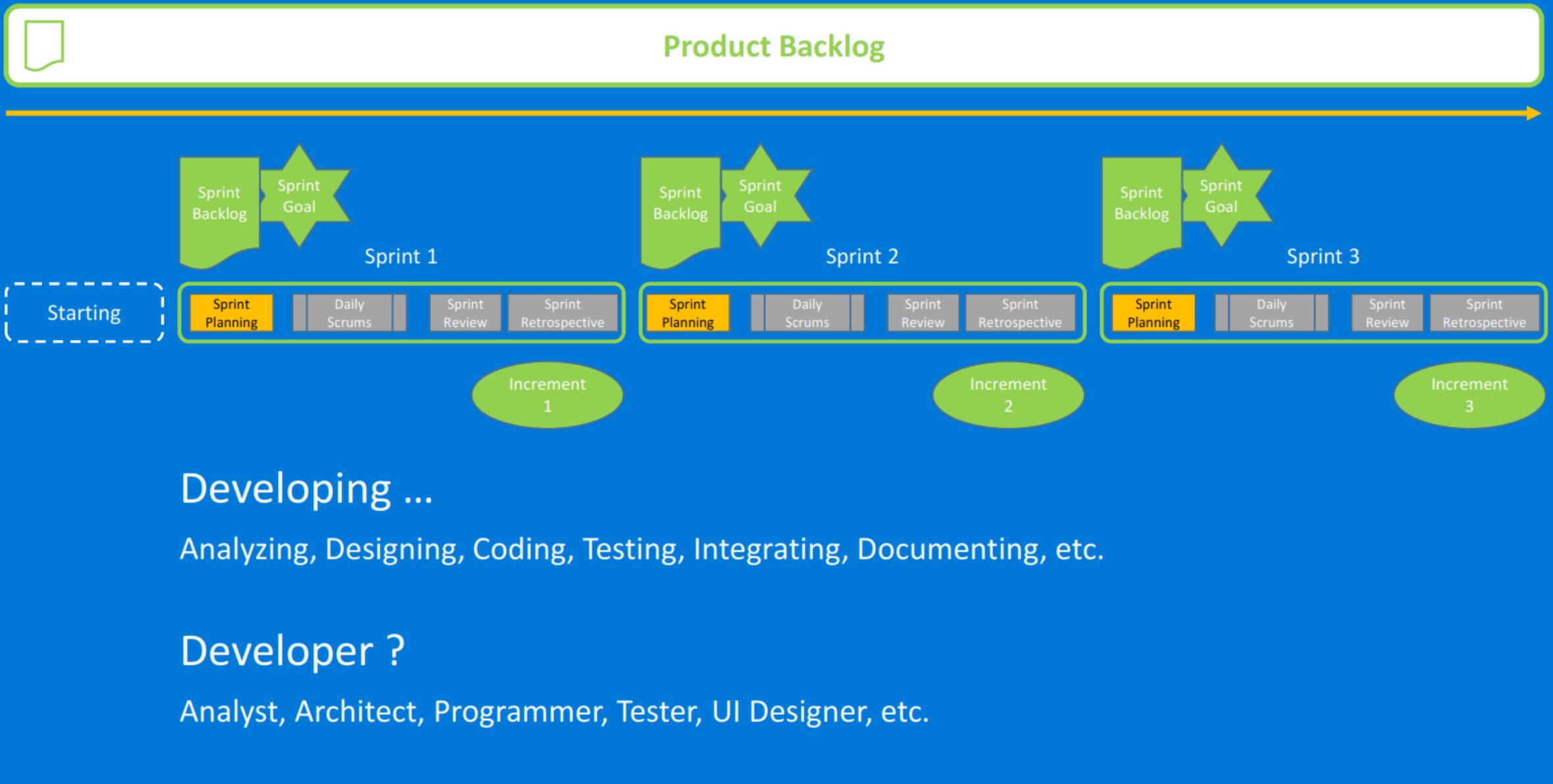
In the Exam:

- Product Backlog Item, PBI, Item, User Story
- Work, Task

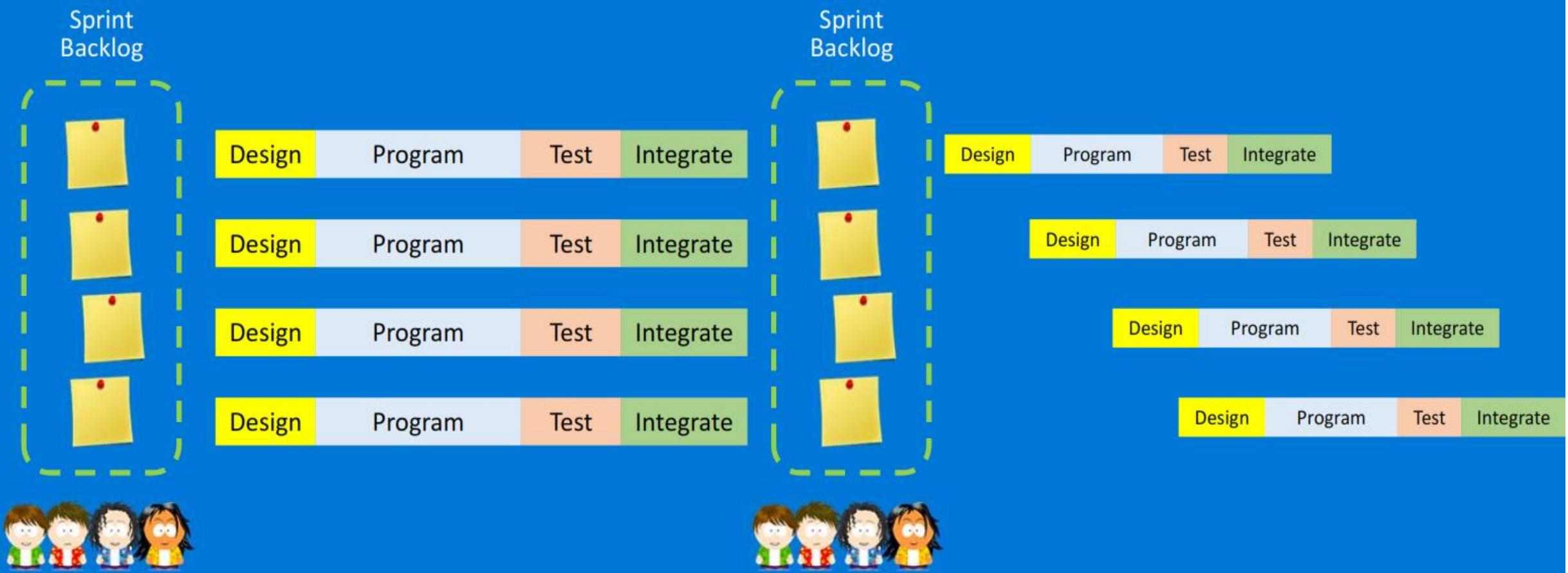


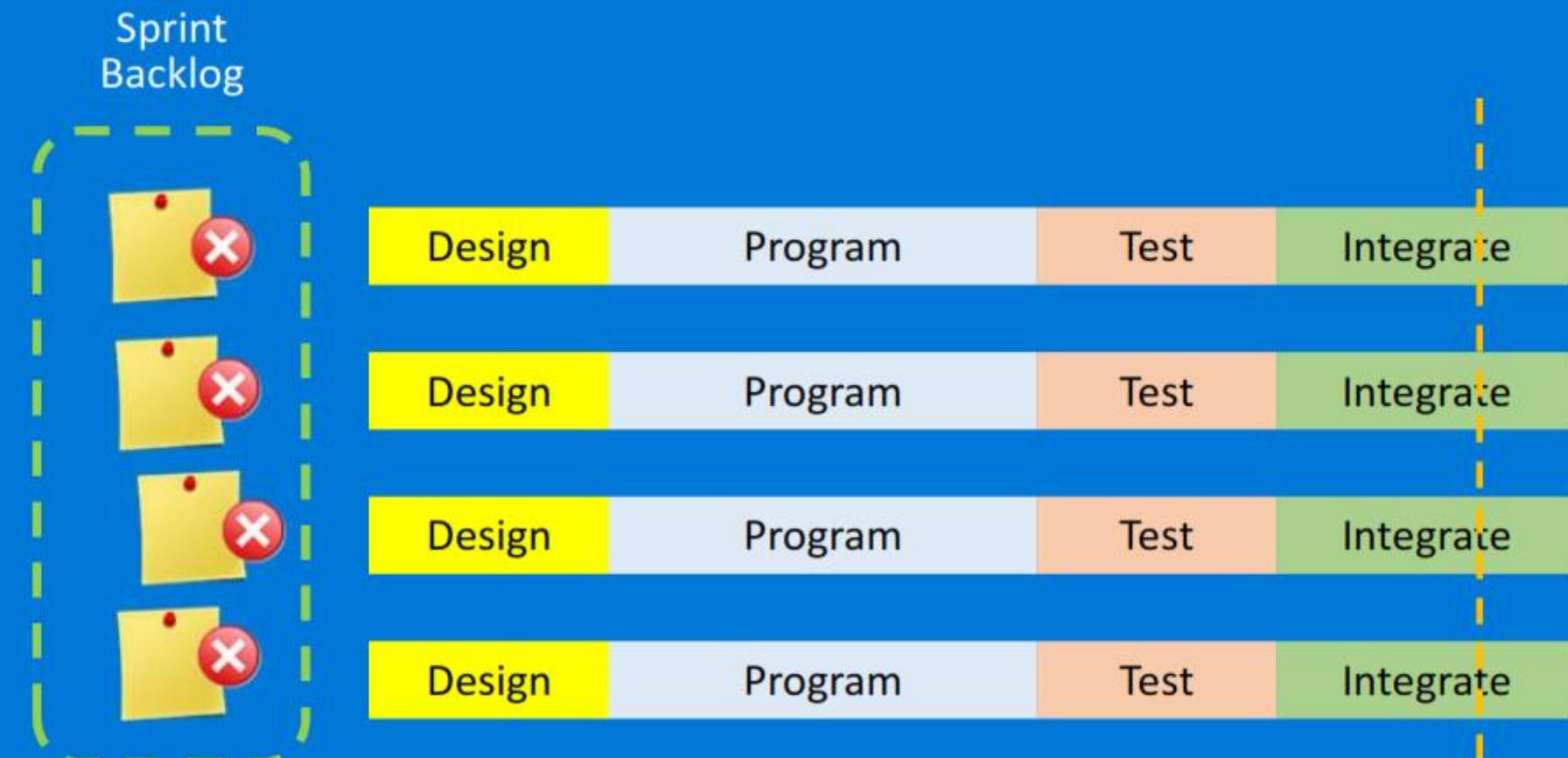
# DURING THE SPRINT

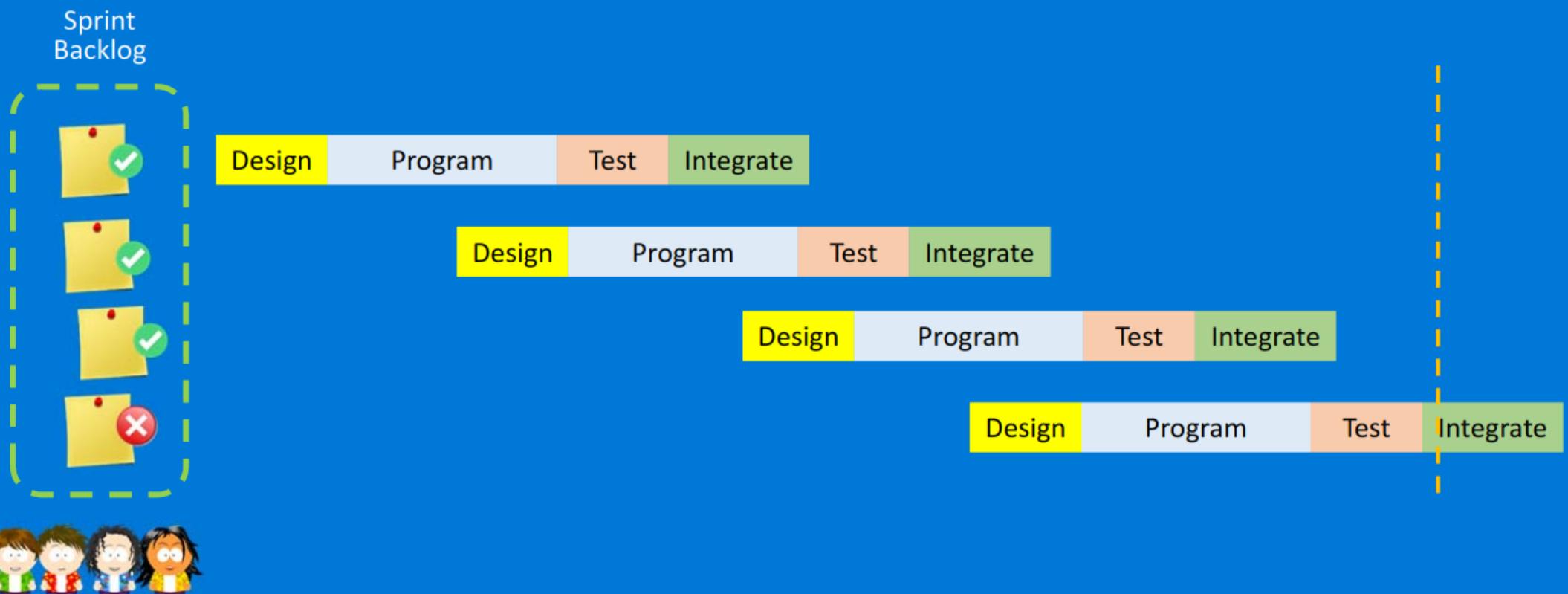




# Which ?



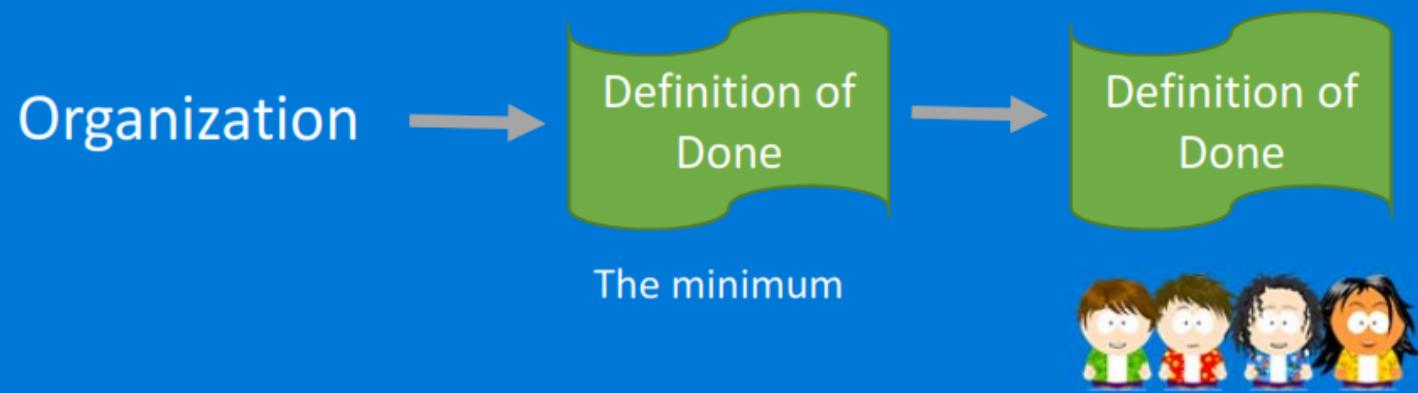




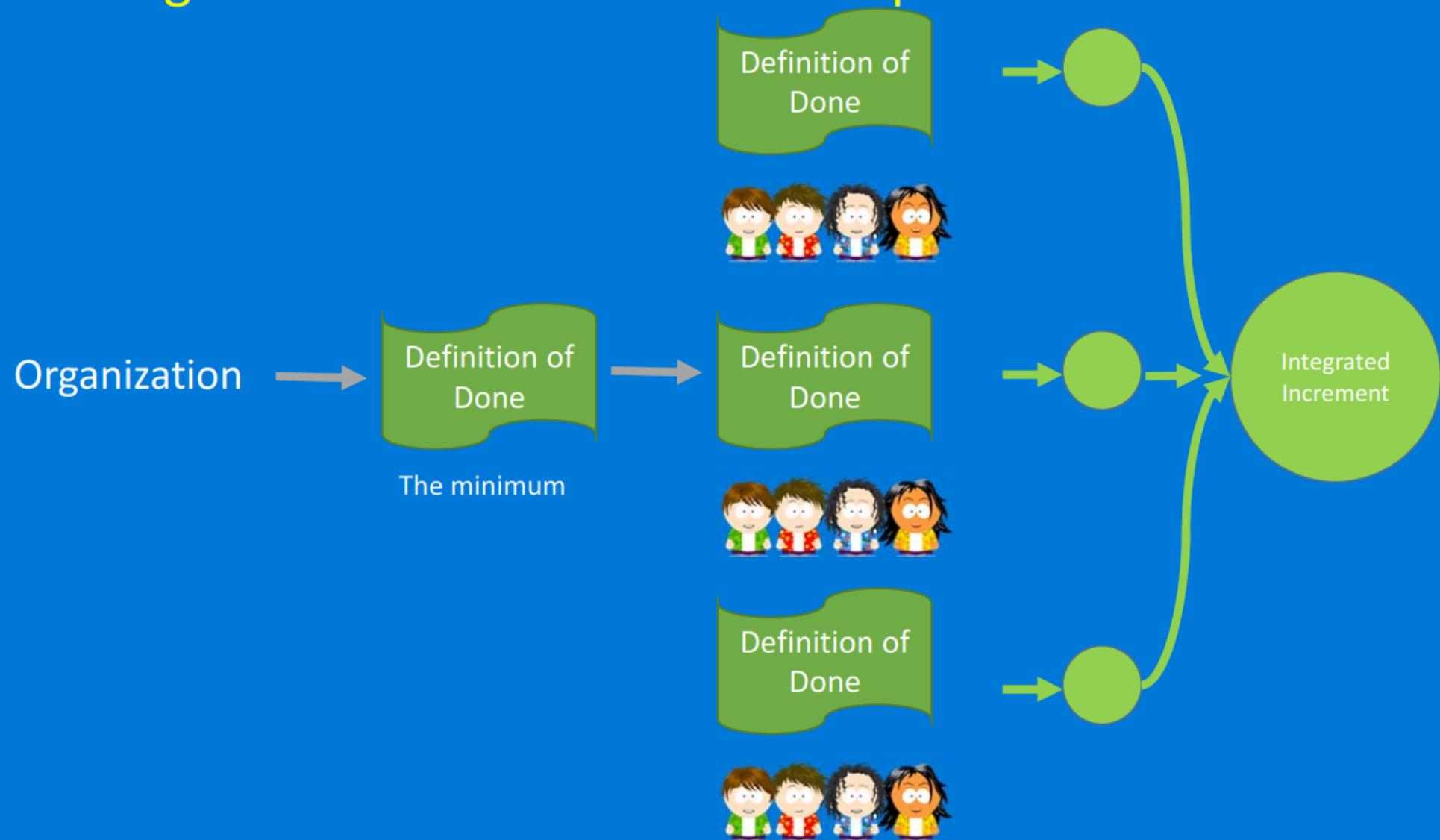
# **DEFINITION OF DONE**



# Creating Definition of Done



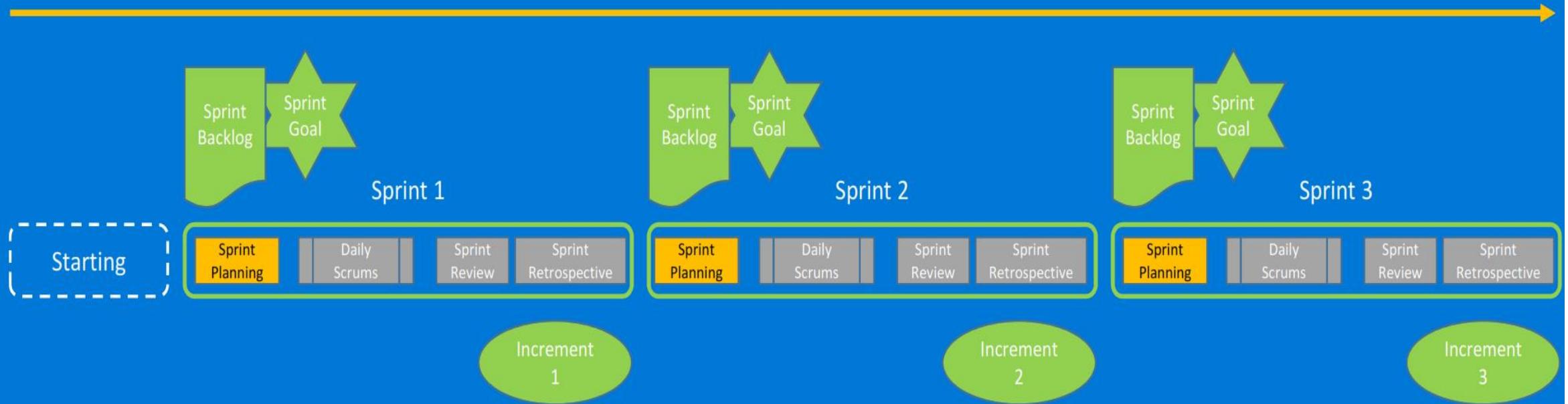
# Creating Definition of Done in a multiple teams environment



# Measuring Sprint Performance



## Product Backlog

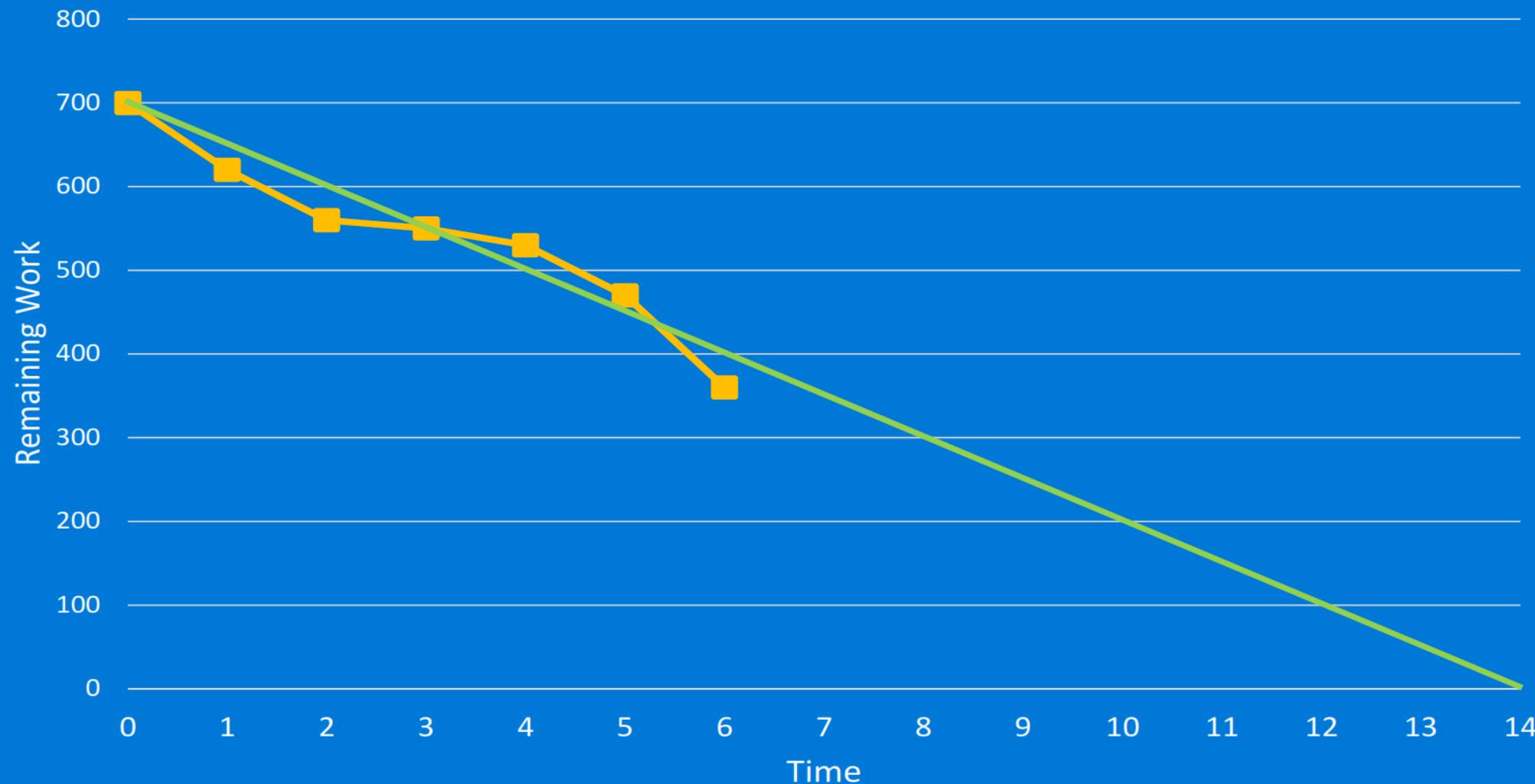


## Measuring Sprint Performance

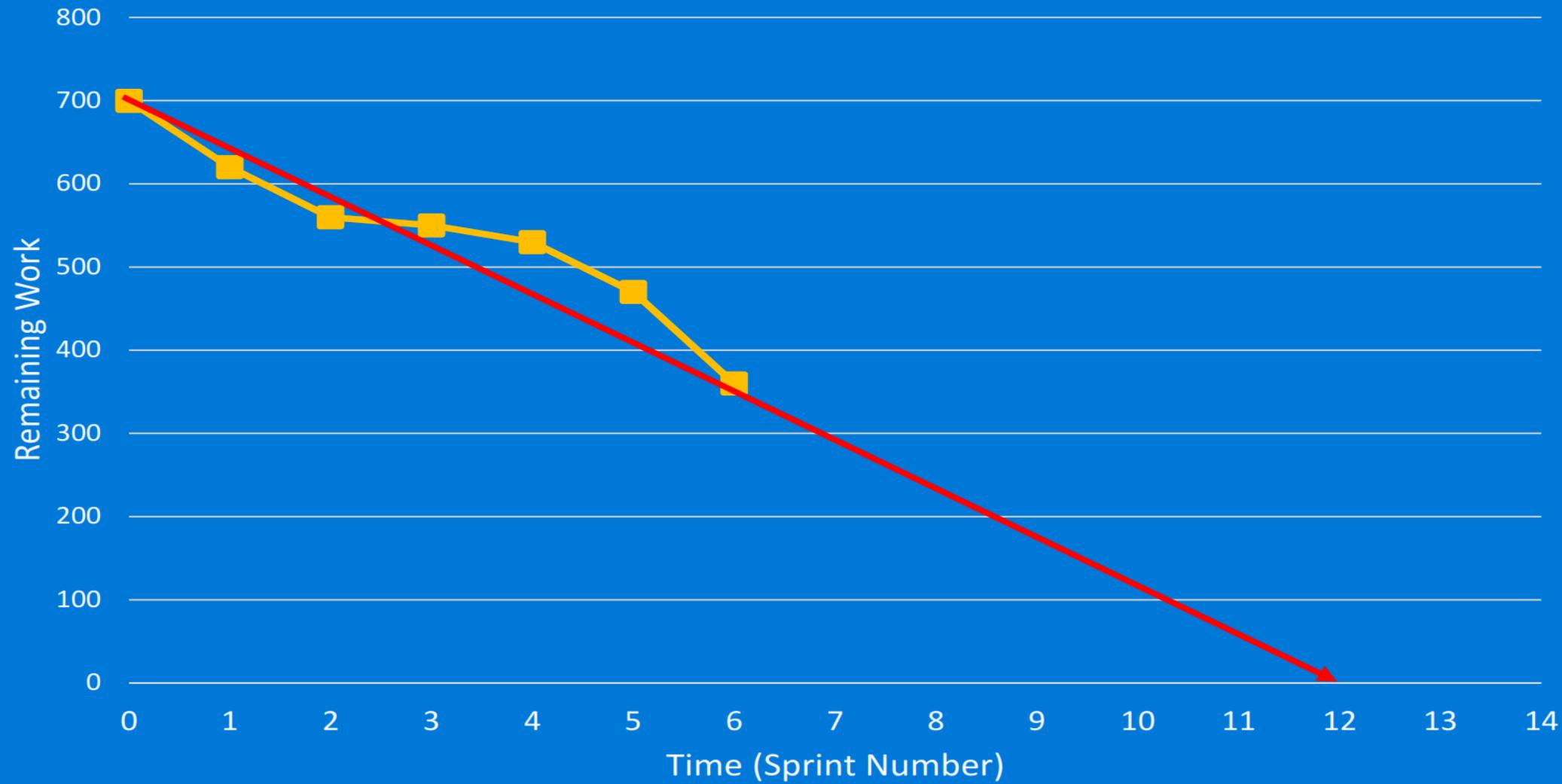
Usually by Burn-down chart



# Sprint Burn-down Chart



# Project Burn-down Chart



## Measuring progress of the **Project**



Product Owner

At least once per Sprint

## Measuring progress of the **Sprint**



Development Team

At least once per day

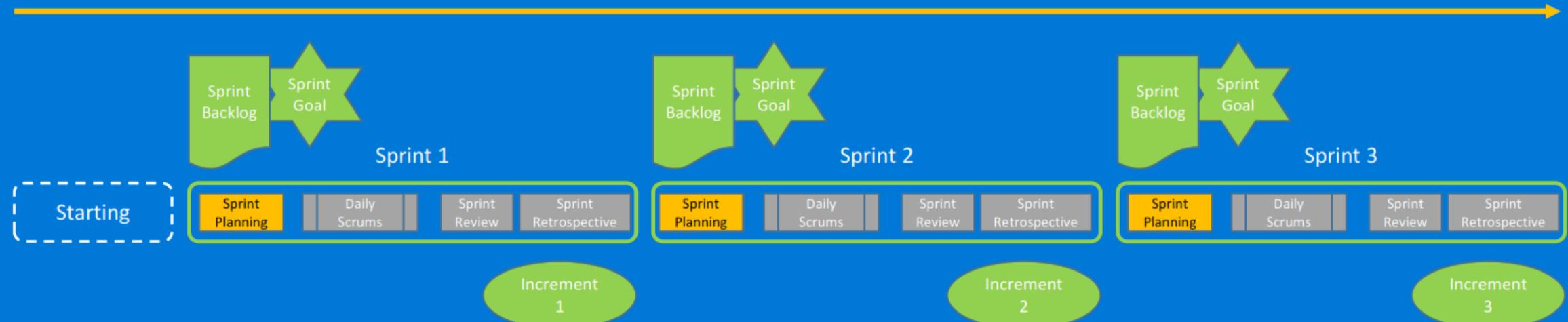


# Daily Scrum





## Product Backlog



## Synchronizing

همگام سازی

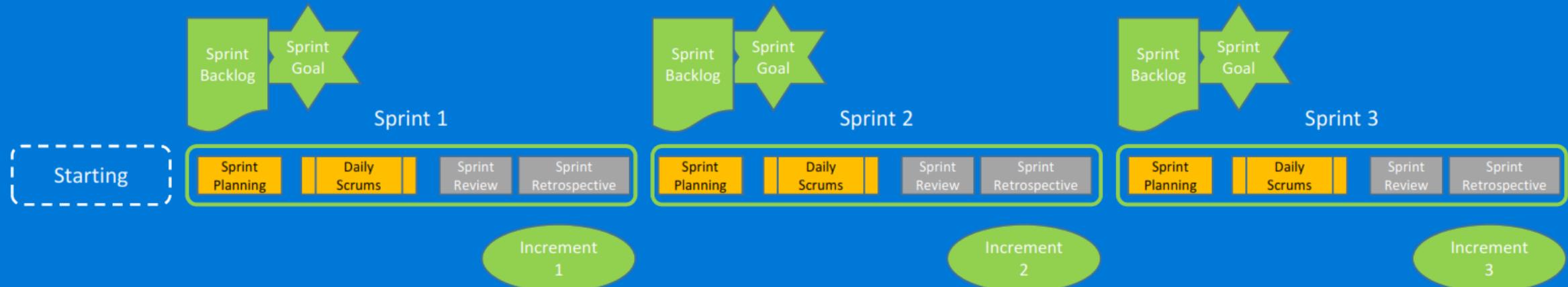
## Daily Scrum

- 15 minutes meeting
- Usually standing up
- It's only for developers





## Product Backlog



## Daily Scrum 3 Questions

- What I did since yesterday
- What I'm going to do until tomorrow
- The problems



# Daily Scrum

**Only** the developers **participate**  
**Others** can **attend**

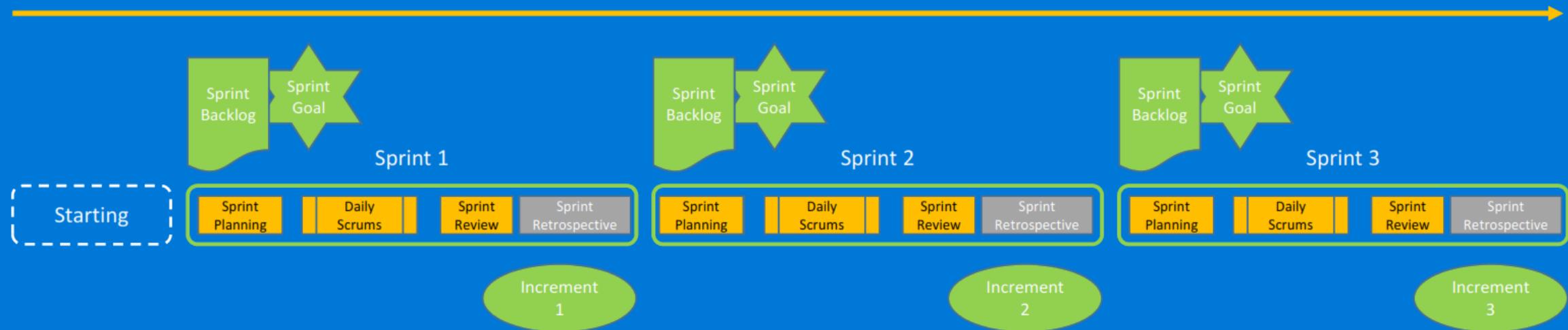


# Sprint Review





## Product Backlog



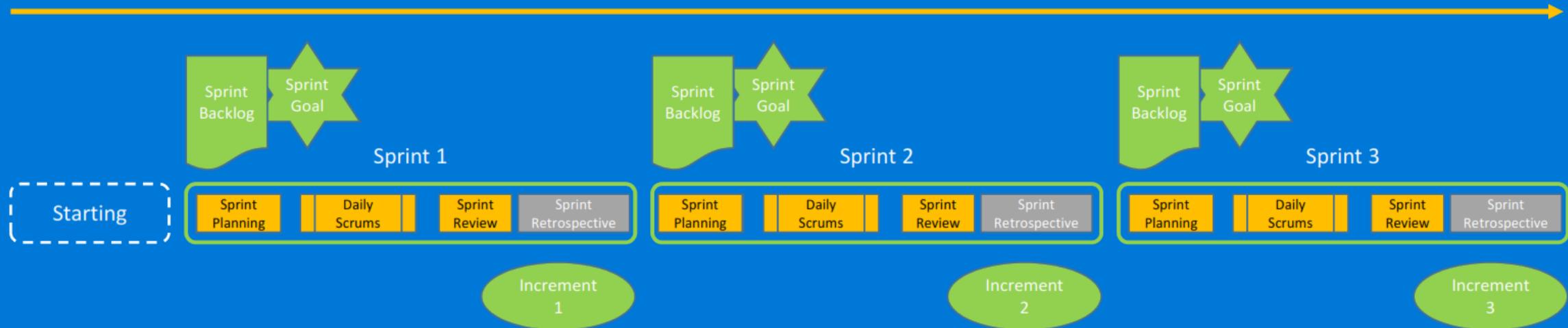
## Sprint Review

- To receive feedback from the customer and stakeholders





## Product Backlog



## Sprint Review Characteristics

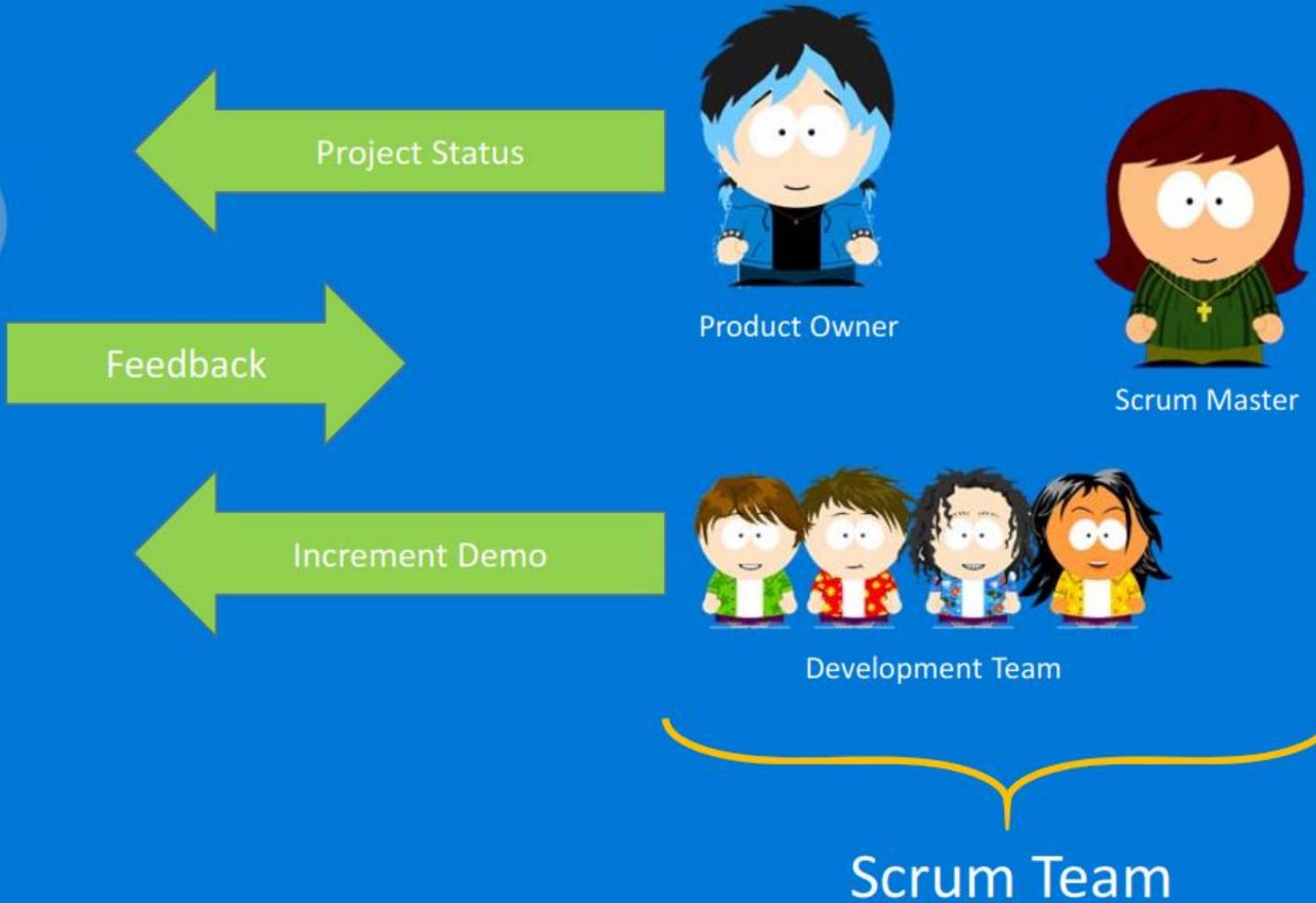
- An informal meeting
- 4 hours in a one-month Sprint and shorter proportionally
- All Scrum team members are in the meeting



# Sprint Review Interactions



Customer

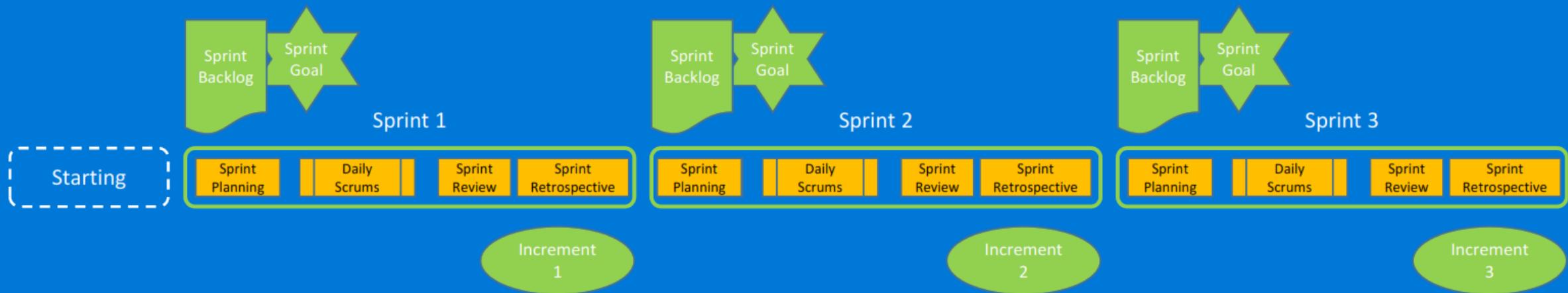


# Sprint Retrospective





## Product Backlog



## Sprint Retrospective

- Continuous improvement



## Improvement in the **Process**

### Retrospective

- Done at the end of each Sprint

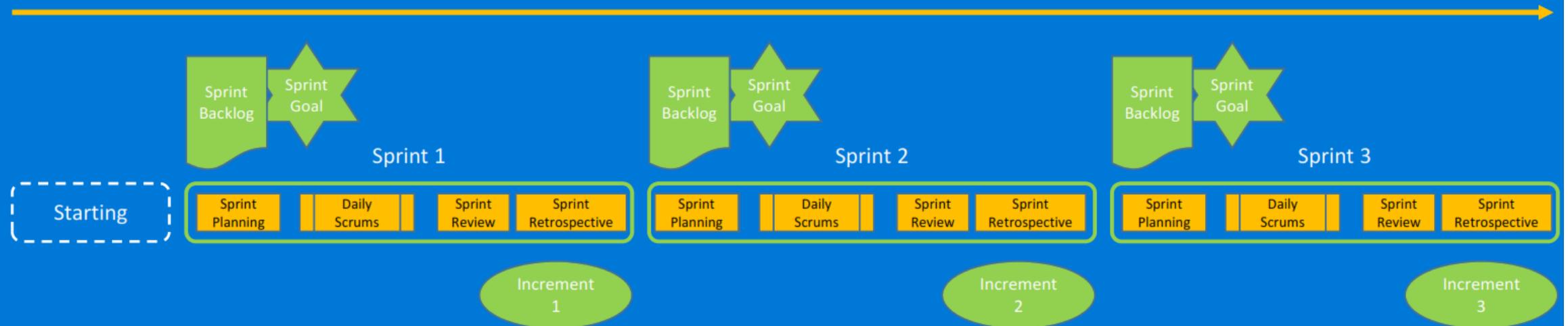
## Improvement in the **Product**

### Refactoring

- Improving the code without changing its behavior
- Done continuously



## Product Backlog



## Sprint Retrospective Characteristics

- 3 hours in a one-month Sprint and shorter proportionally
- All Scrum team members are in the meeting

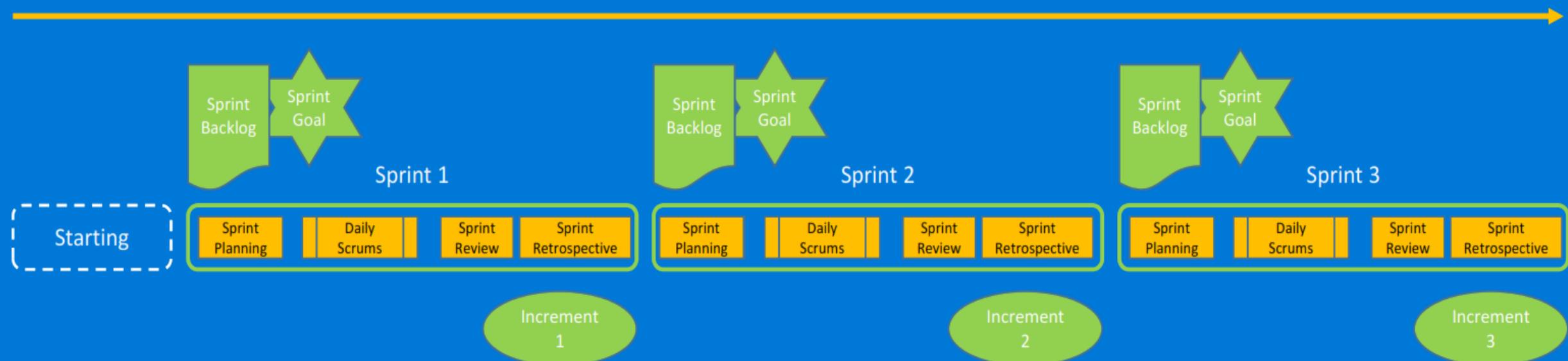


# Product Backlog Refinement





## Product Backlog

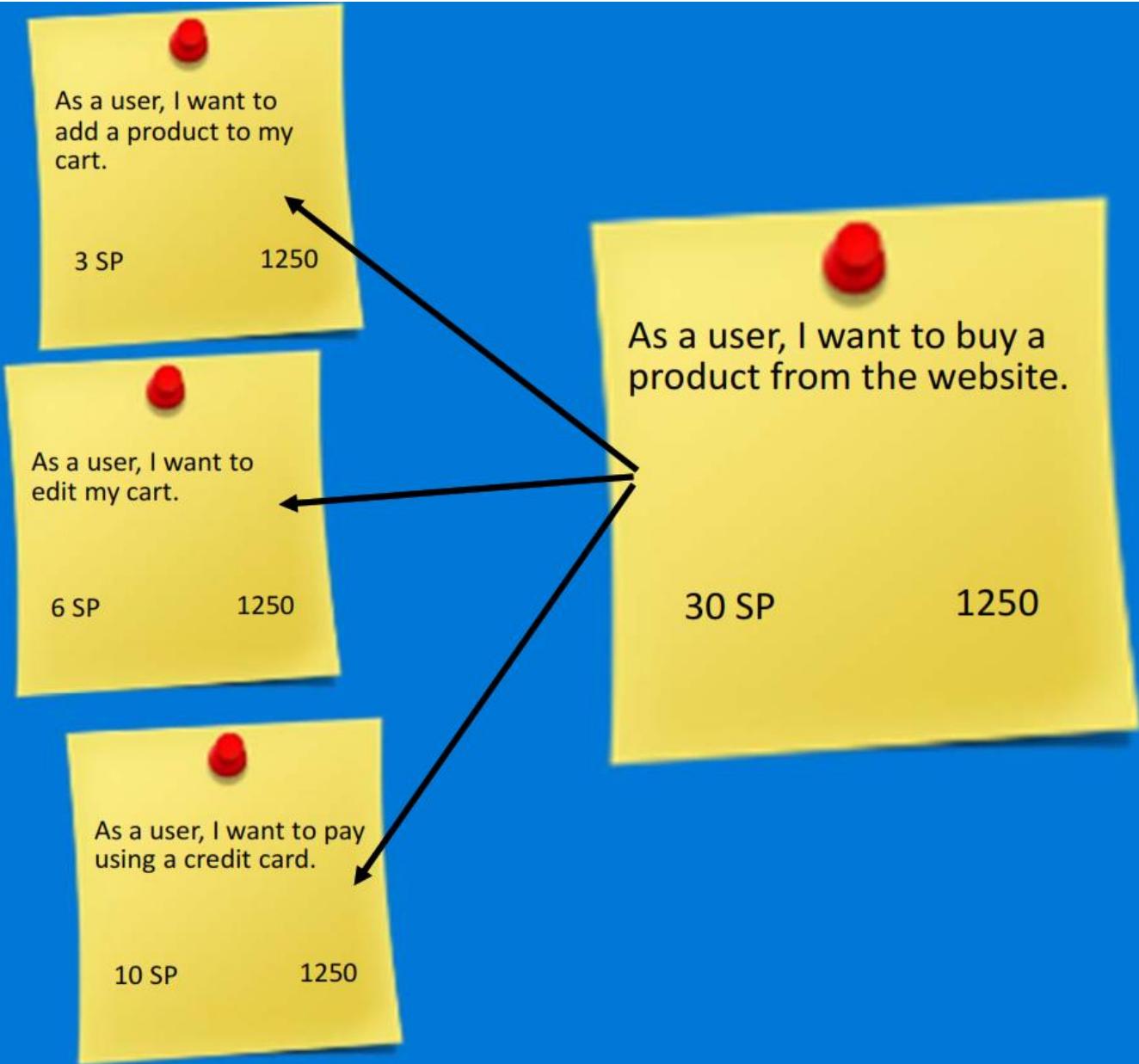


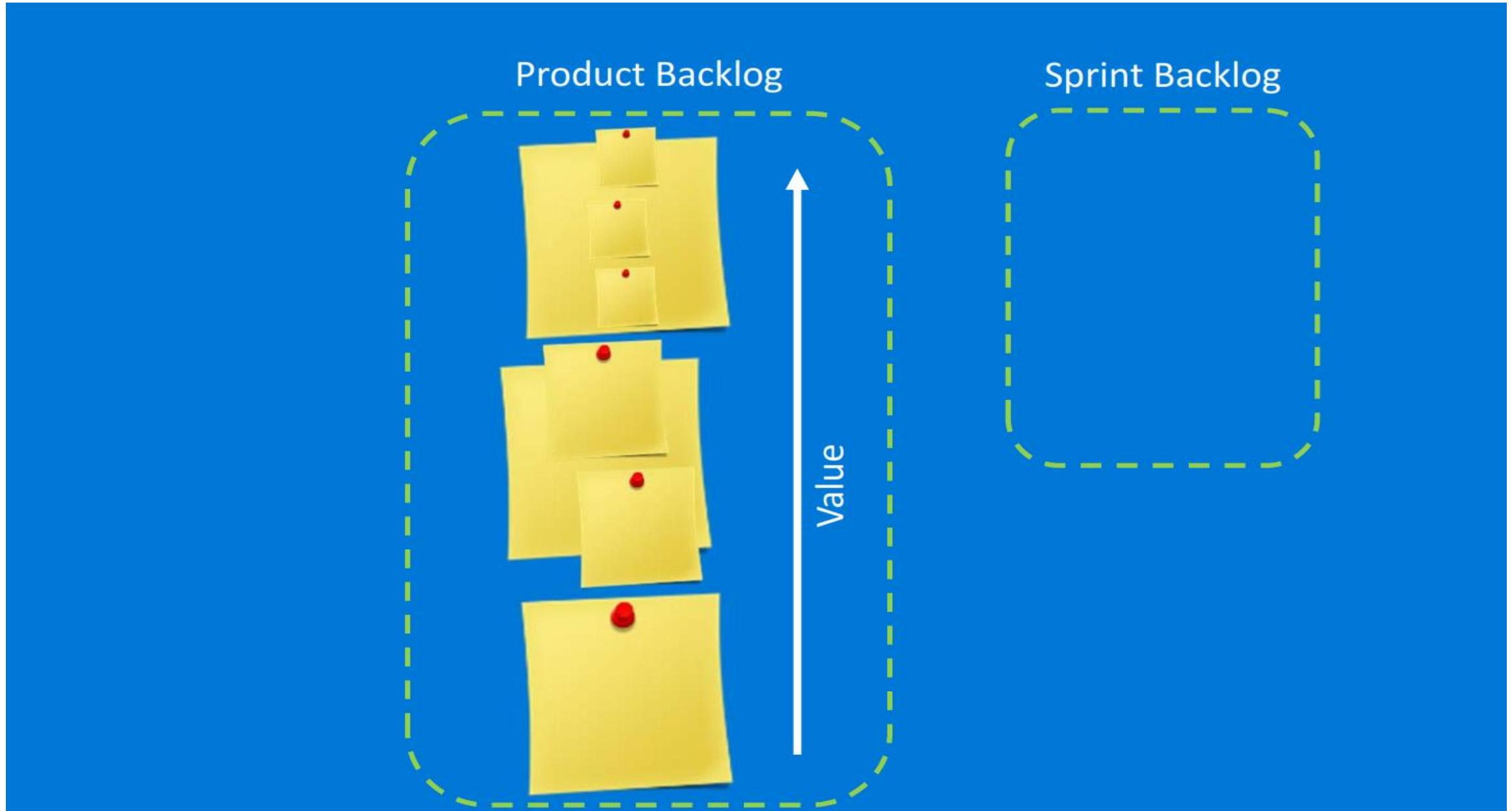
## Product Backlog Refinement (Grooming)

- Done continuously (not timeboxed)



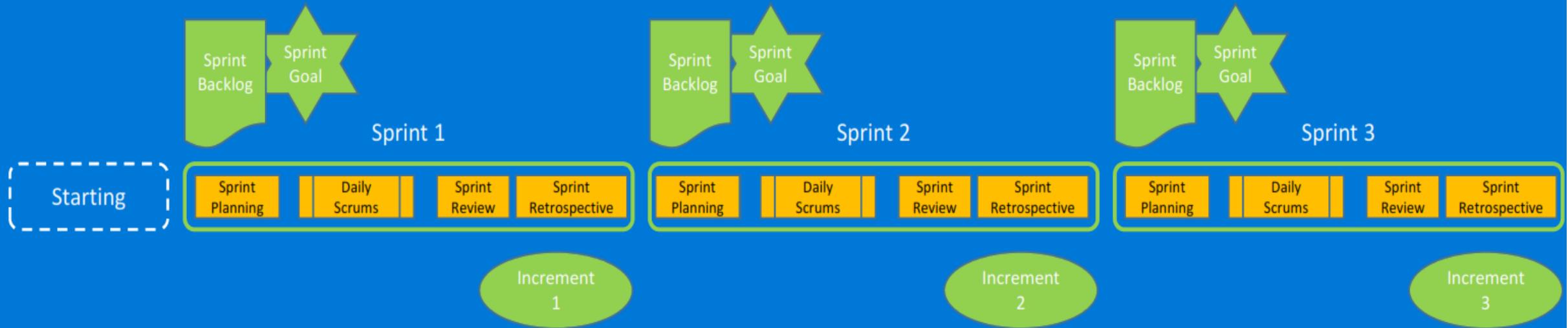
- Breaking down the huge items (Epic)
- Creating new items
- Removing items
- Adding detail to the items
- Estimation
- Valuation
- Prioritizing
- ...







## Product Backlog



## Product Backlog Refinement (Grooming)

- Done continuously (not timeboxed)
- Max 10% of the developers time



# Scrum Roles



# The Scrum Team



Product Owner



Scrum Master



Development Team





Product Owner

# Responsibilities

Main and Primary Responsibility:

**Maximizing Value**

Other Responsibilities:

1. Creates the items in the Product Backlog
2. Assigns value to items
3. Orders the items
4. Explains the items to everyone (developers / customer / ...)
5. Measures project performance
6. Contacts the customer





Product Owner

## Characteristics

1. Owns the Product Backlog
2. Is always 1 and just 1 person
3. Can be influenced by others
4. Is respected by everyone
5. Can delegate some their responsibilities
6. Full-time or part-time job





Scrum Master

# Responsibilities

Main and Primary Responsibility:

**Taking care of the Scrum framework**

Other Responsibilities:

1. Ensure Scrum is understood by everyone
2. Ensure Scrum is enacted
3. Helps others to find techniques
4. May facilitate the events if required or requested
5. Removes Impediments
6. Helps the organization adopt Scrum





Scrum Master

## Characteristics

1. One Scrum Master for each team
2. Full-time or part-time
3. Servant leader
4. A manager\* (They don't manage people, They manage Scrum process)
5. Not a project manager
6. Not a team leader



# Responsibilities



Development Team

Main and Primary Responsibility:

## Create the Increments

Other Responsibilities:

1. Develop!
2. Estimate
3. Select items for Sprint Backlog
4. Decompose Sprint Backlog Items to tasks
5. Measure Sprint performance
6. Calculate velocity
7. Compose / Refine the DoD
8. Make technical decisions



# Characteristics



Development Team

1. 3 to 9 people
2. Have no titles
3. Preferably full-time
4. Self-organized
5. Cross-functional



# Scrum Team

## Characteristics



Product Owner



Scrum Master



Development Team

1. Self-organized
2. Cross-functional
3. Have no additional roles
4. Deliver incrementally and develop iteratively



# Events



# Duration

Max 1 month

Sprint



8 hours

In a 1-month Sprint

15 minutes

4 hours

In a 1-month Sprint

3 hours

In a 1-month Sprint

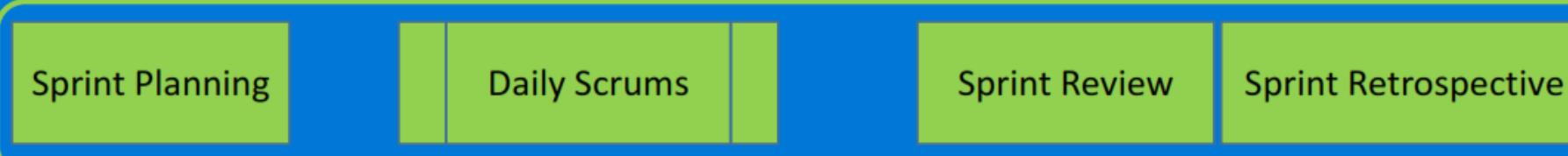
1. There are 5 events
2. All of them are timeboxed
3. Except Sprint, all other events' timeboxing means the events can be finished earlier than the maximum duration



# Main Purpose

Value through creating the  
Increments

## Sprint



Sprint Backlog  
Sprint Goal

Synchronize the  
developers

Feedback  
(Revising the  
Product Backlog)

Process  
continuous  
improvement



# Participants

## Sprint

Sprint Planning

Daily Scrums

Sprint Review

Sprint Retrospective



# Resource

- <https://scrumschool.ir/>

