

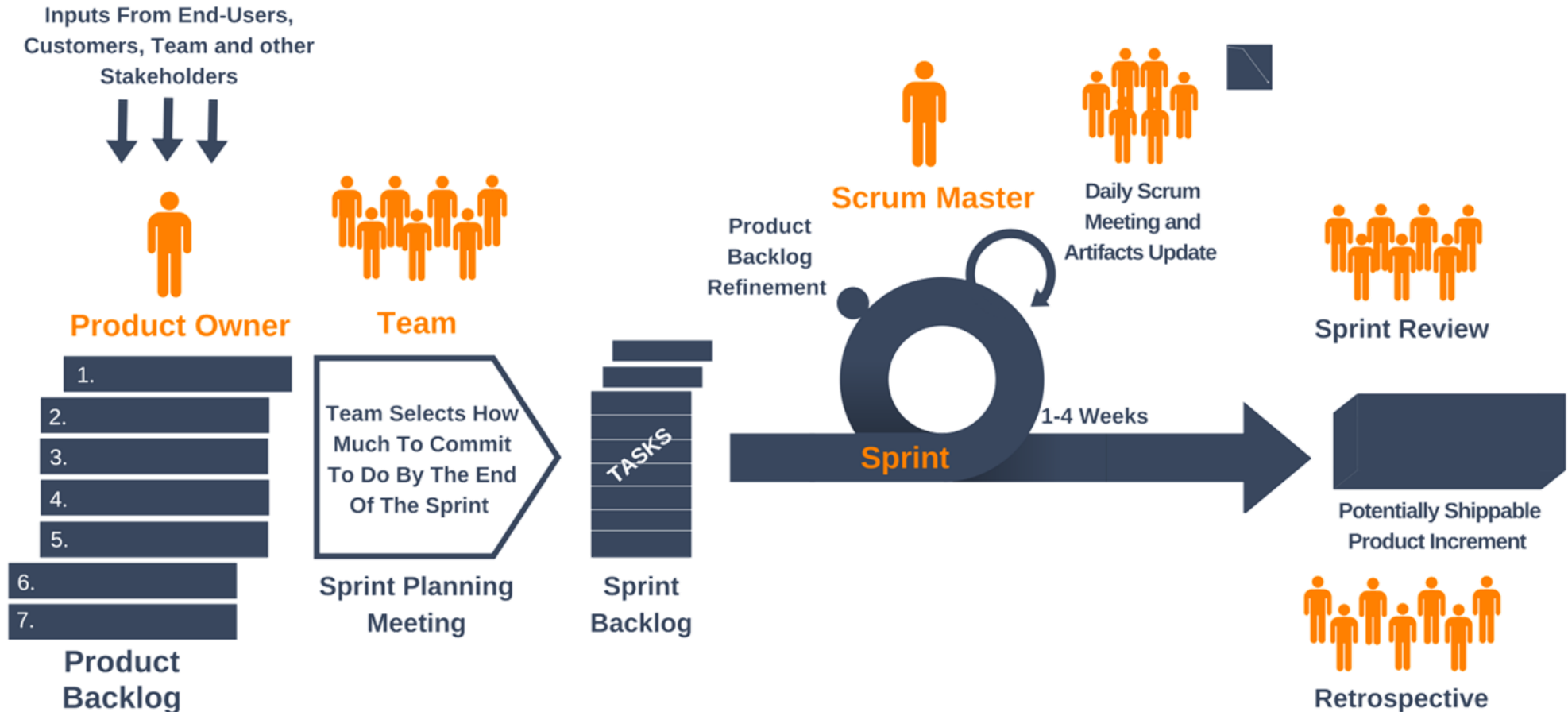
## AGILE SOFTWARE DEVELOPMENT

---

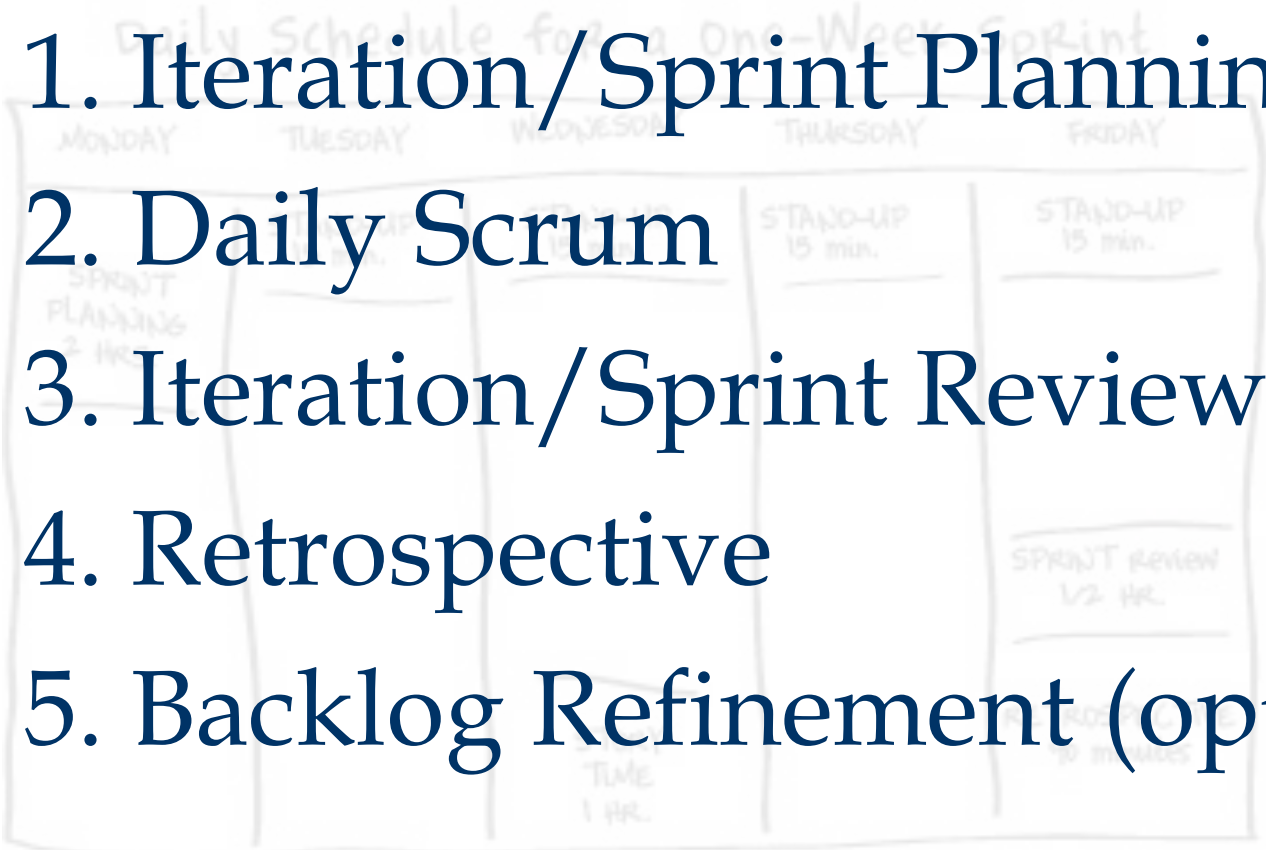


## Scrum Methodology: Ceremonies

# Scrum Ceremonies



# Scrum Ceremonies

- 
- A background image of a hand-drawn 'Daily Schedule for a One-Week Sprint'. The schedule is organized by day (Monday to Friday) and time slots. Key activities include 'SPRINT PLANNING 2 HR.' on Monday, 'STAND-UP 15 min.' on Tuesday, Wednesday, and Thursday, and 'SPRINT REVIEW 1/2 HR.' on Friday. There are also slots for 'TIME 1 HR.' on Wednesday and Thursday.
1. Iteration/Sprint Planning
  2. Daily Scrum
  3. Iteration/Sprint Review
  4. Retrospective
  5. Backlog Refinement (optional)

# 1. Iteration/Sprint Planning

- **Rules**

- Duration: timeboxed to maximum 2h/ sprint week
- Participants: *Product Owner* and *Development Team* (external stakeholders that may contribute are optional)
- Facilitator: *Scrum Master*

- **Goal**

- Project predictability

- **Objectives**

- To create a Sprint Backlog
- To get Team commitment for the Sprint

# 1. Iteration/Sprint Planning

- **Structure**

- Part 1: PO presents the Stories that are prioritized and ready for the Sprint (half of time)
- Part 2: The Team is sizing Stories and slicing them in Tasks, then commits to PO (half of time)

- **Common traps**

- Long and boring meetings with too detailed technical discussions
- Superficial approach, team members do not really understand the requirements
- Wrong and/or manipulated estimations



## 2. Daily Scrum

- **Rules**

- Duration: timeboxed to maximum 15 minutes
- Participants: Development Team (PO is welcome, but optional and must be silent)
- Facilitator: Scrum Master or a Team member

- **Goal**

- Self-organizing

- **Objectives**

- To have a common Team understanding of the Sprint progress
- To collect impediments and decide who will tackle each of them



## 2. Daily Scrum

- **Structure**
  - Each Team member explains what he/she achieved since last Daily Scrum, what he/she will work until next Daily Scrum and what impediments prevent him/her to work (if any)
- **Common traps**
  - Long and boring meetings with too detailed technical discussions
  - Superficial approach, team members do not really understand the Sprint progress
  - False impediments



# 3. Iteration/Sprint Review

- **Rules**

- Duration: timeboxed to maximum 1h/sprint week
- Participants: PO and Development Team (stakeholders are welcome, but optional)
- Facilitator: Scrum Master

- **Goal**

- Delivering incrementally

- **Objectives**

- To review and accept the Stories finalized by the Team
- To discuss potential changes of the Product Backlog resulting from the Sprint Review (priorities)

# 3. Iteration/Sprint Review

- **Structure**

- Team is organizing a demonstration of finalized Stories
- PO is reviewing and accepting depending on DoD and Acceptance Criteria
- PO and team discuss the results of the Sprint and the consequences on subsequent Sprints

- **Common traps**

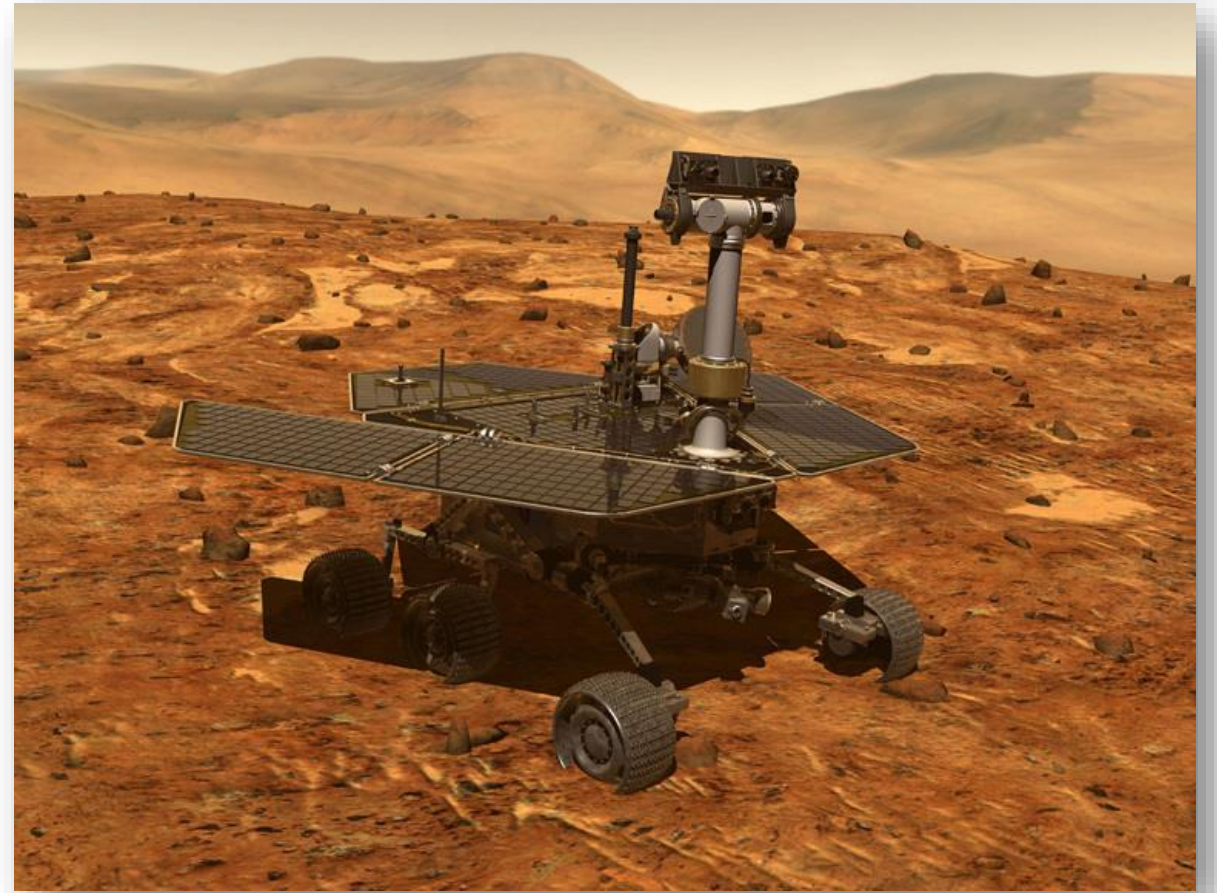
- Acceptance Criteria not checked during review (incomplete testing)
- DoD ignored

### 3. Iteration/Sprint Review

Fast feedback

#### Remote driving Mars Rover

- 10 minutes to send radio signals between Earth and Mars



# 3. Iteration/Sprint Review

## Feedback

- Bugs
- Small cosmetic things
- New ideas

# 4. Retrospective

- **Rules**

- Duration: timeboxed to maximum 45mins/sprint week
- Participants: Development Team (PO only if invited by the Team)
- Facilitator: Scrum Master

- **Goal**

- Continuous improvement

- **Objectives**

- To review the efficiency and effectiveness of the teamwork and methods used
- To determine and decide upon action items that will lead to improving Team's performance

# 4. Retrospective

- **Structure**

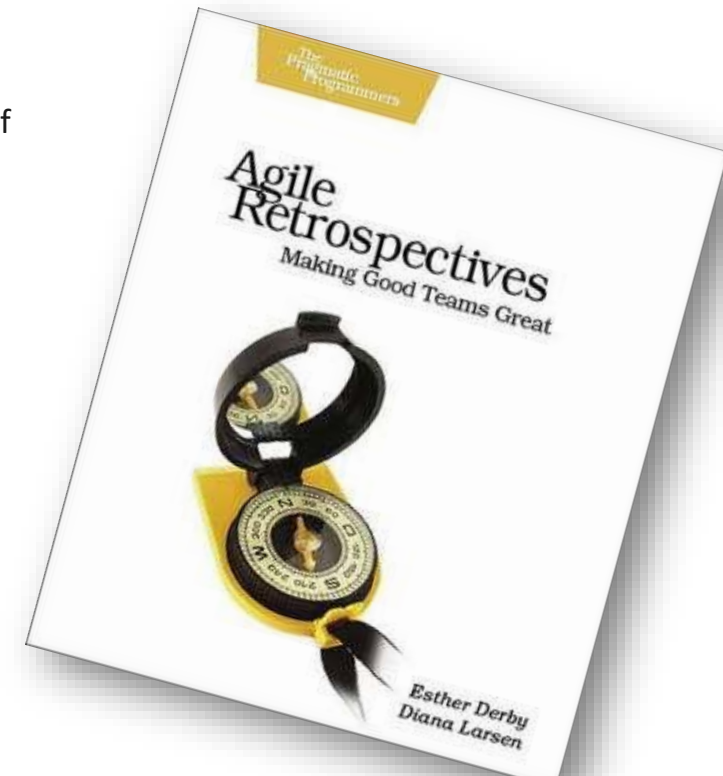
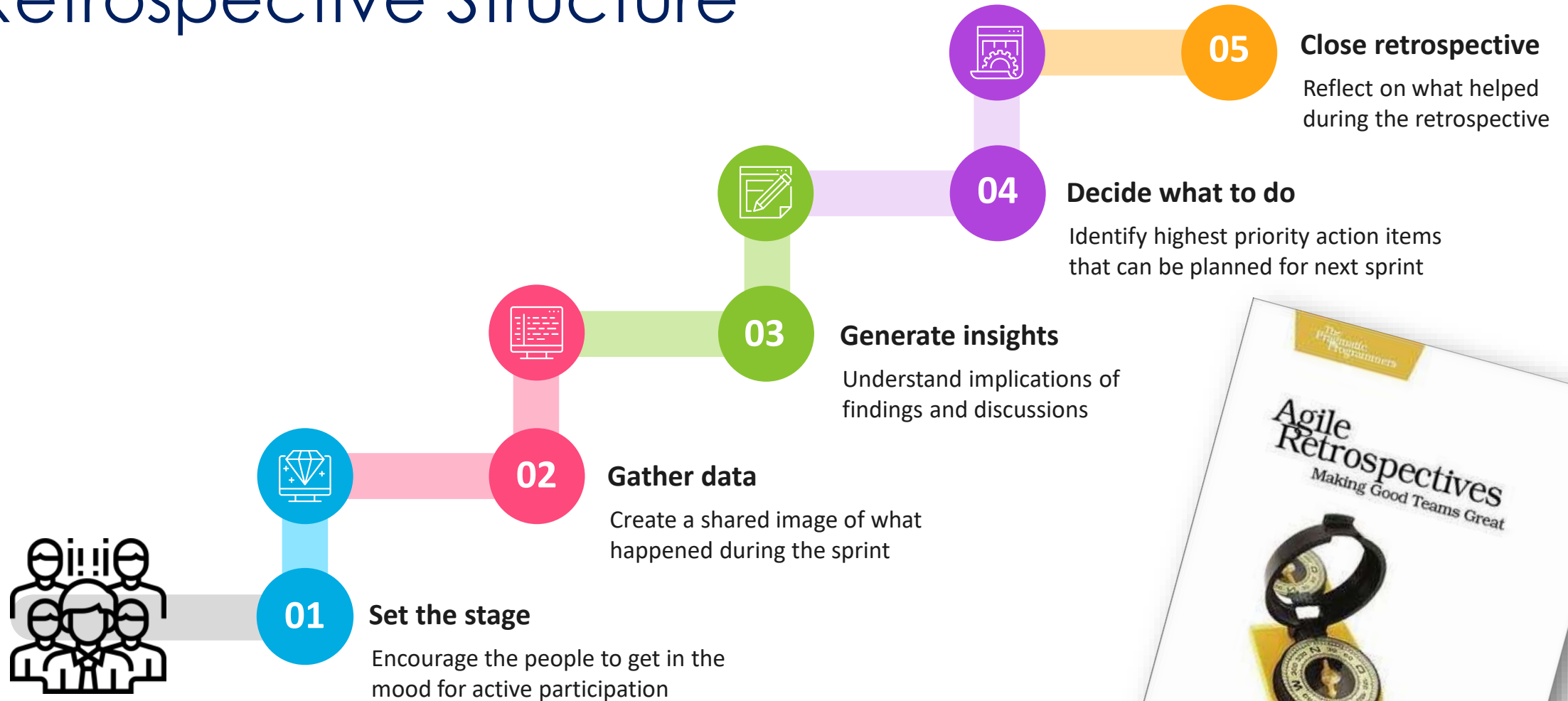
- Collecting Team member's opinions about what they should keep doing, stop doing or doing differently
- Discussing and collectively deciding the improvements applied in the next Sprint

- **Common traps**

- False harmony, avoiding to discuss about “the elephant in the room”
- Finger-pointing, personal conflicts
- Failing to identify effective action items

# 4. Retrospective

## Retrospective Structure





# 4. Retrospective

## Set the stage

**Goal:** getting team members in the right mood for retrospective

### Check-in

---

Two words summary for what they hope to get from retrospective or how they are feeling about retrospective

### Focus on/Focus off

---

Discuss on terms like “Dialogue rather than Debate” or “Conversation rather than Argument”. Discuss what to do to move people on the right/focus on?

### ESVP

---

Explorers: discover new ideas  
Shoppers: look over new useful ideas  
Vacationers: happy to be away from work  
Prisoners: forced to come to retrospective

### Working agreements

---

Brainstorming over working agreements they would like to put in place for the retrospective

# 4. Retrospective

## Gather data

**Goal:** remembering what happened during last iteration

### Follow-up

---

Discuss the success or failure of improvement actions decided during previous retrospective

### Timeline

---

Team members recall good, problematic, significant events happened on a timeline (happy/sad faces bottom side)

### Triple nickel

---

Spend 5 minutes gathering at least 5 ideas related to a specific issue. Team is divided in groups of 5 if team is bigger than 7. Pass written issues to the right and give another 5 minutes to reflect and expand the ideas

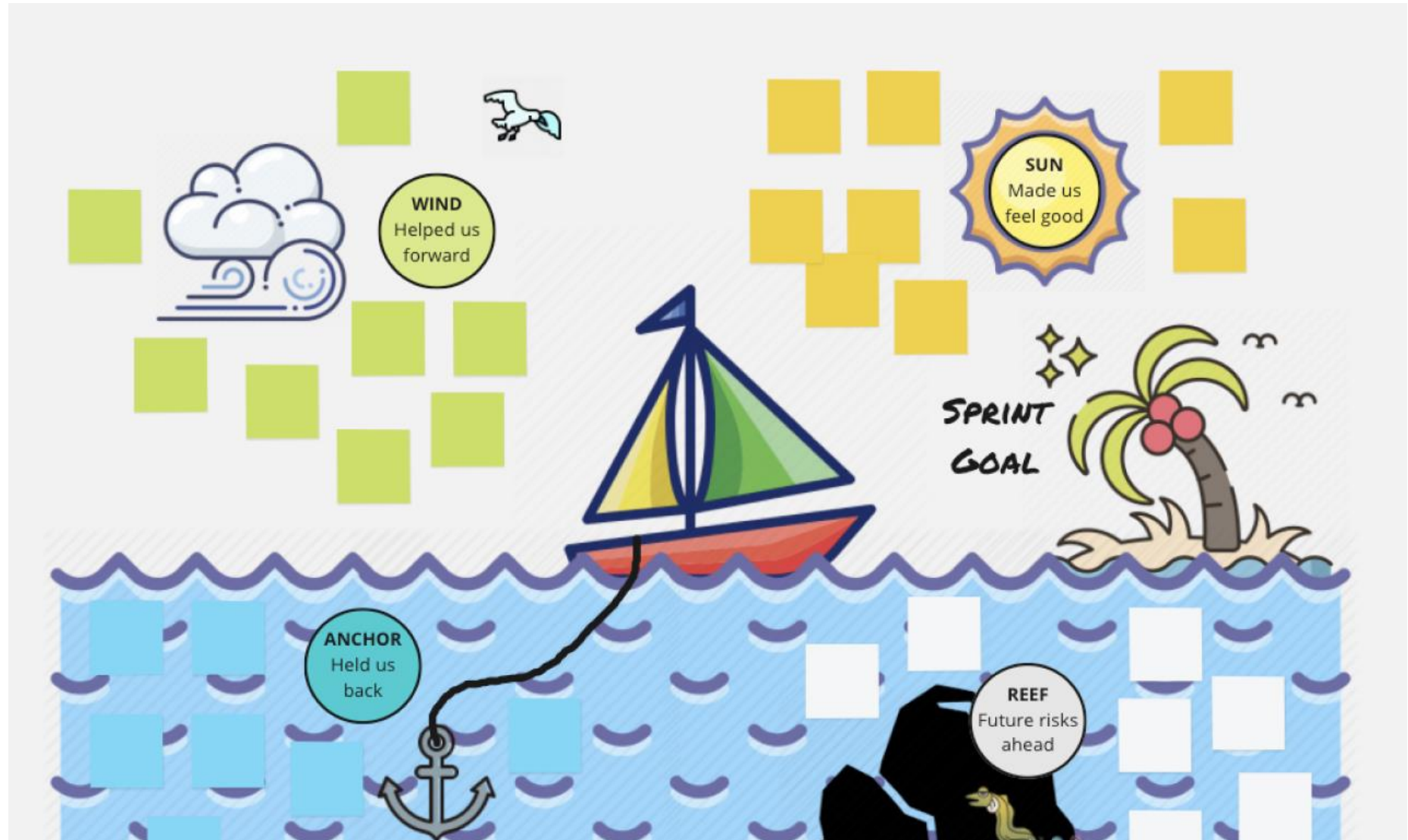
### Others

---

Mad-Sad-Glad,  
Satisfaction histogram,  
Team radar etc

# 4. Retrospective

## Gather data



# 4. Retrospective

## Generate insights

**Goal:** identifying potential improvements of collaboration and teamwork

### **Brainstorming**

---

Quiet writing: silent writing of ideas  
Round-robin: pass token around the group  
Free-for-all: ad-hoc information generation

### **5 Whys**

---

Ask why five times  
to discover cause-and-effect relation

### **Fishbone**

---

Visual tool to display  
root cause analysis of problems: procedures,  
policies, system, skills etc

# 4. Retrospective

## Decide what to do

**Goal:** deciding on actionable improvements for the next iteration

### Short subjects

---

Start doing  
Stop doing  
Continue doing

### SMART goals

---

Specific, Measurable,  
Attainable, Relevant,  
Timely

### Dot voting

---

Prioritize potential improvements  
and focus on  
most relevant 2-3 actions

# 4. Retrospective

## Close retrospective

**Goal:** ending retrospective and conclude upon its success

### **Plus/Delta**

---

Plus: what is going well in retrospectives?  
Delta: what should we change?

### **Helped, Hindered, Hypothesis**

---

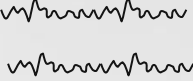



Helped: what is going well  
Hindered: what was a hindrance  
Hypothesis: ideas of improvement

### **Return on time invested**

---

Feedback focused on how  
people believe their  
time was spent

# 4. Retrospective

Process	Technical
	
	
	

What we are going to change  
What we expect to happen  
Who is responsible to implement it



# 5. Backlog Refinement (Grooming)

- **Rules**

- Duration: timeboxed to a duration negotiated by PO and Team
- Participants: PO and Development Team
- Facilitator: Scrum Master

- **Goal**

- Improving the planning process

- **Objectives**

- To communicate significant changes in the Product Backlog
- To slice backlog items that are too big, then optionally to roughly estimate the resulted Stories
- To collect questions about the Stories that are prioritized for next Sprint

# 5. Backlog Refinement (Grooming)

- **Structure**
  - Flexible agenda, proposed by Product Owner
- **Common traps**
  - Getting into too detailed discussions about backlog items
  - Slicing backlog items using technical criteria (Technical Stories instead of User Stories)