Scrum - An Introductory Course to Agile

⚠️ Being Agile

- Playground
  - Senior developers
  - requirements change often
  - small number of developers
  - culture that responds to change
- Pros
  - speed wins
  - energizes employees and customers
  - good and fast release rather than later and perfect release
- Cons
  - developer centric rather than user centric
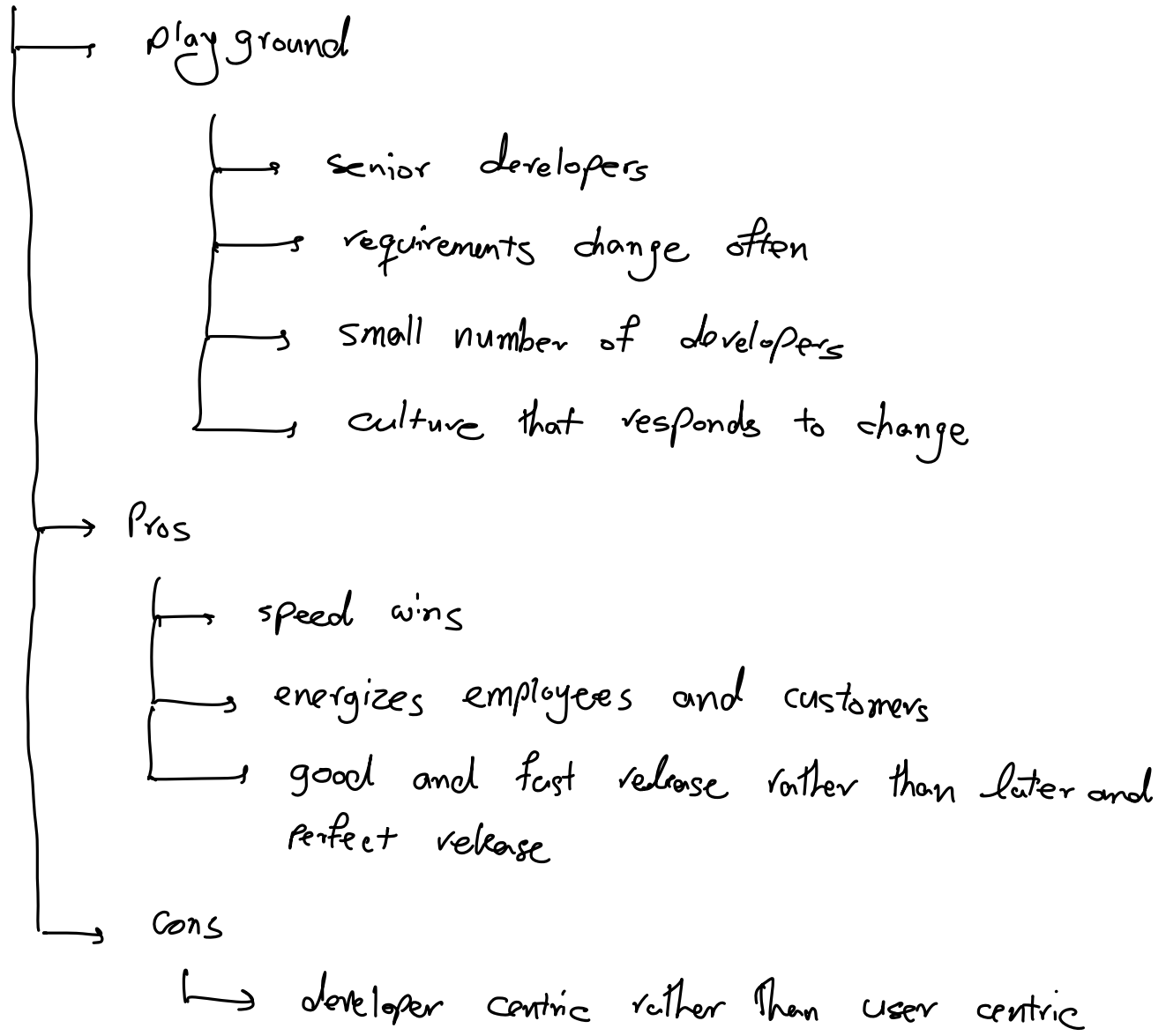
⚠️ Scrum is an agile framework for completing complex projects, Scrum originally was formalized for software development process

⚠️ Scrum focuses on empirical process control

└→ knowledge comes from experience
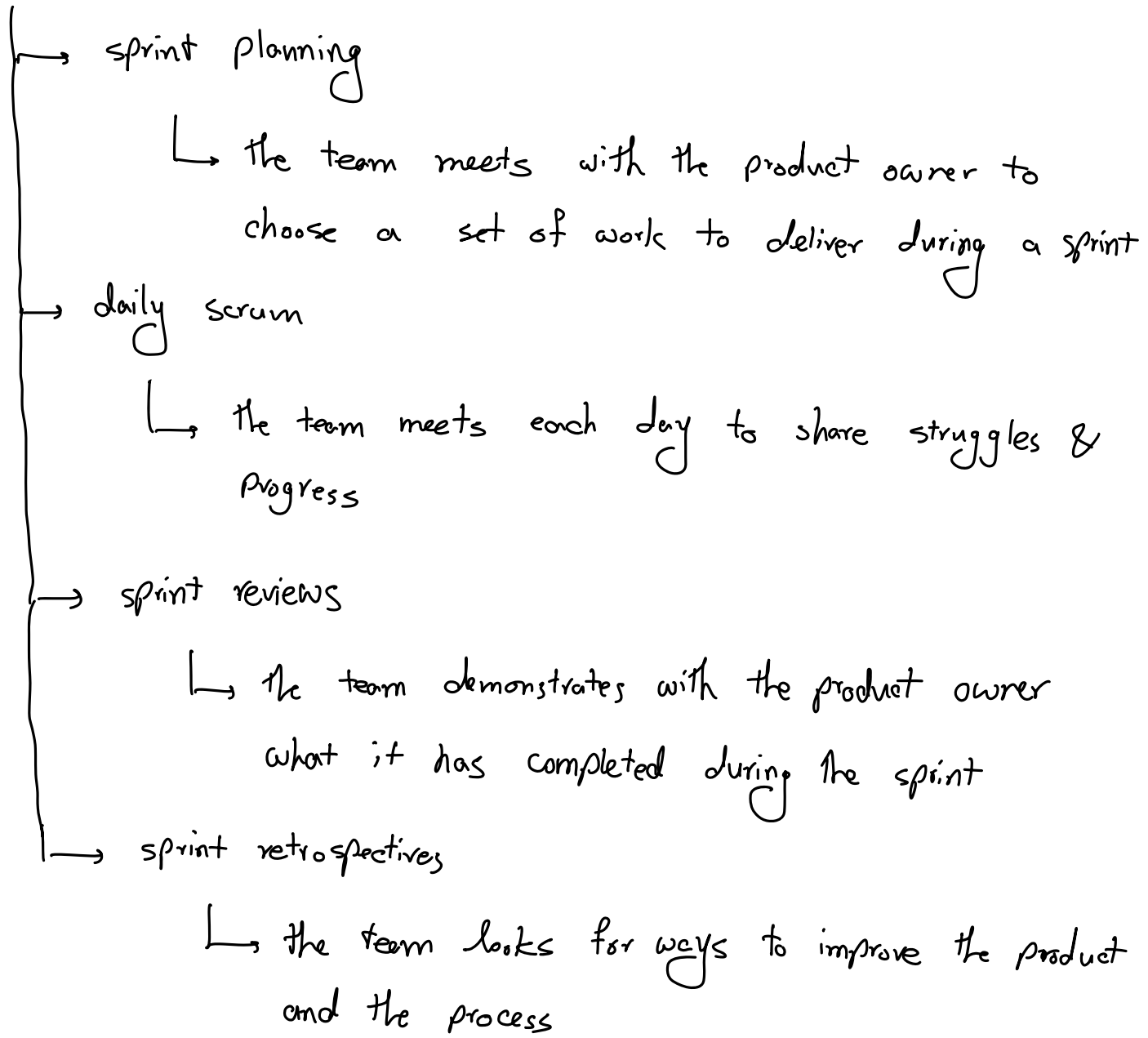└→ decisions are made based on what is known

⚠️ Implementation in Scrum is based on Three principles
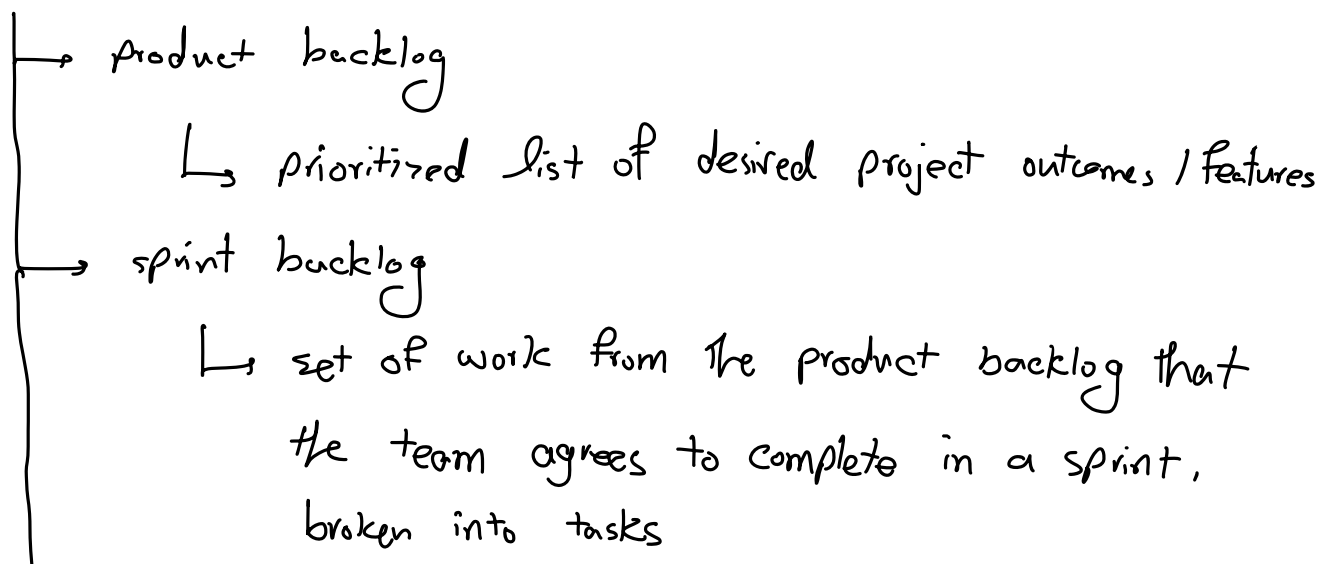
├→ transparency
├→ inspection
└→ adaption

⚠️ Scrum main roles

├→ Scrum Master

    ├→ coach, shelters the team
    └→ do not assign tasks, do not provide
        day-to-day direction

├→ Product owner

    ├→ responsible for the vision of product
    └→ responsible for business value of the project

└→ team member

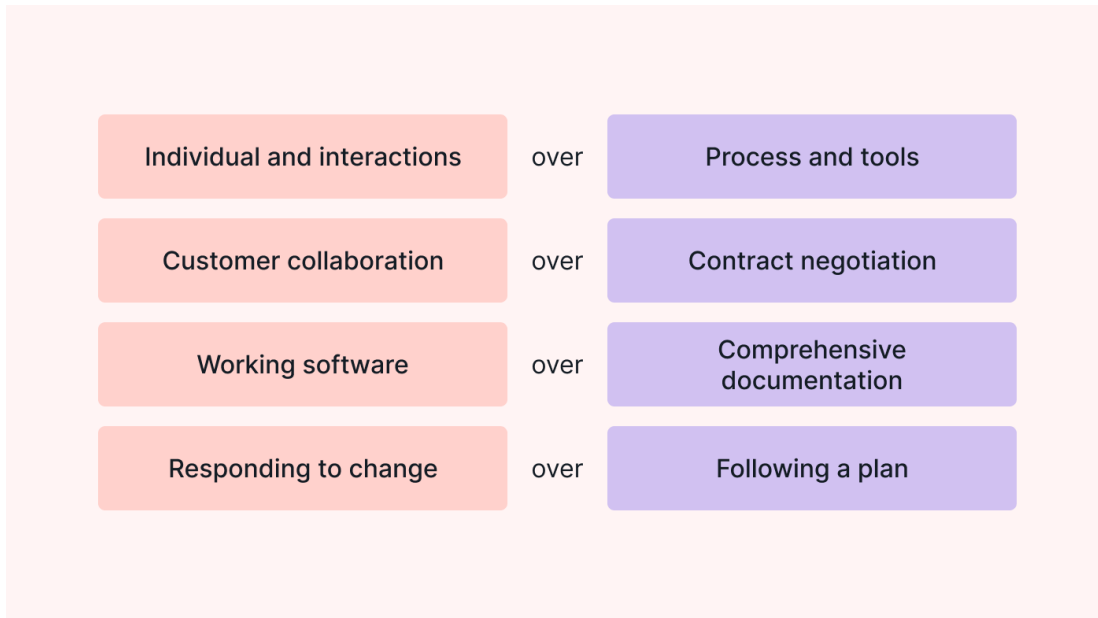   ├→ self-organizing
    └→ plays for the team

Scrum Ceremonies

→ sprint planning
  └→ the team meets with the product owner to choose a set of work to deliver during a sprint

→ daily scrum
  └→ the team meets each day to share struggles & progress

→ sprint reviews
  └→ the team demonstrates with the product owner what it has completed during the sprint

→ sprint retrospectives
  └→ the team looks for ways to improve the product and the process

⚠ Scrum artifacts

→ product backlog
  └→ prioritized list of desired project outcomes / features

→ sprint backlog
  └→ set of work from the product backlog that the team agrees to complete in a sprint, broken into tasks

↳ burn down chart
  ↳ at-a-glance look at the work remaining

⚠ Agile Manifesto

Individual and interactions	over	Process and tools
Customer collaboration	over	Contract negotiation
Working software	over	Comprehensive documentation
Responding to change	over	Following a plan

⚠ Core Scrum foundations

① Empricism   ② self-organization

③ prioritization   ④ rhythm   ⑤ collaboration

⚠ Core Scrum Values

① Communication   ②. Simplicity

③ Courage   ④ Feedback   ⑤ Respect

⚠ Product backlog attributes

① ID   ② story   ③ Prioritization   ④ Estimation

⚠️ Sprint backlog

- a set of product backlog items
- follows sprint goal
- defines work that transform PB items into "Done" increment
- individuals sign up for work of their own choosing
- work is never assigned
- is changed by DEV team throughout sprint
- is highly visible
- PO can not make changes

⚠️ Scrum roles introduction

- The product owner manages the product (and ROI)
- The scrum master manages the process
- the team manages itself
- Scrum teams are self-organizing & cross-functional

⚠️ Product owner is a person not a committee

⚠️ Scrum Master shouldn't be the product owner, too

⚠️ Development team is typically about 5-9

⚠ Everyone answers 3 questions

- → What did you do yesterday?
- → What will you do today?
- → Is anything in your way?

⚠ Two-week sprints are good length to start with

⚠ Scrum Master checklist

- → beginning of sprint
  - → print sprint info page, put it on the wall
  - → e-mail everyone with sprint goal & link to info page
- → every day
  - → make sure the daily scrum meeting is in time
  - → check stories added/removed from the sprint backlog
  - → notify product owner with changes
  - → make sure impediments are solved or reported to PO
- → end of sprint
  - → do an open sprint demo (notify before)
  - → do a sprint retrospective with the whole team
  - → update the sprint statistics document

⚠ How to get a good user story

- → As a <user>, I can <do>, so that <value>
- → may have high level tests attached
- → key attributes (INVEST)
  - → ① independable ② negotiable
    - ③ Valuable ④ Estimable
    - ⑤ Small ⑥ Testable

⚠ Increment

- → sum of PB items that are completed & usable
  - → meets the scrum team's definition of done
  - → it is a measure for sprint planning

⚠ Definition of Done

- → sum of criteria that mark work completed
  - → it is a shared understanding to ensure transparency
  - → varies between different teams
  - → marks potentially releasable functionality
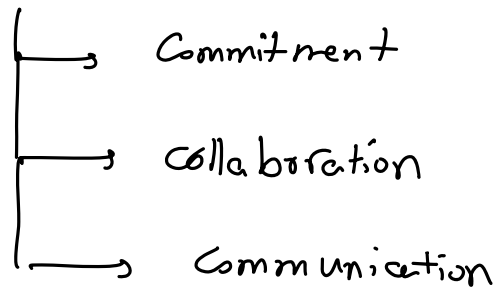 - → can change to include stringent criteria for better quality

⚠ 4 levels of Done

- at the level of task
  - convince your team mate
- at the level of a story
  - commitment to the product owner
- at the level of a sprint
  - sprint goals and commitment for the set of stories
- at the level of a release
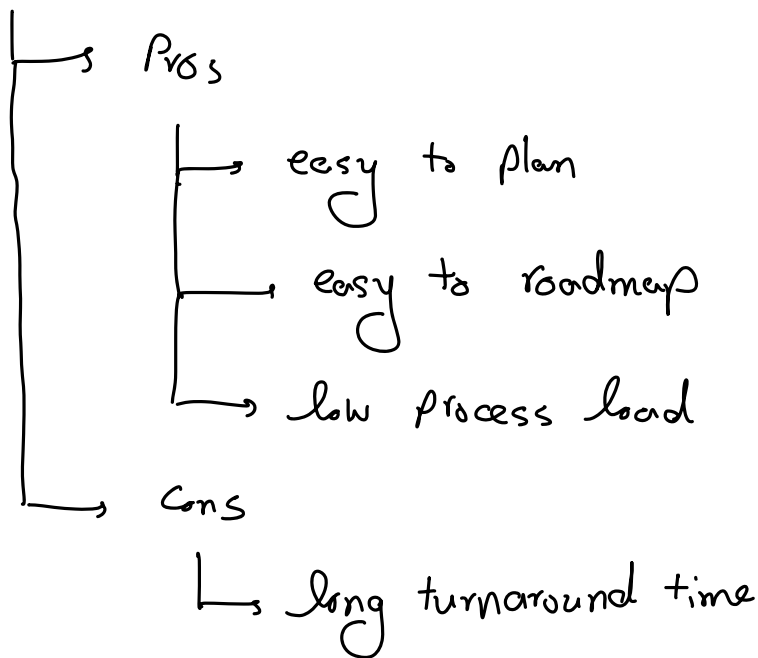 - PO decision about what it take to release

⚠ How to make retrospective relevant

- make retrospective action items visible
- choose a focus
- track outcomes
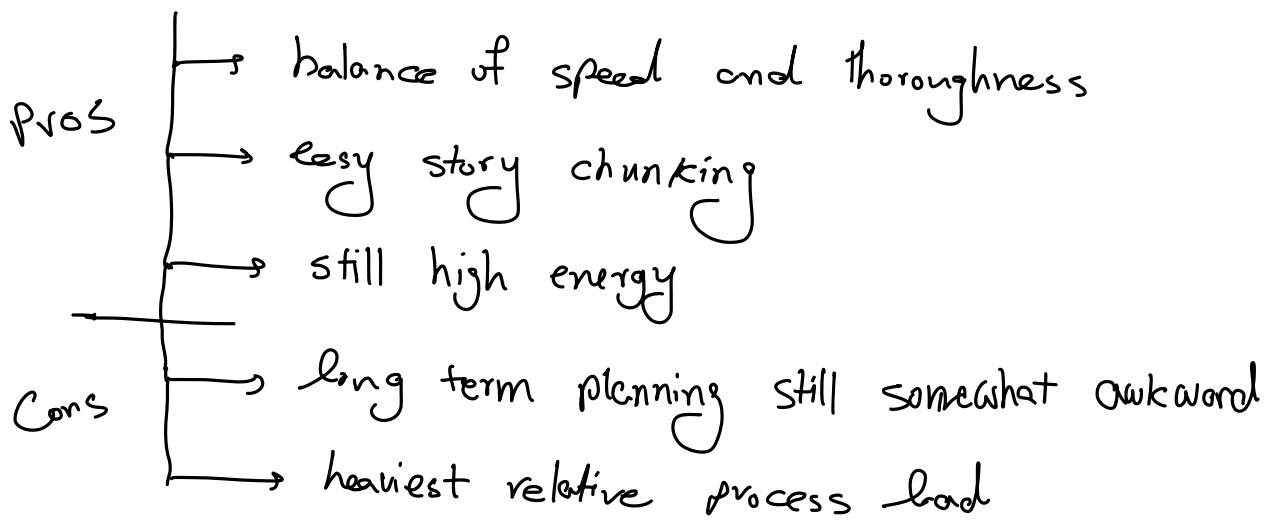- record incremental progress

⚠ How Agile and Scrum contribute to team morale
- → Commitment
- → Collaboration
- → Communication

⚠ 4 weeks sprint
- → Pros
  - → easy to plan
  - → easy to roadmap
  - → low process load
- → Cons
  - → long turnaround time

⚠ 2 weeks sprint

Pros
- → balance of speed and thoroughness
- → easy story chunking
- → still high energy

Cons
- → long term planning still somewhat awkward
- → heaviest relative process load

⚠️ **Sprint & product backlog tips**

→ keep your sprint backlog & product backlog separate, organized & prioritized

→ reprioritize your product backlog each time you add to it

→ do not add more to your sprint backlog than can be reasonably completed during your sprint

⚠️ **Sprint planning meeting tips**

→ make sure you come up with a clear goal for your sprint during the planning session

→ come up with a list of highly commited team members for the sprint

→ only begin a sprint planning session once your product backlog has enough organization and detail

→ once you select the sprint backlog activities, realize that you must commit to them completely

→ the sprint planning meeting should take no longer and no shorter than 4 hours