Managing on Agile Team

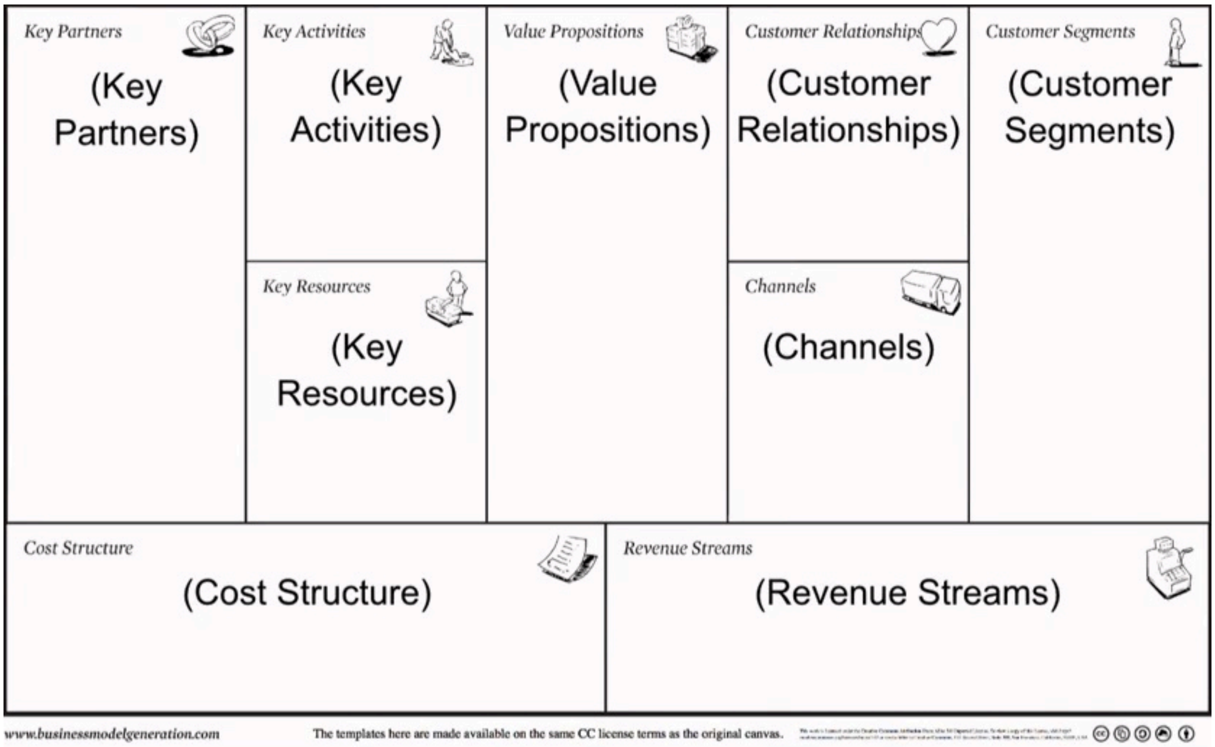
⚠ The job of software development

- learning
  - collaborate on product design
  - acceptance, usability testing
- deciding
  - prioritize and batch tasks
  - manage work in progress
- building
  - code creation and maintainance
  - version control & integration
- managing
  - team management
  - interface to general management

⚠ The business of creating software

- ① proposition design
- ② Product / system support
- ③ product design
- ④ software development
- ⑤ general management

⚠ The Business Model Canvas



Key Partners
(Key Partners)

Key Activities
(Key Activities)

Key Resources
(Key Resources)

Value Propositions
(Value Propositions)

Customer Relationships
(Customer Relationships)

Channels
(Channels)

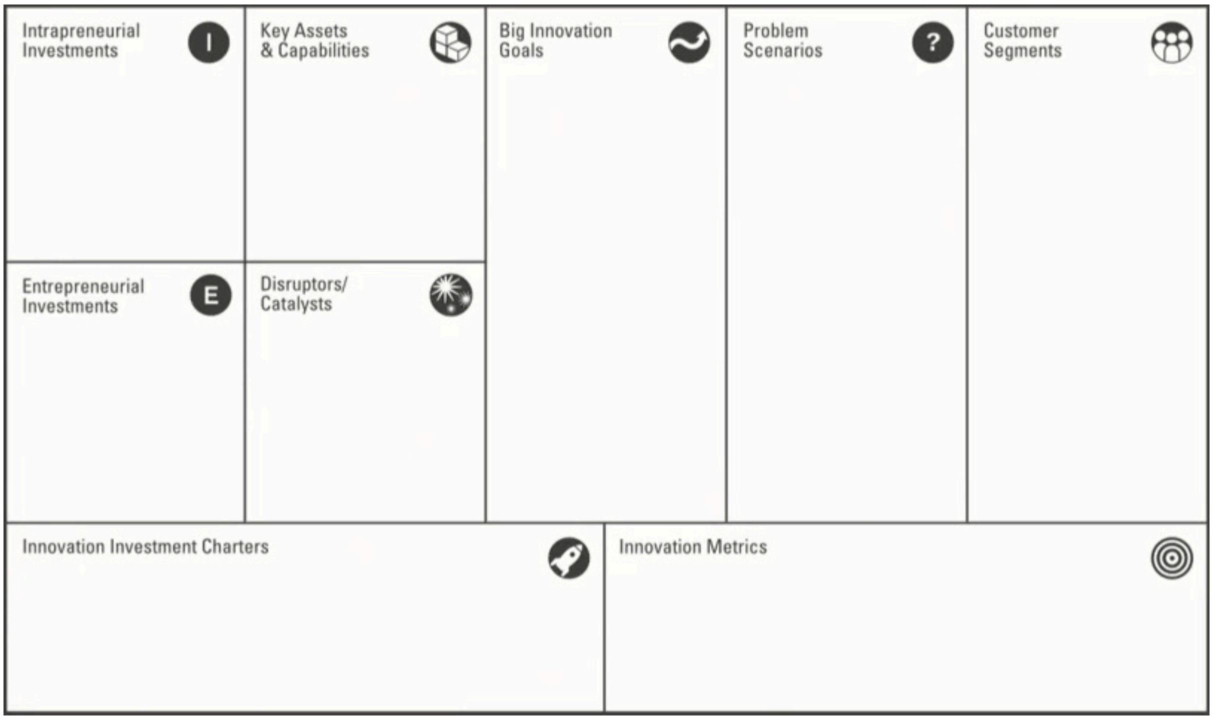
Customer Segments
(Customer Segments)

Cost Structure
(Cost Structure)

Revenue Streams
(Revenue Streams)

www.businessmodelgeneration.com

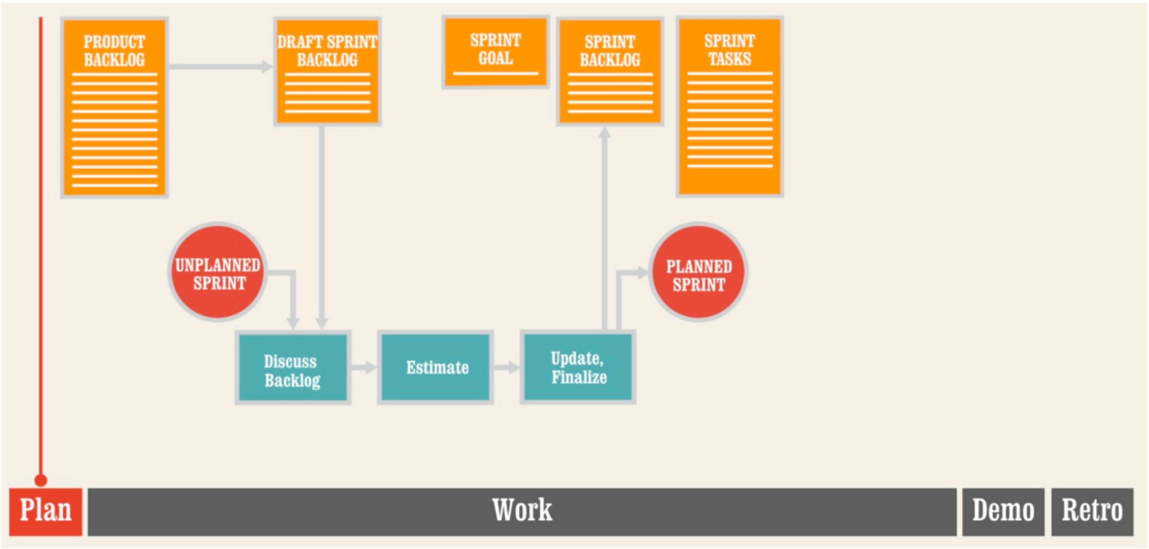
⚠ The Corporate innovation Canvas


Intrapreneurial Investments
I
Key Assets & Capabilities
Big Innovation Goals
Problem Scenarios
?
Customer Segments
Entrepreneurial Investments
E
Disruptors/ Catalysts
Innovation Investment Charters
Innovation Metrics

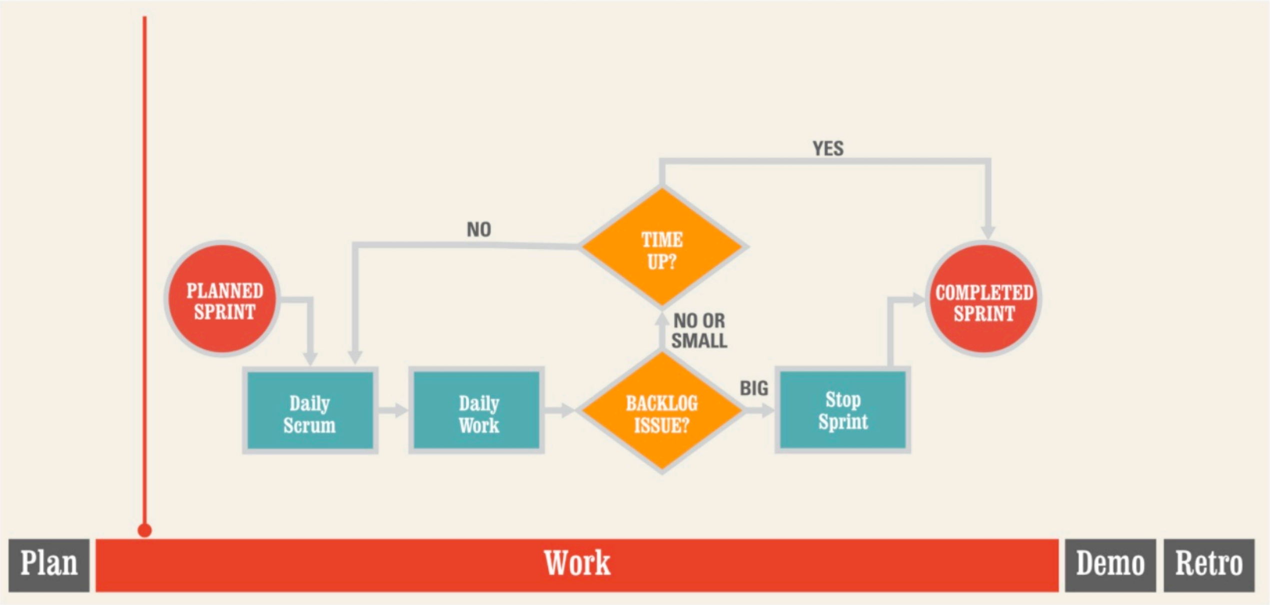
⚠ Roles in Scrum

- → Scrum Master
- → Product owner
- → development team

⚠ Scrum Planning



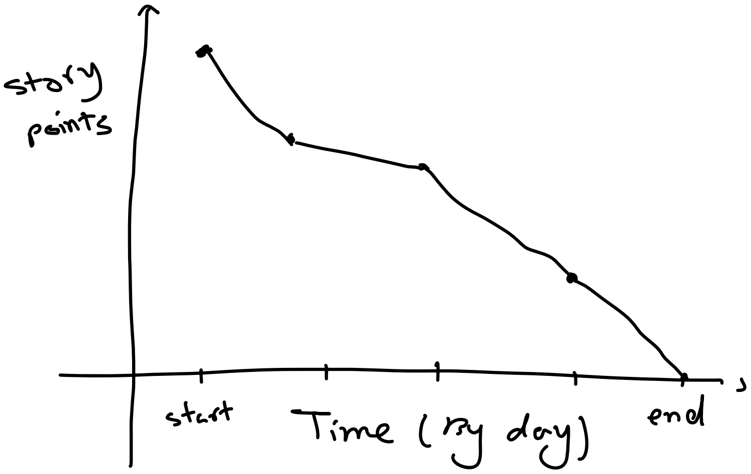
⚠ Sprint Execution



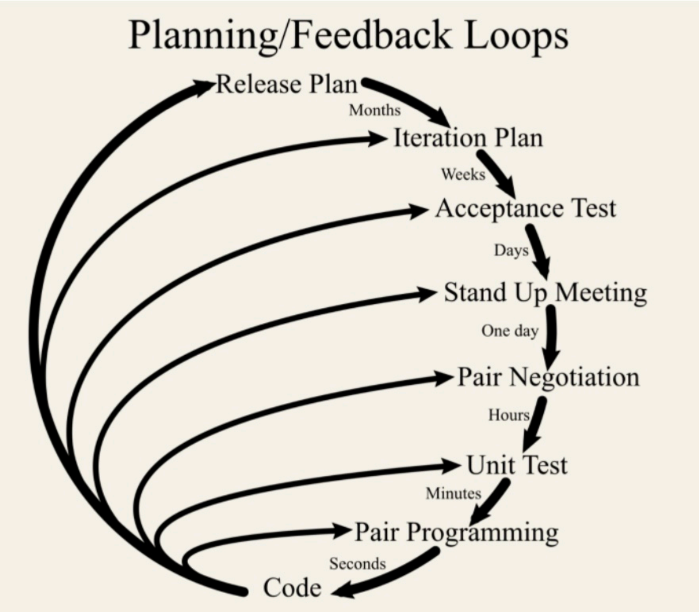
⚠ Burndown chart

↓

to track progress



⚠ XP framework



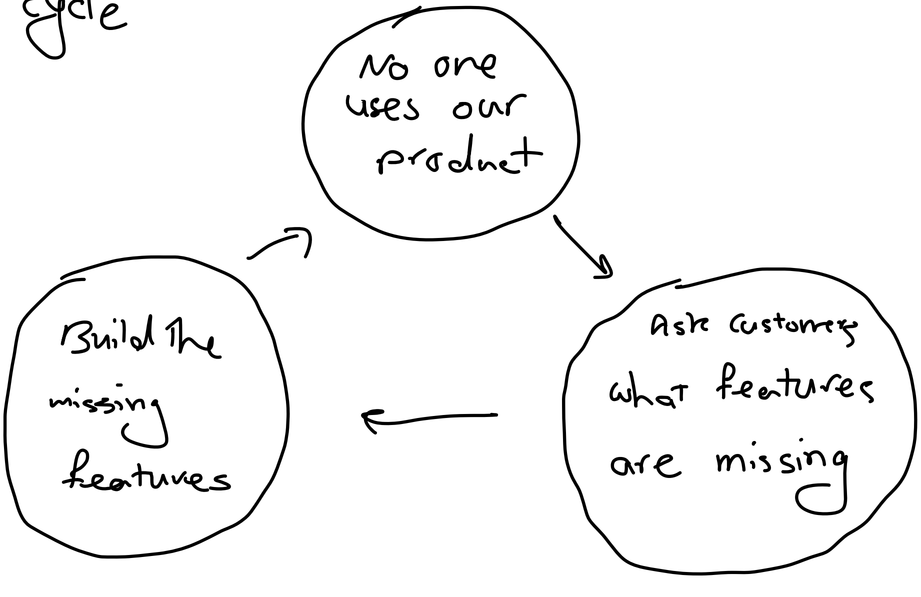
⚠ Values in XP

- Communication
- Simplicity
- Feedback
- Courage

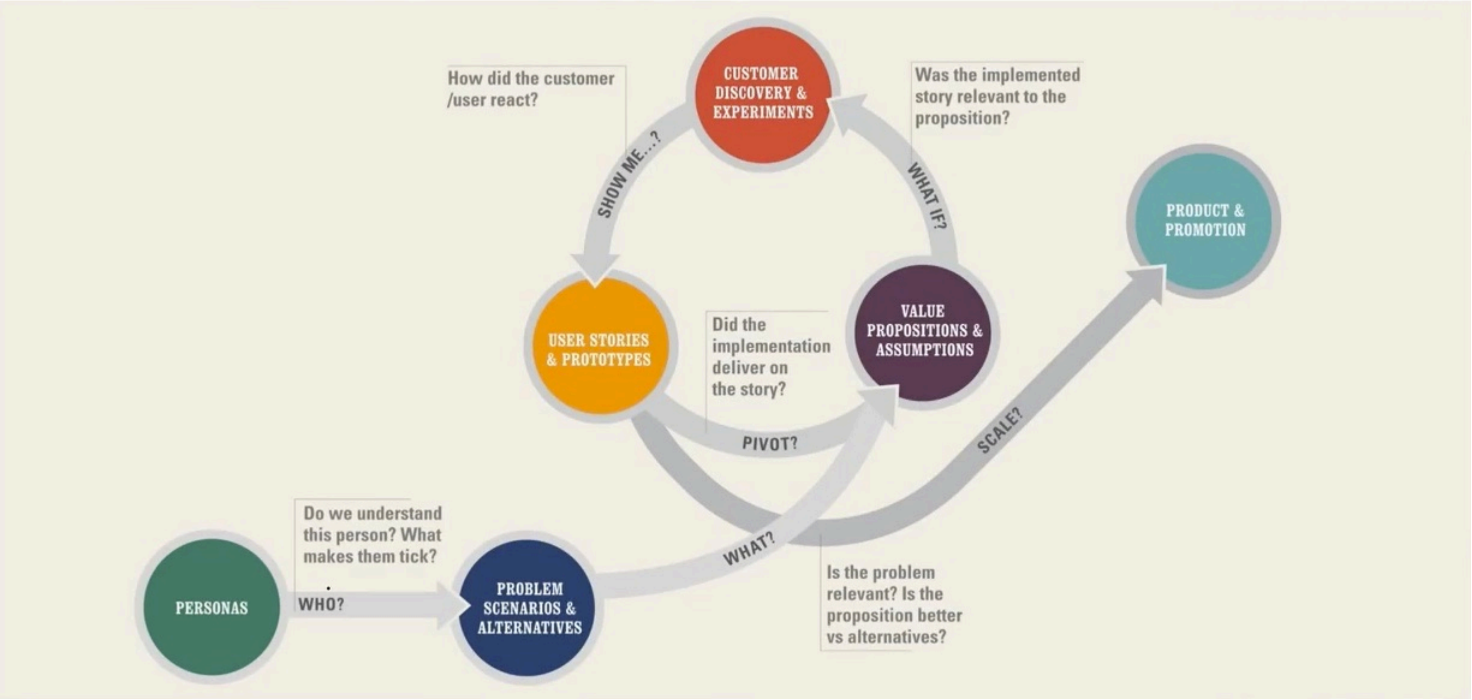
⚠ Six principles in Kanban

- ① visualize work and workflow
- ② make policies explicit (but not permanent)
- ③ Limit work in progress (WIP)
- ④ implement feedback loops
- ⑤ avoid stackups, encourage collaboration
- ⑥ improve collaboratively, evolve experimentally

⚠ Product death cycle


No one uses our product
Ask customers what features are missing
Build The missing features

⚠ The Venture design process


CUSTOMER DISCOVERY & EXPERIMENTS
SHOW ME...?
WHAT IF?
PRODUCT & PROMOTION
USER STORIES & PROTOTYPES
VALUE PROPOSITIONS & ASSUMPTIONS
PIVOT?
SCALE?
WHAT?
PERSONAS
WHO?
PROBLEM SCENARIOS & ALTERNATIVES

How did the customer /user react?

Was the implemented story relevant to the proposition?

Did the implementation deliver on the story?

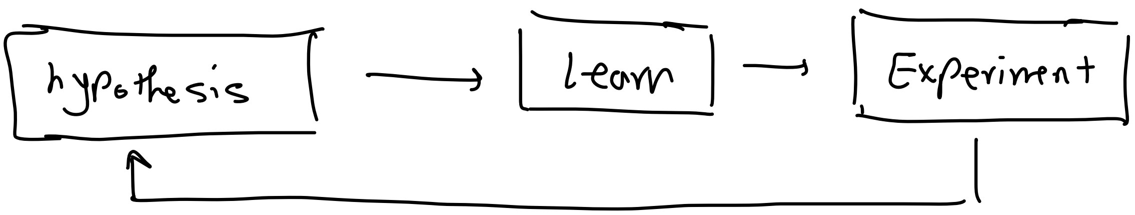
Do we understand this person? What makes them tick?

Is the problem relevant? Is the proposition better vs alternatives?

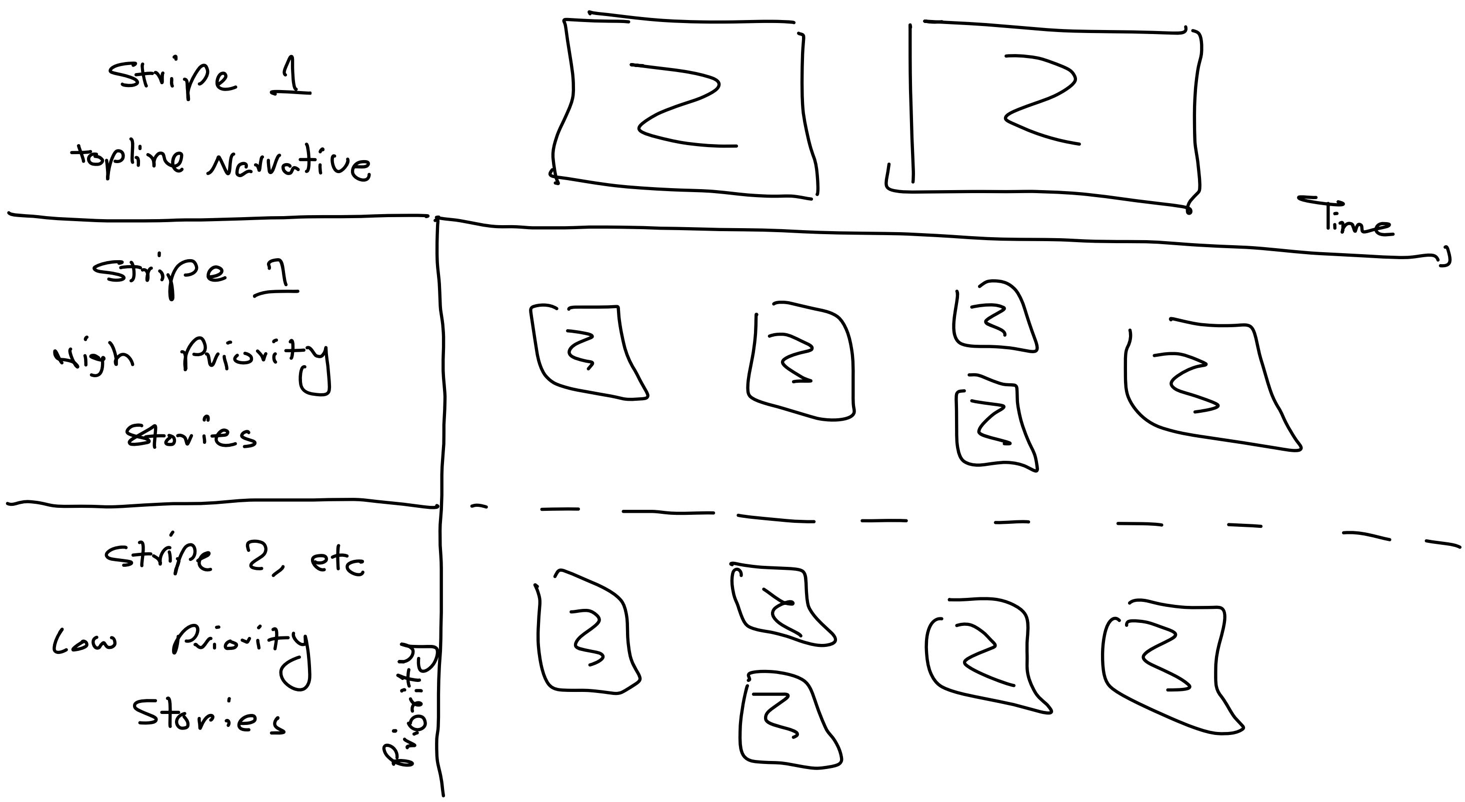
⚠ INVEST-able stories

- independent
- negotiable
- valueble
- estimable
- small
- testable

⚠ learning Loop



⚠ User Story mapping



⚠ Daily Scrum Agenda

- What did I accomplish yesterday?
- what will I accomplish today?
- what issues are impeding my Progress?

⚠ How to define success in software development

- Not accumulating technical debt
- Code has relatively low defect rate
- Prudent use of automation (test, integration)
- releasing is frequent, relatively Painless
- work in progress flows well

⚠ How to define success in managing with agile

- Practices don't require (much) pushing
- principles are visible in outcomes
- evolving collaborativelly, experimentally
- stakeholders are satisfied
- few unplanned suprises (internal)

⚠ How to create an agile friendly environment

- drive narrative collaboration
- release a lot
- invest in outomating repetitive tasks
- create a culture of experimentation