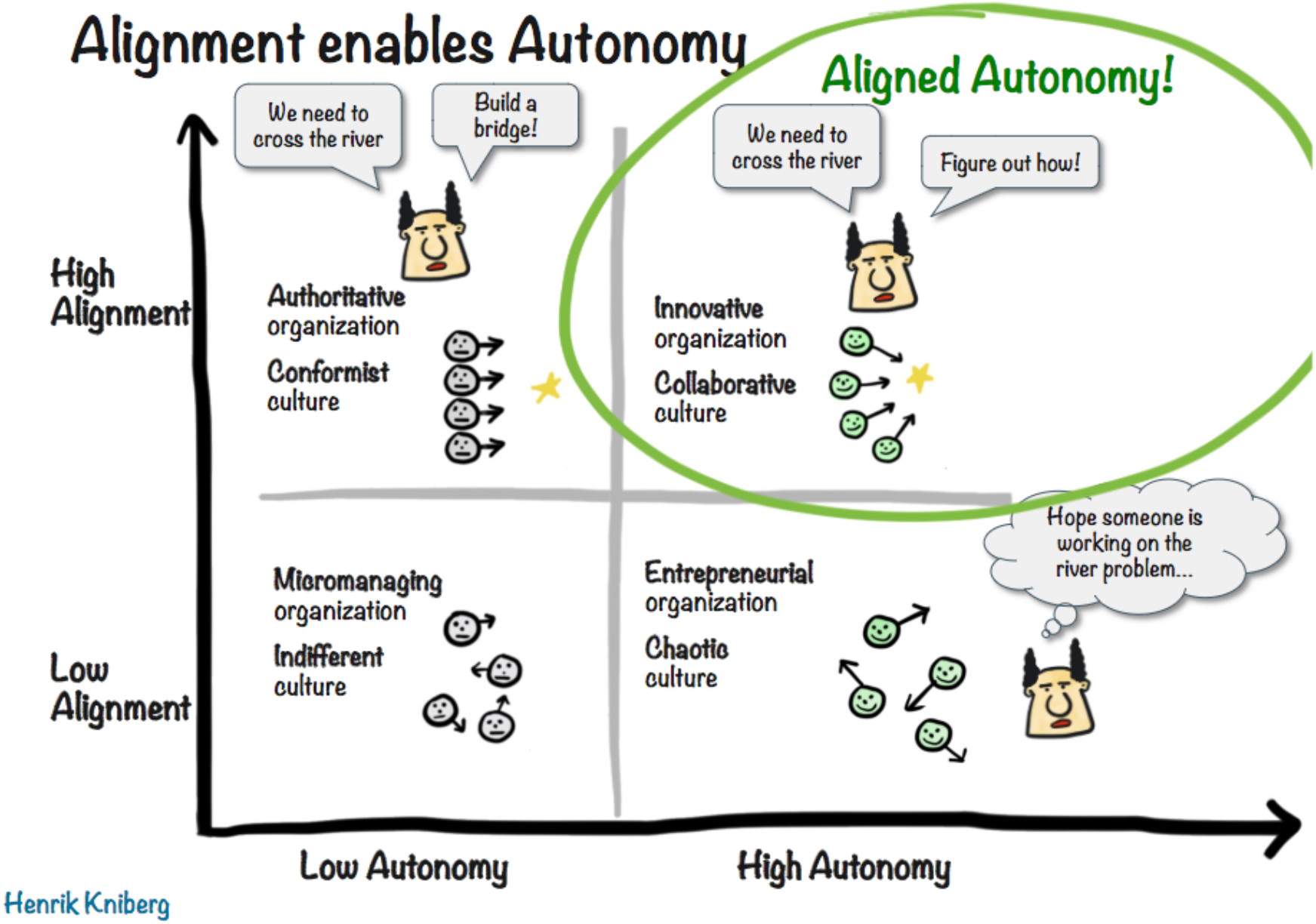
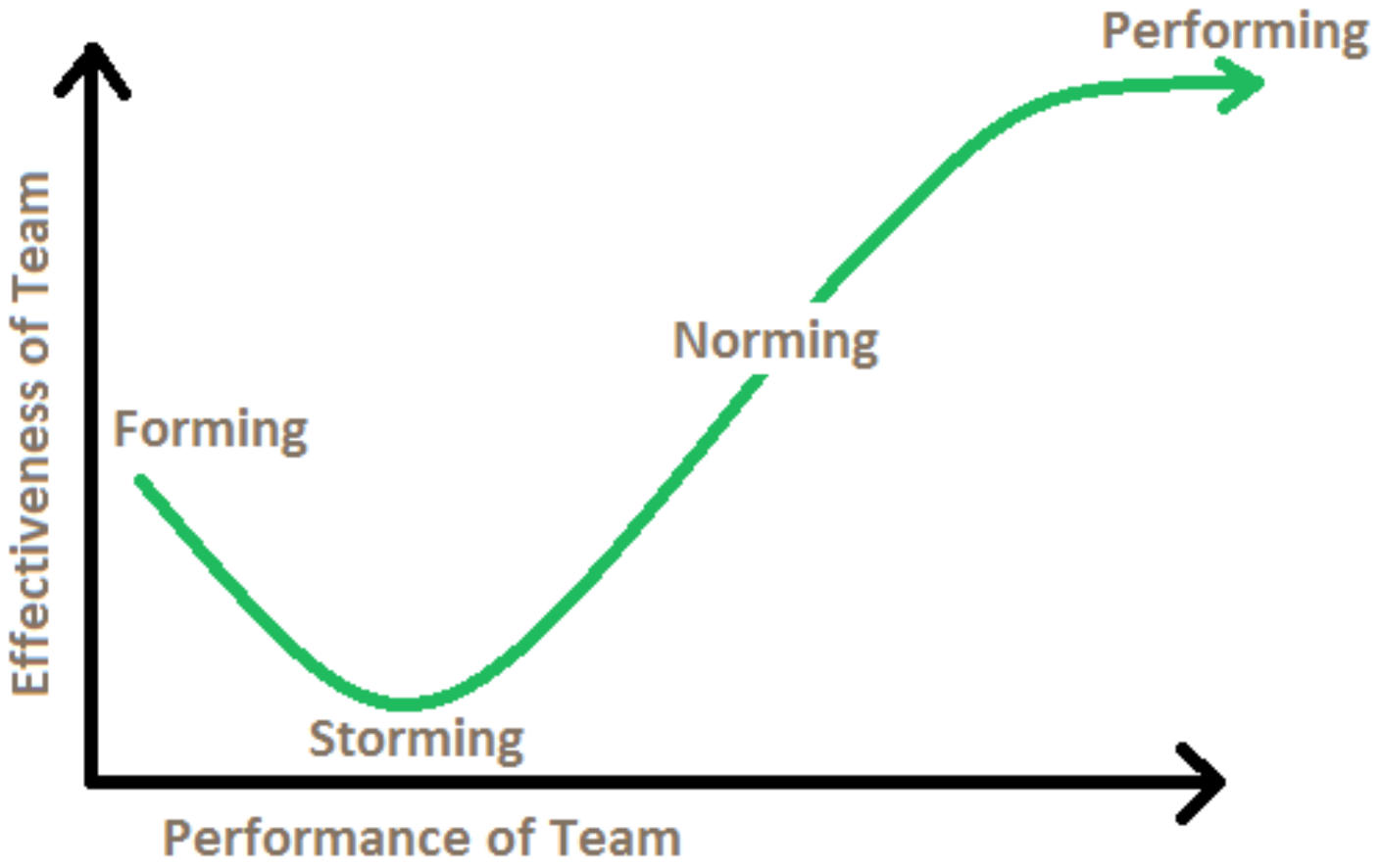
Methods & tools

This is a course where you should apply methods for product- and service design and development.


Alignment enables Autonomy
Aligned Autonomy!
We need to cross the river
Build a bridge!
We need to cross the river
Figure out how!
High Alignment
Low Alignment
Authoritative organization
Conformist culture
Innovative organization
Collaborative culture
Micromanaging organization
Indifferent culture
Entrepreneurial organization
Chaotic culture
Hope someone is working on the river problem...
Low Autonomy
High Autonomy
Henrik Kniberg

Tuckman's Team & Group Development Model



FORMING

Behaviors

- The purpose and goals for the team are unclear.
- Members feel varying degrees of commitment.
- Members are cautious, don't initiate and avoid responsibility.
- Communication is low and a few members often dominate.
- Members are dependent on directive leadership.

Tasks

- Build a common purpose. Clearly establish the expectations of the customers or sponsors.
- Understand personal expectations and interests.
- Clarify accountability, recognition, and rewards.
- Assess resources; see who has what to contribute.
- Leader provides direction and drives the team process.

STORMING

Behaviors

- Differences and confusion arise over goals and roles.
- Struggles erupt over approaches, direction, and control.
- Team members react toward leadership with counterproductive behaviors.
- Team is uncertain about how to deal with issues openly.
- Team wrestles with issues of communication.
- Members act from an independent stance.

Tasks

- Involve everyone in the discussion.
- Inquire into differences; include all ideas and opinions.
- Seek further clarity about purpose and develop a common approach to meeting project objectives.
- Assess and test resource needs; make necessary adjustments.
- Define operational agreements (norms).
- Leader raises difficult issues and coaches team through struggles.

NORMING

Behaviors

- Team gains confidence, feels a sense of momentum.
- "What," "How," "Who," and "When" become clarified.
- Team develops agreements on approaches, goals, communication, and leadership roles.
- Team builds relationships with externals (customers, key stakeholders).
- Members begin to relate interdependently.

Tasks

- Develop processes for information sharing, feedback, and resource distribution.
- Have open forums on tasks and relationships, both internal and external.
- Build appropriate feedback loops with external relationships.
- Work toward consensus on overarching issues. Negotiate where appropriate.
- Leader uses a facilitative style to create the opportunity for others to lead.

PERFORMING

Behaviors

- Members take full responsibility for tasks and relationships.
- Team achieves effective and satisfying results.
- Team takes the initiative to continually assess external forces.
- Team facilitates itself easily through the various stages.
- Members work proactively for the benefit of the team.

Tasks

- Continuously seek to improve tasks and relationships.
- Assess and evaluate results against purpose and external forces.
- Celebrate successes—reward and recognize both team and individuals wins.
- Continuously test for better methods and approaches.
- Leader focuses on purpose, interdependent relationships, and conditions that shift the stages.

www.catalystonline.com

Select

Minimum 1 agile method for product discovery (what)
Minimum 1 agile method for product delivery (how)
As many other methods/tools you need for your project

They should be challenging for all different competences in the team

Write the *Methods* chapter

What methods you selected
Briefly explain the methods and how you plan to use them in your project
Why you selected those methods
(What team members are responsible for what methods?)
(Can be included in the project plan and later in the project report.)

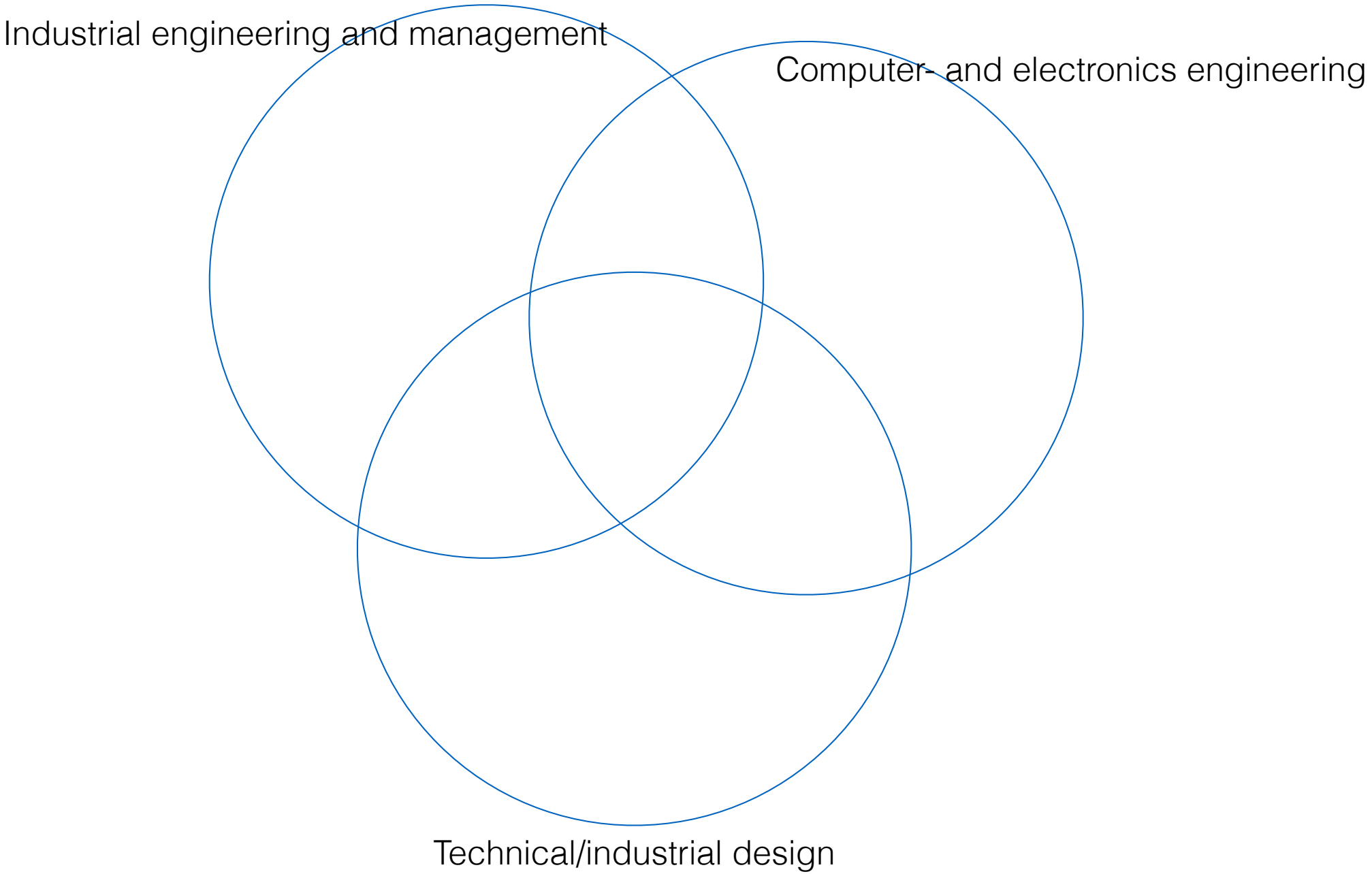
Exercise

1. Discuss in smaller groups (2-3) within the same skills for 10 minutes.
- Industrial engineering and management
- Computer science and electronics (year 3, 5 and international students)
- Technical/industrial design

- What methods and/or tools have you used in previous courses that can be used in product development projects?
- Any methods/tools you have experienced outside school?

2. Write down the name of each method/tool.

3. When presenting - Read the notes out loud.


Industrial engineering and management
Computer- and electronics engineering
Technical/industrial design

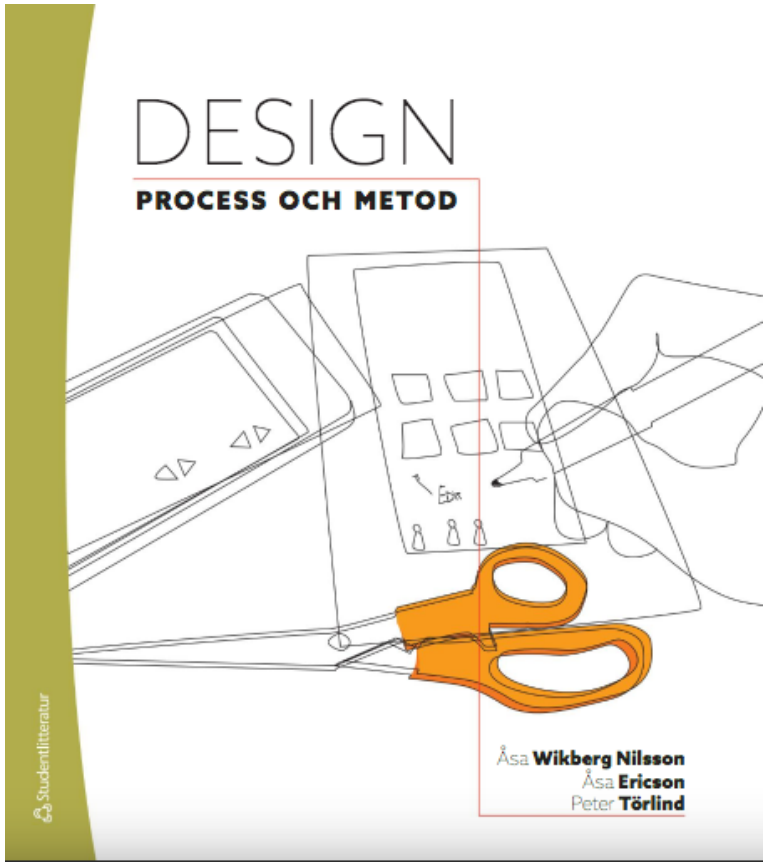
- What methods do they use in the company?

Methods used by Miun Innovation?

Marshmallows challenge
Disappearing conditions (7 minutes - 7 solutions)
NABC
Business Model Canvas
(Value Proposition Canvas)

Some ”agile” methods

The design process
Scrum
Kanban
XP
Lean UX
Specification by example
Pretotyping
Test driven design
Lean Startup
Personas
Customer Journeys
Effektstyrning av IT



METODER

Projektanalys 43
Mindmapping 45
Projektplanering 47
Marknadsanalys 49
Framtidsanalys 51
Hållbarhetsanalys 53
Intervju 83
Deltagande observation 85
Fokusgrupp 87
Workshop 89
Probes 91
Ergonomianalys 93
Personas 95
Behovsanalys 97
Kundresa 99
Kollage 101
Braindraining 121
Kreativitetssnurran 123
Brainstorming 125
Brainwriting 6-3-5 127
Braindrawing 129
Sandbox play 131
Scamper 133
Speedstorming 135
World café 137
Analogier 139
Scenarier 141
Dark horse 143
Six thinking hats 145
Morfologisk matris 147
Occams razor 149
Idéutvärdering 151
Upplevelseprototypa 205
Branding 207
Legometoden 209
Kanomodellen 211
Scenariobaserad utvärdering 213
Konceptvalsmatris 215
Värdemetoden 217
Konceptviktningsmatris 219

Other

Human Computer Interaction
Materialval/krav, kan vara fysiska, miljömässiga, estetiska eller företagsidentitet
Modellteknik/prototypframställning
Relevant ISO-standards
Rapid prototyping, 3d printing
Electronic Kanban/Scrum-boards
Development tools for open source code (versioning, task management, etc)
Data mining-tools,
Data warehousing-platforms
Licences for open source code or open hardware
Crowd sourcing
UML
Kano model
Affärssystem / affärsprocesser (enterprise systems)
Gamification

Leif:

- simulation
- modulation
- faktoranalys
- statistical methods
- multivariat dataanalys

Aron:

- tillfällighetsteori
- venture cup template for business models
- värdering av produkt
- kassaflödesanalys (eng discounted cash flow)
- kapitalbehovanalys (eng venture capital analysis)
- kompetenskartläggning (eng competence mapping)
- SWOT-analysis
- Value based decision theory

Mittuniversitetet
MID SWEDEN UNIVERSITY

Product Development =

Product Discovery
decide WHAT to build

doing the right thing

+

Product Delivery
decide HOW to build it

doing things right

Some previous SIMS-projects…

Mittuniversitetet
MID SWEDEN UNIVERSITY

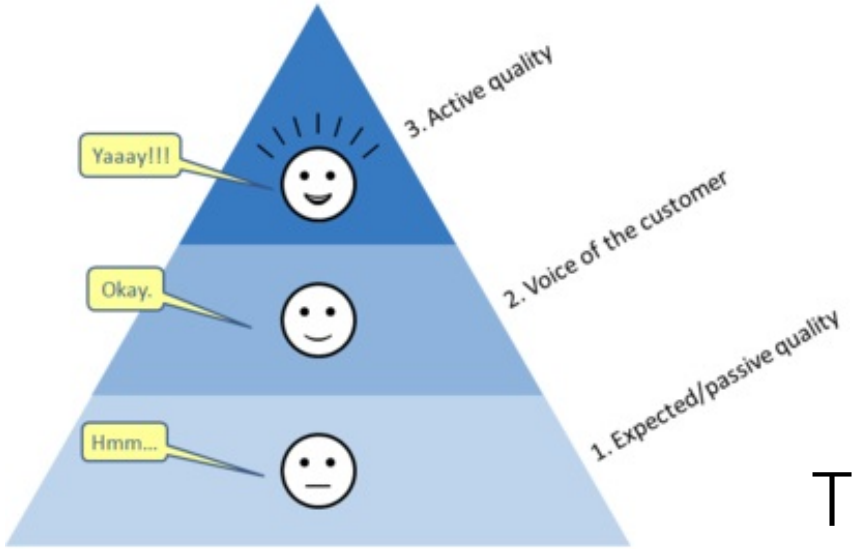
Methods

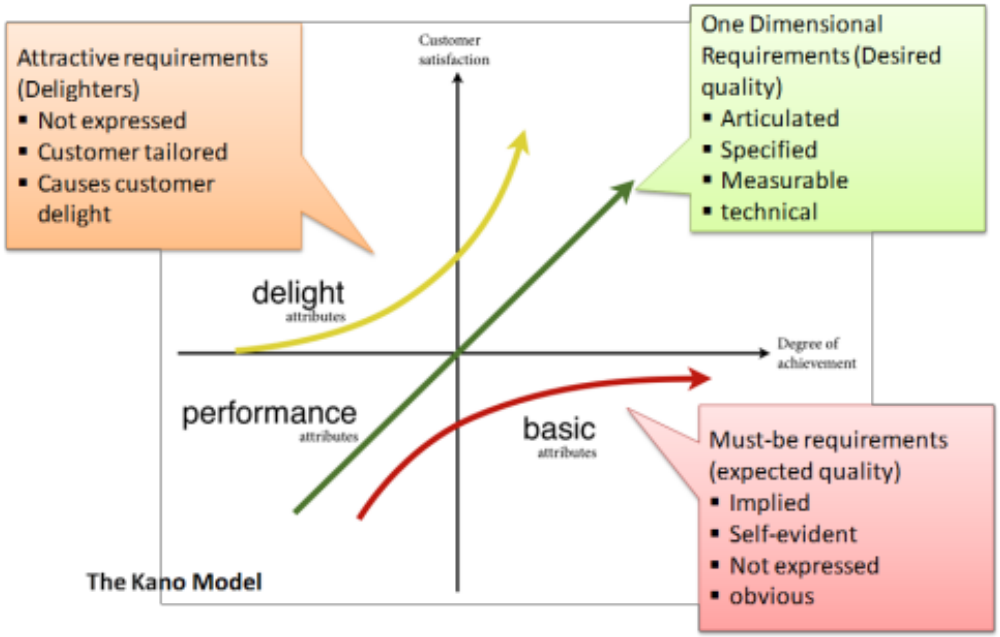
- Blue ocean strategy
- Brainwriting
- Extreme programming
- Kanban
- Kano model
- NABC
- Personas
- SWOT-analysis

2. Metod 2
2.1. Innovation- och sammarbetsövningar 3
2.2. Projektplan 4
2.3. Scrum 6
2.4. Kanban 6
2.5. Marknadsundersökning 6
2.6. 6D 7
2.7. MDI 9
2.8. Användartester 10
Test 1 (16/11-2015) 11
Test 2 (1/12-2015) 12
Test 3 (7/12-2015) 13
2.9. Prototyputveckling 14
2.9.1. Behörigheter 14
2.9.2. Arkitektur 16
2.9.3. Plattform 16
2.9.4. Uppbyggnad 18
2.9.5. Process för skapandet av applikation 18
2.10. Affärsmodell 21

2 Methodology
2.1 Impact mapping
2.2 The 6D Framework
2.3 Gantt Scheme
2.4 Scrum
2.5 Applied methodology
2.6 Motivation
2.7 Ethics
2.8 Previous research
3 Results
3.1 Initiation phase
3.1.1 Impact mapping
3.1.2 6D Gamification Framework
3.2 Planning phase
3.2.1 Gantt scheme
3.2.2 Risk assessment
3.2.3 Design plan
3.2.4 Technical plan
3.2.5 Test Plan
3.2.6 Business plan

Some additional methods (examples)…





The Kano model

The Kano model addresses three types of customer requirements:

Satisfying basic needs: allows a company to get into the market

Satisfying performance needs: allows a company to sustain and stay competitive

Satisfying excitement needs: allows a company to excel and be world class

Pretotyping

Test your assumption as early as possible.

Jeff Hawkins was a Palm co-founder and one of the inventors of the Palm Pilot. Before committing to develop the Pilot, Jeff made and carried a mock-up (a block of wood) in his pocket for several weeks.

https://docs.google.com/file/d/0B0QztbuDlKs_ZTk2M2RhZWItYzk3YS00ZDZmLTgyZjItY2Y2ZWIyYjZkOTE3/edit?hl=en_US

”6D” Gamification Design Framework by Kevin Werbach

1. Define business objectives

2. Delineate target behaviours

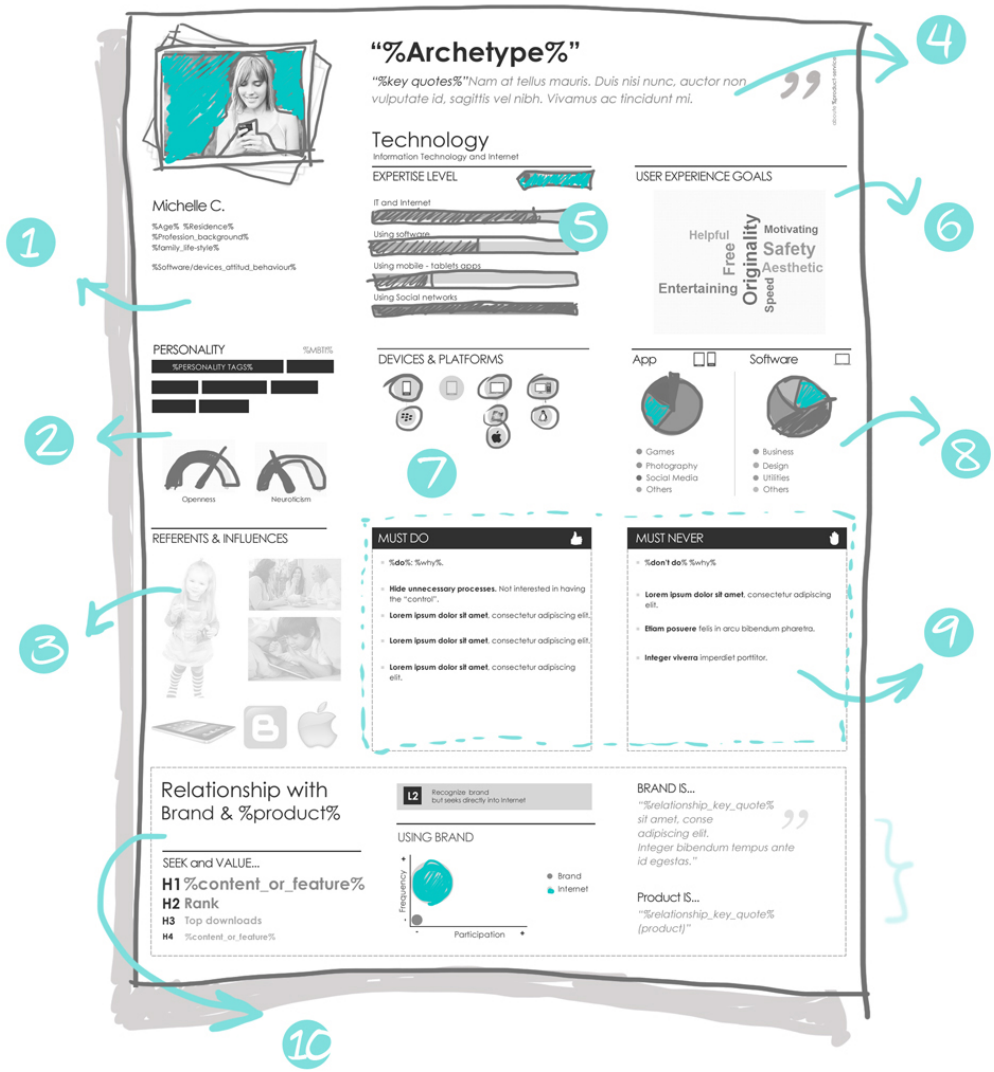
3. Describe your players

4. Devise activity loops

5. Don’t forget the fun! – easy 2 miss when you obsess on the structure

6. Deploy the appropriate tools

10 elements for User Persona



1- Profile

2- Personality

3- Referents & Influences

4- Archetype & quotes

5- Technology expertise

6- User Experience Goals

7- used device and platforms

8- Domain details

9- Must Do - Must Never

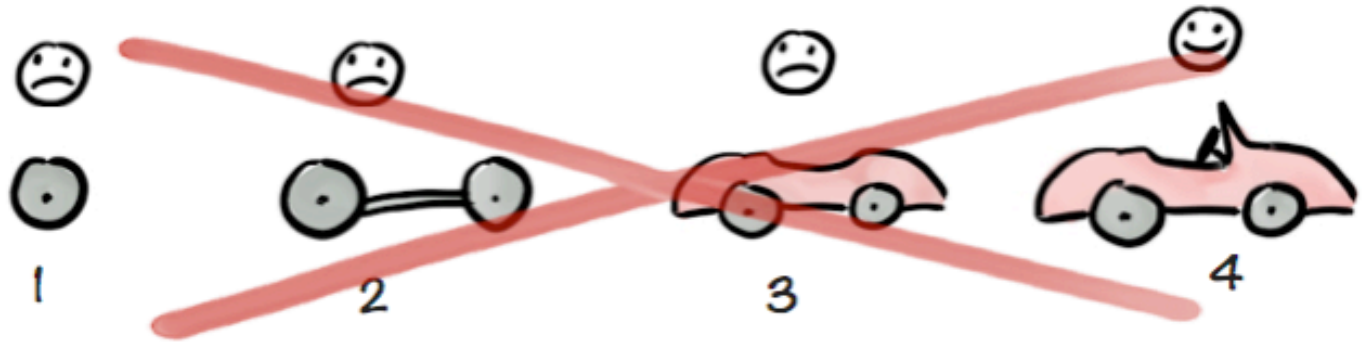
10- Brand & Product relationship

Project plan

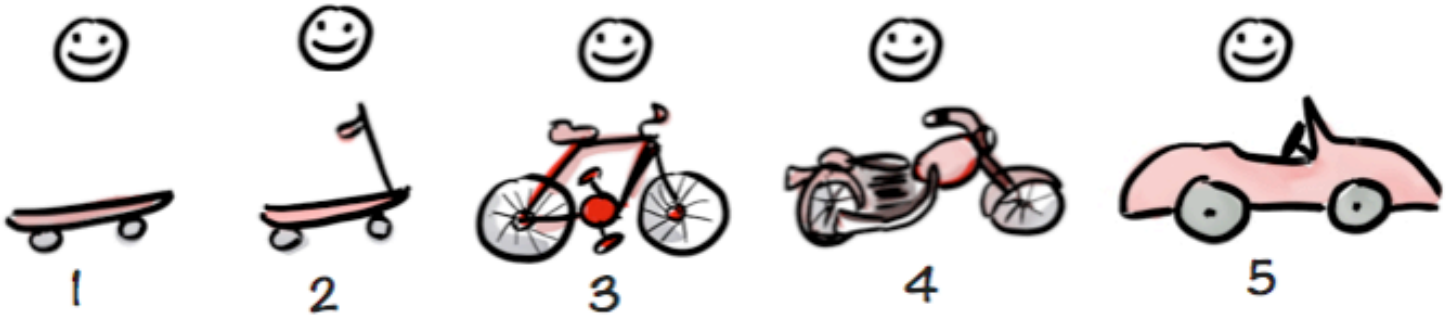
You may use any template or instruction that you've seen in previous courses. Normally they include:

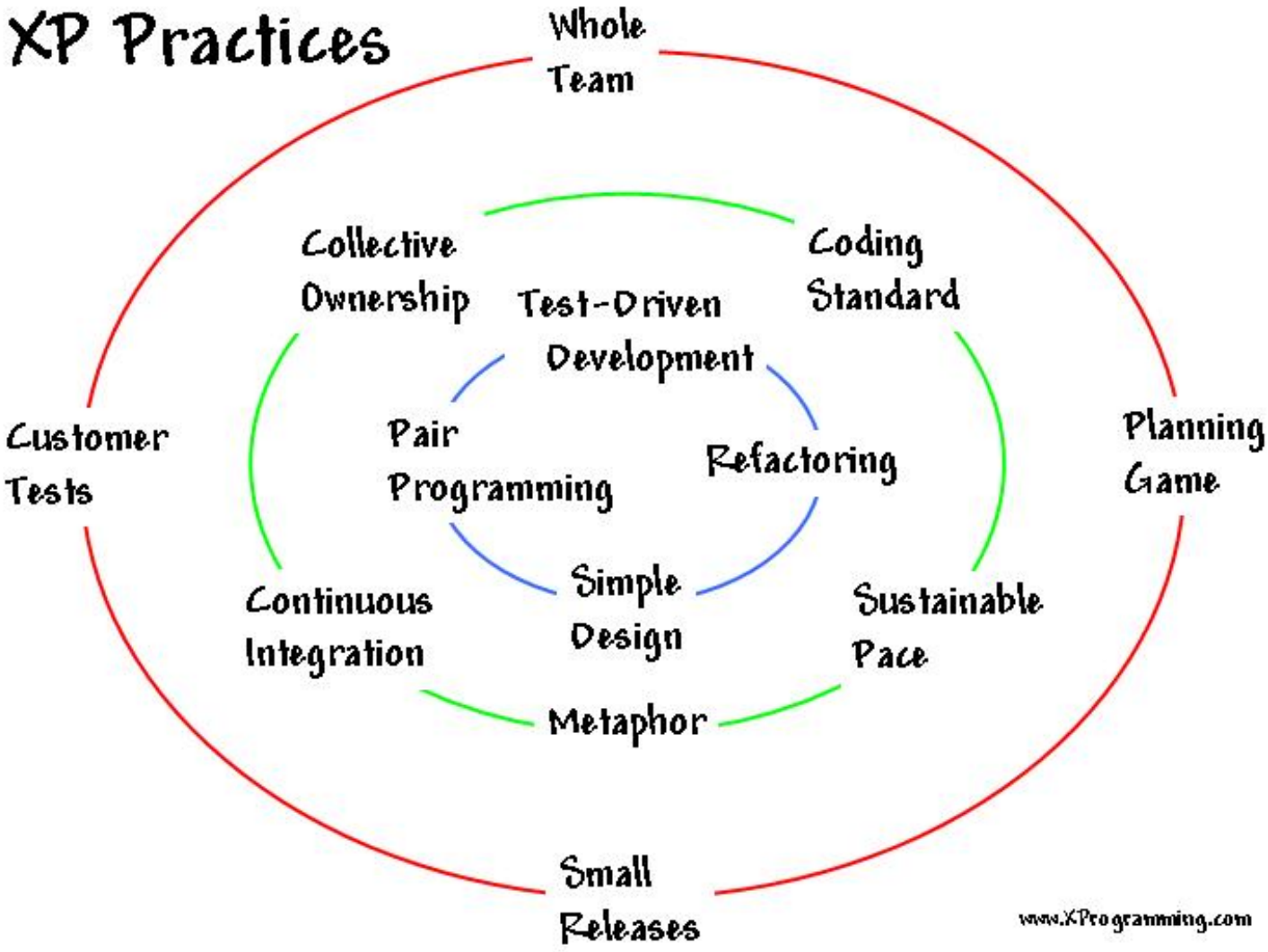
Project goals
Resources
Time
Activities/deadlines
(Methods/tools)

Not like this.....

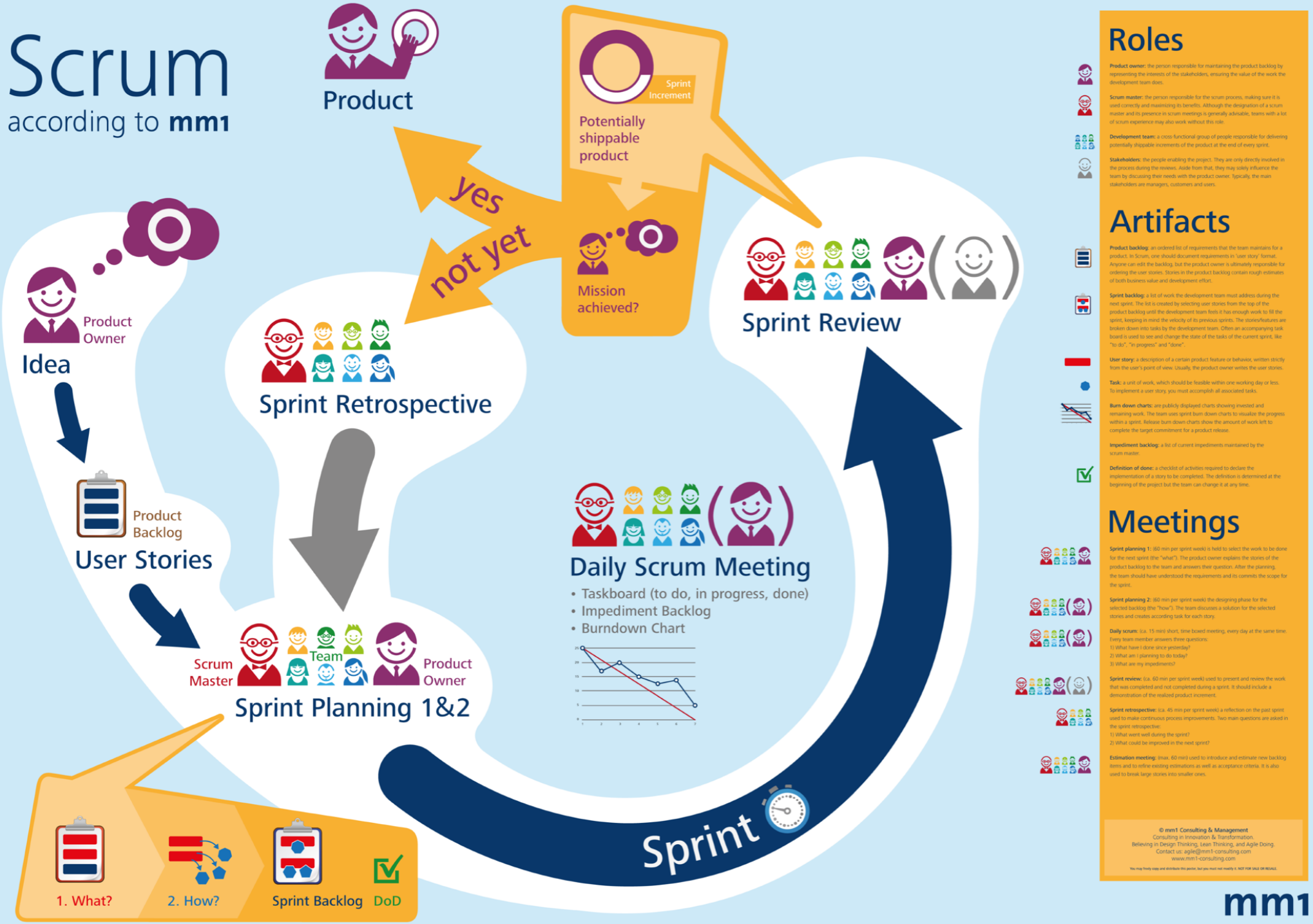


Like this!




XP Practices
Whole Team
Collective Ownership
Coding Standard
Test-Driven Development
Customer Tests
Pair Programming
Refactoring
Planning Game
Continuous Integration
Simple Design
Sustainable Pace
Metaphor
Small Releases
www.XProgramming.com

Scrum
according to **mm1**



Roles

Product owner: the person responsible for maintaining the product backlog by representing the interests of the stakeholders, ensuring the value of the work the development team does.

Scrum master: the person responsible for the scrum process, making sure it is used correctly and maximizing its benefits. Although the designation of a scrum master and its presence in scrum meetings is generally advisable, teams with a lot of scrum experience may also work without this role.

Development team: a cross-functional group of people responsible for delivering potentially shippable increments of the product at the end of every sprint.

Stakeholders: the people enabling the project. They are only directly involved in the process during the reviews. Aside from that, they may solely influence the team by discussing their needs with the product owner. Typically, the main stakeholders are managers, customers and users.

Artifacts

Product backlog: an ordered list of requirements that the team maintains for a product. In Scrum, one should document requirements in "user story" format. Anyone can edit the backlog, but the product owner is ultimately responsible for ordering the user stories. Stories in the product backlog contain rough estimates of both business value and development effort.

Sprint backlog: a list of work the development team must address during the next sprint. The list is created by selecting user stories from the top of the product backlog until the development team feels it has enough work to fill the sprint, keeping in mind the velocity of its previous sprints. The stories/features are broken down into tasks by the development team. Often an accompanying task board is used to see and change the state of the tasks of the current sprint, like "to do", "in progress" and "done".

User story: a description of a certain product feature or behavior, written strictly from the user's point of view. Usually, the product owner writes the user stories.

Task: a unit of work, which should be feasible within one working day or less. To implement a user story, you must accomplish all associated tasks.

Burn down charts: are publicly displayed charts showing invested and remaining work. The team uses sprint burn down charts to visualize the progress within a sprint. Release burn down charts show the amount of work left to complete the target commitment for a product release.

Impediment backlog: a list of current impediments maintained by the scrum master.

Definition of done: a checklist of activities required to declare the implementation of a story to be completed. The definition is determined at the beginning of the project but the team can change it at any time.

Meetings

Sprint planning 1: (60 min per sprint week) is held to select the work to be done for the next sprint (the "what"). The product owner explains the stories of the product backlog to the team and answers their question. After the planning, the team should have understood the requirements and its commits the scope for the sprint.

Sprint planning 2: (60 min per sprint week) the designing phase for the selected backlog (the "how"). The team discusses a solution for the selected stories and creates according task for each story.

Daily scrum: (ca. 15 min) short, time boxed meeting, every day at the same time. Every team member answers three questions:
1) What have I done since yesterday?
2) What am I planning to do today?
3) What are my impediments?

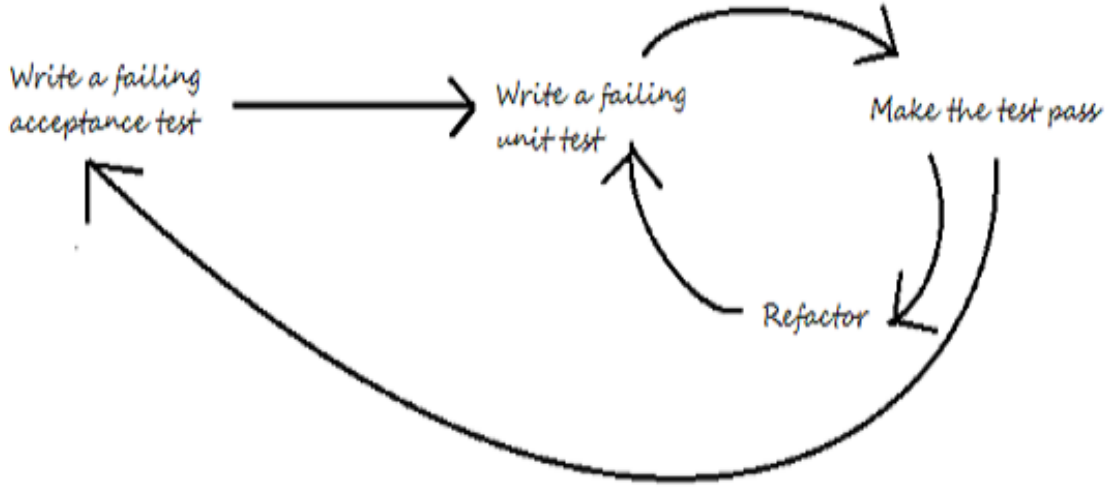
Sprint review: (ca. 60 min per sprint week) used to present and review the work that was completed and not completed during a sprint. It should include a demonstration of the realized product increment.

Sprint retrospective: (ca. 45 min per sprint week) a reflection on the past sprint used to make continuous process improvements. Two main questions are asked in the sprint retrospective:
1) What went well during the sprint?
2) What could be improved in the next sprint?

Estimation meeting: (max. 60 min) used to introduce and estimate new backlog items and to refine existing estimations as well as acceptance criteria. It is also used to break large stories into smaller ones.

Consulting in Innovation & Transformation.
Believing in Design Thinking, Lean Thinking, and Agile Doing.
Contact us: agile@mm1-consulting.com
www.mm1-consulting.com

mm1



TDD - Test driven design