DAT255 / DIT543 SOFTWARE ENGINEERING PROJECT

TODAY

Reflections

Scrum

Practical Stuff

REFLECTIONS

3: No remarks
2: Minor remarks
1: Major remarks
0: Failed delivery

NO REMARKS

What is
What might or should be
Feedback to reduce the gap

MINOR REMARKS

Loss in detail
E.g. no roles in planning phase
or emphasis on What is

MAJOR REMARKS

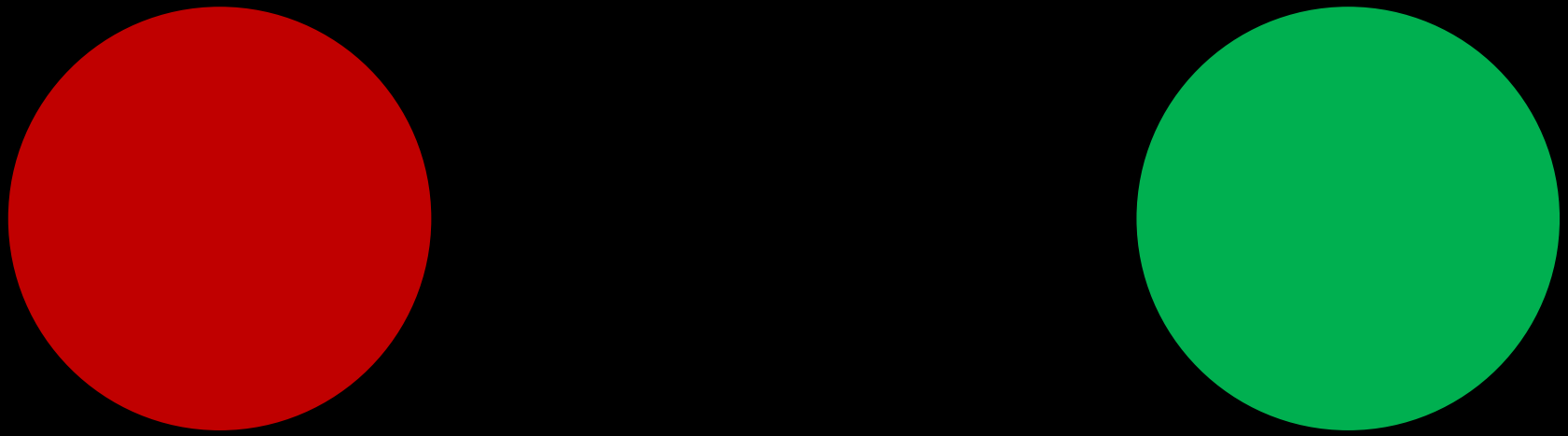
ONLY MENTIONS PROBLEM

“T H A T I S O N L Y T O B E E X P E C T E D

FAILED DELIVERY

“THE SPRINTS WHERE TOO SHORT”

“NEXT TIME WE’LL HAVE A MORE COMPLETE UNDERSTANDING UP-FRONT”



Strategi för kvalité
Under legoövningen blev våra leveranser inte godtagna efter första sprinten, men det blev de i både andra och tredje.
Då kunde vi däremot hela tiden kolla med produktägaren om något vi gjorde höll måtten.
Det gjorde det väldigt tydligt om det vi byggde var stabilt och konsekvent.

Så är inte fallet under detta projekt: det kommer inte vara lika tydligt om produkten i projektet är bra jämfört med om ett legobygge är klart och det är inte möjligt att ständigt fråga produktägaren för att undersöka detta.

Som KPI För att mäta vilken kvalitet leveranserna håller tänker vi att mäta hur stor andel av alla rader av vår kod som körs av de enhetstester som vi bygger.
Desto större andelen är, desto större är sannolikheten att vi vet vad vår kod gör.

En brist med detta KPI är att det är beroende av våra enhetstester, och det kan finnas fall som vi inte har gjort lämpliga enhetstester för.

The last learning is that SCRUM isn't something you just pick up and get to work with in a second, it takes time for the team to learn how to use it. I look forward to learn how to use it during this course and in my future.

Vårt mål är att lägga ner mycket tid tidigt i projektet för att minska arbetet med att refaktorera koden i slutet. KPI blir således att vi kategoriserar den typ av arbete vi gör, framförallt då refaktorering och skapande av ny funktionalitet. Vi tänker att ett optimalt resultat är 0, då allt över det innebär refaktorering. Då vi vet att det kommer behöva läggas tid på refaktorering bör riktmärket sättas högre än så för att skapa ett KPI som faktiskt är rimligt att använda.

The chosen strategies, with related KPIs are:

Daily Scrum

→ Estimation Accuracy, Happiness

Trello

→ Estimation Accuracy, Git

Retrospectives and Surveys

→ Happiness

The chosen KPIs are:

Estimation Accuracy

→ Burn down/up chart

Happiness

→ Survey

Git

→ Autogenerated

It was also important to ask leading questions like
“Do you want the color blue or red?”,
rather than asking open questions like
“Do you want to add something?”
to prevent additional requirements from the product owner, that was not in the initial requirements.

TEACHER PERSPECTIVE

WHAT IS

- New rooms
- Nr of students
- Scheduling

WHAT MIGHT / SHOULD BE

- Lots of space
- Keep to schedule

FEEDBACK

- Keep rooms
- Encourage even distribution
- Plan for 4h

KPI

- Mean survey response
- # of negative comments

QA

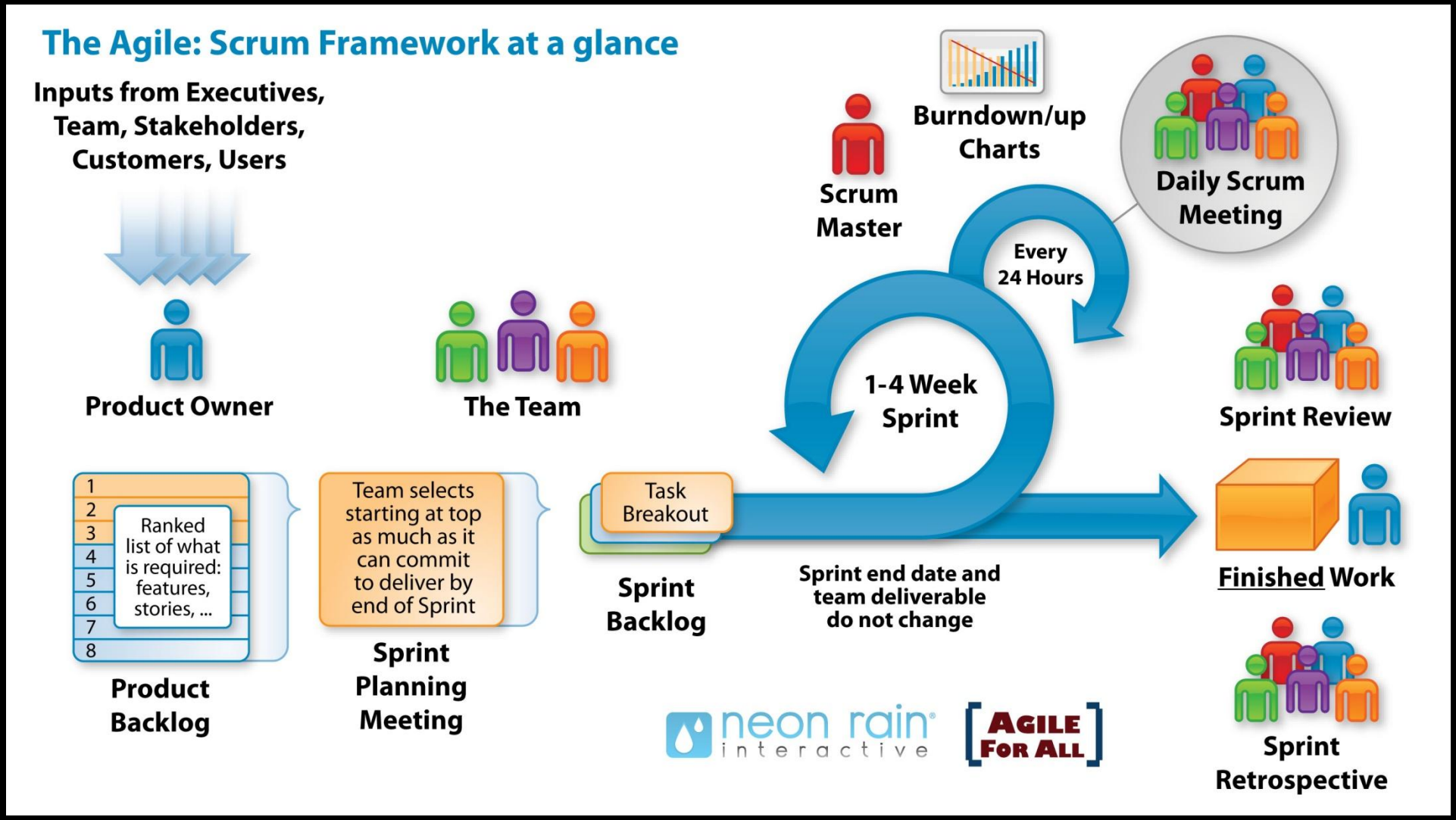
'Questions don't have to make sense, Vincent', said Miss Susan.

'But answers do'

Terry Pratchett
Thief of Time, 2001


SCRUM

SCRUM


The Agile: Scrum Framework at a glance
Inputs from Executives, Team, Stakeholders, Customers, Users
Product Owner
The Team
Scrum Master
Burndown/up Charts
Daily Scrum Meeting
Every 24 Hours
1-4 Week Sprint
Sprint Review
1
2
3
4
5
6
7
8
Ranked list of what is required: features, stories, ...
Product Backlog
Team selects starting at top as much as it can commit to deliver by end of Sprint
Sprint Planning Meeting
Task Breakout
Sprint Backlog
Sprint end date and team deliverable do not change
Finished Work
Sprint Retrospective
neon rain interactive
AGILE FOR ALL

SCRUM BACKLOGS

Product backlog:

Prioritised list of **all** product requirements

Product backlog can be continuously updated by the **Product Owner**

Sprint backlog:

List of requirements selected **for the current sprint**

Sprint backlog contains more detailed information – **tasks**

Sprint backlog is only updated by **Scrum Team**

Backlog Items

User stories
As an **X** I want **Y** since **Z**

"As a **entrepreneur** I want **a portable booth** to be able to **sell sausages to hungry citizens wherever they happen to be**"

"Sausages!"

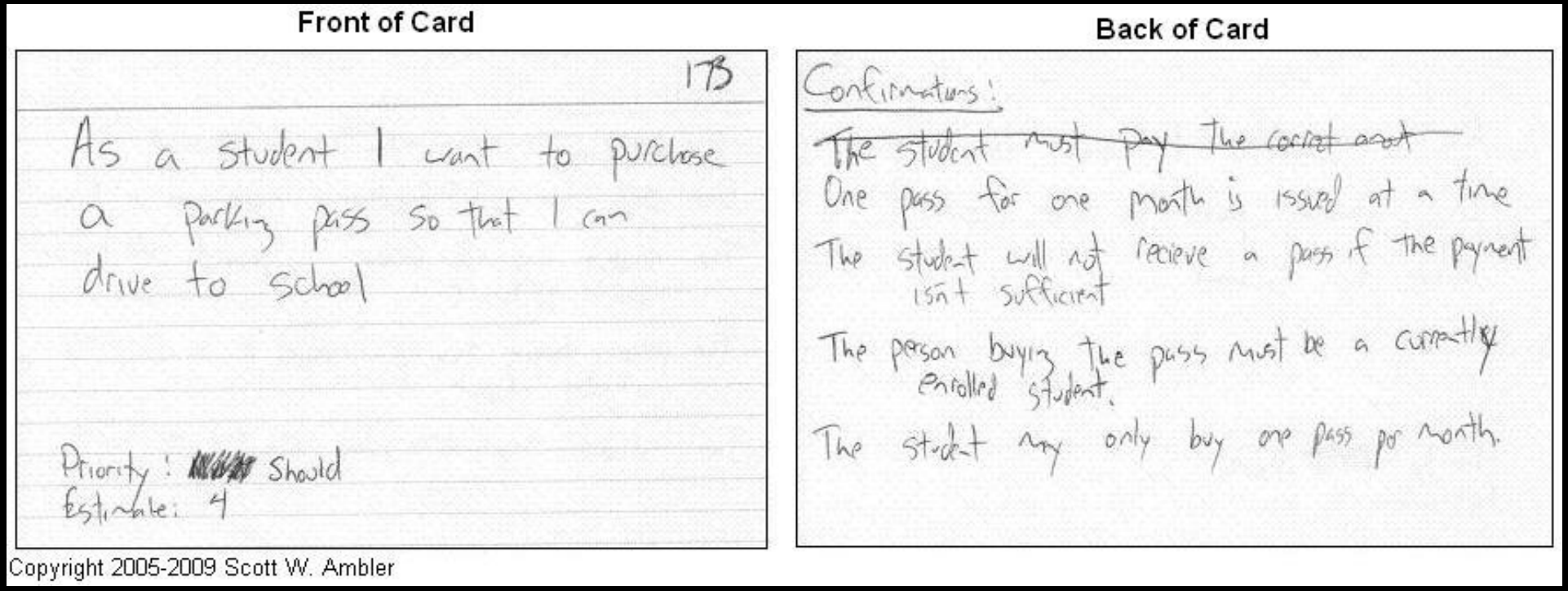
Epics:
Largestoriesthatcan'tbe delivered within one sprint

Themes:
Something that is common to a number of backlog items

- Roads and bridges
- Work places
- Vehicles

User Story Cards

- Describes the requirements and the acceptance criteria
- Can also hold information about the estimate (from Scrum Team) and the priority (from Product Owner)
- Can be updated continuously



http://www.agilemodeling.com/artifacts/userStory.htm

I	Independent	User stories should not overlap and they should be formulated so they can be implemented in any order.
N	Negotiable	A user story should be an invitation for a conversation. It can be changed, augmented, and redacted; of course, always in dialog with the Product Owner!
V	Valuable	Each user story should deliver value, either to the Product Owner or to Scrum Team.
E	Estimable	It must be possible to assign effort to each user story. A story that can not be estimated is not complete!
S	Small	A user story must be a manageable task. If its completion takes longer than 3 or 4 days, it must be broken down!
T	Testable	There must be clear, testable criteria to define when the story is done in the eyes of the Product Owner and the Scrum Team.

[Buglione & Abran, 2013]

http://xp123.com/articles/invest-in-good-stories-and-smart-tasks/

https://en.wikipedia.org/wiki/File:Scrum_task_board.jpg

DoD - Definition of Done

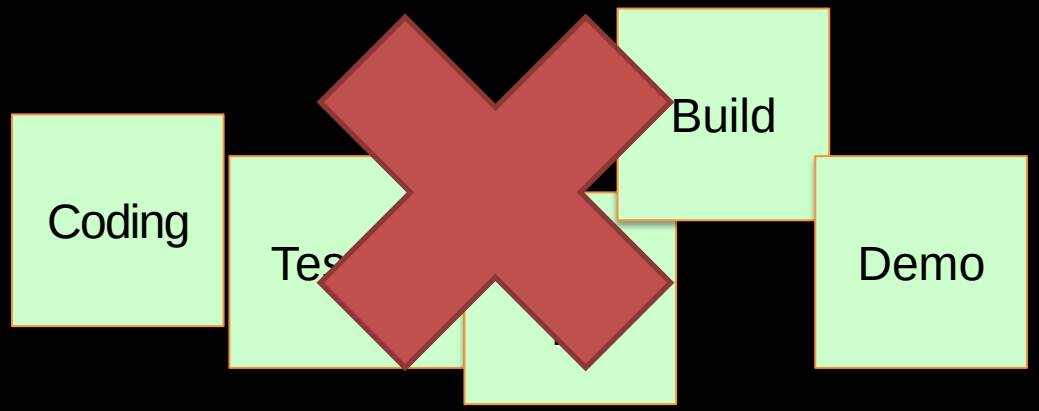
A checklist which usefully guides discussion, estimation and design

Should be helpful and not an obstacle

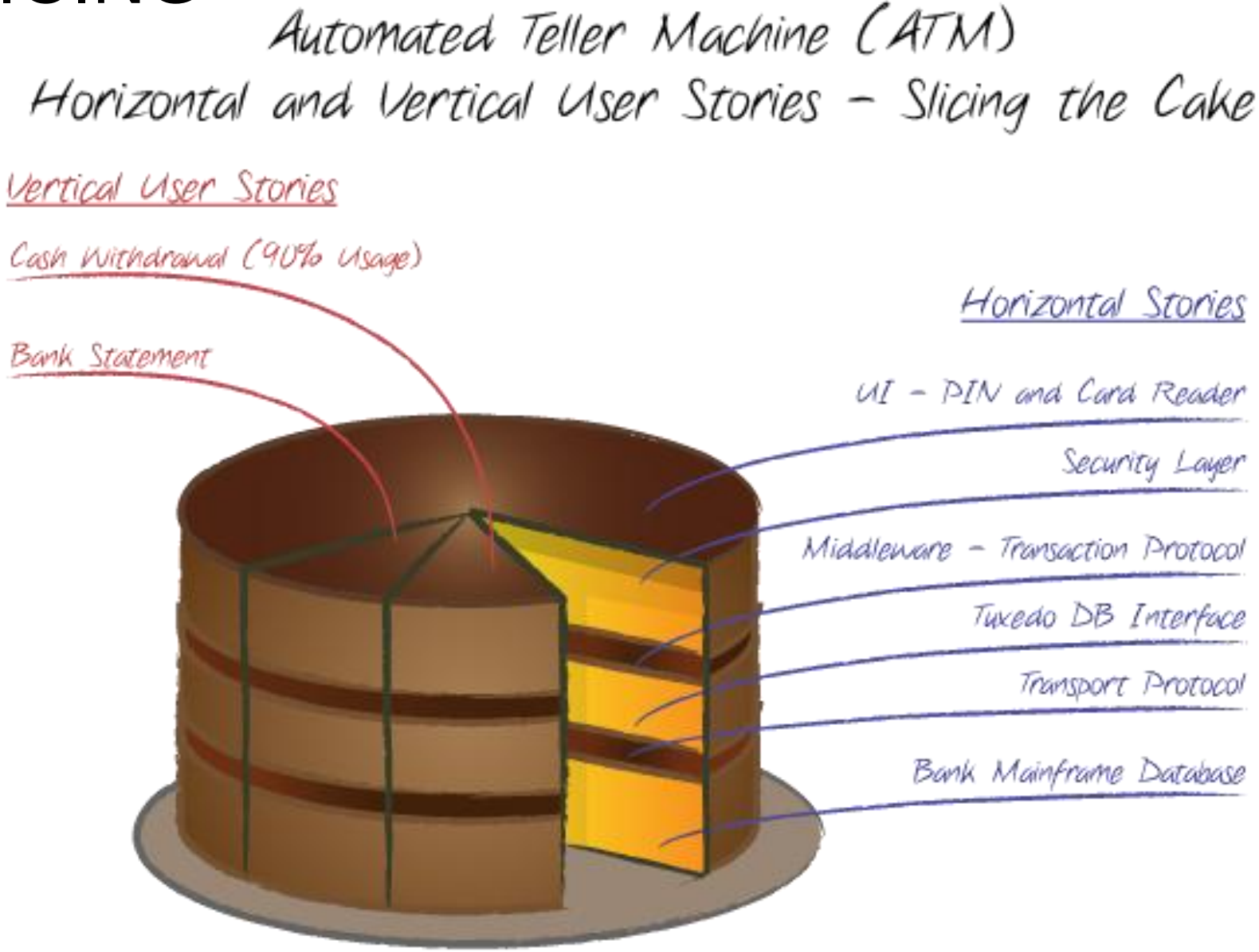
Team level – PO – Other teams

Can change over time

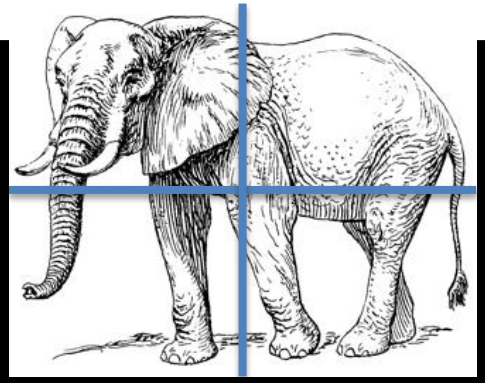
Code – Test – Integrate – Demonstrate – Ship

STORIES	TODO
As a student, I want to purchase a parking pass so that I can drive to school.	
As a student, I want to purchase a parking pass so that I can drive to school.	Integrate parking pass into datamodel Create connection to payment service Design GUI for purchase process Define parking pass validation scheme

SLICING

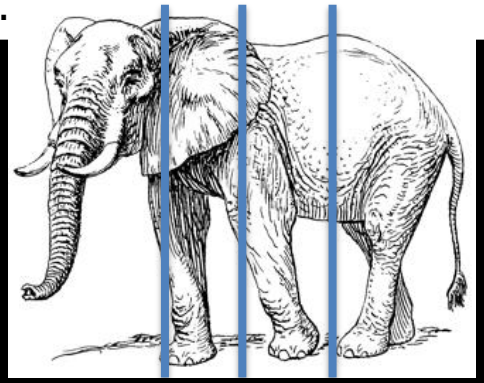

Automated Teller Machine (ATM)
Horizontal and Vertical User Stories – Slicing the Cake
Vertical User Stories
Cash Withdrawal (90% Usage)
Bank Statement
Horizontal Stories
UI – PIN and Card Reader
Security Layer
Middleware – Transaction Protocol
Tuxedo DB Interface
Transport Protocol
Bank Mainframe Database

As a student, I want to purchase a parking pass so that I can drive to school.



- Integrate parking pass into datamodel
- Create connection to payment service
- Design GUI for purchase process
- Define parking pass validation scheme

As a student, I want to purchase a parking pass so that I can drive to school.



- Implement purchase process without payment
- Integrate payment into the purchase process
- Allow users to see their past orders
- Define parking pass validation scheme

ELEPHANT CARPACCIO

Vision: Retail calculator – calculate prices for deliveries in other countries
Three Inputs:
How many items
Price per item
2-letter country code
Output: total price of the order.
Algorithm: Give a discount based on total price,
then add state tax based on country code
and discounted price.

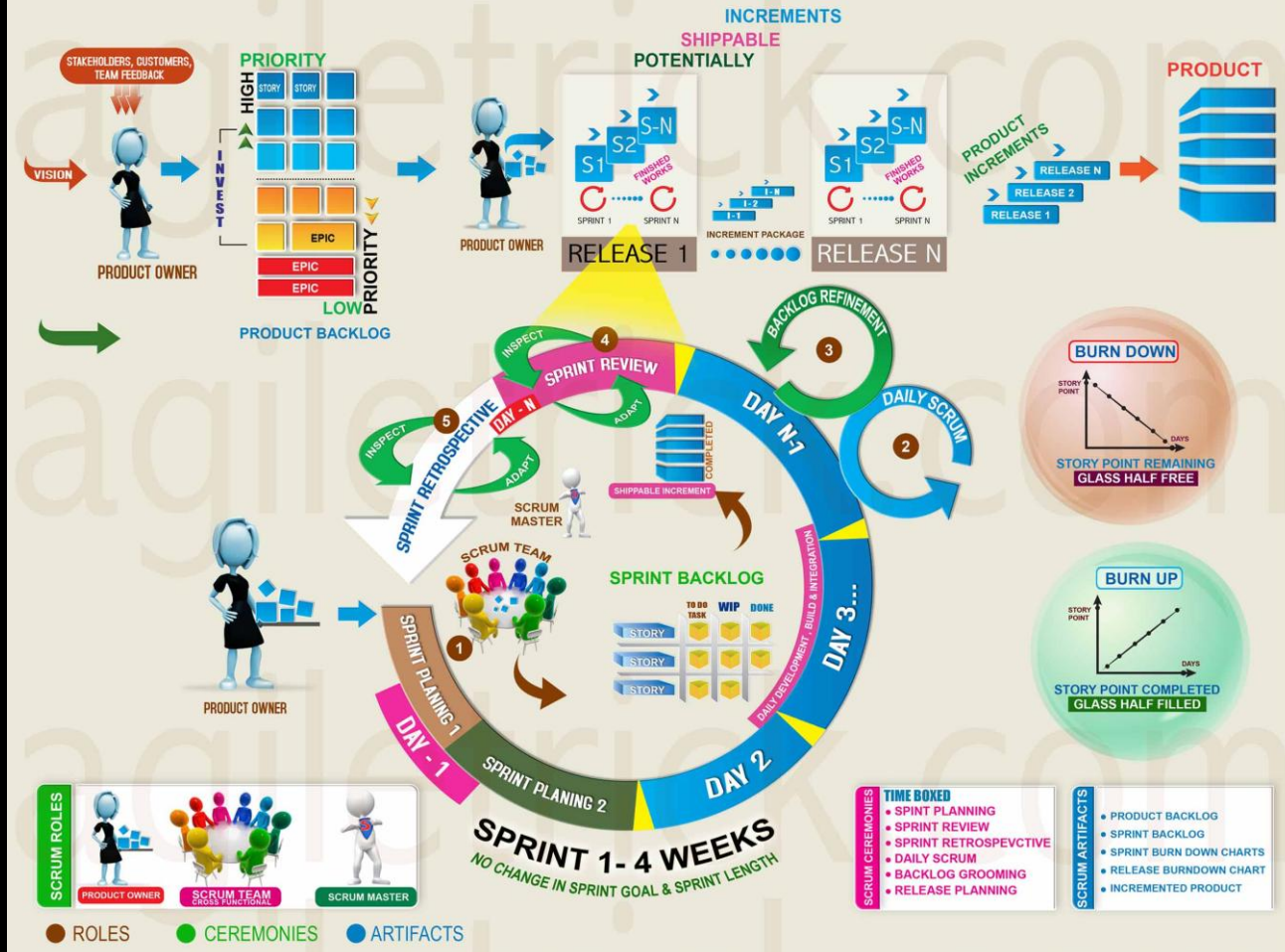
Country	VAT
Belgium (BE)	21%
Germany (DE)	19%
Hungary (HU)	27%
Sweden (SE)	25%
United Kingdom (UK)	20%

Amount	Discount
10.000 SEK	2%
50.000 SEK	3%
100.000 SEK	5%
200.000 SEK	8%
500.000 SEK	10%

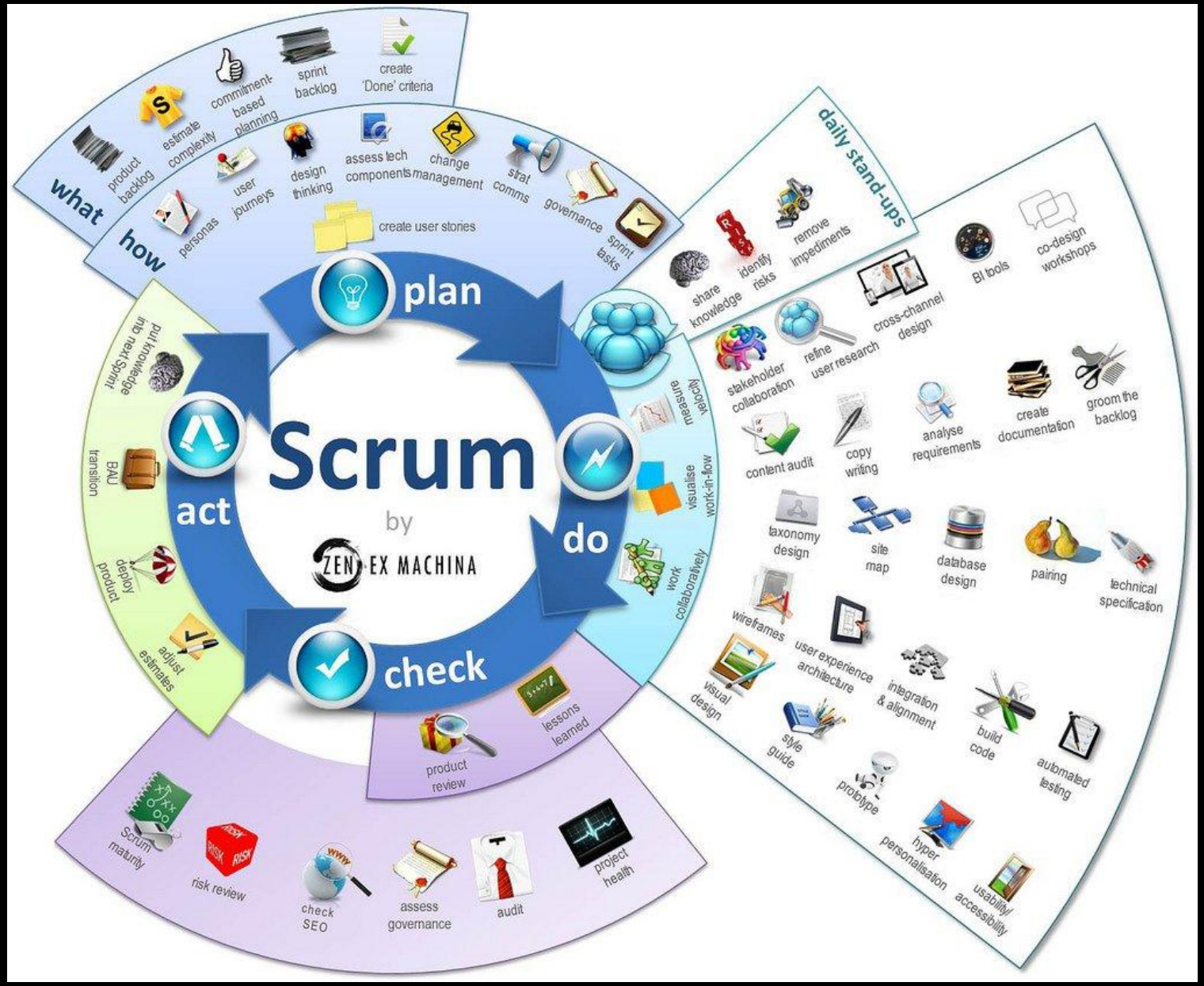
Slice user stories as small as possible!

THE SCRUM FRAMEWORK

agiletrick.com



I - INDEPENDENT | N - NEGOTIABLE | V - VALUABLE | E - ESTIMABLE | S - SMALL ENOUGH | T- TESTABLE


Scrum
by
ZEN EX MACHINA
plan
do
check
act
what
how
product backlog
estimate complexity
commitment based planning
sprint backlog
create 'Done' criteria
personas
user journeys
design thinking
assess tech components
change management
strat comms
governance
sprint tasks
create user stories
daily stand-ups
share knowledge
identify risks
remove impediments
BI tools
co-design workshops
stakeholder collaboration
refine user research
cross-channel design
content audit
copy writing
analyse requirements
create documentation
groom the backlog
taxonomy design
site map
database design
pairing
technical specification
wireframes
user experience architecture
integration & alignment
visual design
style guide
prototype
build code
automated testing
hyper personalisation
usability/ accessibility
measure velocity
visualise work-in-flow
work collaboratively
lessons learned
product review
put knowledge into next Sprint
BAU transition
deploy product
adjust estimates
Scrum maturity
risk review
check SEO
assess governance
audit
project health

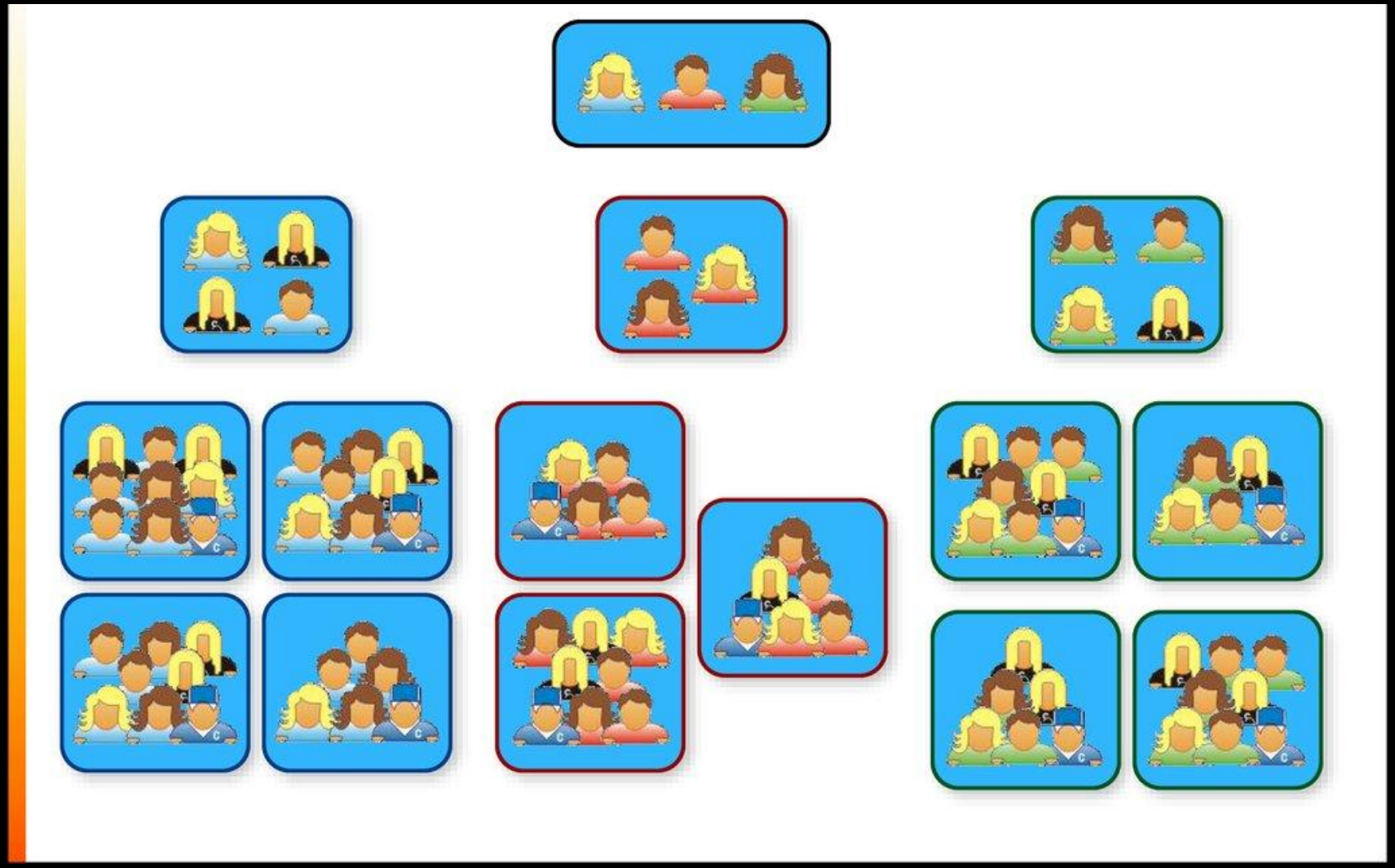
Way of Working

Possible tools and practices:

- Slack / Facebook group
- Epic board
- Git repo
- Meet at lectures and supervision
- Rotate representatives
- Social contract
- DoD
- …

What is
What might or should be
Feedbck to reduce the gap

Scrum of Scrums



Scrum of Scrums

Regular meetings:

1) What has your team done since we last met?
2) What will your team do before we meet again?
3) Is anything slowing your team down or getting in the way?
4) Are you about to put something in another team's way?

REALITY CHECK

What was purpose of lecture?
Which learning objectives were covered? How?
What was the relationship to the course assessment?

QA

'Questions don't have to make sense, Vincent', said Miss Susan.

'But answers do'

Terry Pratchett
Thief of Time, 2001