

CS353FZ
TEAM PROJECT – WEEK4.
SCRUM AND ADMIN

Chris Roadknight

This week:

Lecture on Scrum

Allocate teams

Arrange communication method, Discuss you possible projects

Feed back questions to me

[No need to share '2 minute visions' this week]

Next week:

Allocate supervisors

Share 2 minute visions

Decide provisional project

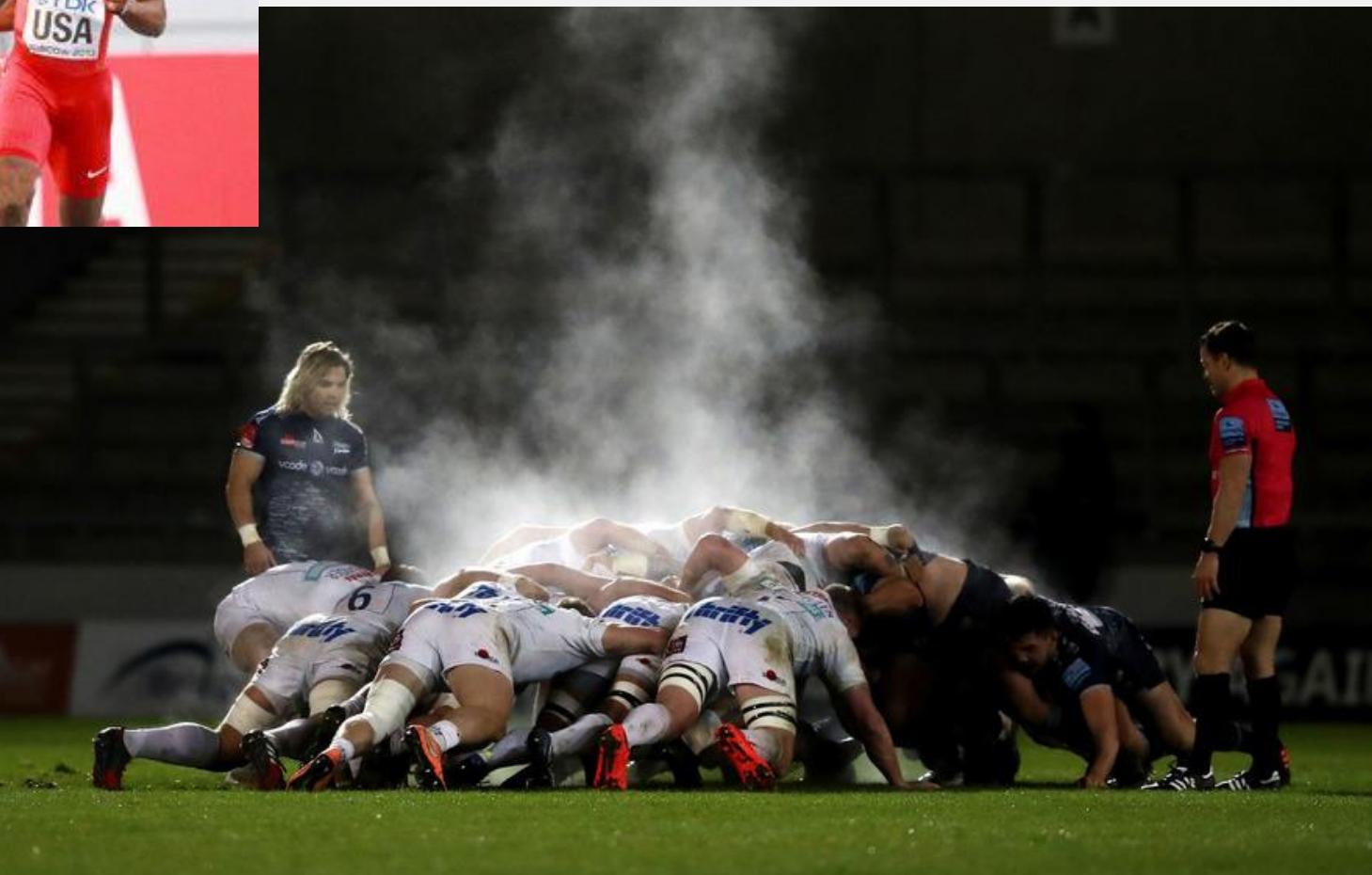
Meet prospective external supervisor

Decide if your team would like to apply for that project [3 external projects available]

We're losing the relay race

"The... 'relay race' approach to product development...may conflict with the goals of maximum speed and flexibility. Instead a holistic or 'rugby' approach—where a team tries to go the distance as a unit, passing the ball back and forth—may better serve today's competitive requirements."

Hirotaka Takeuchi and Ikujiro Nonaka,
"The New New Product Development Game",
Harvard Business Review, January 1986.



Software failures....

Heathrow Terminal 5 Opening – Automated baggage processing software failed, wasn't programmed to handle rare events (manual baggage retrieval by staff)

<https://www.zdnet.com/article/it-failure-at-heathrow-t5-what-really-happened/>

The Mariner 1 Spacecraft – using , instead of . meant the rocket veered off course erratically (rocket cost \$18....software costs unknown)...it had worked in testing

<https://www.wired.com/2009/07/dayintech-0722/>

Ariane 5 – change in datatype [64 bit float - > 16 bit integer] \$460 million

<https://www.bugsnag.com/blog/bug-day-ariane-5-disaster>

NASA's Mars Orbiter – software team who took over (relay??) didn't convert imperial measurements to metric. \$125 million + additional costs

<https://www.bugsnag.com/blog/bug-day-mars-climate-orbiter>

Tesla bike incidents – trained and tested on people riding bikes and people walking but not of people pushing bikes....

<https://cacm.acm.org/magazines/2019/3/234930-metamorphic-testing-of-driverless-cars/fulltext>

Scrum in 100 words

- Scrum is an agile process that allows us to focus on delivering the highest business value in the shortest time.
- It allows us to rapidly and repeatedly inspect actual working software (every two weeks to one month).
- The business sets the priorities. Teams self-organize to determine the best way to deliver the highest priority features.
- Every two weeks to a month anyone can see real working software and decide to release it as is or continue to enhance it for another sprint.

Scrum origins

- Jeff Sutherland
 - Initial scrums at Easel Corp in 1993
 - IDX and 500+ people doing Scrum
- Ken Schwaber
 - ADM
 - Scrum presented at OOPSLA 96 with Sutherland
 - Author of three books on Scrum
- Mike Beedle
 - Scrum patterns in PLOPD4
- Ken Schwaber and Mike Cohn
 - Co-founded Scrum Alliance in 2002, initially within the Agile Alliance



Scrum has been used by:

- Microsoft
- Yahoo
- Google
- Electronic Arts
- Lockheed Martin
- Philips
- Siemens
- Nokia
- IBM
- Capital One
- BBC
- **Paddy Power**
- Intuit
- Nielsen Media
- First American Real Estate
- BMC Software
- Ipswitch
- John Deere
- Lexis Nexis
- Sabre
- Salesforce.com
- Time Warner
- Turner Broadcasting
- Oce

Scrum has been used for:

- Commercial software
- In-house development
- Contract development
- Fixed-price projects
- Financial applications
- ISO 9001-certified applications
- Embedded systems
- 24x7 systems with 99.999% uptime requirements
- the Joint Strike Fighter
- Video game development
- FDA-approved, life-critical systems
- Satellite-control software
- Websites
- Handheld software
- Mobile phones
- Network switching applications
- ISV applications
- Some of the largest applications in use

Agile Culture

Agile requires a new culture

- Adventure Works - American Management Implemented Scrum
- The product began to emerge in high-quality, regular increments. Joris (owner) adopted a sustainable pace of work - everyone worked eight-hour days.
- Company was owned by a Japanese company and eight-hour workdays was unacceptable to Japanese management.
- Demanded return to the 12-hour work days that were normal prior to Scrum were restored.
- Defects rose 60 percent so Joris restored Scrum's eight-hour workdays.
- Japanese managers saw empty parking lots and darkened offices, reported that employees were lazy and recommended selling the company. - Sold Adventure Works to its American management.
- Two months later, company sold product for twice the price of the buyout.

Characteristics

- Self-organizing teams
- Product progresses in a series of month-long “sprints”
- Requirements are captured as items in a list of “product backlog”
- No specific engineering practices prescribed
- Uses generative rules to create an agile environment for delivering projects
 - vs inclusive rules
 - One of the “agile processes”

The Agile Manifesto—a statement of values

Individuals and interactions

over

Process and tools

Working software

over

Comprehensive documentation

Customer collaboration

over

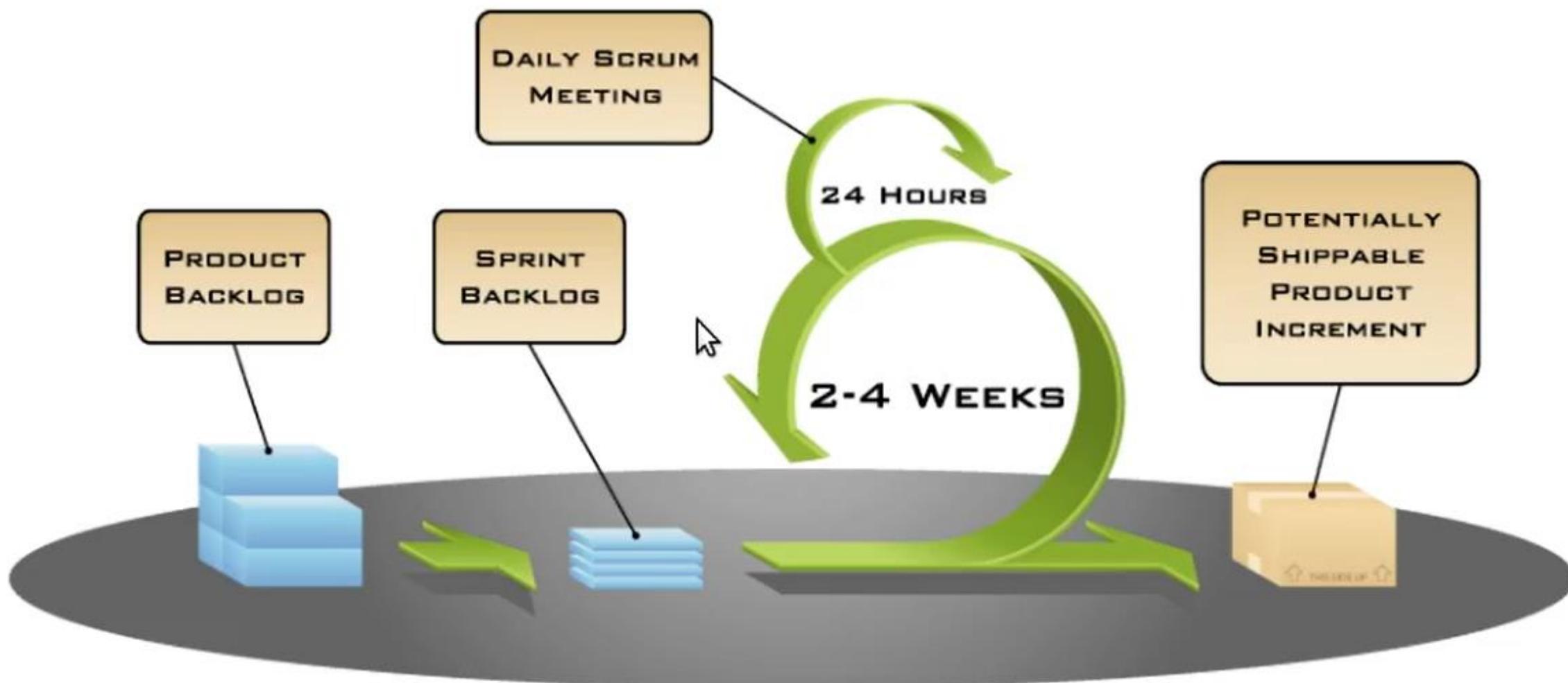
Contract negotiation

Responding to change

over

Following a plan

Putting it all together



Sprints

- Scrum projects make progress in a series of “sprints”
 - Analogous to Extreme Programming iterations
- Typical duration is 2–4 weeks or a calendar month at most
- A constant duration leads to a better rhythm
- Product is designed, coded, and tested during the sprint

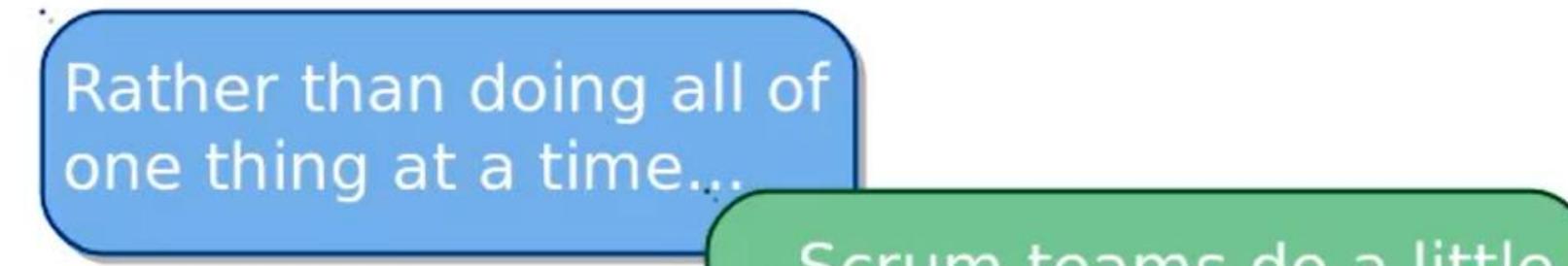
Sequential vs. overlapping development

Requirements

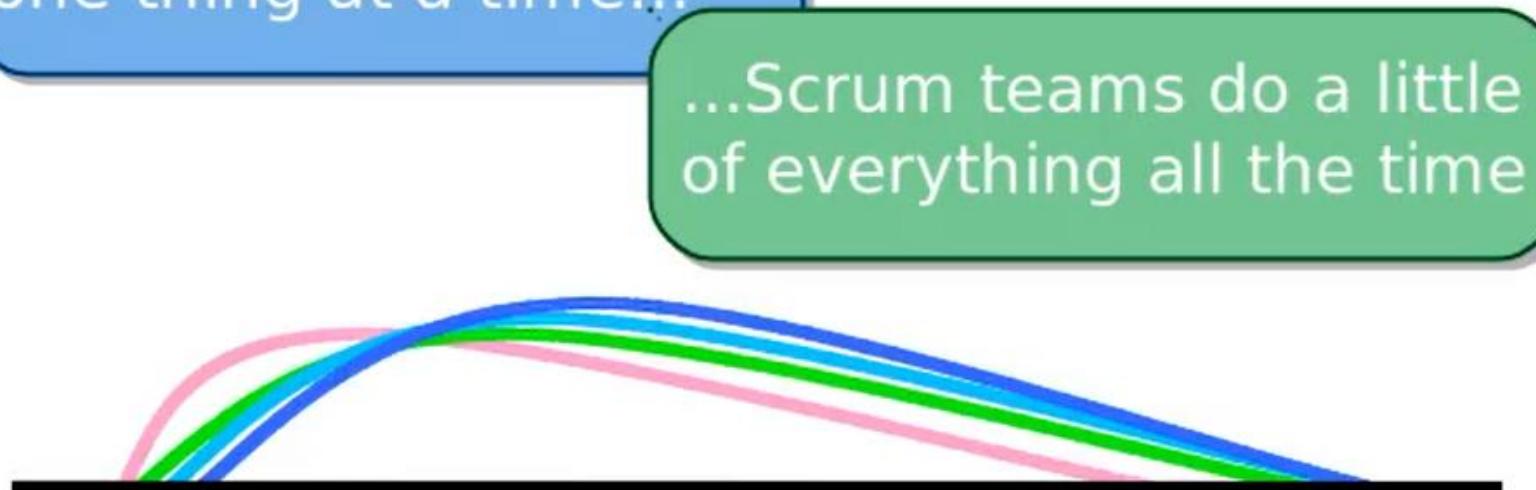
Design

Code

Test



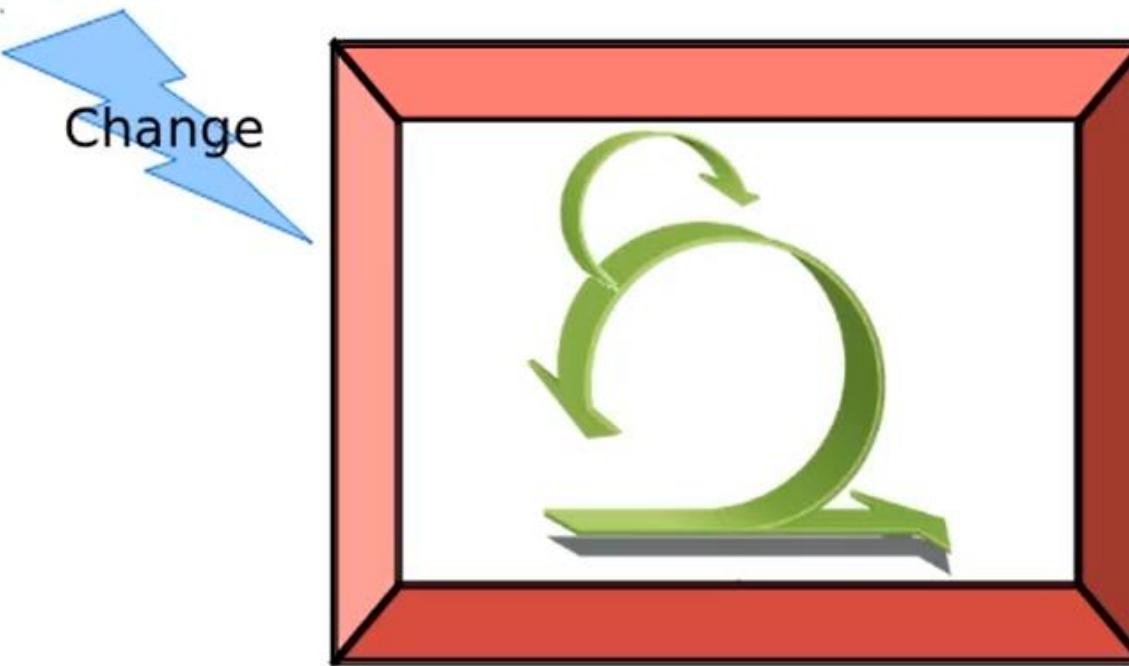
Rather than doing all of
one thing at a time...



...Scrum teams do a little
of everything all the time

Source: "The New New Product Development Game" by Takeuchi and Nonaka. *Harvard Business Review*, January 1986.

No changes during a sprint



- Plan sprint durations around how long you can commit to keeping change out of the sprint

Scrum framework

Roles

- Product owner
- ScrumMaster
- Team

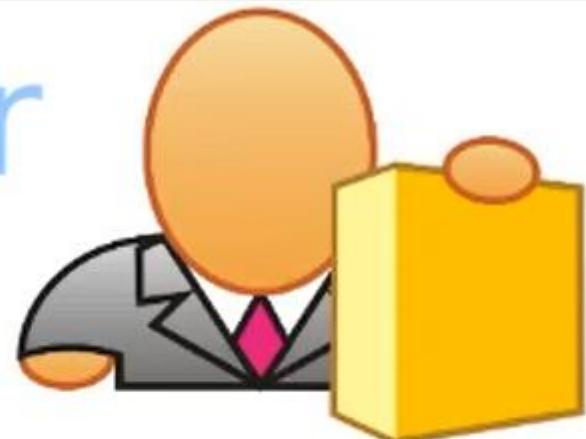
Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

Product owner



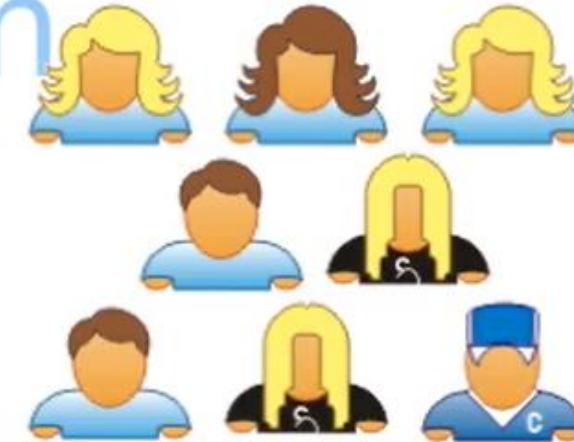
- Define the features of the product
- Decide on release date and content
- Be responsible for the profitability of the product (ROI)
- Prioritize features according to market value
- Adjust features and priority every iteration, as needed
- Accept or reject work results

The ScrumMaster



- Represents management to the project
- Responsible for enacting Scrum values and practices
- Removes impediments
- Ensure that the team is fully functional and productive
- Enable close co-operation across all roles and functions
- Shield the team from external interferences

The team



- Typically 5-9 people
- Cross-functional:
 - Programmers, testers, user experience designers, etc.
- Members should be full-time
 - May be exceptions (e.g., database administrator)
- Teams are self-organizing
 - Ideally, no titles
- Membership should change only between sprints

Scalability

- Typical individual team is 7 ± 2 people
 - Scalability comes from teams of teams
- Factors in scaling
 - Type of application
 - Team size
 - Team dispersion
 - Project duration
- Scrum has been used on multiple 500+ person projects
 - Scrum of scrums
 - Scrum of scrum of scrums ...

Scrum framework

Roles

- Product owner
- ScrumMaster
- Team

Ceremonies

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts



Sprint planning

- Team selects items from the product backlog they can commit to completing
- Sprint backlog is created
 - Tasks are identified and each is estimated (1-16 hours)
 - Collaboratively, not done alone by the ScrumMaster

As a vacation planner, I want to see photos of the hotels.

Code the middle tier (8 hours)
Code the user interface (4)
Write test fixtures (4)
Code the foo class (6)
Update performance tests (4)

The daily scrum

- Parameters
 - Daily
 - 15-minutes
 - Stand-up
- Not for problem solving
 - Whole world is invited
 - Only team members, ScrumMaster, product owner, can talk
- Helps avoid other unnecessary meetings



Everyone answers 3 questions

1

What did you do yesterday?

2

What will you do today?

3

Is anything in your way?

- These are commitments in front of peers

The sprint review

- Team presents what it accomplished during the sprint
- Typically takes the form of a demo of new features or underlying architecture
- Informal
 - 2-hour prep time rule
 - No slides
- Whole team participates
- Invite the world



Sprint retrospective

- Periodically take a look at what is and is not working
- Typically 15–30 minutes
- Done after every sprint
- Whole team participates
 - ScrumMaster
 - Product owner
 - Team
 - Possibly customers and others

Start / Stop / Continue

- Whole team gathers and discusses what they'd like to:

Start doing

Stop doing

Continue doing

This is just one
of many ways
to do a sprint
retrospective.

Scrum framework

Roles

- Product owner
- ScrumMaster
- Team

Ceremonies

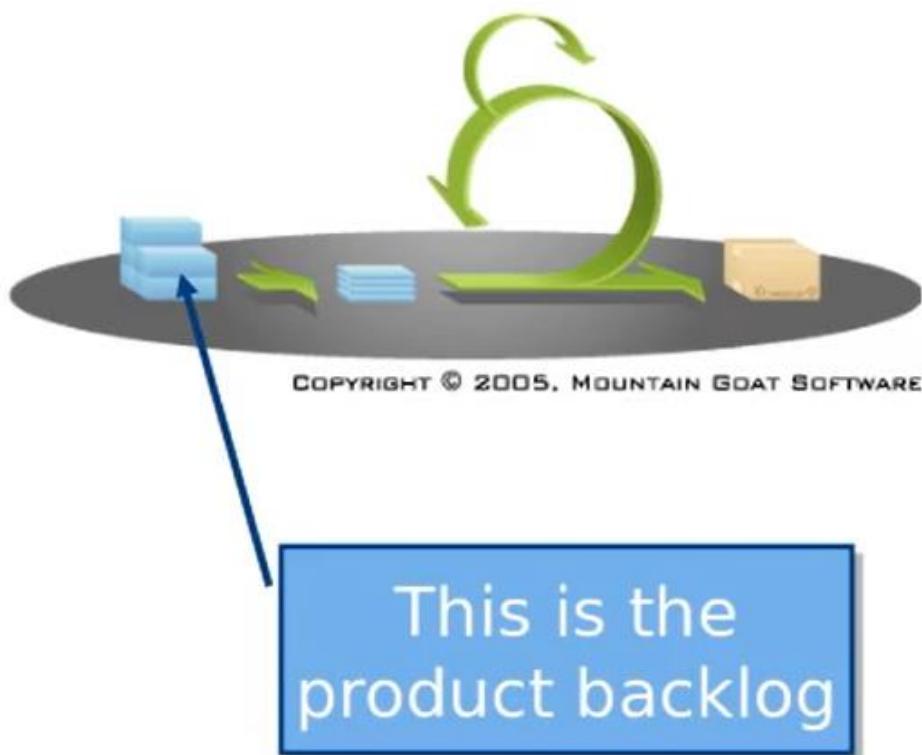
- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artifacts

- Product backlog
- Sprint backlog
- Burndown charts

Product backlog

- The requirements
- A list of all desired work on the project
- Ideally expressed such that each item has value to the users or customers of the product
- Prioritized by the product owner
- Re-prioritized at the start of each sprint



A sample product backlog

Backlog item	Estimate
Allow a guest to make a reservation	3
As a guest, I want to cancel a reservation.	5
As a guest, I want to change the dates of a reservation.	3
As a hotel employee, I can run RevPAR reports (revenue-per-available-room)	8
Improve exception handling	8
...	30
...	50

The sprint goal

- A short statement of what the work will be focused on during the sprint

Database Application

Make the application run on SQL Server in addition to Oracle.

Life Sciences

Support features necessary for population genetics studies.

Financial services

Support more technical indicators than company ABC with real-time, streaming data.

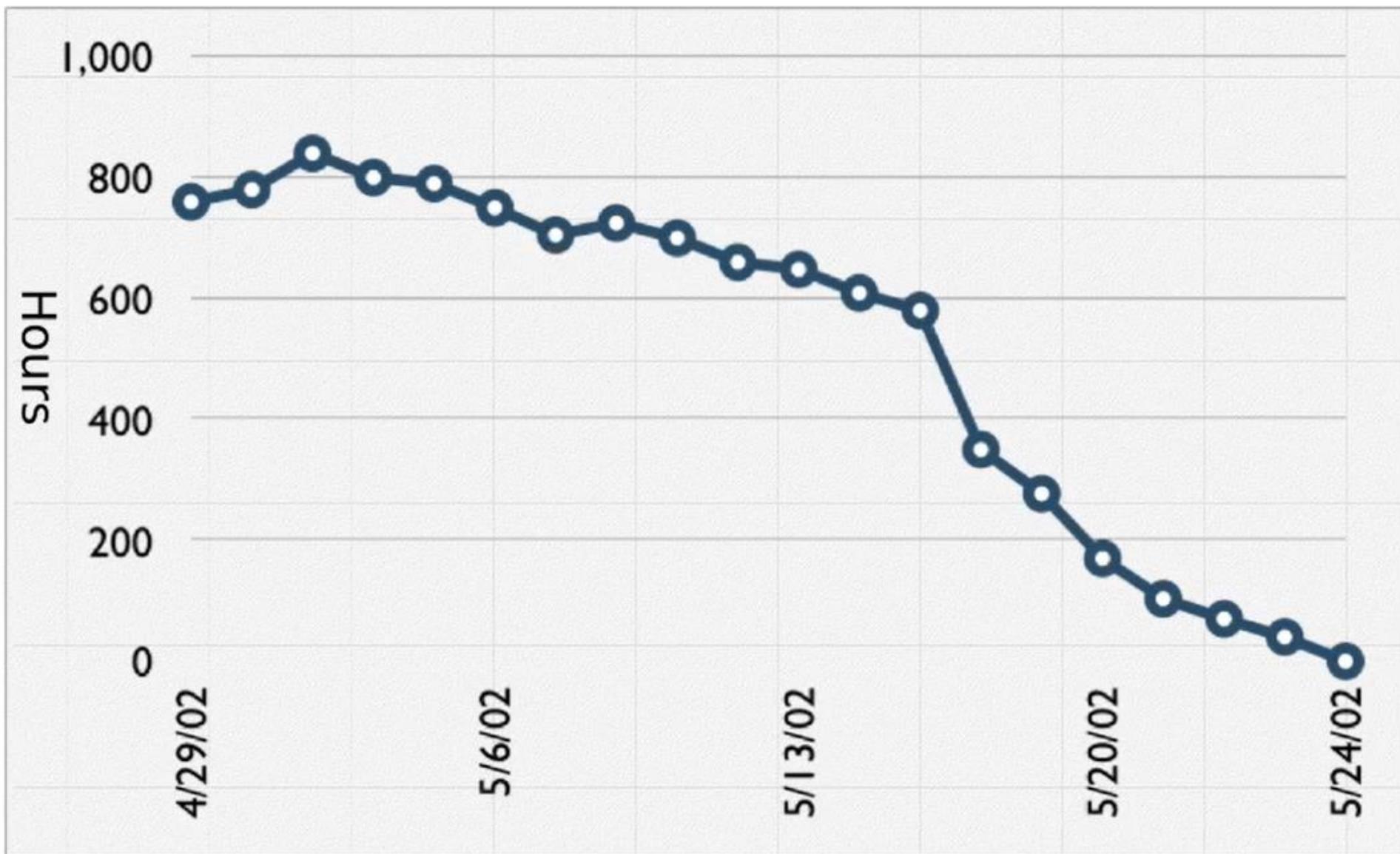
Managing the sprint backlog

- Individuals sign up for work of their own choosing
 - Work is never assigned
- Estimated work remaining is updated daily
- Any team member can add, delete or change the sprint backlog
- Work for the sprint emerges
- If work is unclear, define a sprint backlog item with a larger amount of time and break it down later
- Update work remaining as more becomes known

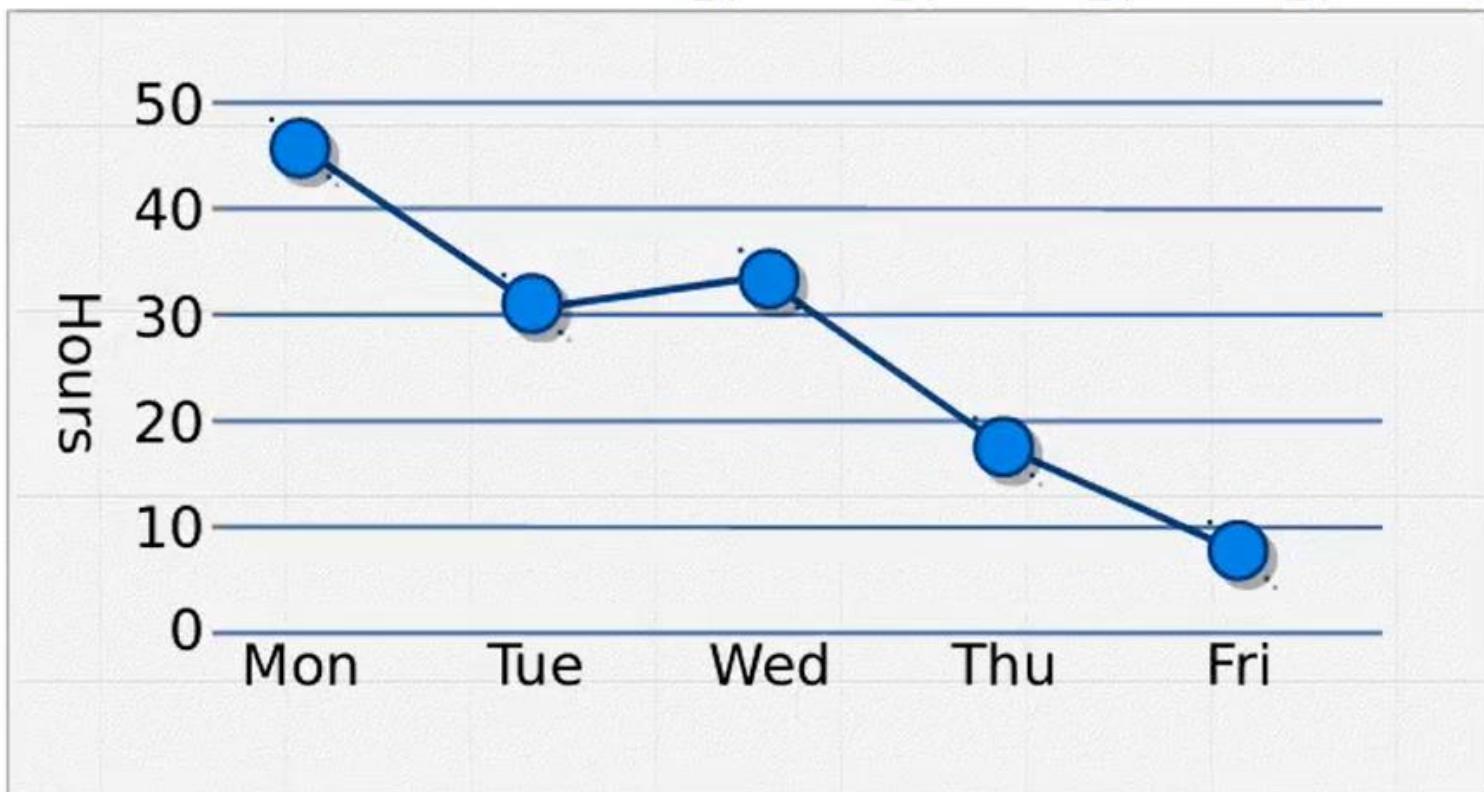
A sprint backlog

Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	4	
Test the middle tier	8	16	16	11	8
Write online help	12				
Write the foo class	8	8	8	8	8
Add error logging			8	4	

A sprint burndown chart



Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	7	
Test the middle tier	8	16	16	11	8
Write online help	12				



Estimating Tasks

- A sprint or product backlog item takes up space on the y-axis
- This space is proportional to how long (or difficult) a backlog item is
- Measured in:
 - Hours
 - Points
 - Gummi bears, foot-pounds, NUTs (Nebulous Units of Time)



Estimating Tasks

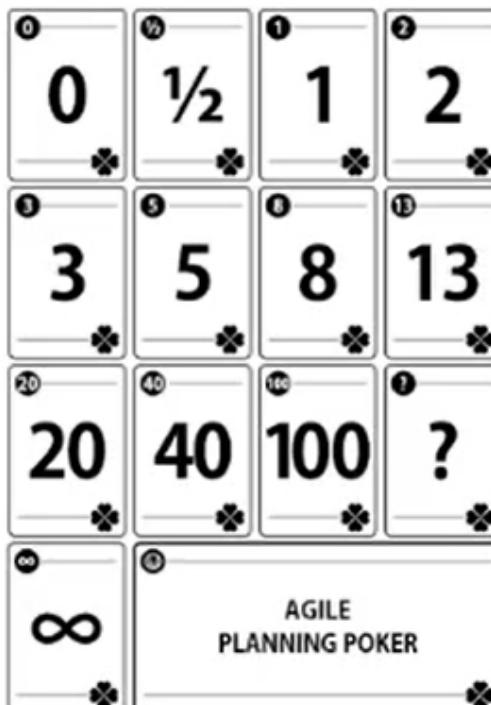
- Two broad categories of estimation methods:
 - 1: Model based - e.g. COCOMO
 - Uses models usually built from observations
 - Perhaps tailored to the current domain
 - 2: Expert (consensus) based - e.g. Wideband Delphi
 - Uses experts and their experience

Planning Poker

- A variation of Wideband Delphi
 - 1: Assemble a group of experts (around 10)
 - 2: Give each estimator a deck of cards (usually 0, 1, 2, 3, 5, 8, 13, 20, 40, and 100)
 - 3: Moderator reads a description of the user story. The product owner can answer questions from estimators.
 - 4: Each estimator selects a card and places it face down on the table. When all estimates are in, the cards are flipped over.
 - 5: If the estimates vary widely, the owners of the high and low estimates discuss the reasons why their estimates are so different. All estimators should participate in the discussion.
 - 6: Repeat from step 4 until estimates converge to within some predetermined threshold.

Planning Poker

- How effective is it?:
 - One study found that estimates obtained through the Planning Poker process were less optimistic and more accurate than estimates obtained through mechanical combination of individual estimates for the same tasks.



Planning poker cards

Your Project

- Project theme:
 - *To build a (server/DB backed) Web Application*
 - **Must use SCRUM**
 - Generate documentation on all SCRUM-related activities
 - **Must use Github** (see <https://education.github.com/pack>)
 - **Must use some form of Test Driven Development**
 - **Must use telegram messenger for communication**
 - Use the opportunity to learn new technologies
 - Ideas: deploy to a cheap VPS, use Firebase
 - For example if you have not use web frameworks before - check them out.
 - JS libraries like Jquery/Bootstrap/React.js/Express (Node.js)

Your Team

Team size

- SCRUM works best for small teams
- You will be working in pre-assigned groups of 5 students
 - Just like industry
 - “You can choose your friends but you can't choose your ~~family~~ co-workers”
 - And just like industry
 - *Make it work*

General Tips

- Use class time for teamwork
 - Recommended: at least one other SCRUM meeting per week
- Appoint a SCRUM Master each week
 - Work out schedule in advance
- Effort
 - At least 4 hours student work outside classroom
 - $7 \text{ hours} \times 10 \text{ weeks} \times 5 \text{ students} = 350 \text{ person hours}$
 - You can build something worthwhile together

Lab Format

- Sign into lab
 - Present your logbook for marking
 - SCRUM meeting - followed by teamwork/dev
 - SCRUM Master updates lecturer on status
 - Pick up your logbook
 - Sign out of the lab
-
- In the early stages, there will be occasional short presentations to help get you started

Your project proposal

- Think of an idea for an application
- Start with the 2-minute vision (see next slide)
- Document the idea. Use the following headings:
 - UI sketches (good for communicating with team)
 - Project scope (what it does – what it does not do)
 - Schedule (*very* rough estimate. Use Gantt chart)
 - Risks (technical? complexity? understanding end-user?)
 - Feasibility (how likely is the project to succeed?)
 - Hardware and software
 - Development tools, libraries etc...

2-minute vision

For (target customer)

Who (needs or where is opportunity)

The (proposed product name)

is a (product category)

That (key benefit/compelling advantage).

Unlike (primary competitive alternative)

Our product (differentiating factors)

Example - 2-minute vision for Moodle

For universities

Who need a way to deliver learning online

The moodle system

is a web based virtual learning
environment

That is open source and free.

Unlike Blackboard

Our product supports plug-ins, LaTeX
markup, etc.

End

- Further reading/viewing:
 - Ken Schwaber's Google Tech Talk
 - Iterative vs Incremental Development
 - Scrum in 10 minutes
 - Jeff Sutherland's SCRUM Handbook
 - The SCRUM Primer



Mountain Goat
Software, LLC

Adapted from material contributed by Mountain Goat Software LLC



TEAMS!!!

After consulting with MC in Ireland and head of school here we have decided to...
...randomly allocate team members.

149 students (149 is a prime number ☺)

24 teams of 6,
1 team of 5

The Github problem...

Access to github seems *unpredictable* in china.

I think Gitee.com is more acceptable to Chinese authorities

There are alternatives

<https://www.guru99.com/github-alternative.html>

Though again, I am unsure how reliable access to these will be.

In fact, implementing version control for a simple project like this is feasible without an online Source Code Repository Tool.

As long as you had access to a shared file structure.

So, I will let you decide the best method for your team over the next week

Special Projects Division!!!

1. Collect a variety of data which will be uploaded to a cloud database
2. A Python client library for sending data to and receiving command from a cloud-based server
3. Design of an efficient message routing scheme for sensor data and control command transmission that can scale to multi-million devices
4. A software system for device and sensor management and sensor data visualisation

What you will receive on successful completion

1. A Certificate of Completion jointly issued by the Maynooth University Future Ready Initiative and the industry collaborator (Climate Response Ireland)
2. A paid micro-internship during Winter break (depending on the performance of the team; the industry collaborator will make decisions on this)

Full information next week.....

What you need to do before next week...including any remaining time today

1. Check your group members on moodle and form an initial wechat group to make contact.
 1. Your designated supervisor will be in contact next week
2. Discuss various logistics and contingency plans
 1. Where will you host your software, what impact will continued covid measures have, when will you have meetings etc
3. Discuss your own ideas briefly – [full sharing of project proposal and deliverable next week with supervisors]
 1. Everyone needs to do this, even if applying for Special project
4. Discuss if you are going to apply, as a group, for one of the Special projects
5. Applications for Special Project will happen after next week's introduction from the stakeholder.
6. Decide an initial project manager/scrum master

week	date	content
3	15/09/2021	Introduction
4	22/09/2021	Scrum, team allocation, initial introductions
5	29/09/2021	Supervisor allocation, project decision , Special Projects
6	06/10/2021	Project sign off, Sprint 1, Week 1. Sprint Planning
7	13/10/2021	Sprint 1, Week 2
8	20/10/2021	Sprint 1, Week 3. Sprint Review, Sprint Retrospective
9	27/10/2021	Sprint 2, Week 1. Sprint Planning
10	03/11/2021	Sprint 2, Week 2
11	10/11/2021	Sprint 2, Week 3. Sprint Review, Sprint Retrospective
12	17/11/2021	Sprint 3, Week 1. Sprint Planning
13	24/11/2021	Sprint 3, Week 2
14	01/12/2021	Sprint 3, Week 3. Sprint Review, Sprint Retrospective
15	08/12/2021	Sprint 4, Week 1. Sprint Planning
16	15/12/2021	Sprint 4, Week 2
17	22/12/2021	Sprint 4, Week 3. Sprint Review, Sprint Retrospective
18	29/12/2021	Wrap up, submission deadline ?

Plagiarism

1. Many web/mobile based applications have been done before
2. You cannot recycle software from other modules/years
3. You cannot copy from other groups
4. You cannot copy from internet resources
 - I. If you do use part of an internet resource, you should reference it
5. Guides to doing them have even been done
6. **YOUR SOFTWARE NEEDS TO BE ORIGINAL**
7. Original from each other and original from internet resources
8. I will be using JPLAG/MOSS to detect plagiarism
9. Penalties for plagiarism are severe
 - I. 0 for the module, and a written warning up to exclusion
10. Remember, most of the marks are about participating and documenting the agile software development process...NOT the resulting product