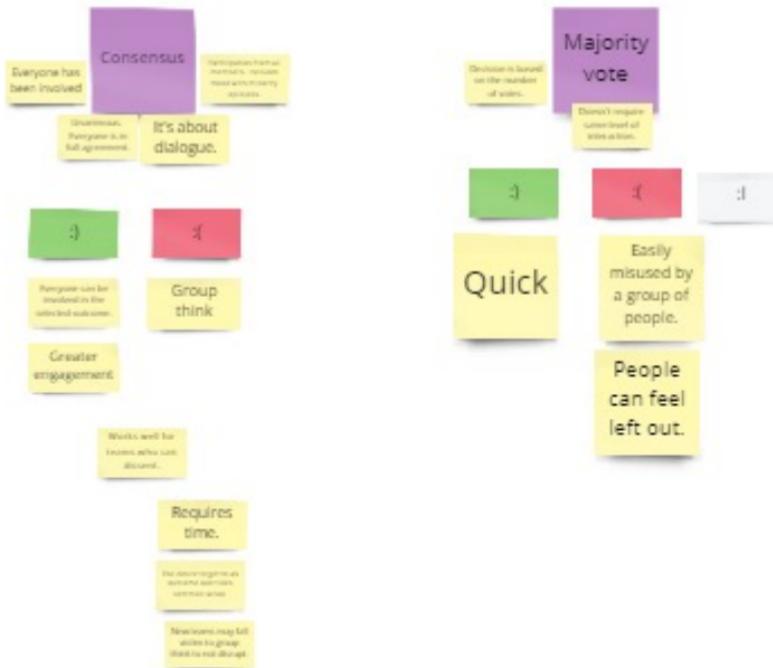


So you're trying to get to an agreement. You could have the majority decide vs a more negotiate say through consensus

Can be anonymous.



Majority vote

miro





Why do we do the Sprint Review?



Name some of activities that occur during this event?



Who should attend?



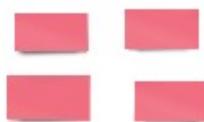
What is the timebox?



What should the ScrumMaster do here?



What are the possible outcomes of this event?



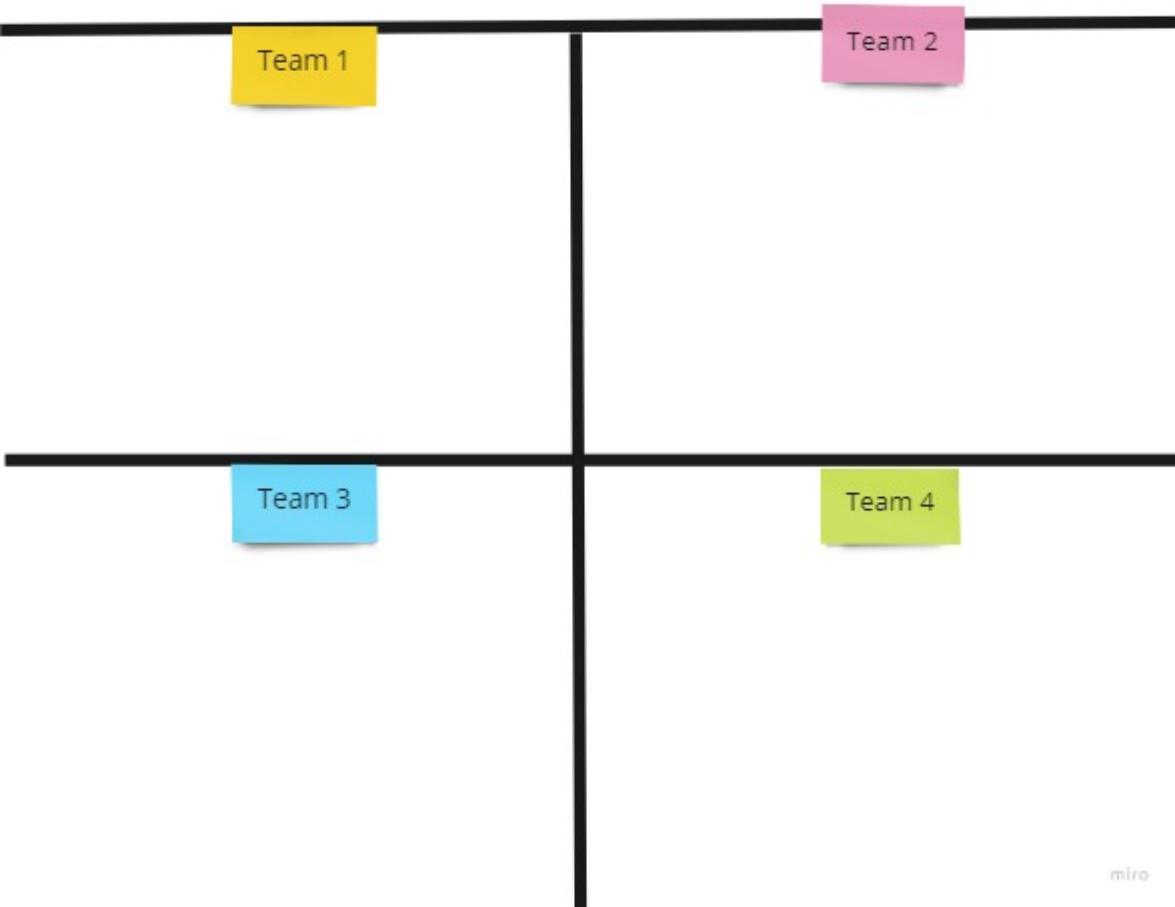
How does this event bring empiricism to life?



Backup board 2

[Click here to watch the video!](#)

1. How much of these 3 intrinsic motivators do you have at work?
2. Give examples of what removing these motivators could look like at work?
3. Give examples of what could be done to introduce these motivators in?



Create post it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organisation?



Create post it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organisation?



Team 1

What differences do you see?

Create post-it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organization?



What differences do you see?

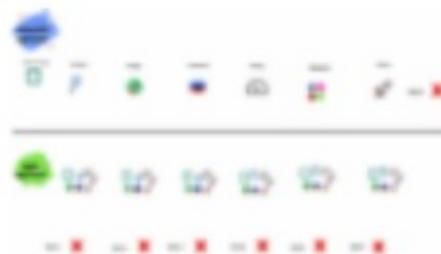
Create post-it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organization?



Team 2

What differences do you see?

Create post-it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organization?



What differences do you see?

Create post-it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organization?



Team 1

Imagine a 12 month project.

The requirement documentation was created in January, capturing what the customer wanted.
What percentage of requirements stayed the same by the time the project was delivered in December?



Activity B

Team 2

Imagine a 12 month project.

The requirement documentation was created in January, capturing what the customer wanted.
What percentage of requirements stayed the same by the time the project was delivered in December?



Team 3

Imagine a 12 month project.

The requirement documentation was created in January, capturing what the customer wanted.
What percentage of requirements stayed the same by the time the project was delivered in December?



Team 4

Imagine a 12 month project.

The requirement documentation was created in January, capturing what the customer wanted.
What percentage of requirements stayed the same by the time the project was delivered in December?



miro

What's your story?

Name

Role

Use of Agile?

As a, I want, So that

MECHANICAL



ORGANIC



1. What are examples of mechanical & organic types of systems ?
2. Discuss the following statements:
 - a. A watch maker should be able to always figure out what is wrong with a watch they designed.
 - b. The weather person should be able to guarantee the weather forecast for the following day?



Team 1

SO, WHAT'S 'BETTER'?



Team2

SO, WHAT'S 'BETTER'?



Team 3

SO, WHAT'S 'BETTER'?



Team 4

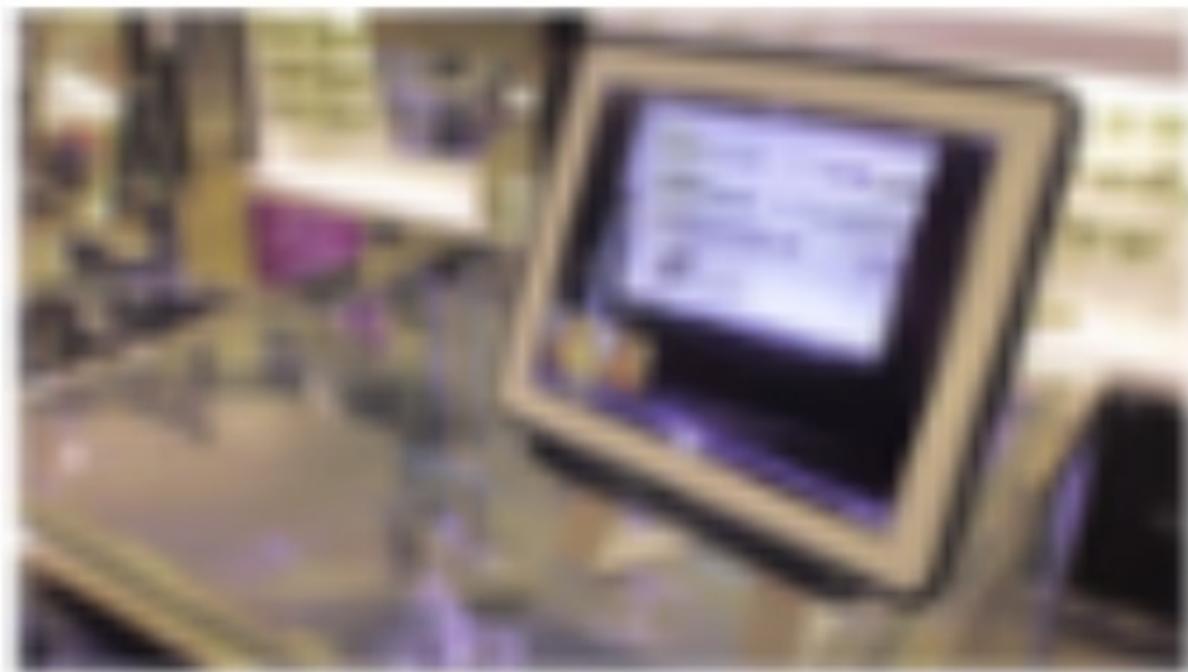
SO, WHAT'S 'BETTER'?



Create post it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organisation?



Nordstorm Innovation labs



What's your story?

Name

Role

Use of Agile?

As a, I want, So that

Create post it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organisation?



What's your story?

Name

Role

Use of Agile?

As a, I want, So that

What's your story?

Name

Role

Use of Agile?

As a, I want, So that

Team 1

What do you need from one another
to be effective during this course?

Team2

What do you need from one another
to be effective during this course?



Team 3

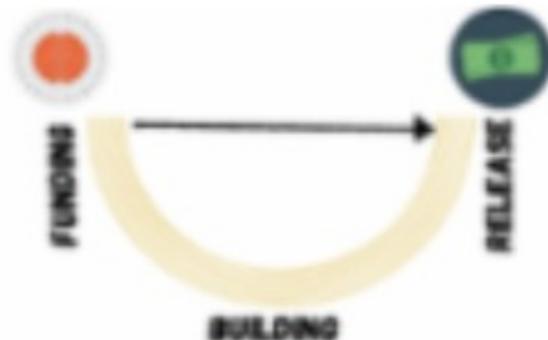
What do you need from one another
to be effective during this course?

Team 4

What do you need from one another
to be effective during this course?

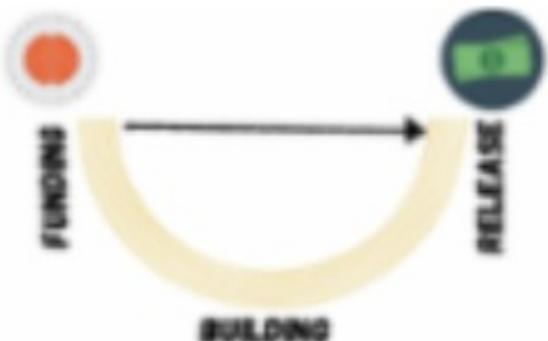
Team 1

How long to go from 'concept to cash'?



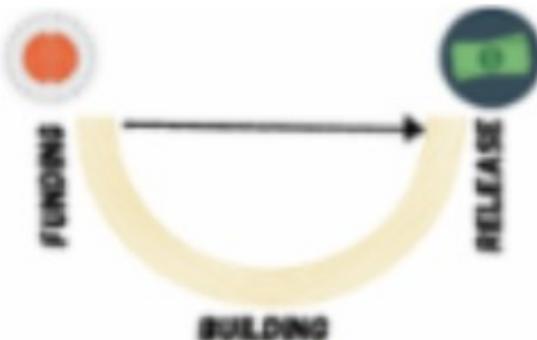
Team 3

How long to go from 'concept to cash'?



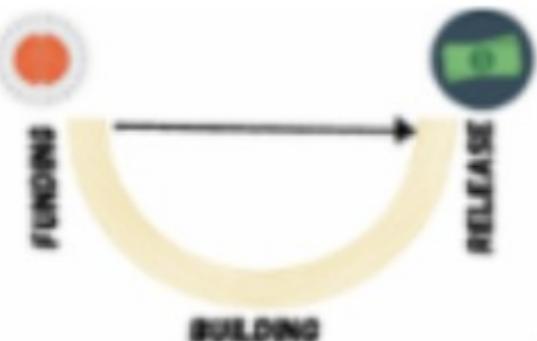
Team2

How long to go from 'concept to cash'?



Team 4

How long to go from 'concept to cash'?



Team 1

Is innovation essential?

Are you defining innovation as improving existing ways to do things better? If that's true, the question below may push you to think & generate your next steps to answer questions.



Question 1

Are there real world
examples of natural
innovation found?

Question 2

Are there real world
examples of what
you or your team can
do?

Team 2

Is innovation essential?

Are you defining innovation as improving existing ways to do things better? If that's true, the question below may push you to think & generate your next steps to answer questions.



Question 1

Are there real world
examples of natural
innovation found?

Are there real world
examples of what
you or your team can
do?

Team 3

Is innovation essential?

Are you defining innovation as improving existing ways to do things better? If that's true, the question below may push you to think & generate your next steps to answer questions.



Question 1

Are there real world
examples of natural
innovation found?

Question 2

Are there real world
examples of what
you or your team can
do?

Team 4

Is innovation essential?

Are you defining innovation as improving existing ways to do things better? If that's true, the question below may push you to think & generate your next steps to answer questions.



Question 1

Are there real world
examples of natural
innovation found?

Are there real world
examples of what
you or your team can
do?

What do all of the below have in common?

Team 1



What do all of the below have in common?

Team 2



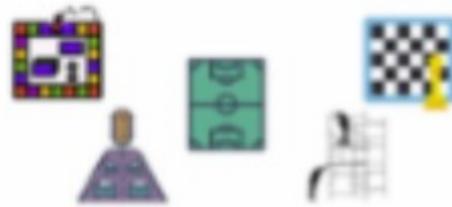
What do all of the below have in common?

Team 3



What do all of the below have in common?

Team 4



Team 1

Which of the two styles have you experienced?

Which leadership style requires less middle managers?

Are there scenarios where command & control is useful?

Team 2

Which of the two styles have you experienced?

Which leadership style requires less middle managers?

Are there scenarios where command & control is useful?

Team 3

Which of the two styles have you experienced?

Which leadership style requires less middle managers?

Are there scenarios where command & control is useful?

Team 4

Which of the two styles have you experienced?

Which leadership style requires less middle managers?

Are there scenarios where command & control is useful?



What is value to you in the team that you work in?



Mini Value Stream

- Pick one team member's value stream
- That is a customer request aka requirement
- List the high level activities
- All the way to the delivery of the request



Team 1



ACTIVITIES



ACTIVITIES

Team 3



ACTIVITIES

Team 4



ACTIVITIES

Team 1

What are the benefits & disadvantages of these approaches?

Everyone will have different approaches to their work, which are best suited to their strengths. Use this time to discuss these different approaches, which you believe to be the best. For each option, discuss the benefits and the disadvantages.



Team 2

What are the benefits & disadvantages of these approaches?

Everyone will have different approaches to their work, which are best suited to their strengths. Use this time to discuss these different approaches, which you believe to be the best. For each option, discuss the benefits and the disadvantages.



Team 3

What are the benefits & disadvantages of these approaches?

Everyone will have different approaches to their work, which are best suited to their strengths. Use this time to discuss these different approaches, which you believe to be the best. For each option, discuss the benefits and the disadvantages.



Team 4

What are the benefits & disadvantages of these approaches?

Everyone will have different approaches to their work, which are best suited to their strengths. Use this time to discuss these different approaches, which you believe to be the best. For each option, discuss the benefits and the disadvantages.



1. Can you name 5 of the Scrum events?

Team 1

Team 2



2. What's the difference between a feature & component team?
When would you used which?

Team 3

Team 4

3. What is the difference between Scrum leadership & command & control?



4. What is empiricism & how is it relevant to working with complex challenges?

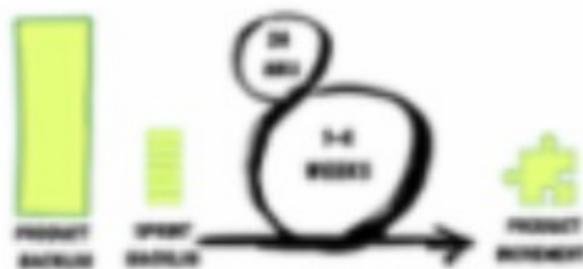
Team 1

Share what you know about each of the elements below.



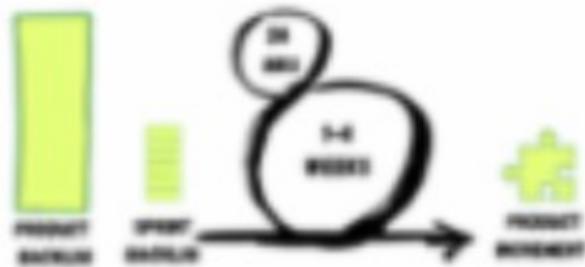
Team 2

Share what you know about each of the elements below.



Team 3

Share what you know about each of the elements below.



Team 4

Share what you know about each of the elements below.



Team 1

A healthy Product Backlog is...



The further you look out...



Team 2

A healthy Product Backlog is...



The further you look out...



Team 3

A healthy Product Backlog is...



The further you look out...



Team 4

A healthy Product Backlog is...



The further you look out—



Time-box exercise

Where do you use a timebox? (Individual or personal, organisational)



Team 1

Guess the values



Can you guess what each pillar stands for?



What does Empiricism have to do with complex work?

Time-box exercise

Where do you use a timebox? (Individual or personal, organisational)



Team 3

Guess the values



Can you guess what each pillar stands for?



What does Empiricism have to do with complex work?

Time-box exercise

Where do you use a timebox? (Individual or personal, organisational)



Team 2

Guess the values



Can you guess what each pillar stands for?



What does Empiricism have to do with complex work?

Time-box exercise

Where do you use a timebox? (Individual or personal, organisational)



Team 4

Guess the values



Can you guess what each pillar stands for?



What does Empiricism have to do with complex work?

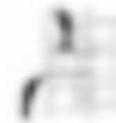
Team 1

- What makes forces proportional to?
- $F \propto$ make up of elements which push/tension
- $F \propto$ makes the displacement linear to a process
- $F \propto$ process brings compression to life
- $F \propto$ makes one the loadbearing changes in length?



Team 2

- What makes forces proportional to?
- $F \propto$ make up of elements which push/tension
- $F \propto$ makes the displacement linear to a process
- $F \propto$ process brings compression to life
- $F \propto$ makes one the loadbearing changes in length?



Team 3

- What makes forces proportional to?
- $F \propto$ make up of elements which push/tension
- $F \propto$ makes the displacement linear to a process
- $F \propto$ process brings compression to life
- $F \propto$ makes one the loadbearing changes in length?



Team 4

- What makes forces proportional to?
- $F \propto$ make up of elements which push/tension
- $F \propto$ makes the displacement linear to a process
- $F \propto$ process brings compression to life
- $F \propto$ makes one the loadbearing changes in length?



Team 1

Build your own Scrum



Scrum Board

Team 2

Build your own Scrum



Scrum Board

Team 3

Build your own Scrum



Scrum Board

Team 4

Build your own Scrum



Scrum Board

Scrum board for Team 1, Team 2, Team 3, Team 4

Team 1



REFACTORING

Refactoring is about changing the structure of existing code without changing its behavior.
It's changing the code without changing the behavior.



AUTOMATION

Testing is a repetitive process and manual tests are time-consuming.
Automated tests are repeatable and can be run quickly.

UNIT TESTING

Unit testing can be used to verify the behavior of specific code units which could be more difficult to test.



CONTINUOUS INTEGRATION

Testing is done by refactoring in discrete parts.
It's also known as 'continuous integration'.

Team 2



REFACTORING

Refactoring is about changing the structure of existing code without changing its behavior.
It's changing the code without changing the behavior.



AUTOMATION

Testing is a repetitive process and manual tests are time-consuming.
Automated tests are repeatable and can be run quickly.

UNIT TESTING

Unit testing can be used to verify the behavior of specific code units which could be more difficult to test.



CONTINUOUS INTEGRATION

Testing is done by refactoring in discrete parts.
It's also known as 'continuous integration'.

Team 3



REFACTORING

Refactoring is about changing the structure of existing code without changing its behavior.
It's changing the code without changing the behavior.



AUTOMATION

Testing is a repetitive process and manual tests are time-consuming.
Automated tests are repeatable and can be run quickly.

UNIT TESTING

Unit testing can be used to verify the behavior of specific code units which could be more difficult to test.



CONTINUOUS INTEGRATION

Testing is done by refactoring in discrete parts.
It's also known as 'continuous integration'.

Team 4



REFACTORING

Refactoring is about changing the structure of existing code without changing its behavior.
It's changing the code without changing the behavior.



AUTOMATION

Testing is a repetitive process and manual tests are time-consuming.
Automated tests are repeatable and can be run quickly.

UNIT TESTING

Unit testing can be used to verify the behavior of specific code units which could be more difficult to test.



CONTINUOUS INTEGRATION

Testing is done by refactoring in discrete parts.
It's also known as 'continuous integration'.

Team 1

Team 2

Team 4

Team 3



Planning:

True

False



Daily Scrum

True

False



Retrospective:

True

False



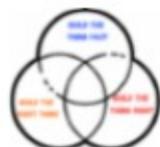
Sprint Review:

True

False



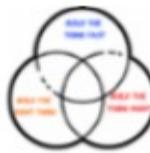
Team 1



BUILD THE RIGHT THING

Chocolate
Ice Cream

Team 2

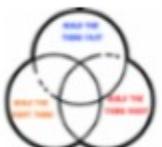


BUILD THE RIGHT THING



BUILD THE THING RIGHT

Team 3

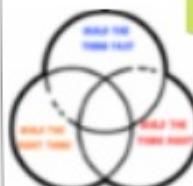


BUILD THE RIGHT THING



BUILD THE THING RIGHT

Team 4



BUILD THE RIGHT THING



BUILD THE THING RIGHT

What could be the impact of a team not meeting the definition of done for the increment? How would you go about influencing a team who were new to the idea of getting the increment to D.O.D to see the benefit?

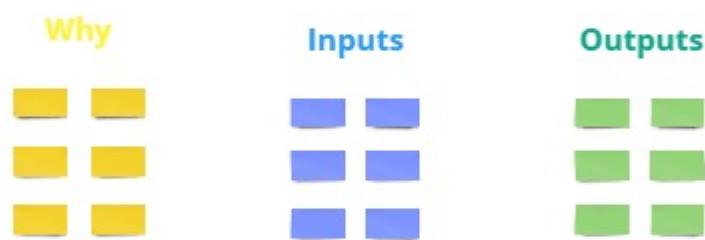
Team 1

Team 2

Team 3

Team 4

Sprint Planning



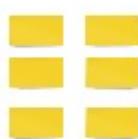
How does this event bring empiricism to life?



Daily Scrum



What is the purpose of the Daily Scrum?



What should be discussed?



Who should attend?



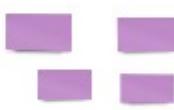
What is the timebox for this event?



What is the outcome of this event?



What kind of impediments would a ScrumMaster help with?



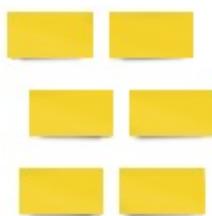
How does this event bring empiricism to life?



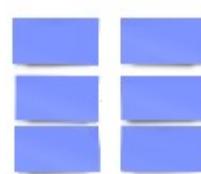
Sprint Retro



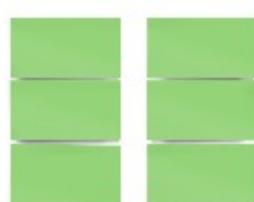
Why do we do this event?



Who should attend?



What should the ScrumMaster do here?



What is presented here?



What is the timebox?



What is the outcome of this event?



How does this event bring empiricism to life?



Sprint Review



Why do we do the Sprint Review?



Name some of activities that occur during this event?



Who should attend?



What is the timebox?



What should the ScrumMaster do here?



What are the possible outcomes of this event?



How does this event bring empiricism to life?



Planning:

True

False



Daily Scrum

True

False



Retrospective:

True

False



Sprint Review:

True

False



Planning:

True

False



Daily Scrum:

True

False



Retrospective:

True

False



Sprint Review:

True

False



Planning:

True

False



Daily Scrum

True

False



Retrospective:

True

False



Sprint Review:

True

False



Team 1

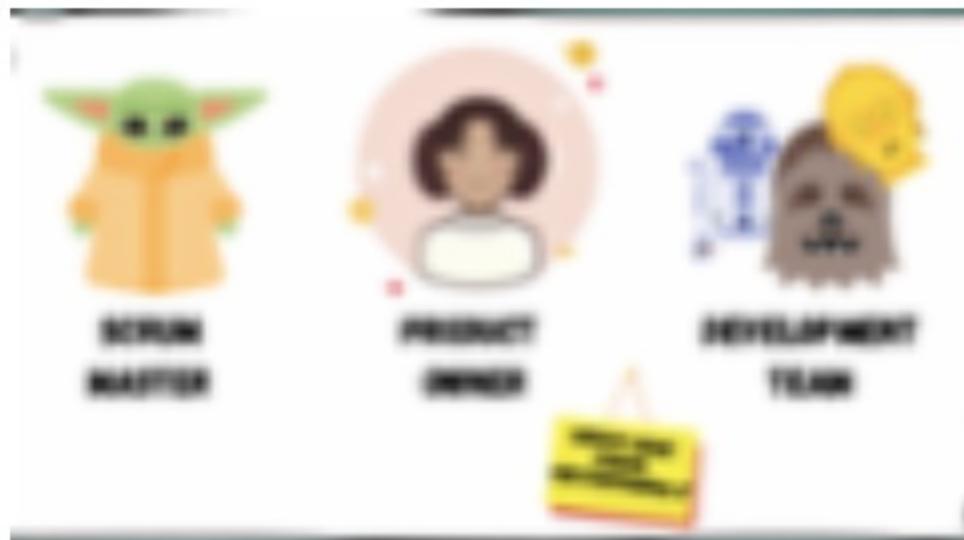
Team 2

Team 3

Team 4



What metaphors come to mind when you think of the below Scrum Roles? Create post it notes for your metaphors below



Team 1

THE HEART OF SCRUM

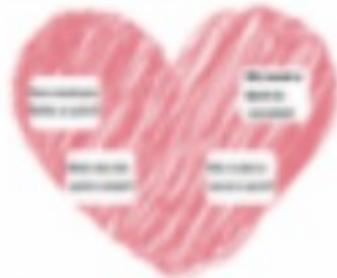


- No change after changing something
- Identify goals, keep the course
- Understand changes & act on it immediately



Team 2

THE HEART OF SCRUM



- No change after changing something
- Identify goals, keep the course
- Understand changes & act on it immediately



Team 4

THE HEART OF SCRUM



- No change after changing something
- Identify goals, keep the course
- Understand changes & act on it immediately



Team 3

THE HEART OF SCRUM



- No change after changing something
- Identify goals, keep the course
- Understand changes & act on it immediately



Team 1

Team 2

Change
Driver

**SIGNS OF A HEALTHY
SPRINT BACKLOG**



**SIGNS OF A HEALTHY
SPRINT BACKLOG**



Team 4

Team 3



**SIGNS OF A HEALTHY
SPRINT BACKLOG**



**SIGNS OF A HEALTHY
SPRINT BACKLOG**



Team 1

Exercise #2

Check out the below example of a fictional goal. As the sprint begins, things change & we now need new options!

THE GOAL		NEW OPTIONS
SPRINT	GOALS	SPRINT
SPRINT 1	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction
SPRINT 2	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction

Use the Post-It notes to create new options
without prototyping the Sprint Goal!

Team 2

Exercise #2

Check out the below example of a fictional goal. As the sprint begins, things change & we now need new options!

THE GOAL		NEW OPTIONS
SPRINT	GOALS	SPRINT
SPRINT 1	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction
SPRINT 2	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction

Use the Post-It notes to create new options
without prototyping the Sprint Goal!

Team 4

Exercise #2

Check out the below example of a fictional goal. As the sprint begins, things change & we now need new options!

THE GOAL		NEW OPTIONS
SPRINT	GOALS	SPRINT
SPRINT 1	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction
SPRINT 2	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction

Use the Post-It notes to create new options
without prototyping the Sprint Goal!

Team 3

Exercise #2

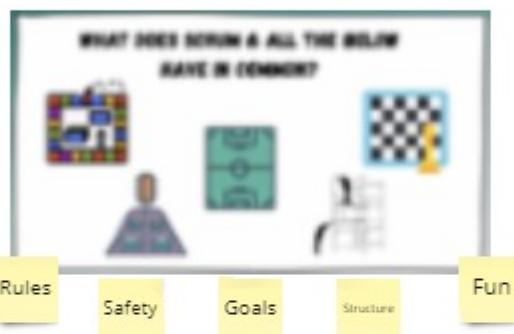
Check out the below example of a fictional goal. As the sprint begins, things change & we now need new options!

THE GOAL		NEW OPTIONS
SPRINT	GOALS	SPRINT
SPRINT 1	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction
SPRINT 2	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction

Use the Post-It notes to create new options
without prototyping the Sprint Goal!

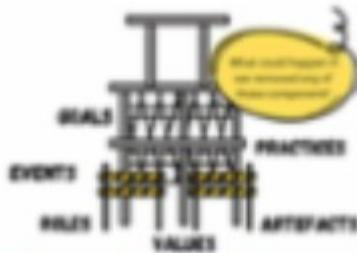
Scrum as a Game Quiz

Think about what it is that allows a game to be played - what stops people from playing football outside of the white lines & what encourages cars to drive on a certain side of the road & take action when they see different coloured lights.



The Half Scrum exercise

Imagine Scrum as a scaffolding create structure for a team to inspect & adapt, to learn, to grow - to be effective! What could happen then if we were to remove elements?



Daily Scrum

Sprint Planning

Refinement

Sprint Review

Sprint Retro

Creating potentially releasable increment

Not sticking to the time-box

Removing the Product Owner role

Blend the PO & SM role

Allowed the Sprint Goal to change

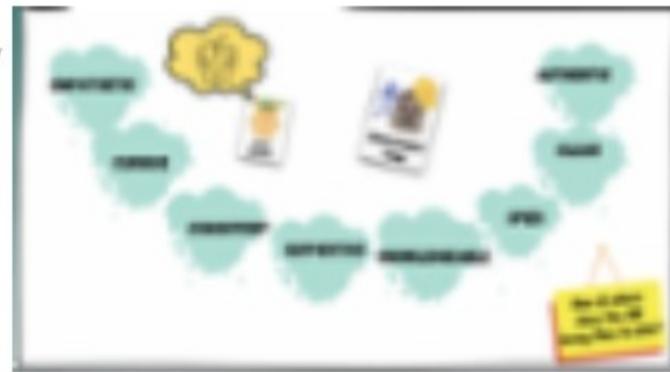
Let anyone enter Retrospective

Demo work which doesn't meet D.O.D

Exercise #1

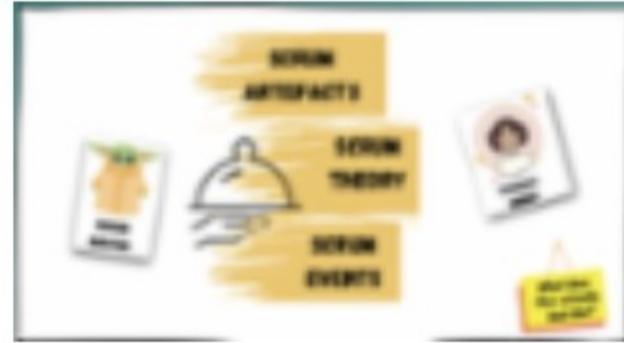
Service to Team

Create cards to describe how the ScrumMaster as a servant leader may bring these ideas to life to serve the team.



Service to Product Owner

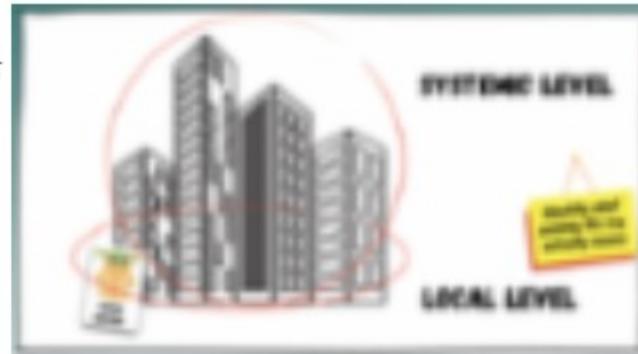
Create cards to describe how the ScrumMaster serve the Product Owner. Consider using the hints!



Exercise #3

Service to Organisation

Create cards to describe how the ScrumMaster serve the organisation.



Wrap-up

Either drawing from personal experience or through creating a fictional scenario, explore the following question: When you tried to serve an organisation as a ScrumMaster what changes did the organisation need to take to give Scrum a chance of success? What organisational challenges got in the way which stopped or could stop Scrum from working?

◦

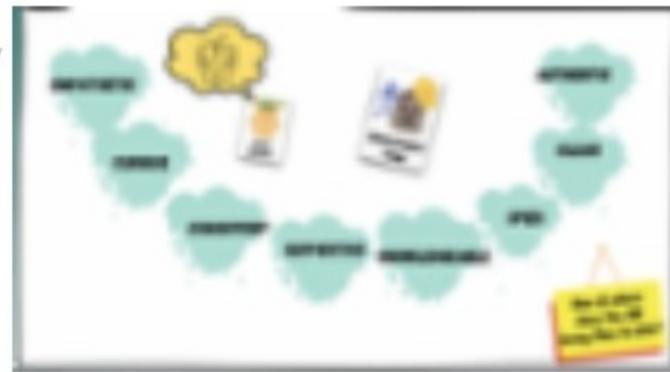
Organisational changes needed to help Scrum work

Organisational blockers getting in the way of Scrum working

Exercise #1

Service to Team

Create cards to describe how the ScrumMaster as a servant leader may bring these ideas to life to serve the team.



Service to Product Owner

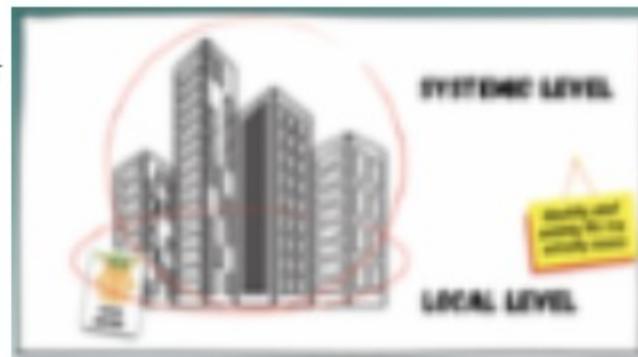
Create cards to describe how the ScrumMaster serve the Product Owner. Consider using the hints!



Exercise #3

Service to Organisation

Create cards to describe how the ScrumMaster serve the organisation.



Wrap-up

Either drawing from personal experience or through creating a fictional scenario, explore the following question: When you tried to serve an organisation as a ScrumMaster what changes did the organisation need to take to give Scrum a chance of success? What organisational challenges got in the way which stopped or could stop Scrum from working?

◦

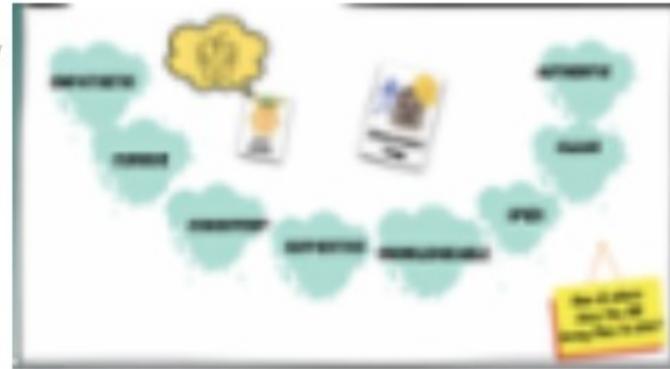
Organisational changes needed to help Scrum work

Organisational blockers getting in the way of Scrum working

Exercise #1

Service to Team

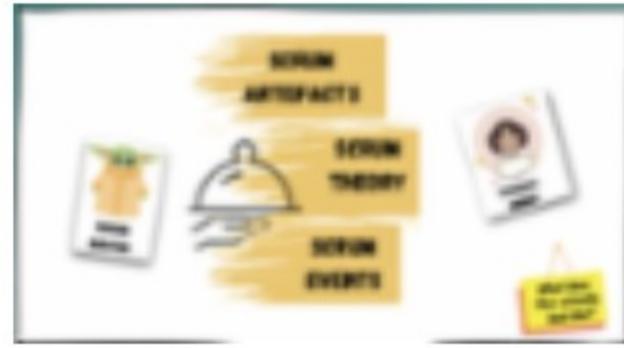
Create cards to describe how the ScrumMaster as a servant leader may bring these ideas to life to serve the team.



Exercise #2

Service to Product Owner

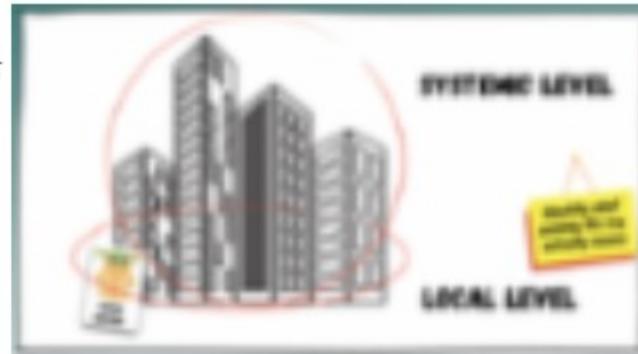
Create cards to describe how the ScrumMaster serve the Product Owner. Consider using the hints!



Exercise #3

Service to Organisation

Create cards to describe how the ScrumMaster serve the organisation.



Wrap-up

Either drawing from personal experience or through creating a fictional scenario, explore the following question: When you tried to serve an organisation as a ScrumMaster what changes did the organisation need to take to give Scrum a chance of success? What organisational challenges got in the way which stopped or could stop Scrum from working?

◦

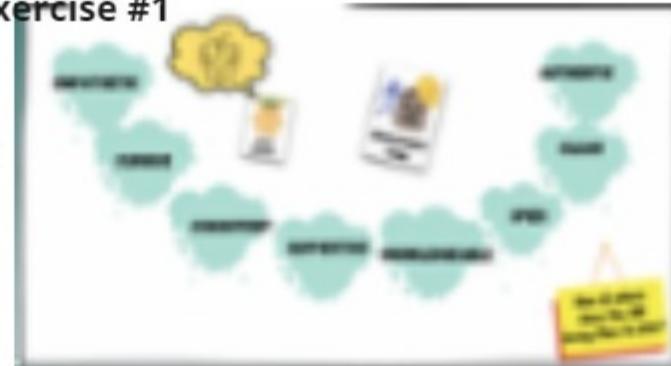
Organisational changes needed to help Scrum work

Organisational blockers getting in the way of Scrum working

Exercise #1

Service to Team

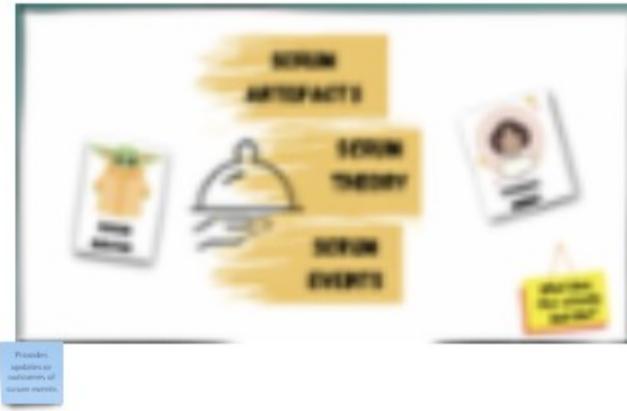
Create cards to describe how the ScrumMaster as a servant leader may bring these ideas to life to serve the team.



Exercise #2

Service to Product Owner

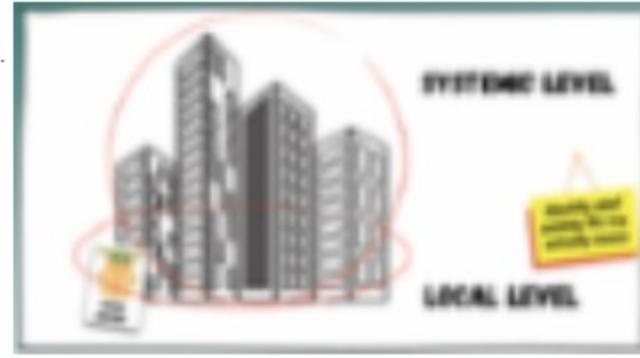
Create cards to describe how the ScrumMaster serve the Product Owner. Consider using the hints!



Exercise #3

Service to Organisation

Create cards to describe how the ScrumMaster serve the organisation.



Wrap-up

Either drawing from personal experience or through creating a fictional scenario, explore the following question: When you tried to serve an organisation as a ScrumMaster what changes did the organisation need to take to give Scrum a chance of success? What organisational challenges got in the way which stopped or could stop Scrum from working?

◦

Organisational changes needed to help Scrum work

Organisational blockers getting in the way of Scrum working

Team 1

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.

1	3	8
2	5	13



BANANA

APPLE

KIWI

MANGO

COCONUT

Team 2

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.

1	3	8
2	5	13



BANANA

APPLE

KIWI

MANGO

COCONUT

Team 3

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.

1	3	8
2	5	13



BANANA

MANGO

APPLE

KIWI

COCONUT

Team 4

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.

1	3	8
2	5	13



BANANA

KIWI

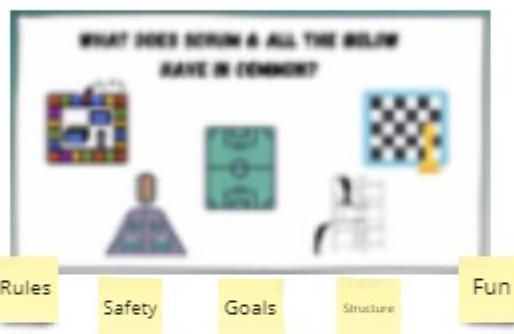
APPLE

MANGO

COCONUT

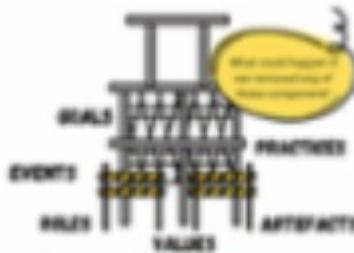
Scrum as a Game Quiz

Think about what it is that allows a game to be played - what stops people from playing football outside of the white lines & what encourages cars to drive on a certain side of the road & take action when they see different coloured lights.



The Half Scrum exercise

Imagine Scrum as a scaffolding create structure for a team to inspect & adapt, to learn, to grow - to be effective! What could happen then if we were to remove elements?



Daily Scrum

Sprint Planning

Refinement

Sprint Review

Sprint Retro

Creating potentially releasable increment

Not sticking to the time-box

Removing the Product Owner role

Blend the PO & SM role

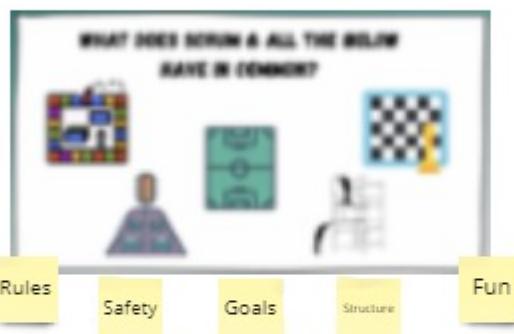
Allowed the Sprint Goal to change

Let anyone enter Retrospective

Demo work which doesn't meet D.O.D

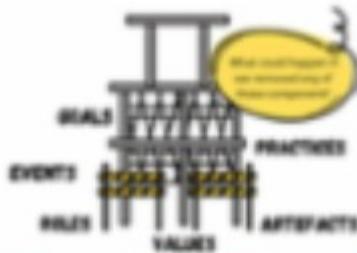
Scrum as a Game Quiz

Think about what it is that allows a game to be played - what stops people from playing football outside of the white lines & what encourages cars to drive on a certain side of the road & take action when they see different coloured lights.



The Half Scrum exercise

Imagine Scrum as a scaffolding create structure for a team to inspect & adapt, to learn, to grow - to be effective! What could happen then if we were to remove elements?



Daily Scrum

Sprint Planning

Refinement

Sprint Review

Sprint Retro

Creating potentially releasable increment

Not sticking to the time-box

Removing the Product Owner role

Blend the PO & SM role

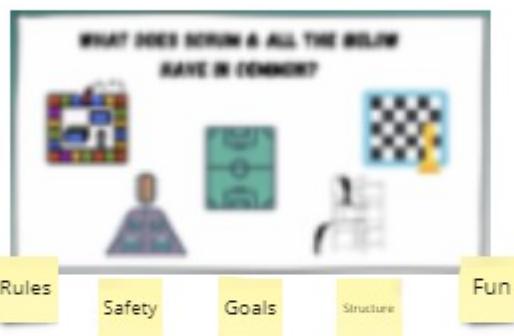
Allowed the Sprint Goal to change

Let anyone enter Retrospective

Demo work which doesn't meet D.O.D

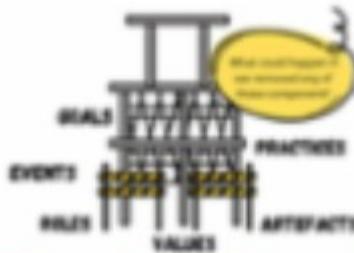
Scrum as a Game Quiz

Think about what it is that allows a game to be played - what stops people from playing football outside of the white lines & what encourages cars to drive on a certain side of the road & take action when they see different coloured lights.



The Half Scrum exercise

Imagine Scrum as a scaffolding create structure for a team to inspect & adapt, to learn, to grow - to be effective! What could happen then if we were to remove elements?



Daily Scrum

Sprint Planning

Refinement

Sprint Review

Sprint Retro

Creating potentially releasable increment

Not sticking to the time-box

Removing the Product Owner role

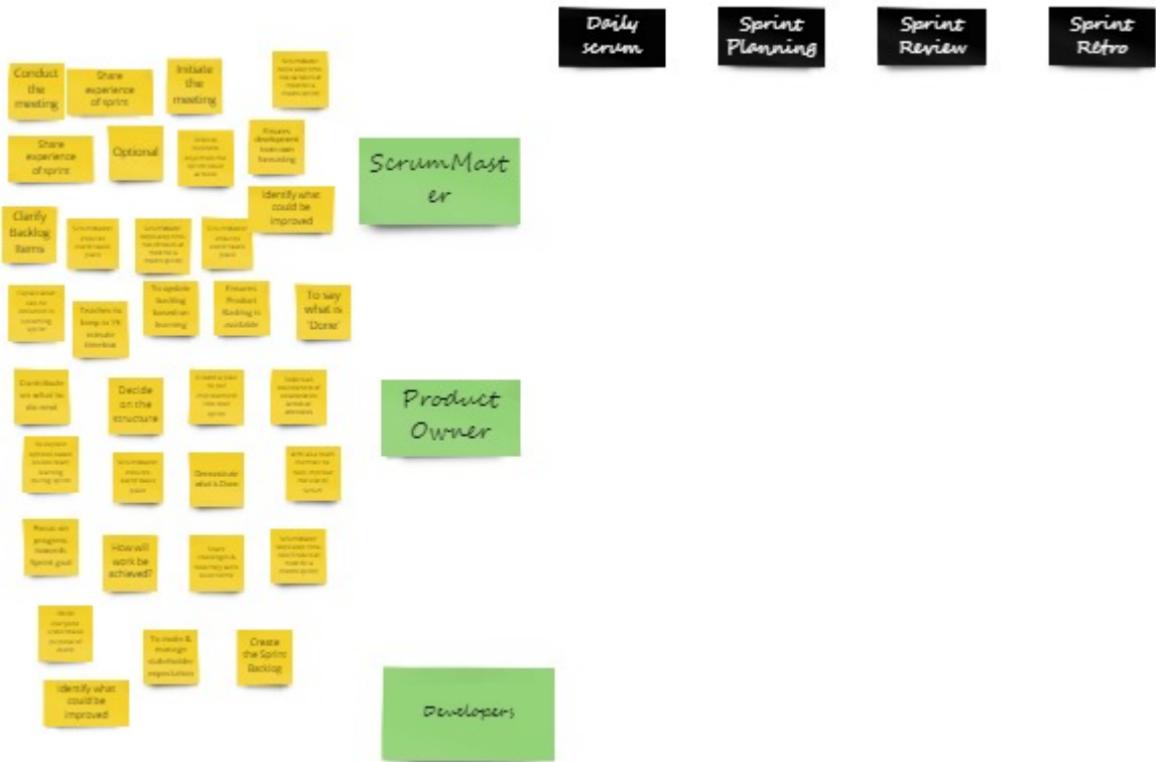
Blend the PO & SM role

Allowed the Sprint Goal to change

Let anyone enter Retrospective

Demo work which doesn't meet D.O.D

Roles & Event Mapping



The ScrumMaster wears more than one hat. Different situations benefit from different hats.

Ensures that the team bring in the retrospective item into new sprint

Teaching how to use empirical data to make decisions

Showing a team how to write a PBI

Help the organization see that certain incentives can affect Scrum adoption

Showing a team how to give a swift daily scrum update

Handling conflict between two developers to see if they can resolve the underlying issue

Showing a player how to push back against external requests, whilst in sprint

Reflecting back the fact that teams always have hang over of work

Teaching the P.O how to focus on the 'what' & 'why'

Help keep the timebox for the events

Helps the team remember to focus on sprint backlog at the Daily Scrum

Educating the team on the purpose of the events



Using the definitions above sort the yellow cards into the appropriate hat

How do the Scrum roles come together to deliver value?

Check out the hints below....

The diagram illustrates the Scrum roles and their interaction:

- Product Owner:** Represented by a yellow character with a green headband. Text above: "Who is responsible for maximizing the value of the product backlog? Who is responsible for defining the product vision? Who is responsible for the final product?"
- Scrum Master:** Represented by a pink character with a white headband. Text above: "Who is responsible for removing obstacles to the development team? Who is responsible for facilitating the Scrum process?"
- Development Team:** Represented by a blue character with a brown headband. Text above: "Who is responsible for creating the increment? Who is responsible for the self-organization of the team? Who is responsible for the technical aspects of the product?"

Below the characters is a yellow box containing the text: "Add to the poster - visualising how the roles work together to deliver the increment".



Refresh yourself with the Agile principles & values...

12 Principles of Agile Software Development

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Individuals and interactions over processes and tools
6. Working software over comprehensive documentation
7. Customer collaboration over contract negotiation
8. Responding to change over following a plan
9. Individuals and interactions over processes and tools
10. Working software over comprehensive documentation
11. Customer collaboration over contract negotiation
12. Responding to change over following a plan



Individuals & Interactions



Business & Technology



Marketing & Software



Customer & Stakeholder Communication



Customer Collaboration



Continuous Integration



Responding to Change



Following a Plan

Think about how often we deliver in Scrum, the roles which create interaction between the business & technology, how the developers are

Create cards demonstrating how Scrum brings Agile to life?

Handling challenges

Check out the below speech bubbles demonstrating challenges which pop up for a ScrumMaster. Either using these examples or your own identify 5 challenges from the Scrum team which go against Scrum rules & then provide ideas on how to resolve such challenges.



The ScrumMaster wears more than one hat. Different situations benefit from different hats.

Ensures that the team bring in the retrospective items into new sprint

Teaching how to use empirical data to make decisions

Showing a team how to write a PBI

Help the organization see that certain incentives can affect Scrum adoption

Showing a team how to give a swift daily scrum update

Handling conflict between two developers to see if they can resolve the underlying issue

Showing a player how to push back against external requests, whilst in sprint

Reflecting back the fact that teams always have hang over of work

Teaching the P.O how to focus on the 'what' & 'why'

Help keep the timebox for the events

Helps the team remember to focus on sprint backlog at the Daily Scrum

Educating the team on the purpose of the events



Using the definitions above sort the yellow cards into the appropriate hat

The ScrumMaster wears more than one hat. Different situations benefit from different hats.

Ensures that the team bring in the retrospective items into new sprint	Teaching how to use empirical data to make decisions
Showing a team how to write a PBI	Help the organization see that certain incentives can affect Scrum adoption
Showing a team how to give a swift daily scrum update	Handling conflict between two developers to see if they can resolve the underlying issue
Showing a player how to push back against external requests, whilst in sprint	Reflecting back the fact that teams always have hang over of work
Teaching the P.O how to focus on the 'what' & 'why'	Help keep the timebox for the events
Helps the team remember to focus on sprint backlog at the Daily Scrum	Educating the team on the purpose of the events



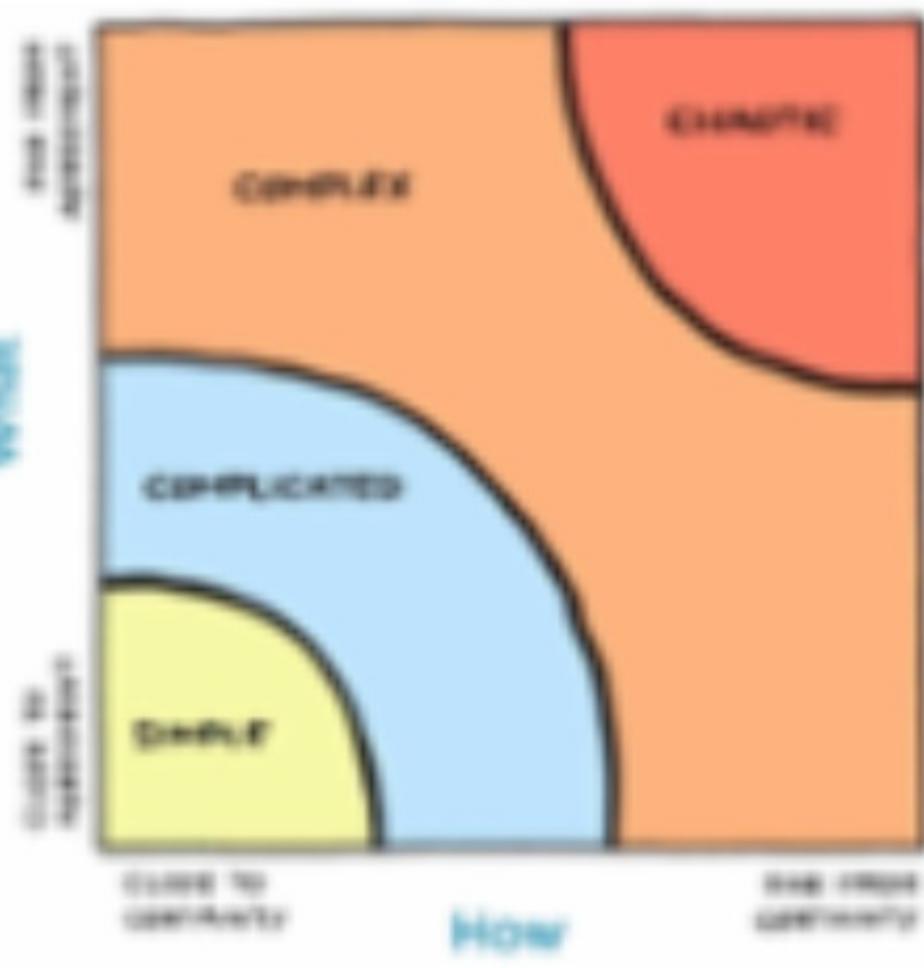
Using the definitions above sort the yellow cards into the appropriate hat

The ScrumMaster wears more than one hat. Different situations benefit from different hats.

Ensures that the team bring in the retrospective item into new sprint	Teaching how to use empirical data to make decisions
Showing a team how to write a PBI	Help the organization see that certain incentives can affect Scrum adoption
Showing a team how to give a swift daily scrum update	Handling conflict between two developers to see if they can resolve the underlying issue
Showing a player how to push back against external requests, whilst in sprint	Reflecting back the fact that teams always have hang over of work
Teaching the P.O how to focus on the 'what' & 'why'	Help keep the timebox for the events
Helps the team remember to focus on sprint backlog at the Daily Scrum	Educating the team on the purpose of the events

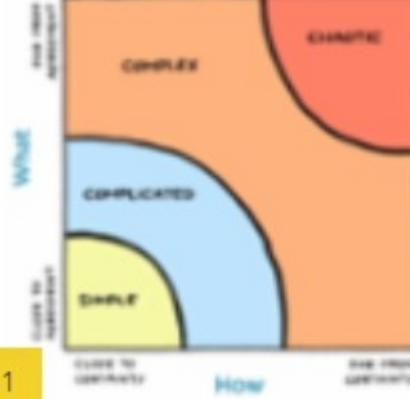


Using the definitions above sort the yellow cards into the appropriate hat



- Using the Stacey Matrix below, generate requirements & place them on the appropriate level of complexity
- These requirements can be work or non-work related
- Finally create a "Waterfall" & "Agile" post-it and place it to where you believe it belongs on the below model
- Bonus questions: Where does innovation live on this model & why?

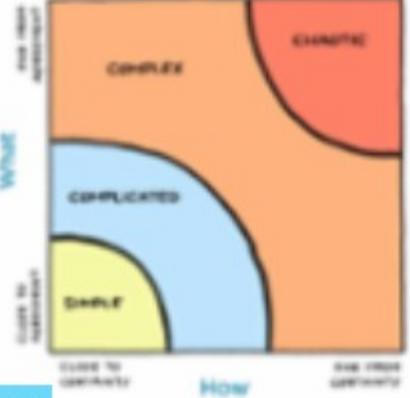
What type of requirements are you working with?



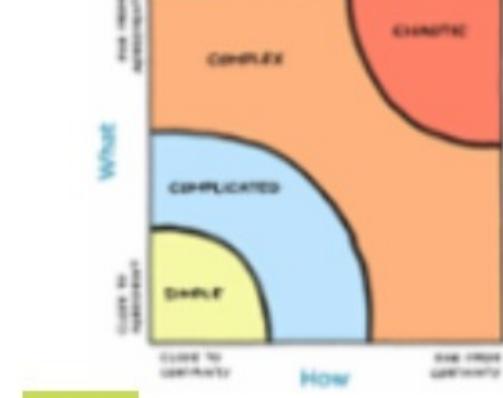
Team 1



Team 2

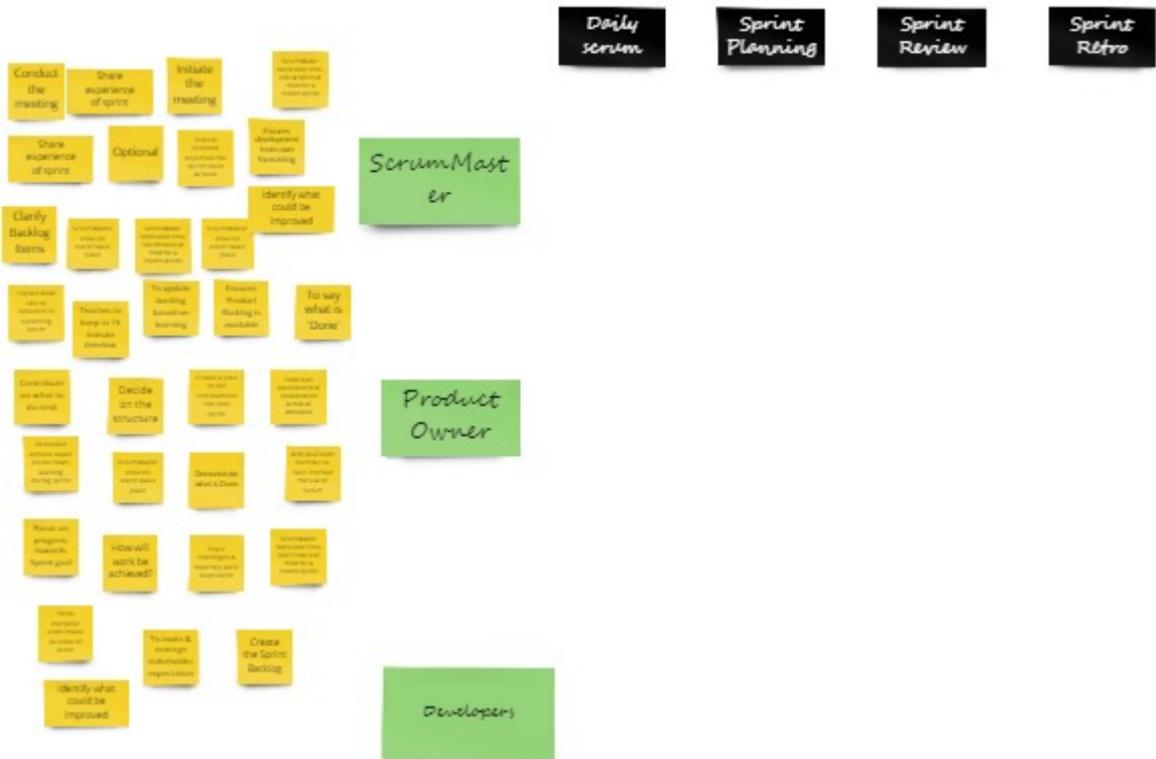


Team 3

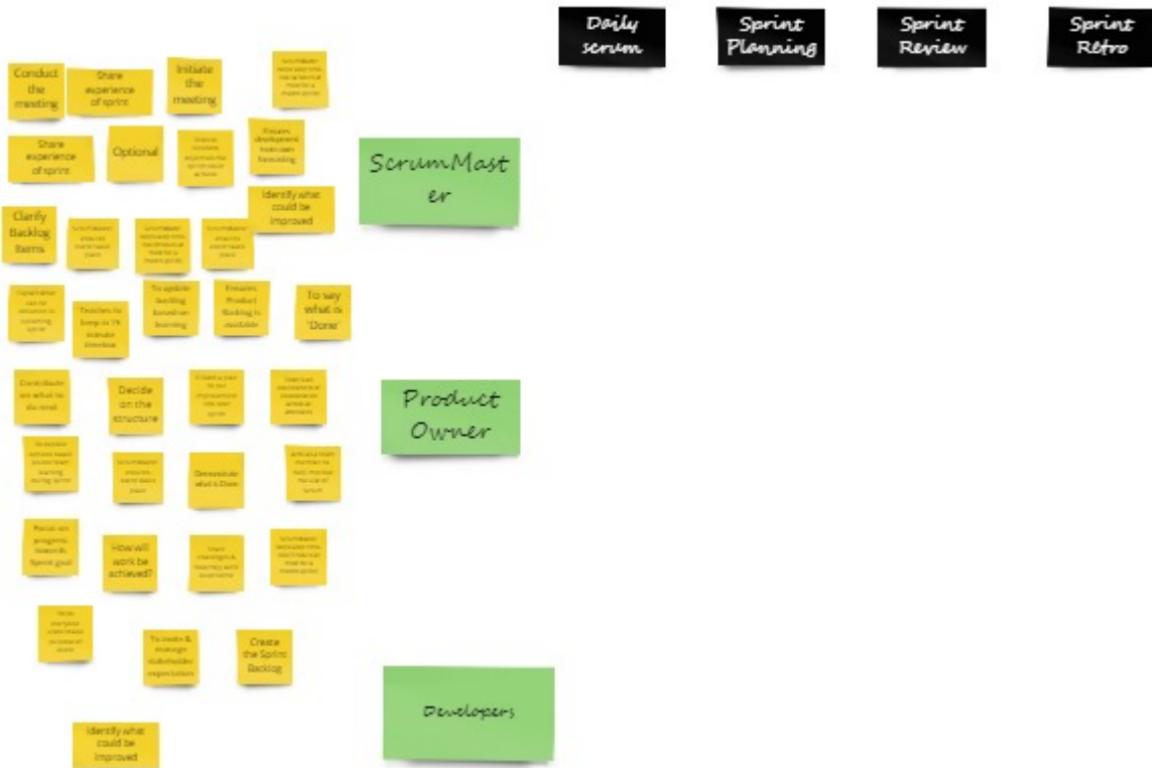


Team 4

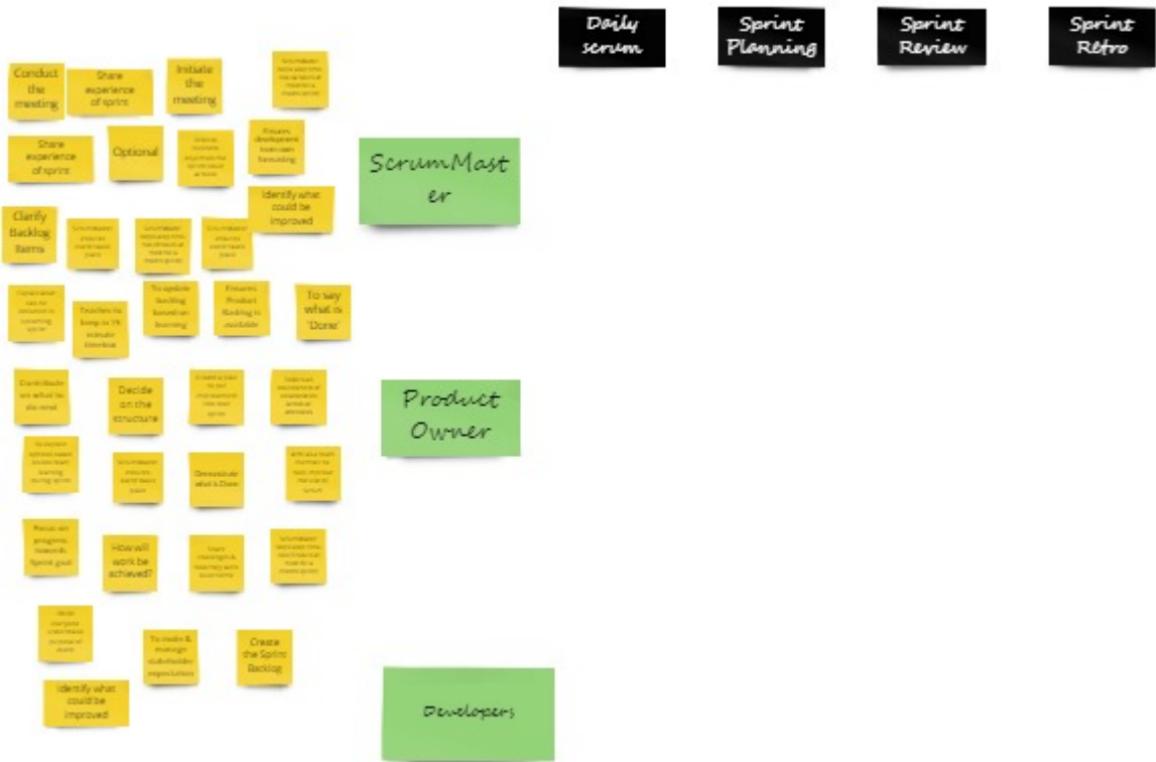
Roles & Event Mapping



Roles & Event Mapping



Roles & Event Mapping



Refresh yourself with the Agile principles & values...

12 Principles of Agile Software Development

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Individuals and interactions over processes and tools
6. Working software over comprehensive documentation
7. Customer collaboration over contract negotiation
8. Responding to change over following a plan
9. Individuals and interactions over processes and tools
10. Working software over comprehensive documentation
11. Customer collaboration over contract negotiation
12. Responding to change over following a plan



Individuals & Interactions



Process & Tools



Working software



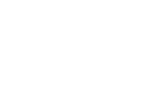
Comprehensive documentation



Customer collaboration



Responding to change



Planning



Publishing a plan

Think about how often we deliver in Scrum, the roles which create interaction between the business & technology, how the developers are

Create cards demonstrating how Scrum brings Agile to life?

Refresh yourself with the Agile principles & values...

12 Principles of Agile Software Development

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Individuals and interactions over processes and tools
6. Working software over comprehensive documentation
7. Customer collaboration over contract negotiation
8. Responding to change over following a plan
9. Individuals and interactions over processes and tools
10. Working software over comprehensive documentation
11. Customer collaboration over contract negotiation
12. Responding to change over following a plan



Think about how often we deliver in Scrum, the roles which create interaction between the business & technology, how the developers are

Create cards demonstrating how Scrum brings Agile to life?

How do the Scrum roles come together to deliver value?

Check out the hints below....



How do the Scrum roles come together to deliver value?

Check out the hints below....



How do the Scrum roles come together to deliver value?

Check out the hints below....



Handling challenges

Check out the below speech bubbles demonstrating challenges which pop up for a ScrumMaster. Either using these examples or your own identify 5 challenges from the Scrum team which go against Scrum rules & then provide ideas on how to resolve such challenges.



Handling challenges

Check out the below speech bubbles demonstrating challenges which pop up for a ScrumMaster. Either using these examples or your own identify 5 challenges from the Scrum team which go against Scrum rules & then provide ideas on how to resolve such challenges.



Handling challenges

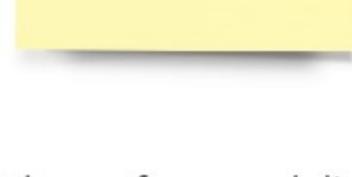
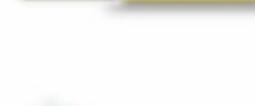
Check out the below speech bubbles demonstrating challenges which pop up for a ScrumMaster. Either using these examples or your own identify 5 challenges from the Scrum team which go against Scrum rules & then provide ideas on how to resolve such challenges.



Refresh yourself with the Agile principles & values...

12 Principles of Agile Software Development

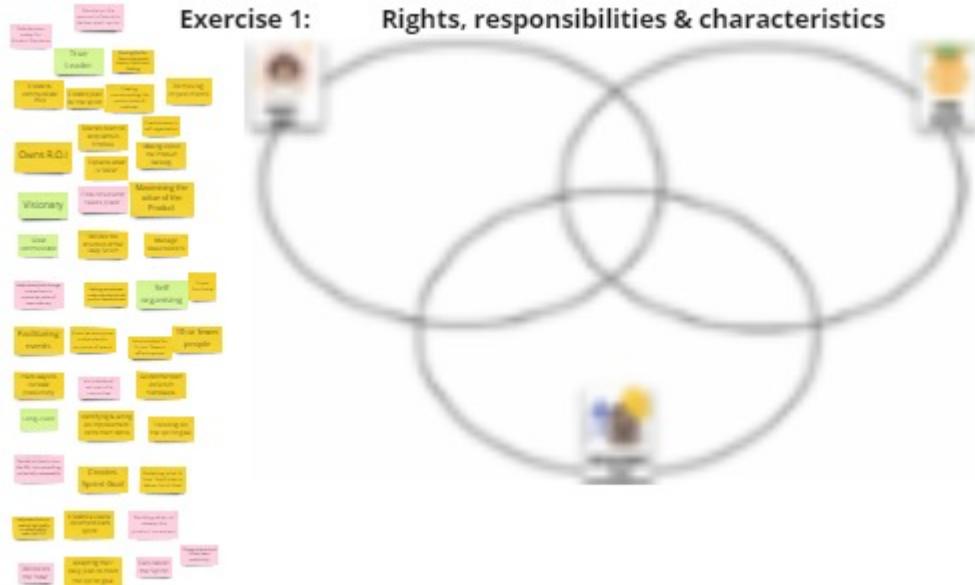
1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Individuals and interactions over processes and tools
6. Working software over comprehensive documentation
7. Customer collaboration over contract negotiation
8. Responding to change over following a plan
9. Individuals and interactions over processes and tools
10. Working software over comprehensive documentation
11. Customer collaboration over contract negotiation
12. Responding to change over following a plan



Think about how often we deliver in Scrum, the roles which create interaction between the business & technology, how the developers are

Create cards demonstrating how Scrum brings Agile to life?

Exercise 1: Rights, responsibilities & characteristics

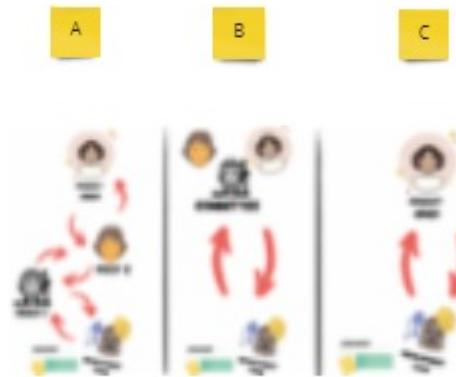


Zoom in: Product Owner as one role

Exercise 2:

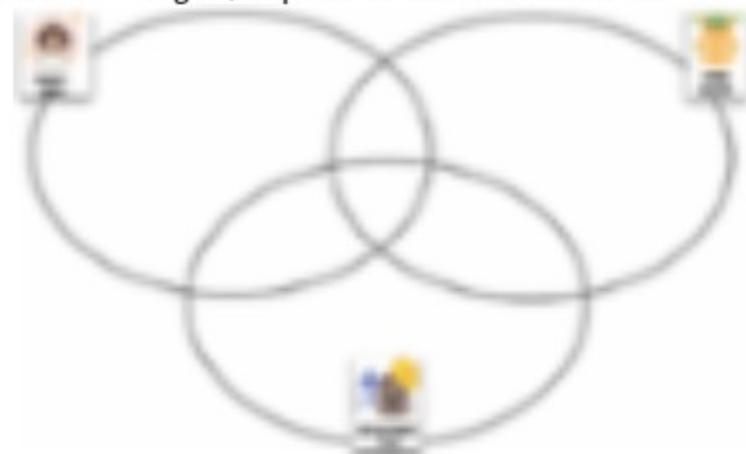
Observe the below scenarios for Product Owner & Dev team set up. What are the pros & cons of each scenario? Create cards for each answer.

Slower decisions	Too many cooks!
Good for support between PO & Dev teams	Might be hard to get PO time
Misunderstanding	Slower decisions
One decision maker	Potential misalignment
Many feedback loops	Faster responses
Not enough decision-making power from proxy	Weak support between PO & Dev teams
One decision-making power from proxy	Product owner becomes a bottleneck



Exercise 1: Rights, responsibilities & characteristics

CHARACTERISTICS

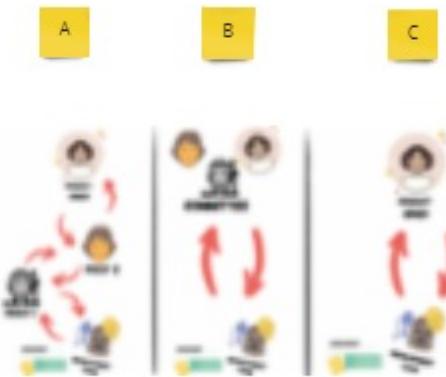


Zoom in: Product Owner as one role

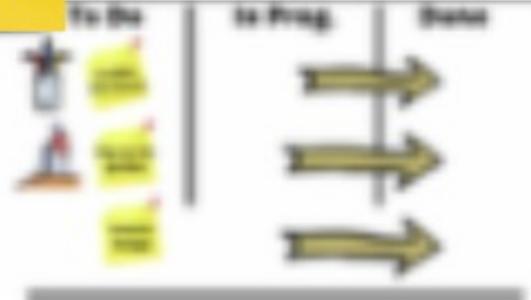
Exercise 2:

Observe the below scenarios for Product Owner & Dev team set up. What are the pros & cons of each scenario? Create cards for each answer.

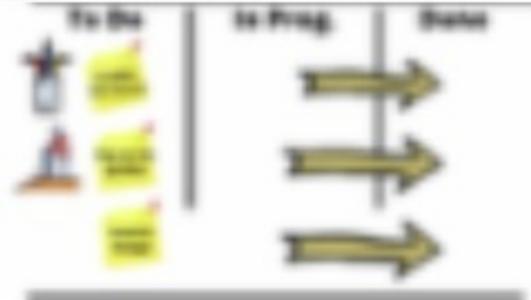
Slower decisions	Too many cooks!
Good for support between PO & Dev teams	Might be hard to get PO time
Misunderstanding	Slower decisions
One decision maker	Potential misalignment
Many feedback loops	Faster responses
Not enough decision-making power from proxy	Weak support between PO & Dev teams
One decision-making power from proxy	PO has better visibility to the team's work & morale



Team 1



Team 4



Team 1



Team 2



Team 4

1 2 3 5 8 13 21



Team 3



Team 1

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.



What factors can impact our estimation?

Answers: surface

Do we all peel the fruit the same way?

No

Do we all use the same equipment or?

No

Are we measuring that there is only one variety of coconut or mango?

No

Team 2

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.



What factors can impact our estimation?

Answers: surface

Do we all peel the fruit the same way?

No

Do we all use the same equipment or?

No

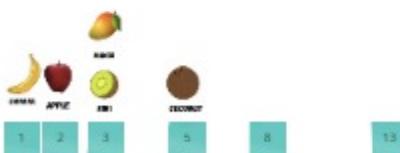
Are we measuring that there is only one variety of coconut or mango?

No

Team 4

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.



What factors can impact our estimation?

Answers: surface

Do we all peel the fruit the same way?

No

Do we all use the same equipment or?

No

Are we measuring that there is only one variety of coconut or mango?

No

Team 3

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.



What factors can impact our estimation?

Answers: tools required?

time

expertise

Do we all peel the fruit the same way?

Answers: yes or mostly

Do we all use the same equipment or?

Answers: no

Are we measuring that there is only one variety of coconut or mango?

Answers: we did

Team 1



Team 2



Team 4



Team 3





Agile Principles in 3 words

Look at the 12 Agile Principles, & describe them in 3 words using post-it notes that best captures the essence of the principle.

Team 1

12 Principles of Agile Software Development:

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Self-organizing teams
6. Regular iterations (1-4 weeks) over long-term planning
7. Simplicity, elegance, and functionality over unnecessary complexity
8. Sustainable development over maximizing productivity
9. Continuous integration over separate and infrequent integration
10. Refactoring over upfront design
11. Sustainable pace over schedules and timelines
12. Evolutionary development over finished products

1. responsive communication & collaboration
2. customer involvement
3. iterative development
4. self-organizing teams
5. regular iterations
6. simplicity, elegance, functionality
7. sustainable development
8. continuous integration
9. evolutionary development
10. communication & collaboration
11. sustainable pace
12. evolutionary development

Team 2

12 Principles of Agile Software Development:

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Self-organizing teams
6. Regular iterations (1-4 weeks) over long-term planning
7. Simplicity, elegance, and functionality over unnecessary complexity
8. Sustainable development over maximizing productivity
9. Continuous integration over separate and infrequent integration
10. Refactoring over upfront design
11. Sustainable pace over schedules and timelines
12. Evolutionary development over finished products



Agile Principles in 3 words

Team 3

Look at the 12 Agile Principles, & describe them in 3 words using post-it notes that best captures the essence of the principle.

12 Principles of Agile Software Development:

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Self-organizing teams
6. Regular iterations (1-4 weeks) over long-term planning
7. Simplicity, elegance, and functionality over unnecessary complexity
8. Sustainable development over maximizing productivity
9. Continuous integration over separate and infrequent integration
10. Refactoring over upfront design
11. Sustainable pace over schedules and timelines
12. Evolutionary development over finished products

7. Stories, tickets, delivered
8. Marathon pace sprint

1. Be proactive
2. Backlog, flexible

3. Rapid prototyping software
4. Stand up, grooming

5. Self-organized teams
6. Verbal standups

9. bug/defect free
10. Prioritize first

11. Accountable, responsible teams
12. Adapt to change

Agile Principles in 3 words

Team 4

Look at the 12 Agile Principles, & describe them in 3 words using post-it notes that best captures the essence of the principle.

12 Principles of Agile Software Development:

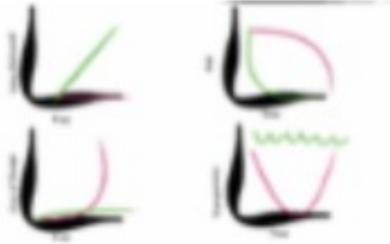
1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Self-organizing teams
6. Regular iterations (1-4 weeks) over long-term planning
7. Simplicity, elegance, and functionality over unnecessary complexity
8. Sustainable development over maximizing productivity
9. Continuous integration over separate and infrequent integration
10. Refactoring over upfront design
11. Sustainable pace over schedules and timelines
12. Evolutionary development over finished products



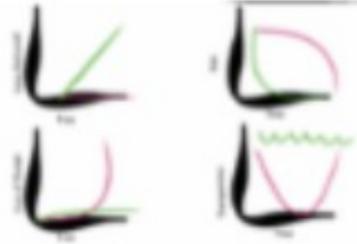
Why Agile - What do you see?

Below the colors, a line represents an adaptive agile approach. The purple line represents a traditional, prescriptive approach. Below each line you see the agile approach used in that team's approach.

Team 1



Team 2



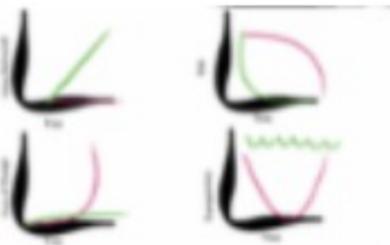
- Faster deployment
- Less cost
- Constant transparency

- Clear risks
- No change through plan development

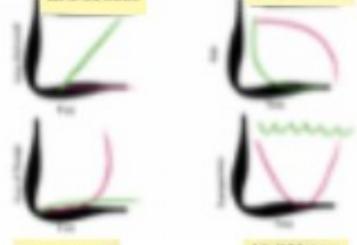
Why Agile - What do you see?

Below the colors, a line represents an adaptive agile approach. The purple line represents a traditional, prescriptive approach. Below each line you see the agile approach used in that team's approach.

Team 3



Team 4



Why Agile - What do you see?

Below the colors, a line represents an adaptive agile approach. The purple line represents a traditional, prescriptive approach. Below each line you see the agile approach used in that team's approach.

Benefits of Agile Approach:
more value delivered,
low risk
less cost of change,
very high
transparency

Benefits of
waterfall:
- If scope very
limited low
cost of change

Agile: low risk
Waterfall: high risk
Agile: high
transparency

Agile: low risk
over time
Waterfall: none

Agile: low and steady cost of
change
Waterfall: high cost of change if
scope changes

Agile: high frequency
of communication
Waterfall: none



Why do we do the Sprint Review?



Who should attend?



What should the ScrumMaster do here?



Name some of activities that occur during this event?



What is the timebox?



What are the possible outcomes of this event?



How does this event bring empiricism to life?



Demo
It gives a sense of transparency because it allows the stakeholders to have a clear picture of what have we done, what is happening while it impacts, allowing us to adapt.

CSM 25th-26th

Nov

A satisfying project experience

Think back to a positive experience you had with a project delivered. Generate Post It notes identifying reasons as to why using the factors below:



Individuals & interactions

Trying to understand the issue



Working software

Website Itself



Customer collaboration

Feedback- Every 2 Weeks

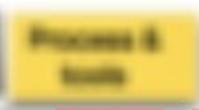


Responding to change

Adopt to change based on Product Owner/Customer



Free Website Host-Domain



AWS

JIRA-LINUX BOX or Server



Help/Design Document for End Users



Delivered documentation



3rd Party Website Creator



AWS-Contract



JIRA-Contract

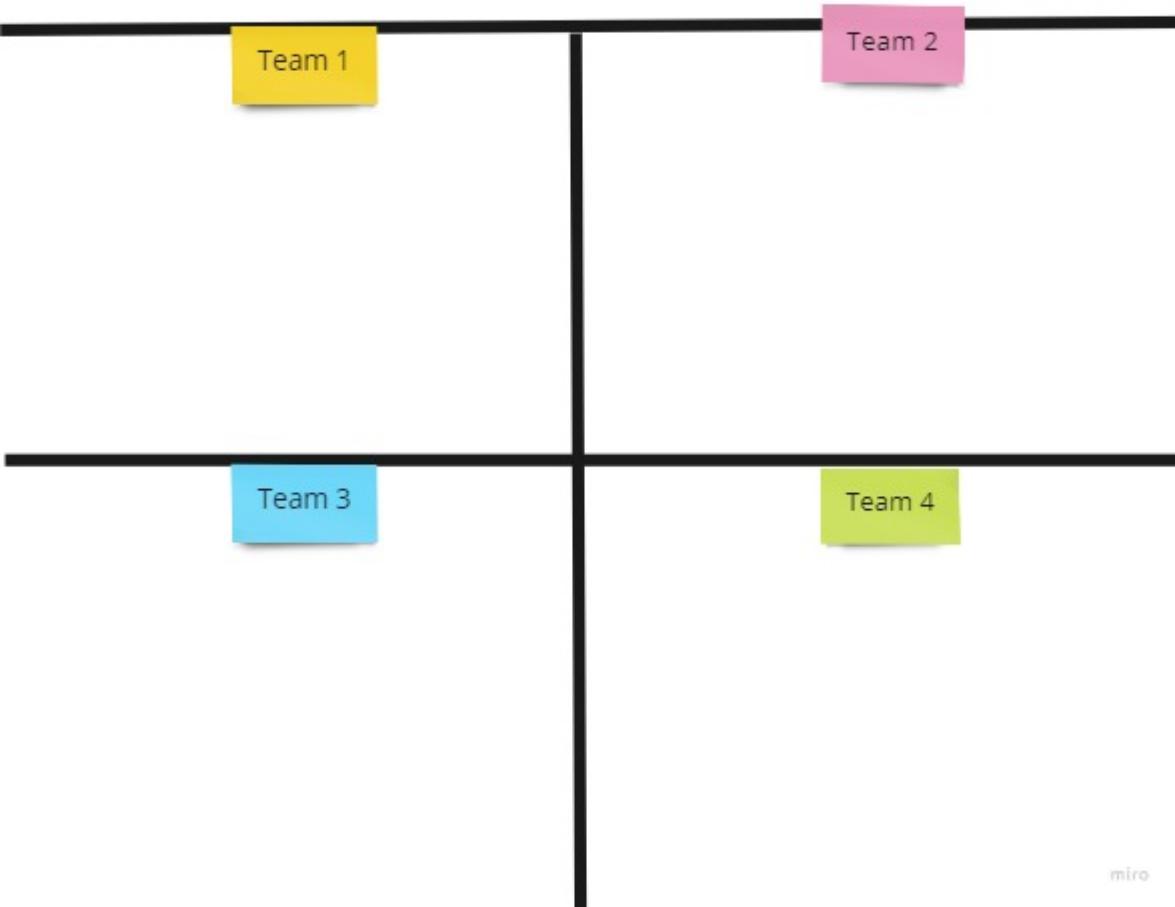


Following a plan

Available to Public

[Click here to watch the video!](#)

1. How much of these 3 intrinsic motivators do you have at work?
2. Give examples of what removing these motivators could look like at work?
3. Give examples of what could be done to introduce these motivators in?



Agile Principles in 3 words

Look at the 12 Agile Principles & describe them in 3 words using post-it notes that best captures the essence of the principle.

Team 2

12 Principles of Agile Software Development

1. Satisfy the customer through early and continuous delivery.
2. Welcome changing requirements, even late in development.
3. Deliver working software frequently
4. Business people and developers work together daily
5. Build projects around motivated individuals.
6. Convey information via face-to-face conversation.
7. Working software is the primary measure of progress.
8. Maintain a constant pace indefinitely.
9. Give continuous attention to technical excellence
10. Simplify: maximizing the amount of work not done
11. Teams self-organize.
11. Teams retrospect and tune behavior

3 words to describe everything

- Collaboration
- Self-management
- Quality

1.
- Communication
- Transparency
- Satisfaction

2.
- Adaptability
- Flexibility
- Quick learning

3.
- Timeframe
- Verified (UKT)
- Continuous Improvement

4.
- Involvement
- Collaboration
- Communication

Agile Principles in 3 words

Team 4

Look at the 12 Agile Principles & describe them in 3 words using post-it notes that best captures the essence of the principle.

12 Principles of Agile Software Development

1. Satisfy the customer through early and continuous delivery.
2. Welcome changing requirements, even late in development.
3. Deliver working software frequently.
4. Business people and developers work together daily.
5. Build projects around motivated individuals.
6. Convey information via face-to-face conversation.
7. Working software is the primary measure of progress.
8. Maintain a constant pace indefinitely.
9. Give continuous attention to technical excellence.
10. Simplify: maximizing the amount of work not done.
11. Teams self-organize.
12. Teams retrospect and tune behavior.

1. fast,
consistent,
proactive

2. flexible,
adjustable,
understanding

3. testable,
quality,
committed

4. collaboration,
open-minded,
progress

5. trust,
organized,
inspired

6. respect,
communication,
transparency

7. demo,
quality
assurance

8. focus,
avoiding
burnout

9. bug
free,
details

10. efficient,
effective,
streamlined

11. independence,
accountability
buddy

12. awareness,
improvement,
feedback

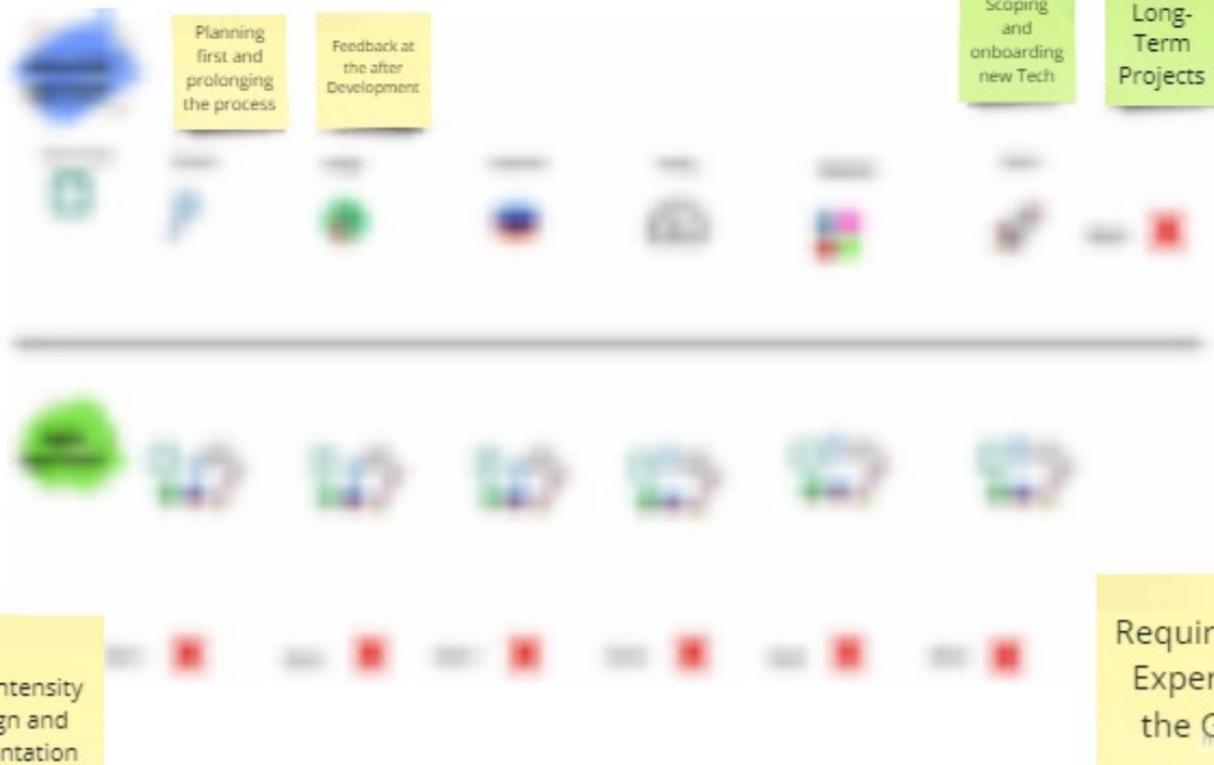
Create post it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organisation?

Planning first and prolonging the process

Feedback at the after Development

Scoping and onboarding new Tech

Long-Term Projects



Create post it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organisation?

Used for projects with few changes

For first time project

Backend development

Large projects Kick-off

Disadvantages
- Takes time to learn from features
- Difficult to make changes during development

Works sequentially
It is smooth

It is easier to make changes

Less chance to make mistakes at the delivery of the project

Get feedback

Team 1

What differences do you see?

Create post-it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organization?



Lack of intimacy of design and documentation

Requires an Expert on the Goal

What differences do you see?

Create post-it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organization?



Advantages: - Clear milestones, - transparent documentation

- Low maintenance costs

Time boxed

Retrospected

- Dynamic transparency - increasing alignment and acceptance

Team 4

What differences do you see?

Create post-it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organization?



Advantages: - Clear milestones, - transparent documentation

Advantages: - Room for modification - Transparency along the project

Making a car Faster

Team 1

Imagine a 12 month project.

The requirement documentation was created in January, capturing what the customer wanted.
What percentage of requirements stayed the same by the time the project was delivered in December?



Budget Objective of Project Prototype stays the same Scope Regulations

Requirements are Ever-changing

15-20% Stay the Same

Activity B

Team 2

Imagine a 12 month project.

The requirement documentation was created in January, capturing what the customer wanted.
What percentage of requirements stayed the same by the time the project was delivered in December?



15-20% Stay the Same
Budget change
Design changes
changes in how it is used to get all requirements at once
Competition



Team 3

Imagine a 12 month project.

The requirement documentation was created in January, capturing what the customer wanted.
What percentage of requirements stayed the same by the time the project was delivered in December?



- Technology and regulations could change
- Customer preferences can change
- Budget and scope of project could change
- Demand can change

25-75% could change

Stakeholders change



Team 4

Imagine a 12 month project.

The requirement documentation was created in January, capturing what the customer wanted.
What percentage of requirements stayed the same by the time the project was delivered in December?



Funding

End-user desirable outcome

Technology

35% remained the same



What's your story?

Name

Role

Use of Agile?

As a, I want, So that

MECHANICAL



ORGANIC



1. What are examples of mechanical & organic types of systems ?

2. Discuss the following statements:

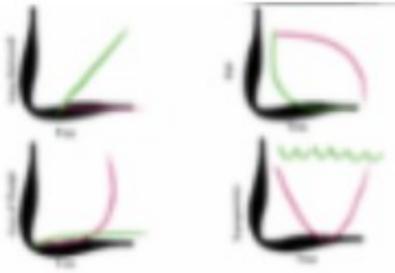
- A watch maker should be able to always figure out what is wrong with a watch they designed.
- The weather person should be able to guarantee the weather forecast for the following day?

Prescription Refill
ATM Money
Withdraw
Automated Pick
up at store

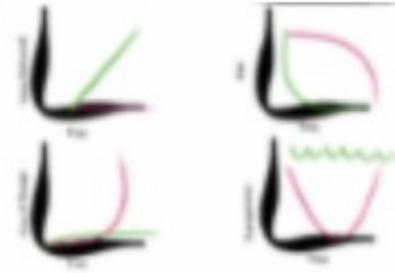
Why Agile - What do you see?

Same task assigned to two different teams an Agile approach. Use purple line represents a traditional, planned approach. What do you see for an Agile approach and many, from a traditional approach?

Team 1



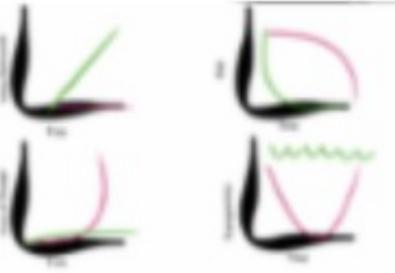
Team 2



Why Agile - What do you see?

Same task assigned to two different teams an Agile approach. Use purple line represents a traditional, planned approach. What do you see for an Agile approach and many, from a traditional approach?

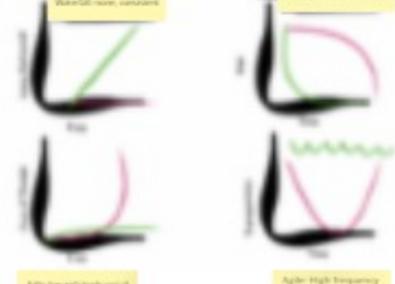
Team 3



Why Agile - What do you see?

Same task assigned to two different teams an Agile approach. Use purple line represents a traditional, planned approach. What do you see for an Agile approach and many, from a traditional approach?

Team 4



Benefits of Agile Approach
- more value delivered
- low risk
- low cost of change
- very high transparency

Benefits of waterfall
- if scope very limited low cost of change

Agile: low and steady cost of change
Waterfall: low cost of change & high transparency/visibility

Agile: high frequency of learning and continuous value delivery
Waterfall: linear

Team 1

SO, WHAT'S 'BETTER'?



Team 2

SO, WHAT'S 'BETTER'?



Team 3

SO, WHAT'S 'BETTER'?



Team 4

SO, WHAT'S 'BETTER'?



Create post it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organisation?

Given all
resources/tools
within 12month
timeline

Customer
does not
review until
final product

More
Documentation
is considered an
advantage

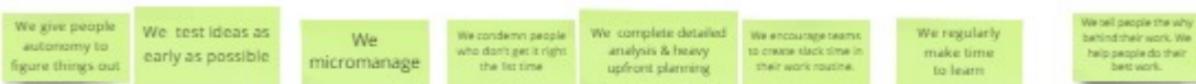
Inspect and
adjust

high degree
of customer
involvement

Advantages
-Room for modification
- Transparency along
the project

What does x belief look like?

Look at the (yellow) belief cards below & identify which action (green) card it matches. Once all cards have been matched, attempt to create more (green) belief cards which match one or more of the (yellow) belief cards.



Team 1



I believe that people can be trusted.

I believe that innovation is complex

I believe that people are lazy and need to always be managed.

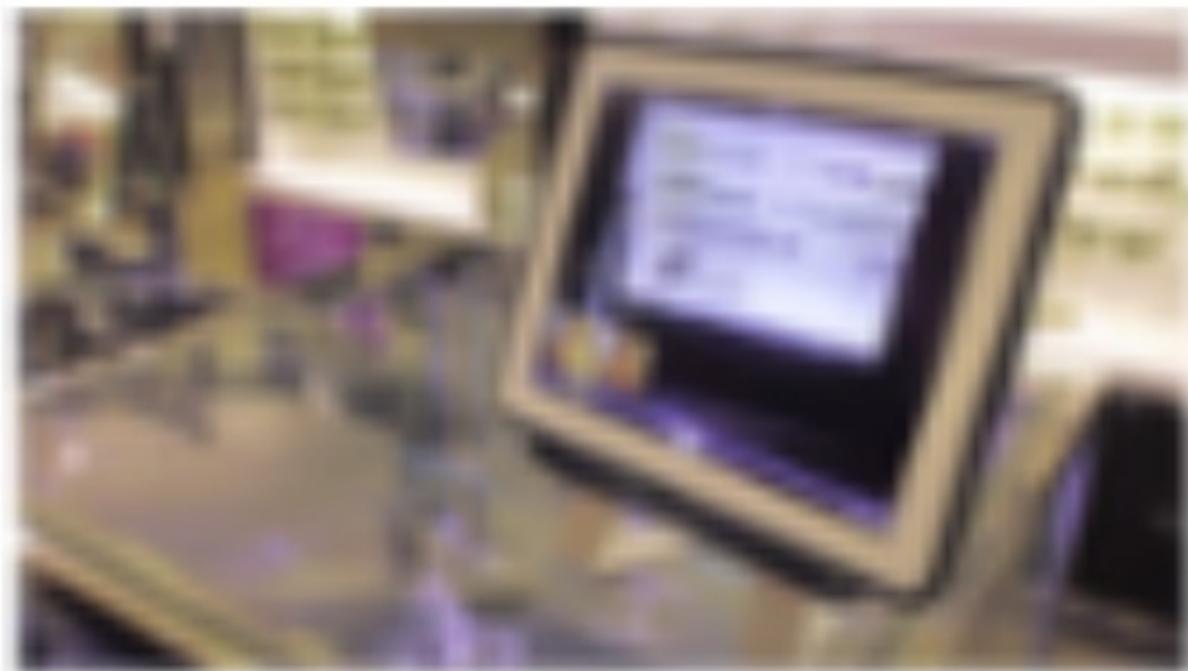
I believe that we must get things right the first time round.

I believe that we can always figure out what customers want at the start of a project.

I believe that we should always try to continuously improve.

I believe that people benefit from intrinsic motivators

Nordstorm Innovation labs



Agile Principles in 3 words

Team 3

Look at the 12 Agile Principles & describe them in 3 words using post-it notes that best captures the essence of the principle.

12 Principles of Agile Software Development

1. Satisfy the customer through early and continuous delivery.
2. Welcome changing requirements, even late in development.
3. Deliver working software frequently.
4. Business people and developers work together daily.
5. Build projects around motivated individuals.
6. Convey information via face-to-face conversation.
7. Working software is the primary measure of progress.
8. Maintain a constant pace indefinitely.
9. Give continuous attention to technical excellence.
10. Simplify: maximizing the amount of work not done.
11. Teams self-organize.
12. Teams retrospect and tune behavior.

1. Be proactive
2. Backlog, flexible

3. Rapid prototyping software
4. Stand up, grooming

5. Self organized teams
6. Verbal meetings

7. Stories, tickets delivered
8. Marathon pace sprint

9. Bug/defect free
10.

11. Accountable, responsible teams

What's your story?

Name

Role

Use of Agile?

As a, I want, So that

Create post it notes & identify the differences between the two approaches. What advantages & disadvantages are there? Which of the two approaches are you currently working with in your organisation?

- Frozen requirements
- No demos until the end

- Not as customer centric not as much customer interaction

- Heavy documentation

Advantage:
Good when you can not interact as much with customers

Time boxed

Restroinspect

- Dynamic requirements
- Showcasing demos after each sprint

What's your story?

Name

Role

Use of Agile?

As a, I want, So that

What's your story?

Name

Role

Use of Agile?

As a, I want, So that

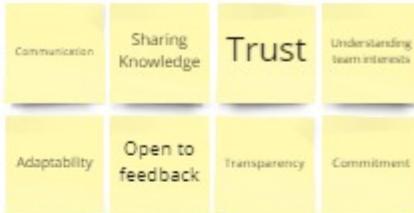
Team 1

What do you need from one another to be effective during this course?

1. Effective Communication
2. Team player
3. Adaptable/Flexible
4. Openness
5. Approachable
6. Creativity

Team2

What do you need from one another to be effective during this course?



Team 3

What do you need from one another to be effective during this course?

- Effective communication
- Trust in team members
- Collaboration
- Shared goal
- Punctual / on time
- Respect for each others opinions

Team 4

What do you need from one another to be effective during this course?



1. What Agile values do you feel your company currently embraces, which could do with introducing/refreshing? How would you sell the benefit?

Team 1

Team 2

2. What could be the challenges of introducing the Agile characteristics?



3. You enter the Dragons Den to convince them of using Agile, what is your elevator pitch?

Team 3

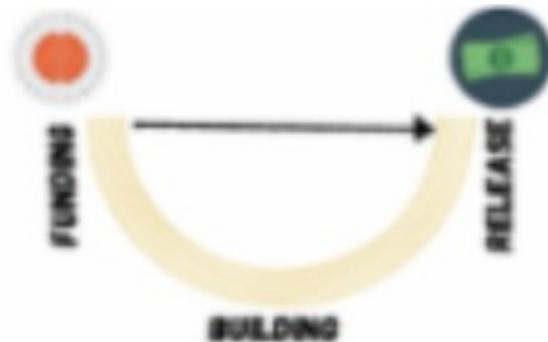
Team 4

4. How do you know you're building the right thing?

5. How do you know your team is improving?

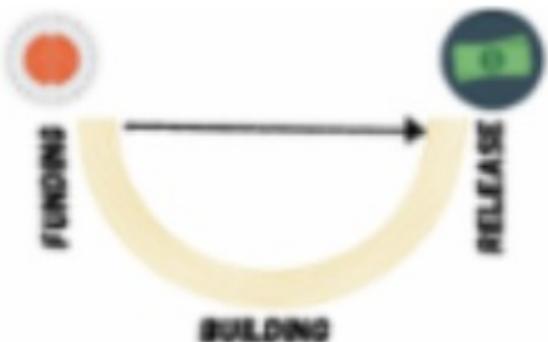
Team 1

How long to go from 'concept to cash'?



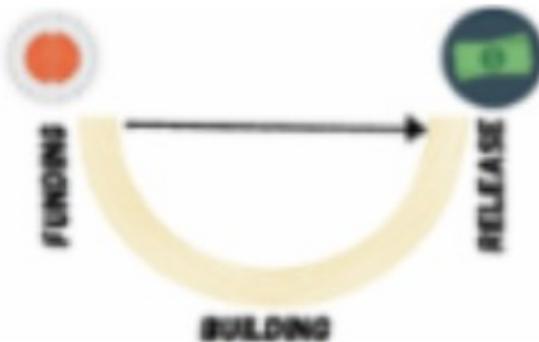
Team 3

How long to go from 'concept to cash'?



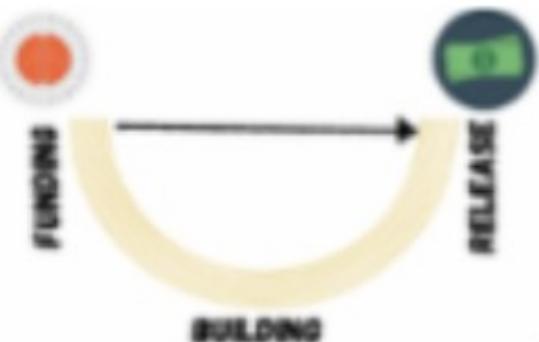
Team2

How long to go from 'concept to cash'?



Team 4

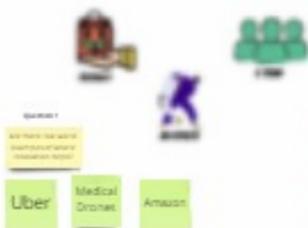
How long to go from 'concept to cash'?



Team 1

Is innovation essential?

we are defining innovation as: improving existing ways to do things better. It's not just the innovations that push markets forward & generate your sales, but to win against



Team 2

Is innovation essential?

we are defining innovation as: improving existing ways to do things better. It's not just the innovations that push markets forward & generate your sales, but to win against



- Nokia & Blackberry

Team 3

Is innovation essential?

we are defining innovation as: improving existing ways to do things better. It's not just the innovations that push markets forward & generate your sales, but to win against



Temperature scanners
Transportation by Train, Airport Check-in, Getting appointment in Hospitals (Digital care), Nokia - went out from in a bad place due to lack of innovation

Team 4

Is innovation essential?

we are defining innovation as: improving existing ways to do things better. It's not just the innovations that push markets forward & generate your sales, but to win against



Agile Principles in 3 words

Look at the 12 Agile Principles & describe them in 3 words using post-it notes that best captures the essence of the principle.

Team 1

12 Principles of Agile Software Development

1. Satisfy the customer through early and continuous delivery.
2. Welcome changing requirements, even late in development.
3. Deliver working software frequently
4. Business people and developers work together daily
5. Build projects around motivated individuals.
6. Convey information via face-to-face conversation.
7. Working software is the primary measure of progress.
8. Maintain a constant pace indefinitely.
9. Give continuous attention to technical excellence
10. Simplify: maximizing the amount of work not done
11. Teams self-organize.
12. Teams retrospect and have behavior

1. responsive, Commitment, Customer-focus
2. Adaptive, Agile, flexible
3. Reliable, Deliverable, Timely
4. Collaboration, Planning-Meeting, Synergy
5. Empowerment, Teamwork, Trust
6. Soft-skills, Effective Communication, Team Viewer
7. Velocity, Workability, Minimal Viable Product
8. Consistency, creating-a-baseline, steady
9. expertise, development, quality
10. Eliminating-waste, lean-approach, efficiency
11. cross-functional teams, experts, Self-sustainable
12. lessons learned, retrospective review, experience-based learning

Team 1



Cynical

Their are motivated by self-interest



Supportive

Providing encouragement/Cross-functional

Miscommunication
Between Customer
and PO

Team 2



Cynical

- Disconnection between leadership and scrum teams.
- Team dynamics
- Working software as a primary measure of progress.



Supportive

- Team support
- Team self-organize
- Move product forward (Maintain a constant pace)

Team 3



Cynical

- Lack of documentation
- Not much reference points from people joining late
- Technically team might not be equally sound



Supportive

- Have working software at the end of each sprint
- Flexible
- High productivity and low risk

Team 4



Cynical

- Lack of follow through
- time consuming



Supportive

- Open to change
- Quick turnaround
- Exciting to see projects through to completion during sprints
- Communication/ collaboration

What does x belief look like?

Look at the (yellow) belief cards below & identify which action (green) card it matches. Once all cards have been matched, attempt to create more (green) belief cards which match one or more of the (yellow) belief cards.

We give people autonomy to figure things out

We test ideas as early as possible

We micromanage

We condemn people who don't get it right the 1st time

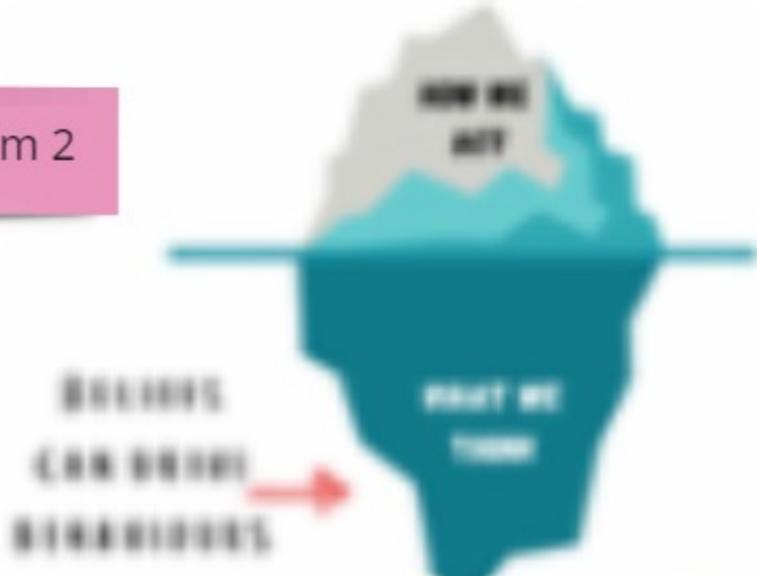
We complete detailed analysis & heavy upfront planning

We encourage teams to create slack time in their work routine

We regularly make time to learn

We tell people the why behind their work. We help people do their best work.

Team 2



I believe that people can be trusted.

I believe that innovation is complex

I believe that people are lazy and need to always be managed.

I believe that we must get things right the first time round.

I believe that we can always figure out what customers want at the start of a project.

I believe that we should always try to continuously improve.

I believe that people benefit from intrinsic motivators

What does x belief look like?

Look at the (yellow) belief cards below & identify which action (green) card it matches. Once all cards have been matched, attempt to create more (green) belief cards which match one or more of the (yellow) belief cards.

We give people autonomy to figure things out

We test ideas as early as possible

We micromanage

We condemn people who don't get it right the 1st time

We encourage teams to create slack time in their work routine

We regularly make time to learn

We tell people the why. We tell people the why behind their work. We help people do their best work.

Team 4



I believe that people can be trusted.

I believe that innovation is complex

I believe that people are lazy and need to always be managed.

I believe that we must get things right the first time round.

I believe that we can always figure out what customers want at the start of a project.

I believe that we should always try to continuously improve.

I believe that people benefit from intrinsic motivation.

What does x belief look like?

Look at the (yellow) belief cards below & identify which action (green) card it matches. Once all cards have been matched, attempt to create more (green) belief cards which match one or more of the (yellow) belief cards.

We give people autonomy to figure things out

We test ideas as early as possible

We micromanage

We condemn people who don't get it right the first time

We complete detailed analysis & heavy upfront planning

We encourage teams to create slack time in their work routine

We regularly make time to learn

We tell people the why behind their work. We help people do their best work.

Team 3



I believe that people can be trusted.

I believe that innovation is complex

I believe that people are lazy and need to always be managed.

I believe that we must get things right the first time round.

I believe that we can always figure out what customers want at the start of a project.

I believe that we should always try to continuously improve.

I believe that people benefit from intrinsic motivators

What do all of the below have in common?

Team 1



What do all of the below have in common?

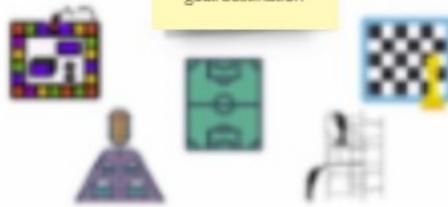
Team 2



What do all of the below have in common?

Team 3

Multiplayer
minimum 2
Systems/rules
Strategy
Human intervention
Shared
goal/destination



What do all of the below have in common?

Team 4



Team 1

Which of the two styles have you experienced?

Which leadership style requires less middle managers?

Are there scenarios where command & control is useful?

1. Both
2. Servant
leadership
3. None
4. Command & control
5. Both
Servant and styles

SERVANT LEADERSHIP



COMMAND & CONTROL



Team 3

Which of the two styles have you experienced?

Which leadership style requires less middle managers?

Are there scenarios where command & control is useful?

SERVANT LEADERSHIP



COMMAND & CONTROL



Have experienced both styles

Servant leadership requires less middle

1. Low skilled team who requires direction
2. Very time sensitive project

Team 2

Which of the two styles have you experienced?

Which leadership style requires less middle managers?

Are there scenarios where command & control is useful?

1. Both
2. Servant leadership
3. Yes, based on the needs and the team

SERVANT LEADERSHIP



COMMAND & CONTROL



COMMAND & CONTROL



Team 4

Which of the two styles have you experienced? **Both**

Which leadership style requires less middle managers? **Servant**

Are there scenarios where command & control is useful?

Involving roles command and control is not suitable for most installations, though it can be useful in some situations.
Below are some examples.

In roles or companies where compliance is at the forefront i.e. compliance, finance, budget etc.

SERVANT LEADERSHIP



COMMAND & CONTROL



COMMAND & CONTROL



What is value to you in the team that you work in?



Mini Value Stream

- Pick one team member's value stream
- That is a customer request aka requirement
- List the high level activities
- All the way to the delivery of the request



Team 1



CUSTOMER REQUEST



DELIVERED BY
SUPPLIER

ACTIVITIES



CUSTOMER REQUEST



DELIVERED BY
SUPPLIER

ACTIVITIES

Team 3



CUSTOMER REQUEST



DELIVERED BY
SUPPLIER

ACTIVITIES



CUSTOMER REQUEST



DELIVERED BY
SUPPLIER

ACTIVITIES

Team 2

Team 4

Team 1

What are the benefits & disadvantages of these approaches?

Everyone can have different approaches to their work, which can have varying strengths and weaknesses. It's up to the team to decide what's best for them. If you only have one idea for a solution, it's the wrong solution!



Team 2

What are the benefits & disadvantages of these approaches?

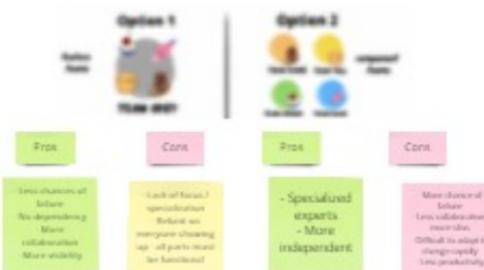
Everyone can have different approaches to their work, which can have varying strengths and weaknesses. It's up to the team to decide what's best for them. If you only have one idea for a solution, it's the wrong solution!



Team 3

What are the benefits & disadvantages of these approaches?

Everyone can have different approaches to their work, which can have varying strengths and weaknesses. It's up to the team to decide what's best for them. If you only have one idea for a solution, it's the wrong solution!



Team 4

What are the benefits & disadvantages of these approaches?

Everyone can have different approaches to their work, which can have varying strengths and weaknesses. It's up to the team to decide what's best for them. If you only have one idea for a solution, it's the wrong solution!



1. Can you name 5 of the Scrum events?

Team 1

Team 2



2. What's the difference between a feature & component team?
When would you used which?



3. What is the difference between Scrum leadership & command & control?

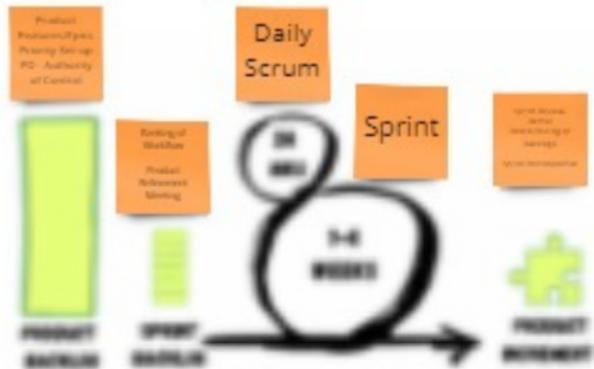
Team 3

Team 4

4. What is empiricism & how is it relevant to working with complex challenges?

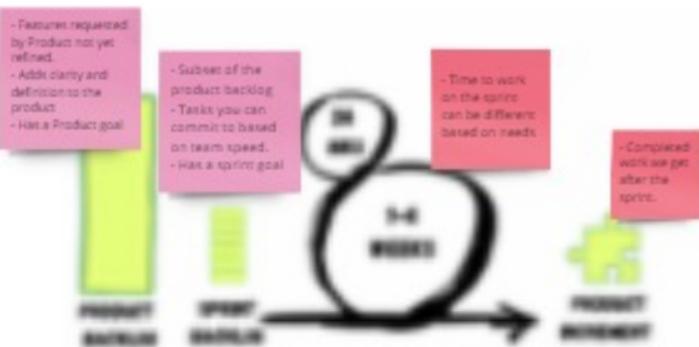
Team 1

Share what you know about each of the elements below.



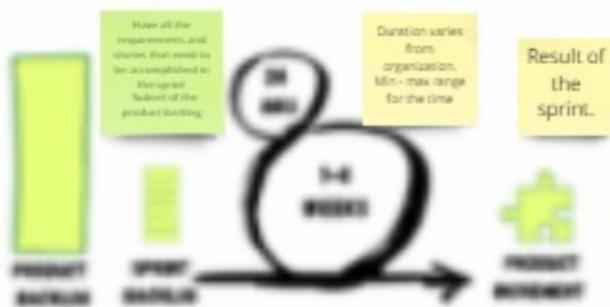
Team 2

Share what you know about each of the elements below.



Team 3

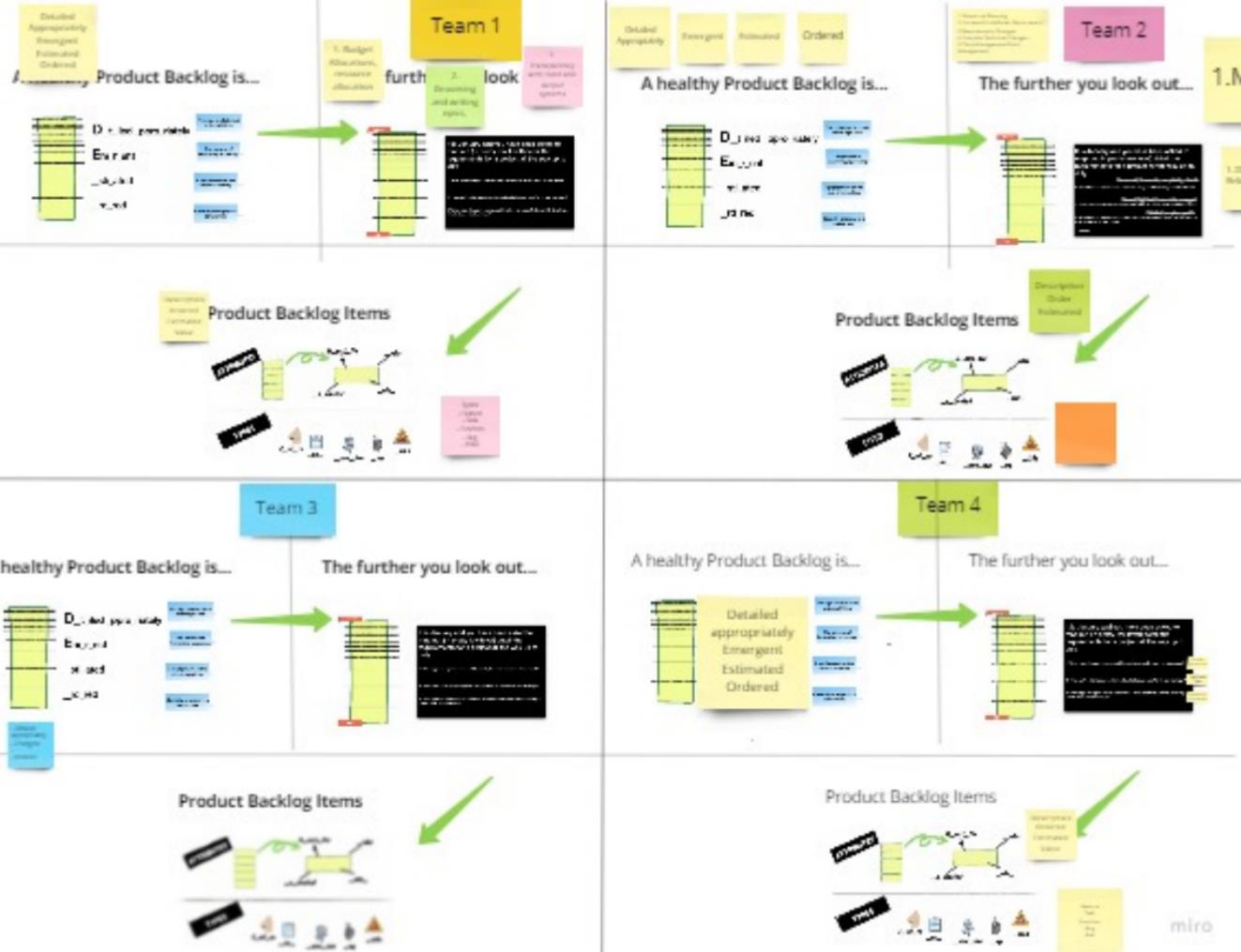
Share what you know about each of the elements below.



Team 4

Share what you know about each of the elements below.





<p>Time-box exercise</p> <p>Where do you use transparency in personal contexts?</p> <p>Personal: Building a staircase Personal: School Drop-off Personal: Daily stand-ups - keeping up with health</p>	<p>Team 1</p> <p>Guess the values</p> <p>Courage Commitment Focus Respect Openness Inspection</p> <p>SCUM VALUES</p> <p>Can you guess what each pillar stands for?</p> <p>Transparency Adaptation Inspection</p> <p>EMPIRICISM</p> <p>Transparency Adaptation Inspection</p> <p>What does Empiricism have to do with complex work?</p>
<p>What do all 3 items have in common?</p> <p>Minimum required Transparency</p>	<p>Can you guess what each pillar stands for?</p> <p>Minimum required Transparency</p> <p>What do all 3 items have in common?</p>
<p>Time-box exercise</p> <p>Where do you use transparency in professional contexts?</p> <p>Cook Customer Daily stand-ups</p>	<p>Team 3</p> <p>Guess the values</p> <p>Courage Commitment Focus Respect Openness Inspection</p> <p>SCUM VALUES</p> <p>Can you guess what each pillar stands for?</p> <p>Transparency Adaptation Inspection</p> <p>EMPIRICISM</p> <p>Transparency Adaptation Inspection</p> <p>What does Empiricism have to do with complex work?</p>
<p>What do all 3 items have in common?</p> <p>Transparency</p> <p>Can you guess what each pillar stands for?</p> <p>What does Empiricism have to do with complex work?</p> <p>They all give us feedback or facts about ourself</p>	<p>Team 4</p> <p>Guess the values</p> <p>Courage Commitment Focus Respect Openness Inspection</p> <p>SCUM VALUES</p> <p>Can you guess what each pillar stands for?</p> <p>Transparency Adaptation Inspection</p> <p>What does Empiricism have to do with complex work?</p> <p>They all give us feedback or facts about ourself</p>

Team 1

- What makes Scrum a framework ?
This is about creating a system, system of the framework

• It is made up of elements which give it structure
- framework elements

• Within the framework there is a process
- what is the process?

• This process brings empiricism to life
- how does it do that?

• Where are the feedbacks loops in Scrum?

Sprint



Retro perspective - product review

• Iteration and Incremental
- Iterative development
- Continuous improvement
- Adaptive process

• Product increment and sprint review
- Iteration improvement activity

- 1.Roles
- 2.Rules

- 1.Sprint

- What makes Scrum a framework ?
This is about creating a system, system of the framework

• It is made up of elements which give it structure
- framework elements

• Within the framework there is a process
- what is the process?

• This process brings empiricism to life
- how does it do that?

• Where are the feedbacks loops in Scrum?

Team 2



• Iteration

- 1.Retro/ Review

Team 3

- What makes Scrum a framework ?
This is about creating a system, system of the framework

• It is made up of elements which give it structure
- framework elements

• Within the framework there is a process
- what is the process?

• This process brings empiricism to life
- how does it do that?

• Where are the feedbacks loops in Scrum?



- What makes Scrum a framework ?
This is about creating a system, system of the framework

• It is made up of elements which give it structure
- framework elements

• Within the framework there is a process
- what is the process?

• This process brings empiricism to life
- how does it do that?

• Where are the feedbacks loops in Scrum?

• Iteration
- it has structure

• Roles, Activities, Deliverables
- Regular feedback

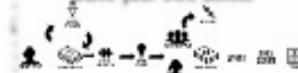
• Sprint planning, review and retrospective



Team 1

Build your own Scrum

Team 2

Build your own Scrum

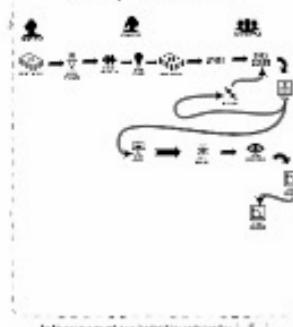
- [] - [] - [] - [] - []



Team 3

Build your own Scrum

Team 4

Build your own Scrum

For the next exercise, use the template of the board you prefer.

miro

Team 1



REFACTORING

Refactoring is about changing the structure of existing code without changing its behavior.
It's changing the code without changing the behavior.



AUTOMATION

Testing is a repetitive process and manual tests are time-consuming.
Automated tests are repeatable and can be run quickly.

UNIT TESTING

Unit testing can be used to verify the behavior of specific code units which could be more difficult to test.



CONTINUOUS INTEGRATION

Testing is done by refactoring in discrete parts.
It's also known as 'continuous integration'.

Team 2



REFACTORING

Refactoring is about changing the structure of existing code without changing its behavior.
It's changing the code without changing the behavior.



AUTOMATION

Testing is a repetitive process and manual tests are time-consuming.
Automated tests are repeatable and can be run quickly.

UNIT TESTING

Unit testing can be used to verify the behavior of specific code units which could be more difficult to test.



CONTINUOUS INTEGRATION

Testing is done by refactoring in discrete parts.
It's also known as 'continuous integration'.

Team 3



REFACTORING

Refactoring is about changing the structure of existing code without changing its behavior.
It's changing the code without changing the behavior.



AUTOMATION

Testing is a repetitive process and manual tests are time-consuming.
Automated tests are repeatable and can be run quickly.

UNIT TESTING

Unit testing can be used to verify the behavior of specific code units which could be more difficult to test.



CONTINUOUS INTEGRATION

Testing is done by refactoring in discrete parts.
It's also known as 'continuous integration'.

Team 4



REFACTORING

Refactoring is about changing the structure of existing code without changing its behavior.
It's changing the code without changing the behavior.



AUTOMATION

Testing is a repetitive process and manual tests are time-consuming.
Automated tests are repeatable and can be run quickly.

UNIT TESTING

Unit testing can be used to verify the behavior of specific code units which could be more difficult to test.



CONTINUOUS INTEGRATION

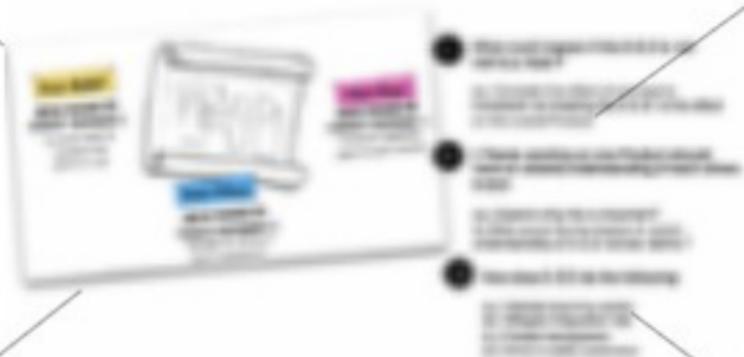
Testing is done by refactoring in discrete parts.
It's also known as 'continuous integration'.

Team 1

Team 2

Team 4

Team 3



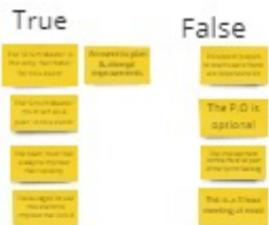
Planning:



Daily Scrum



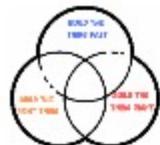
Retrospective:



Sprint Review:



Team 1



Chocolate
Ice Cream

BUILD THE RIGHT THING

BUILD THE THING RIGHT

Build what is
needed for ex.
car
but may not be
perfect, could be
missing safety,
comfort etc.

Build car
right -
quality and
safety

Team 2



BUILD THE RIGHT THING

BUILD THE THING RIGHT

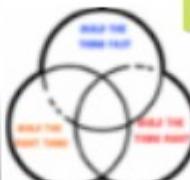
Team 3



BUILD THE RIGHT THING

BUILD THE THING RIGHT

Team 4



BUILD THE RIGHT THING

BUILD THE THING RIGHT

Home
Construction:
Build a house to
blueprint ie one
story, 2 bedroom,
2 bathroom

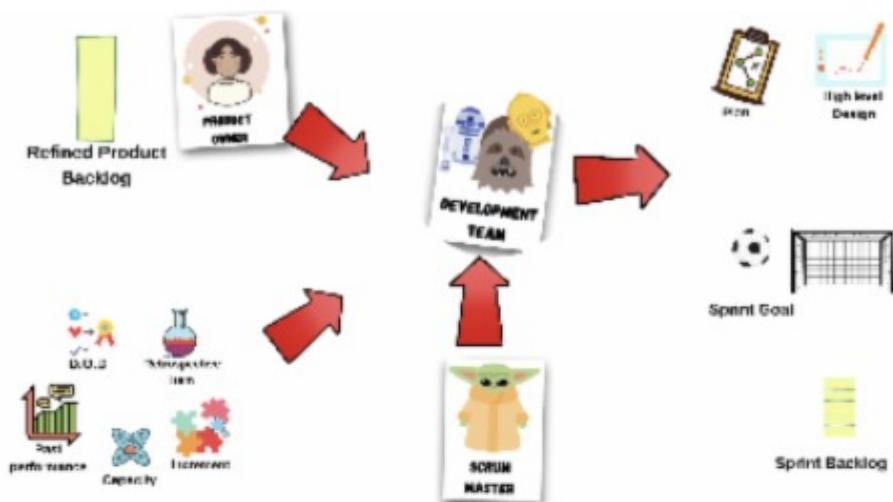
Home Construction:
Each bedroom
needs a window,
closet and door.
Each bathroom has
window, door and
proper ventilation

Team 1

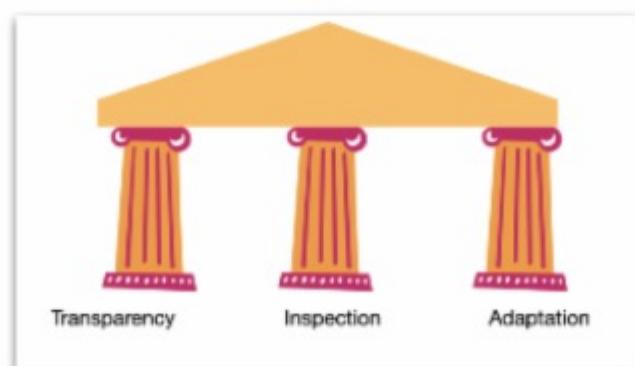
Team 2

Team 3

Team 4



How does this event bring empiricism to life?



Transparency: Sprint backlog provides clear instructions refined product backlog. Goal is also clear and visible

Inspection: High level design of and capacity

Adaptation: Product increment and product retrospective

Daily Scrum



What is the purpose of the Daily Scrum?



What should be discussed?



Who should attend?



What is the timebox for this event?

15 min

Daily

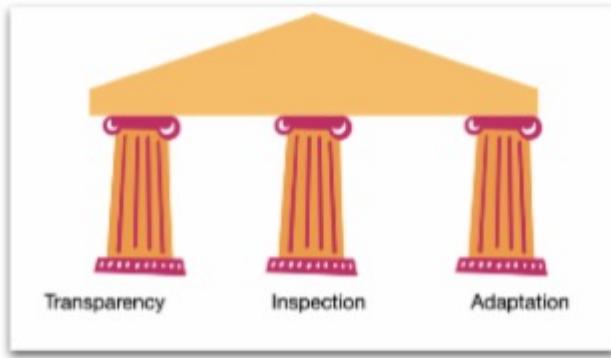
What is the outcome of this event?



What kind of impediments would a ScrumMaster help with?



How does this event bring empiricism to life?



Clarity

Verification/Review/Audit

Culturable

Sprint Retro



Why do we do this event?

- Increase quality of future sprints
- Increase effectiveness of future sprints
- ...
- ...

Who should attend?

- | | |
|------------------|--|
| Scrum Master | |
| Product Owner | |
| Development Team | |

What should the ScrumMaster do here?

Coach

- Encourages the scrum team to improve its process and practices

Facilitation

- Inspect how the previous sprint went

What is presented here?

Agenda

Key experiences for the team

Sprint feedback

Challenges of the team

What is the timebox?

3 hours

What is the outcome of this event?

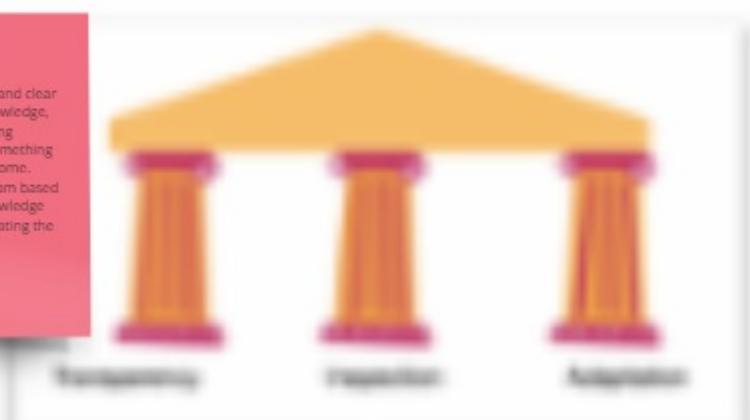
Retro report

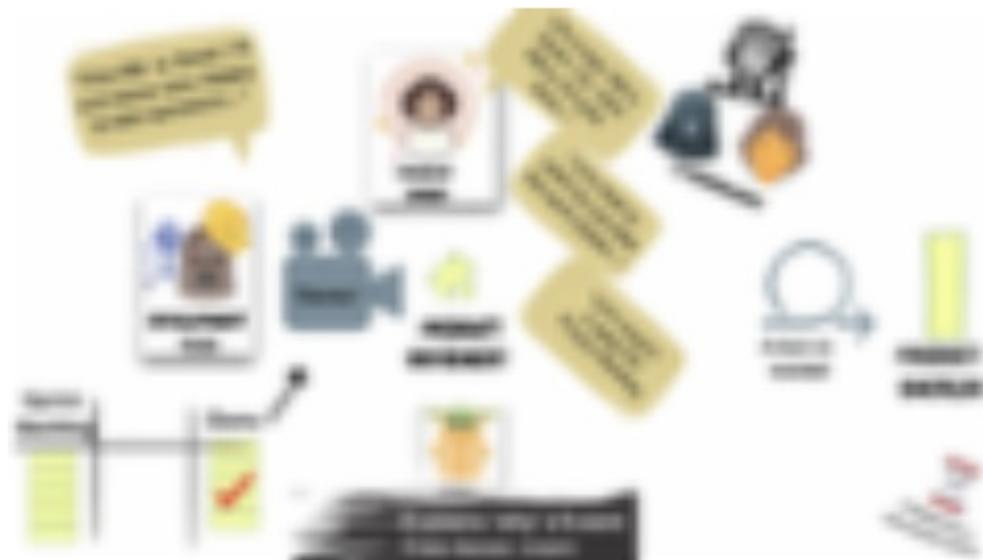
Plan of Action

Improvement Items

How does this event bring empiricism to life?

- 1) Transparency through open and clear communication, access to knowledge, and shared understanding
- 2) Inspection through trying something out and observing the outcome.
- 3) Adaption through aligning team based on the results of the new knowledge from the retrospective and creating the plan of action.





Why do we do the Sprint Review?

- ```

graph TD
 A[Identify the
customer] --> B[Qualify
the customer]
 B --> C[Present
the product]
 C --> D[Product
demonstration]
 D --> E[Measure
the results]
 E --> F[Review the
product]
 F --> A

```

## Who should attend?

- |                      |          |
|----------------------|----------|
| Autorenkennung       | Schemata |
| Operations (Options) |          |
|                      |          |

What should the ScrumMaster do here?

- |                        |                                            |
|------------------------|--------------------------------------------|
| Keep track of time-box | Establish communication                    |
| Enable and coach       | Define roles and set expectations for team |
| Chart review           | Establish communication                    |

Name some of  
activities that  
occur during this  
event?

- ```

graph TD
    Demo[Demo] --> Feedback[Feedback]
    Feedback --> Adoption[Determine future adoption]
    
```

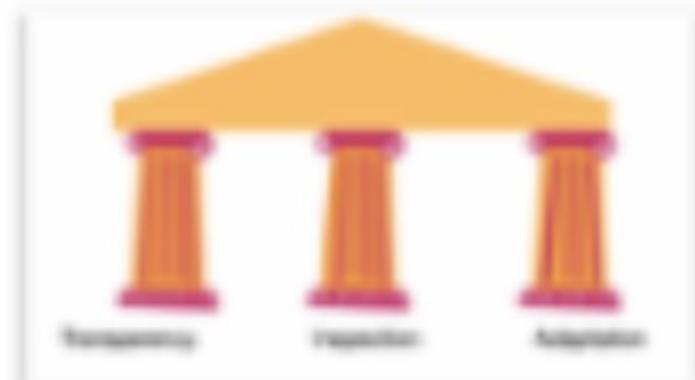
What is the timebox?

- 2 hours
[1 Month(s) earlier]

What are the possible outcomes of this event?

-

How does this event bring empiricism to life?



10

- It gives a sense of transparency because it allows the team/its stakeholders to have a clear picture of what have we done, what is happening while it inspects, allowing us to adapt.

Team 1

Activity:

- For shaping the PBL
- Add details & order

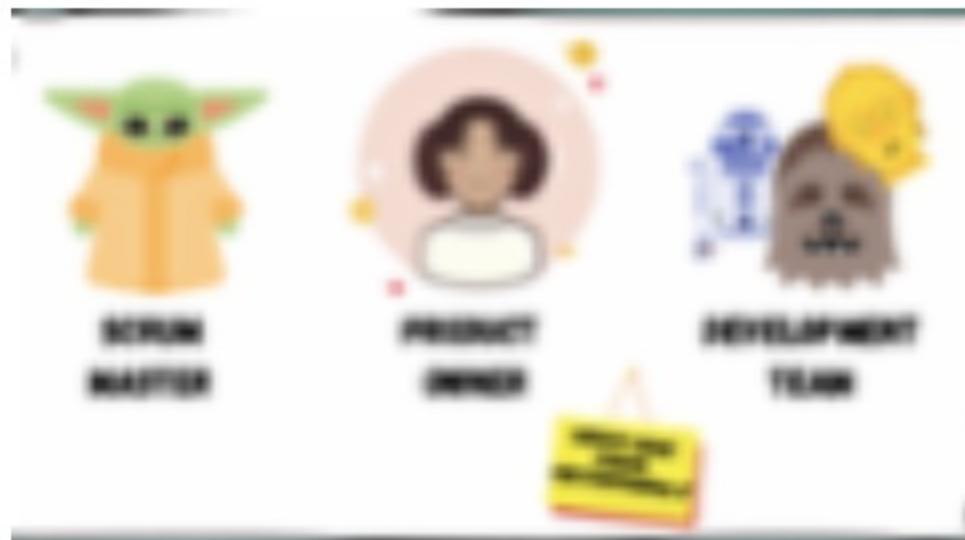
Team 2

Team 3

Team 4



What metaphors come to mind when you think of the below Scrum Roles? Create post it notes for your metaphors below



America
ment |
day work

Team 1
1. Team owner
(product owner)
Coach (Scrum master)
Players (development
team)
2. restaurant example

Team 4
X-Men Metaphor
1. Professor X (Scrum Master)
2. Cyclops (Product Owner)
3. X-Men (Dev Team)

Rowing Metaphor
1. Bosun (Scrum Master)
2. Captain (Product Owner)
3. Crew (Dev Team)

Team 1

THE HEART OF SCRUM

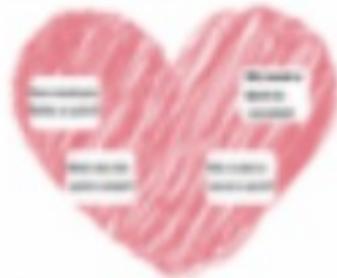


- No change after changing something
- Identify goals, keep the course
- Understand changes & act on it immediately



Team 2

THE HEART OF SCRUM



- No change after changing something
- Identify goals, keep the course
- Understand changes & act on it immediately



Team 4

THE HEART OF SCRUM



- No change after changing something
- Identify goals, keep the course
- Understand changes & act on it immediately



Team 3

THE HEART OF SCRUM



- No change after changing something
- Identify goals, keep the course
- Understand changes & act on it immediately



Team 1

Team 2

Change
Driver

**SIGNS OF A HEALTHY
SPRINT BACKLOG**



**SIGNS OF A HEALTHY
SPRINT BACKLOG**



Team 4

Team 3



**SIGNS OF A HEALTHY
SPRINT BACKLOG**



**SIGNS OF A HEALTHY
SPRINT BACKLOG**



Team 1

Exercise #2

Check out the below example of a fictional goal. As the sprint begins, things change & we now need new options!

THE GOAL		NEW OPTIONS
SPRINT	GOALS	SPRINT
SPRINT 1	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction
SPRINT 2	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction

Use the Post-It notes to create new options
without prototyping the Sprint Goal!

Team 2

Exercise #2

Check out the below example of a fictional goal. As the sprint begins, things change & we now need new options!

THE GOAL		NEW OPTIONS
SPRINT	GOALS	SPRINT
SPRINT 1	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction
SPRINT 2	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction

Use the Post-It notes to create new options
without prototyping the Sprint Goal!

Team 4

Exercise #2

Check out the below example of a fictional goal. As the sprint begins, things change & we now need new options!

THE GOAL		NEW OPTIONS
SPRINT	GOALS	SPRINT
SPRINT 1	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction
SPRINT 2	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction

Use the Post-It notes to create new options
without prototyping the Sprint Goal!

Team 3

Exercise #2

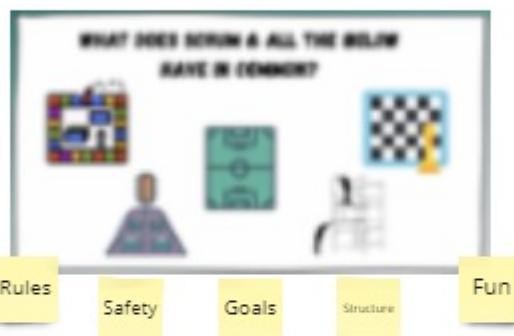
Check out the below example of a fictional goal. As the sprint begins, things change & we now need new options!

THE GOAL		NEW OPTIONS
SPRINT	GOALS	SPRINT
SPRINT 1	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction
SPRINT 2	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction	1. Increase sales 2. Reduce cost 3. Improve customer satisfaction

Use the Post-It notes to create new options
without prototyping the Sprint Goal!

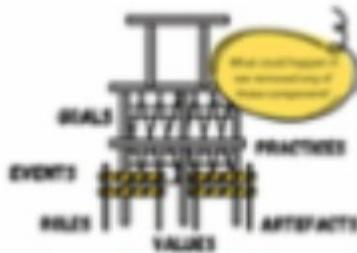
Scrum as a Game Quiz

Think about what it is that allows a game to be played - what stops people from playing football outside of the white lines & what encourages cars to drive on a certain side of the road & take action when they see different coloured lights.



The Half Scrum exercise

Imagine Scrum as a scaffolding create structure for a team to inspect & adapt, to learn, to grow - to be effective! What could happen then if we were to remove elements?



Daily Scrum

Sprint Planning

Refinement

Sprint Review

Sprint Retro

Creating potentially releasable increment

Not sticking to the time-box

Removing the Product Owner role

Blend the PO & SM role

Allowed the Sprint Goal to change

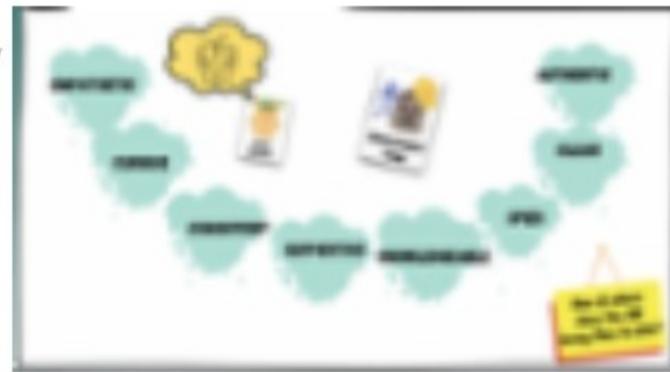
Let anyone enter Retrospective

Demo work which doesn't meet D.O.D

Exercise #1

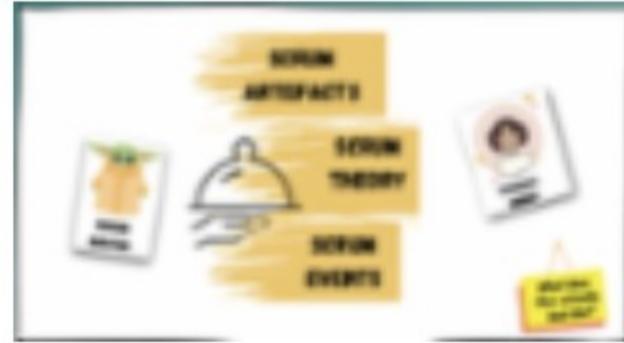
Service to Team

Create cards to describe how the ScrumMaster as a servant leader may bring these ideas to life to serve the team.



Service to Product Owner

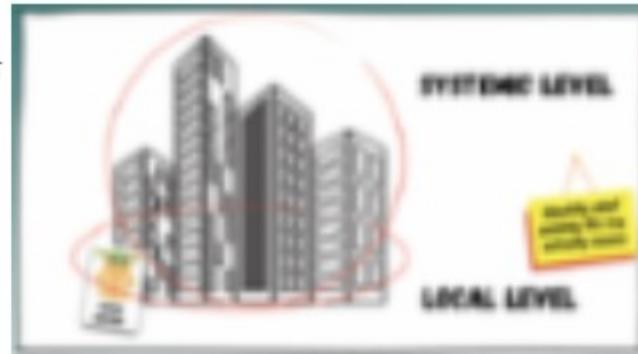
Create cards to describe how the ScrumMaster serve the Product Owner. Consider using the hints!



Exercise #3

Service to Organisation

Create cards to describe how the ScrumMaster serve the organisation.



Wrap-up

Either drawing from personal experience or through creating a fictional scenario, explore the following question: When you tried to serve an organisation as a ScrumMaster what changes did the organisation need to take to give Scrum a chance of success? What organisational challenges got in the way which stopped or could stop Scrum from working?

◦

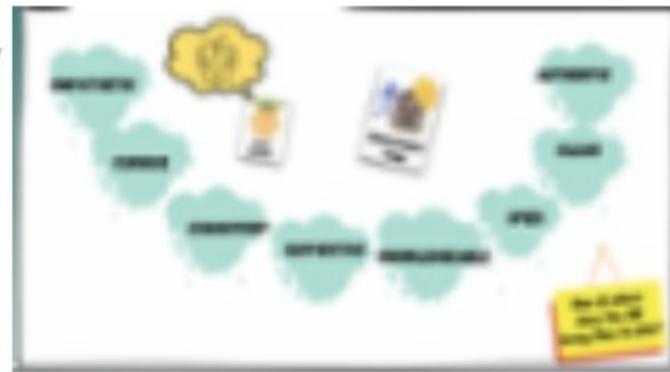
Organisational changes needed to help Scrum work

Organisational blockers getting in the way of Scrum working

Exercise #1

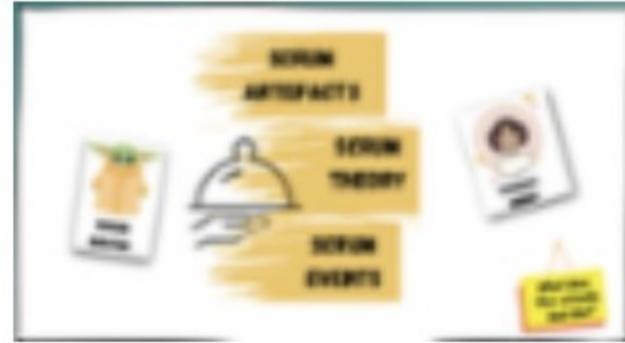
Service to Team

Create cards to describe how the ScrumMaster as a servant leader may bring these ideas to life to serve the team.



Service to Product Owner

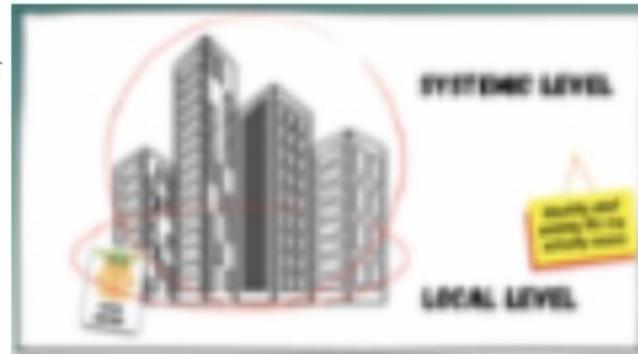
Create cards to describe how the ScrumMaster serve the Product Owner. Consider using the hints!



Exercise #3

Service to Organisation

Create cards to describe how the ScrumMaster serve the organisation.



Wrap-up

Either drawing from personal experience or through creating a fictional scenario, explore the following question: When you tried to serve an organisation as a ScrumMaster what changes did the organisation need to take to give Scrum a chance of success? What organisational challenges got in the way which stopped or could stop Scrum from working?

◦

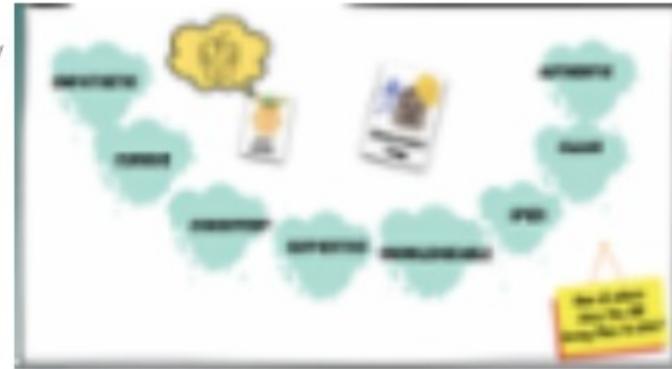
Organisational changes needed to help Scrum work

Organisational blockers getting in the way of Scrum working

Exercise #1

Service to Team

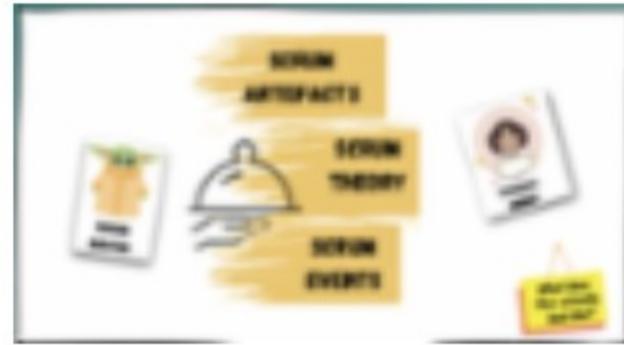
Create cards to describe how the ScrumMaster as a servant leader may bring these ideas to life to serve the team.



Exercise #2

Service to Product Owner

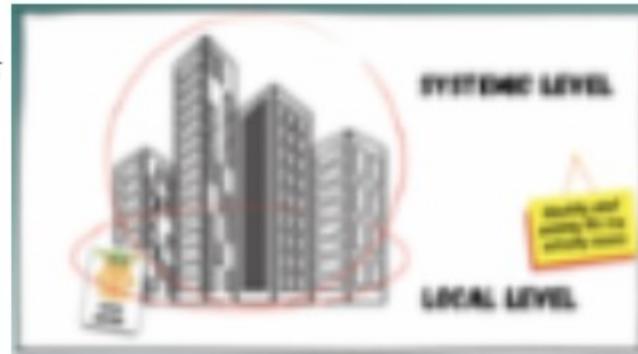
Create cards to describe how the ScrumMaster serve the Product Owner. Consider using the hints!



Exercise #3

Service to Organisation

Create cards to describe how the ScrumMaster serve the organisation.



Wrap-up

Either drawing from personal experience or through creating a fictional scenario, explore the following question: When you tried to serve an organisation as a ScrumMaster what changes did the organisation need to take to give Scrum a chance of success? What organisational challenges got in the way which stopped or could stop Scrum from working?

◦

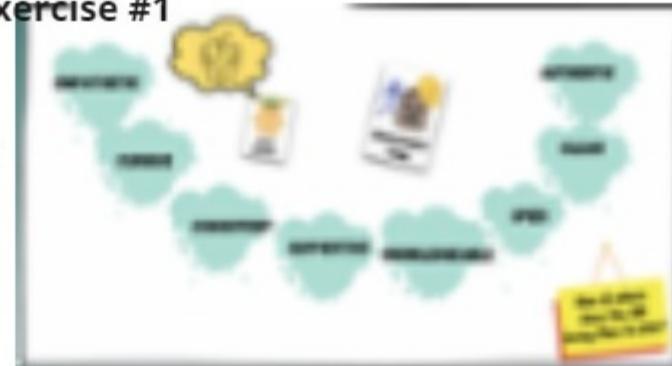
Organisational changes needed to help Scrum work

Organisational blockers getting in the way of Scrum working

Exercise #1

Service to Team

Create cards to describe how the ScrumMaster as a servant leader may bring these ideas to life to serve the team.



Exercise #2

Service to Product Owner

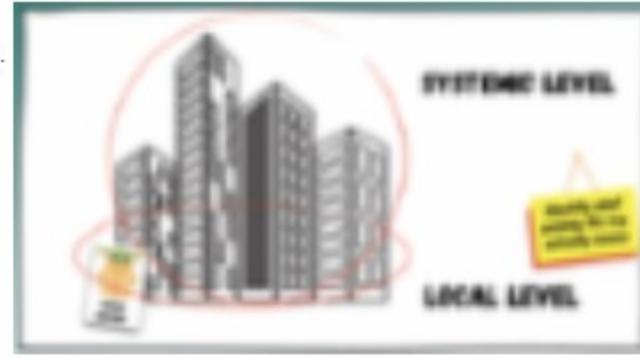
Create cards to describe how the ScrumMaster serve the Product Owner. Consider using the hints!



Exercise #3

Service to Organisation

Create cards to describe how the ScrumMaster serve the organisation.



Wrap-up

Either drawing from personal experience or through creating a fictional scenario, explore the following question: When you tried to serve an organisation as a ScrumMaster what changes did the organisation need to take to give Scrum a chance of success? What organisational challenges got in the way which stopped or could stop Scrum from working?

◦

Organisational changes needed to help Scrum work

Organisational blockers getting in the way of Scrum working

Team 1

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.

1	3	8
2	5	13



BANANA



APPLE



KIWI



MANGO



COCONUT

Team 2

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.

1	3	8
2	5	13



BANANA



APPLE



KIWI



MANGO



COCONUT

Team 3

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.

1	3	8
2	5	13



BANANA



MANGO



APPLE



KIWI



PINEAPPLE



COCONUT

Team 4

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.

1	3	8
2	5	13



BANANA



KIWI



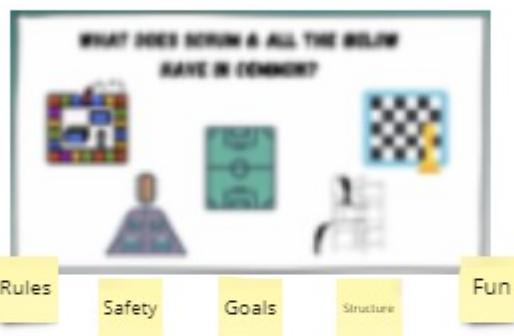
APPLE



COCONUT

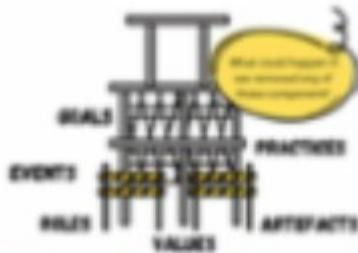
Scrum as a Game Quiz

Think about what it is that allows a game to be played - what stops people from playing football outside of the white lines & what encourages cars to drive on a certain side of the road & take action when they see different coloured lights.



The Half Scrum exercise

Imagine Scrum as a scaffolding create structure for a team to inspect & adapt, to learn, to grow - to be effective! What could happen then if we were to remove elements?



Daily Scrum

Sprint Planning

Refinement

Sprint Review

Sprint Retro

Creating potentially releasable increment

Not sticking to the time-box

Removing the Product Owner role

Blend the PO & SM role

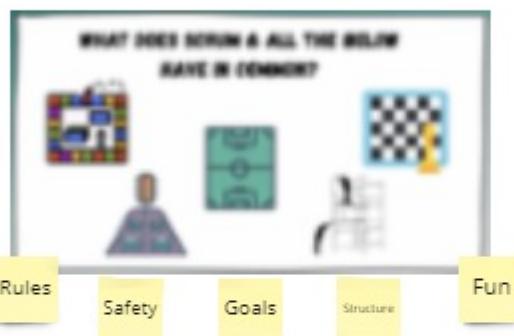
Allowed the Sprint Goal to change

Let anyone enter Retrospective

Demo work which doesn't meet D.O.D

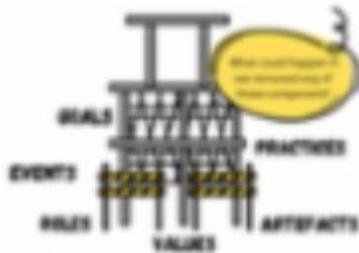
Scrum as a Game Quiz

Think about what it is that allows a game to be played - what stops people from playing football outside of the white lines & what encourages cars to drive on a certain side of the road & take action when they see different coloured lights.



The Half Scrum exercise

Imagine Scrum as a scaffolding create structure for a team to inspect & adapt, to learn, to grow - to be effective! What could happen then if we were to remove elements?



Daily Scrum

Sprint Planning

Refinement

Sprint Review

Sprint Retro

Creating potentially releasable increment

Not sticking to the time-box

Removing the Product Owner role

Blend the PO & SM role

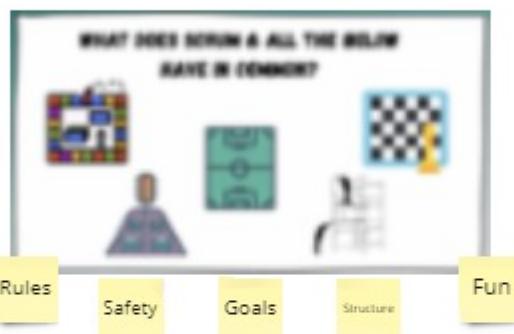
Allowed the Sprint Goal to change

Let anyone enter Retrospective

Demo work which doesn't meet D.O.D

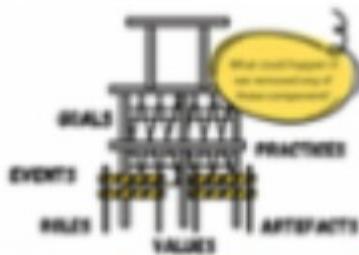
Scrum as a Game Quiz

Think about what it is that allows a game to be played - what stops people from playing football outside of the white lines & what encourages cars to drive on a certain side of the road & take action when they see different coloured lights.



The Half Scrum exercise

Imagine Scrum as a scaffolding create structure for a team to inspect & adapt, to learn, to grow - to be effective! What could happen then if we were to remove elements?



Creating potentially releasable increment

Not sticking to the time-box

Removing the Product Owner role

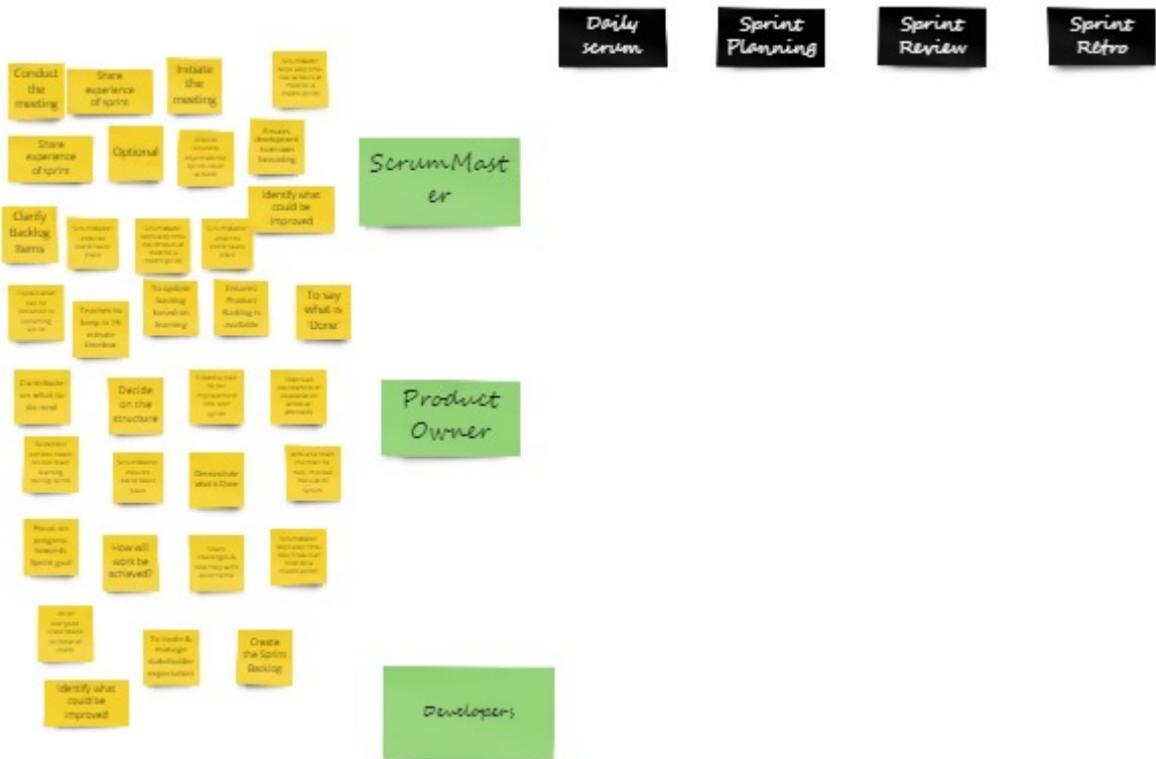
Blend the PO & SM role

Allowed the Sprint Goal to change

Let anyone enter Retrospective

Demo work which doesn't meet D.O.D

Roles & Event Mapping



The ScrumMaster wears more than one hat. Different situations benefit from different hats.

Ensures that the team bring in the retrospective item into new sprint

Teaching how to use empirical data to make decisions

Showing a team how to write a PBI

Help the organisation see that certain incentives can affect Scrum adoption

Showing a team how to give a swift daily scrum update

Handling conflict between two developers to see if they can resolve the underlying issue

Showing a player how to push back against external requests, whilst in sprint

Reflecting back the fact that teams always have hang over of work

Teaching the P.O how to focus on the 'what' & 'why'

Help keep the timebox for the events

Helps the team remember to focus on sprint backlog at the Daily Scrum

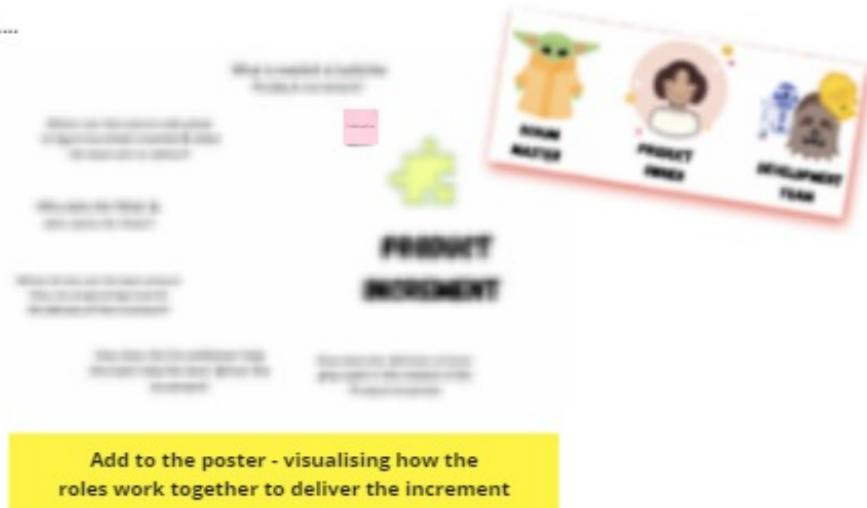
Educating the team on the purpose of the events



Using the definitions above sort the yellow cards into the appropriate hat

How do the Scrum roles come together to deliver value?

Check out the hints below....



Refresh yourself with the Agile principles & values...

12 Principles of Agile Software Development

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Individuals and interactions over processes and tools
6. Working software over comprehensive documentation
7. Customer collaboration over contract negotiation
8. Responding to change over following a plan
9. Individuals and interactions over processes and tools
10. Working software over comprehensive documentation
11. Customer collaboration over contract negotiation
12. Responding to change over following a plan



Individuals & Interactions



Business & Technology



Marketing & Software



Customer & Stakeholder Communication



Customer Collaboration



Continuous Integration



Responding to Change



Following a Plan

Think about how often we deliver in Scrum, the roles which create interaction between the business & technology, how the developers are

Create cards demonstrating how Scrum brings Agile to life?

Handling challenges

Check out the below speech bubbles demonstrating challenges which pop up for a ScrumMaster. Either using these examples or your own identify 5 challenges from the Scrum team which go against Scrum rules & then provide ideas on how to resolve such challenges.



The ScrumMaster wears more than one hat. Different situations benefit from different hats.

Ensures that the team bring in the retrospective items into new sprint

Teaching how to use empirical data to make decisions

Showing a team how to write a PBI

Help the organization see that certain incentives can affect Scrum adoption

Showing a team how to give a swift daily scrum update

Handling conflict between two developers to see if they can resolve the underlying issue

Showing a player how to push back against external requests, whilst in sprint

Reflecting back the fact that teams always have hang over of work

Teaching the P.O how to focus on the 'what' & 'why'

Help keep the timebox for the events

Helps the team remember to focus on sprint backlog at the Daily Scrum

Educating the team on the purpose of the events



Using the definitions above sort the yellow cards into the appropriate hat

The ScrumMaster wears more than one hat. Different situations benefit from different hats.

Ensures that the team bring in the retrospective item into new sprint

Teaching how to use empirical data to make decisions

Showing a team how to write a PBI

Help the organization see that certain incentives can affect Scrum adoption

Showing a team how to give a swift daily scrum update

Handling conflict between two developers to see if they can resolve the underlying issue

Showing a player how to push back against external requests, whilst in sprint

Reflecting back the fact that teams always have hang over of work

Teaching the P.O how to focus on the 'what' & 'why'

Help keep the timebox for the events

Helps the team remember to focus on sprint backlog at the Daily Scrum

Educating the team on the purpose of the events



Using the definitions above sort the yellow cards into the appropriate hat

The ScrumMaster wears more than one hat. Different situations benefit from different hats.

Ensures that the team bring in the retrospective items into new sprint

Teaching how to use empirical data to make decisions

Showing a team how to write a PBI

Help the organization see that certain incentives can affect Scrum adoption

Showing a team how to give a swift daily scrum update

Handling conflict between two developers to see if they can resolve the underlying issue

Showing a player how to push back against external requests, whilst in sprint

Reflecting back the fact that teams always have hang over of work

Teaching the P.O how to focus on the 'what' & 'why'

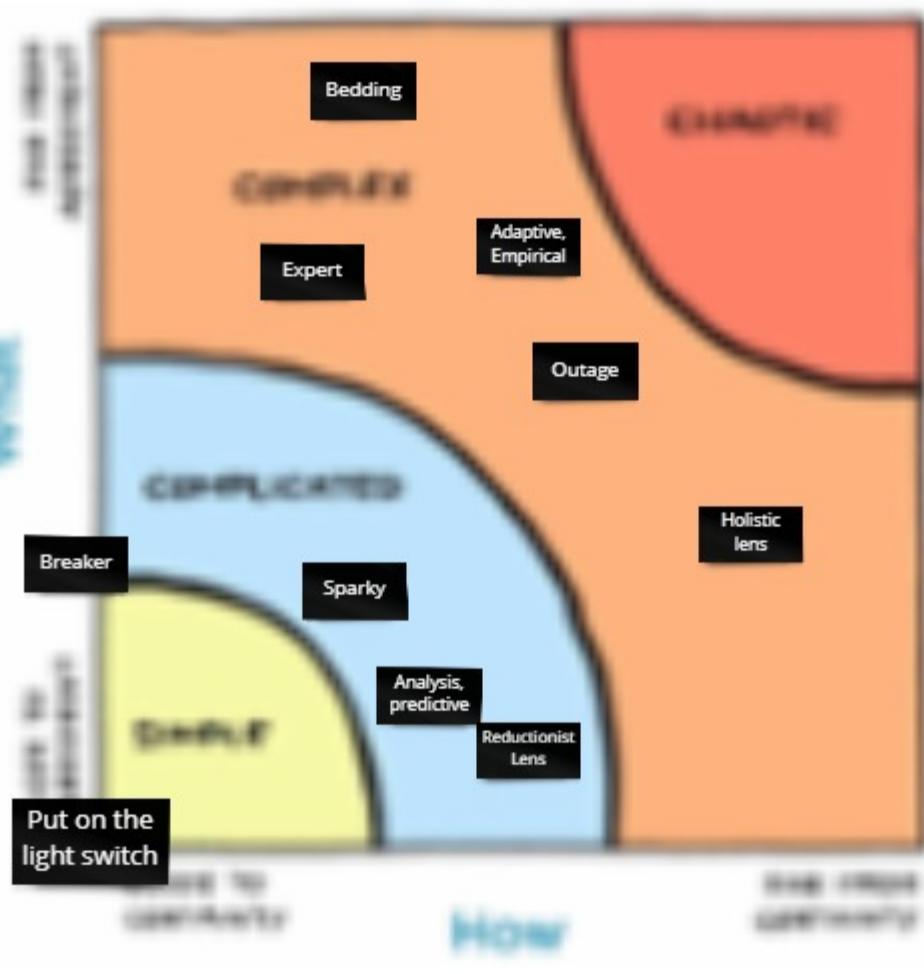
Help keep the timebox for the events

Helps the team remember to focus on sprint backlog at the Daily Scrum

Educating the team on the purpose of the events

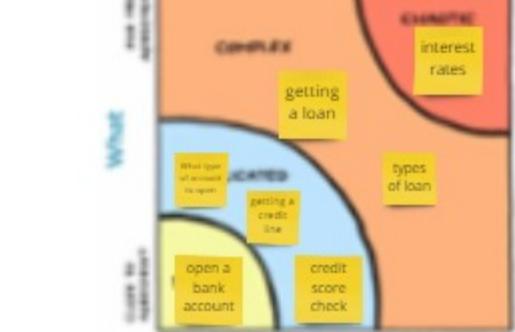
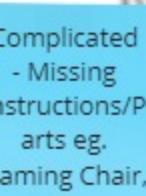
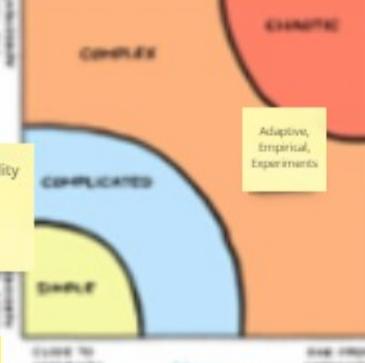


Using the definitions above sort the yellow cards into the appropriate hat

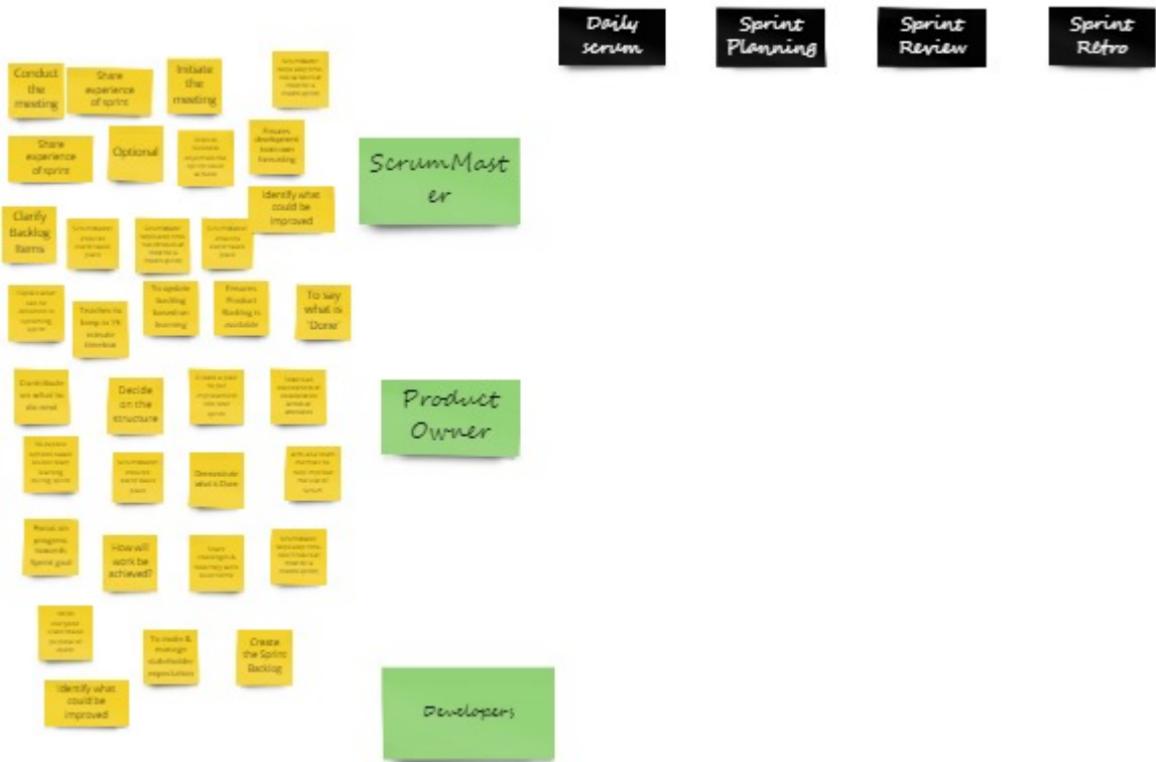


- Using the Stacey Matrix below, generate requirements & place them on the appropriate level of complexity
 - These requirements can be work or non-work related
 - Finally create a "Waterfall" & "Agile" post-it and place it to where you believe it belongs on the below model
 - Bonus questions: Where does innovation live on this model & why?

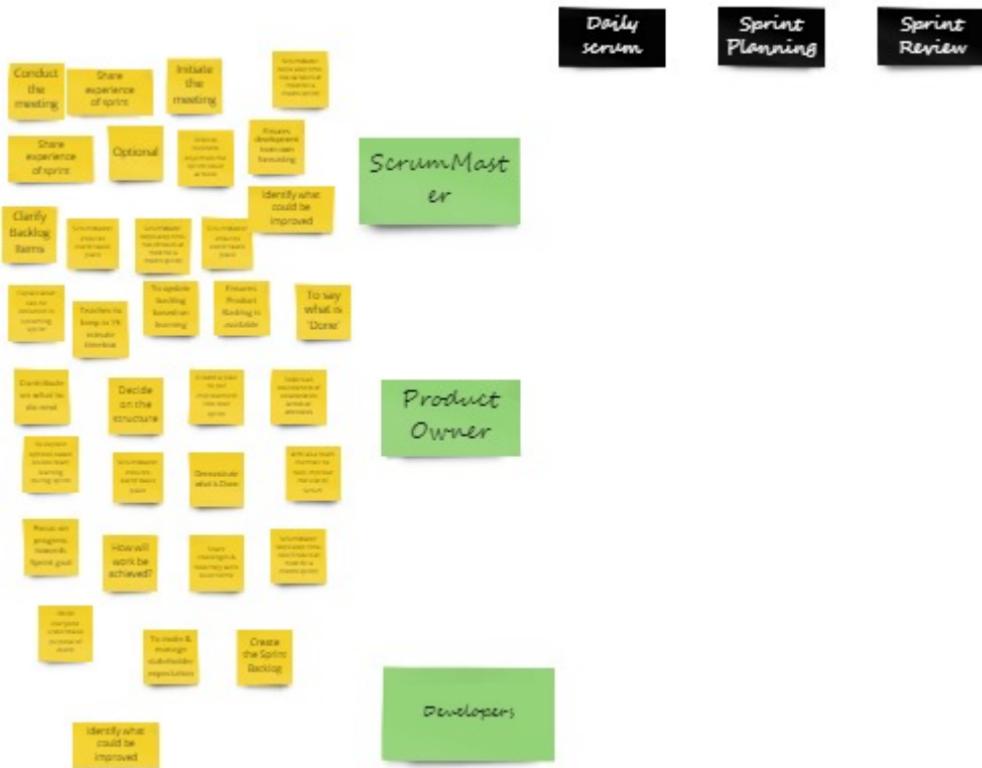
type of requirements are you working with?



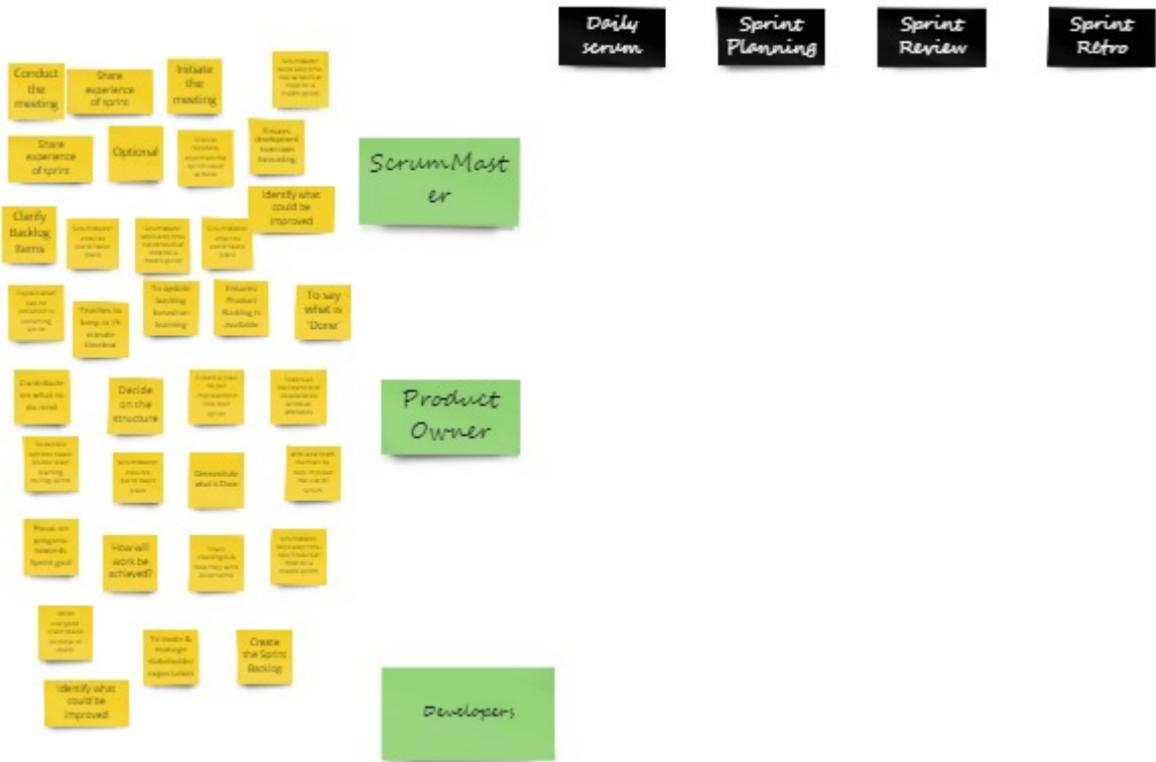
Roles & Event Mapping



Roles & Event Mapping



Roles & Event Mapping



Refresh yourself with the Agile principles & values...

12 Principles of Agile Software Development

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Individuals and interactions over processes and tools
6. Working software over comprehensive documentation
7. Customer collaboration over contract negotiation
8. Responding to change over following a plan
9. Individuals and interactions over processes and tools
10. Working software over comprehensive documentation
11. Customer collaboration over contract negotiation
12. Responding to change over following a plan



Individuals & Interactions



Process & Tools



Working software



Comprehensive documentation



Customer collaboration



Responding to change



→ Publishing a plan

Think about how often we deliver in Scrum, the roles which create interaction between the business & technology, how the developers are

Create cards demonstrating how Scrum brings Agile to life?

Refresh yourself with the Agile principles & values...

12 Principles of Agile Software Development

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Individuals and interactions over processes and tools
6. Working software over comprehensive documentation
7. Customer collaboration over contract negotiation
8. Responding to change over following a plan
9. Individuals and interactions over processes and tools
10. Working software over comprehensive documentation
11. Customer collaboration over contract negotiation
12. Responding to change over following a plan



Individuals & Interactions



Business & Technology



Marketing & Software



Customer & Stakeholder Communication



Customer Collaboration



Continuous Integration



Responding to Change



Following a Plan

Think about how often we deliver in Scrum, the

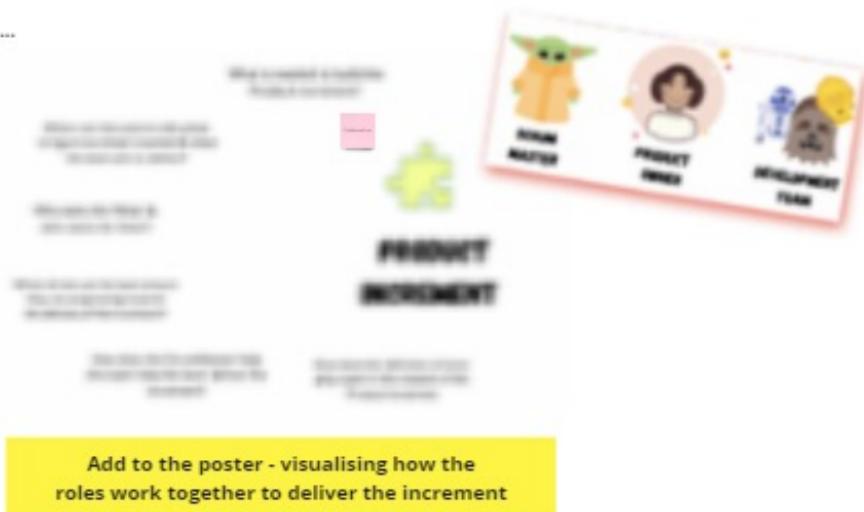
roles which create interaction between the

business & technology, how the developers are

Create cards demonstrating how Scrum brings Agile to life?

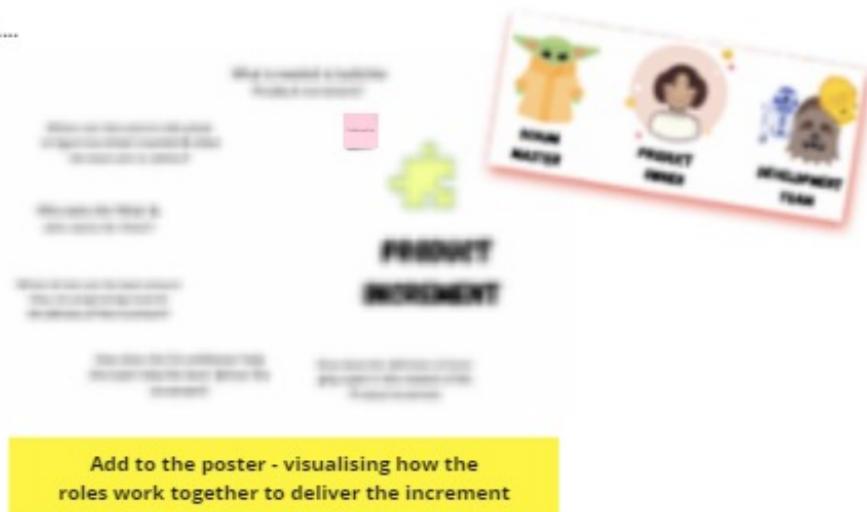
How do the Scrum roles come together to deliver value?

Check out the hints below....



How do the Scrum roles come together to deliver value?

Check out the hints below....



How do the Scrum roles come together to deliver value?

Check out the hints below....



Handling challenges

Check out the below speech bubbles demonstrating challenges which pop up for a ScrumMaster. Either using these examples or your own identify 5 challenges from the Scrum team which go against Scrum rules & then provide ideas on how to resolve such challenges.



Handling challenges

Check out the below speech bubbles demonstrating challenges which pop up for a ScrumMaster. Either using these examples or your own identify 5 challenges from the Scrum team which go against Scrum rules & then provide ideas on how to resolve such challenges.



Handling challenges

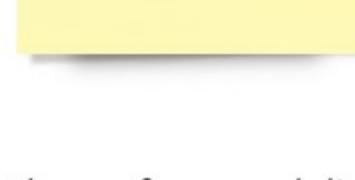
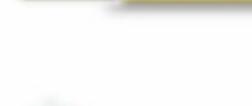
Check out the below speech bubbles demonstrating challenges which pop up for a ScrumMaster. Either using these examples or your own identify 5 challenges from the Scrum team which go against Scrum rules & then provide ideas on how to resolve such challenges.



Refresh yourself with the Agile principles & values...

12 Principles of Agile Software Development

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan
5. Individuals and interactions over processes and tools
6. Working software over comprehensive documentation
7. Customer collaboration over contract negotiation
8. Responding to change over following a plan
9. Individuals and interactions over processes and tools
10. Working software over comprehensive documentation
11. Customer collaboration over contract negotiation
12. Responding to change over following a plan



Think about how often we deliver in Scrum, the roles which create interaction between the business & technology, how the developers are

Create cards demonstrating how Scrum brings Agile to life?

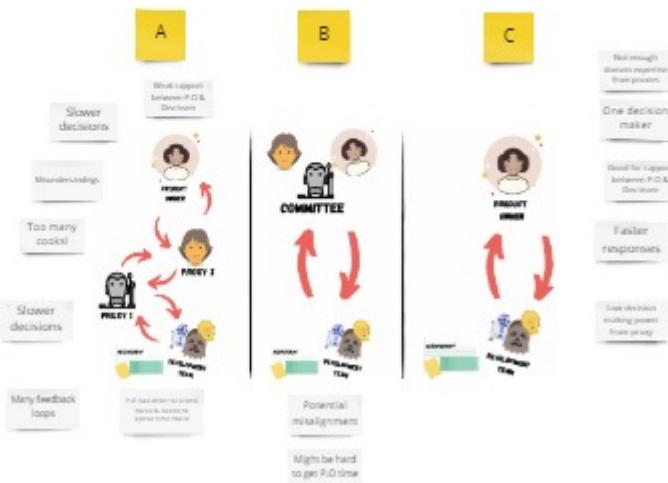
Exercise 1: Rights, responsibilities & characteristics



Zoom in: Product Owner as one role

Exercise 2:

Observe the below scenarios for Product Owner & Dev team set up. What are the pros & cons of each scenario? Create cards for each answer.



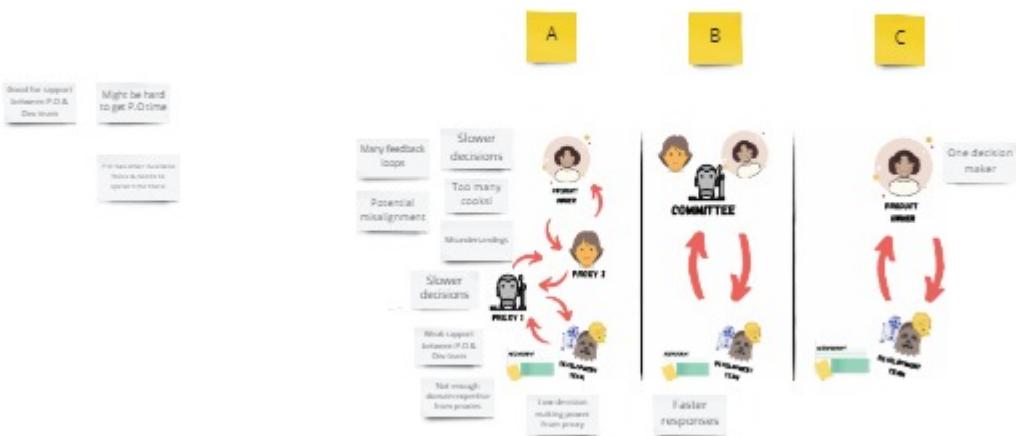
Exercise 1: Rights, responsibilities & characteristics



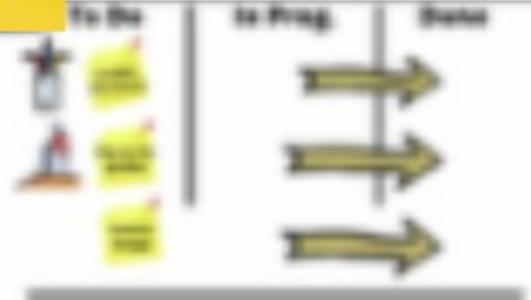
Zoom in: Product Owner as one role

Exercise 2:

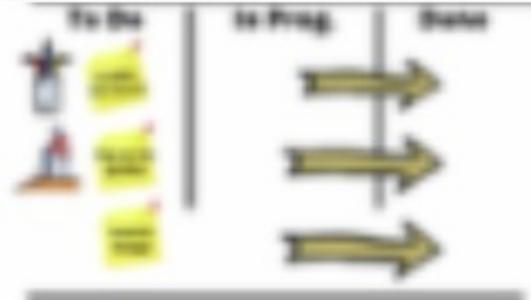
Observe the below scenarios for Product Owner & Dev team set up. What are the pros & cons of each scenario? Create cards for each answer.



Team 1



Team 4



Team 1



Team 2



Team 4

1 2 3 5 8 13 21



Team 3



Team 1

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.



What factors can impact our estimation?

Answers
various

Do we all peel the fruit the same way?

No

Do we all use the same equipment or?

No

Are we measuring that there is only one variety of coconut or mango?

No

Team 2

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.



What factors can impact our estimation?

Answers
various

Do we all peel the fruit the same way?

No

Do we all use the same equipment or?

No

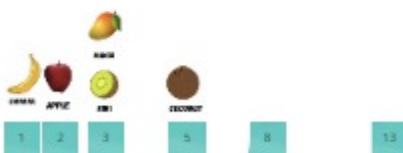
Are we measuring that there is only one variety of coconut or mango?

No

Team 4

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.



What factors can impact our estimation?

Answers
various

Do we all peel the fruit the same way?

No

Do we all use the same equipment or?

No

Are we measuring that there is only one variety of coconut or mango?

No

Team 3

Estimation game

What is the effort to peel these fruit? Using relative sizing move the fruit to the appropriate number.



What factors can impact our estimation?

Answers
various

Do we all peel the fruit the same way?

Yes or mostly

Do we all use the same equipment or?

No

Are we measuring that there is only one variety of coconut or mango?

We did

tools
required!
expertise
time

Team 1



Team 2



Team 4



Team 3



Planning:

True

- The ScrumMaster is the only facilitator in the room
- Product Backlog items are prioritized by the Product Owner
- The ScrumMaster and Product Owner are the only ones in the room
- Product Backlog items are prioritized by the team
- Product Backlog items are prioritized by the Product Owner and the team

False

- The ScrumMaster is the only facilitator in the room
- Product Backlog items are prioritized by the Product Owner
- The ScrumMaster and Product Owner are the only ones in the room
- Product Backlog items are prioritized by the team
- Product Backlog items are prioritized by the Product Owner and the team

Daily Scrum

True

- The ScrumMaster facilitates the Daily Scrum
- Attendees of the Daily Scrum are the team and the Product Owner
- The ScrumMaster facilitates the Daily Scrum
- The ScrumMaster facilitates the Daily Scrum
- The ScrumMaster facilitates the Daily Scrum

False

- The ScrumMaster facilitates the Daily Scrum
- Attendees of the Daily Scrum are the team and the Product Owner
- The ScrumMaster facilitates the Daily Scrum
- The ScrumMaster facilitates the Daily Scrum
- The ScrumMaster facilitates the Daily Scrum

Retrospective:

True

- The ScrumMaster is the only facilitator in the room
- The Product Owner has a role in the Retrospective
- Retrospective is optional

False

- The ScrumMaster is the only facilitator in the room
- The Product Owner has a role in the Retrospective
- Retrospective is optional
- The Retrospective serves the team
- The Retrospective serves the Product Owner
- The P.O. is optional
- This is a 1-hour meeting of most

Sprint Review:

True

- The P.D. is optional
- The ScrumMaster is the only facilitator in the room
- Product Owner is the only facilitator in the room
- The ScrumMaster facilitates the Sprint Review
- The Product Owner facilitates the Sprint Review

False

- This is a 1-hour meeting of most
- The ScrumMaster is the only facilitator in the room
- The Product Owner is the only facilitator in the room
- The ScrumMaster facilitates the Sprint Review
- The Product Owner facilitates the Sprint Review

Planning:

True

False

- The Product Owner is the only person involved in planning.
- The Product Owner is the only person involved in backlog refinement.
- The Product Owner is the only person involved in feature prioritization.
- The Product Owner is the only person involved in feature planning.

- The Product Owner is not the only person involved in backlog refinement.
- The Product Owner is not the only person involved in feature prioritization.
- The Product Owner is not the only person involved in feature planning.
- The Product Owner is not the only person involved in feature development.

Daily Scrum:

True

False

- The Product Owner is the only person involved in the Daily Scrum.
- The Product Owner is the only person involved in the Daily Scrum.
- The Product Owner is the only person involved in the Daily Scrum.
- The Product Owner is the only person involved in the Daily Scrum.

- The Product Owner is not the only person involved in the Daily Scrum.
- The Product Owner is not the only person involved in the Daily Scrum.
- The Product Owner is not the only person involved in the Daily Scrum.
- The Product Owner is not the only person involved in the Daily Scrum.

Retrospective:

True

False

- The retrospective is optional.
- The retrospective is mandatory.
- The retrospective is not a formal meeting.
- The retrospective is a formal meeting.
- The retrospective is not a formal meeting.
- The retrospective is a formal meeting.
- The retrospective is not a formal meeting.
- The retrospective is a formal meeting.
- There is no formal meeting at the retrospective.
- There is a formal meeting at the retrospective.

Sprint Review:

True

False

- The Product Owner is the only person involved in the Sprint Review.
- The Product Owner is the only person involved in the Sprint Review.
- The Product Owner is the only person involved in the Sprint Review.
- The Product Owner is the only person involved in the Sprint Review.

- The Product Owner is not the only person involved in the Sprint Review.
- The Product Owner is not the only person involved in the Sprint Review.
- The Product Owner is not the only person involved in the Sprint Review.
- The Product Owner is not the only person involved in the Sprint Review.

Planning:

True

- The Product Owner is the sole authority for prioritizing work
- The Product Owner is the sole authority for accepting work
- The Product Owner is the sole authority for changing requirements
- The Product Owner is the sole authority for changing scope
- The Product Owner is the sole authority for changing delivery date

False

- The Product Owner is the sole authority for prioritizing work
- The Product Owner is the sole authority for accepting work
- The Product Owner is the sole authority for changing requirements
- The Product Owner is the sole authority for changing scope
- The Product Owner is the sole authority for changing delivery date

Daily Scrum

True

- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting

False

- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting
- The Daily Scrum is a 15-minute stand-up meeting

Retrospective:

True

- This is a 1-hour meeting all round
- Accomplished 2-3 things at a time
- The team can make a plan for next iteration
- The team can make a plan for next iteration
- The team can make a plan for next iteration

False

- The P.O. is optional
- The ScrumMaster is optional
- The team can make a plan for next iteration
- The team can make a plan for next iteration
- The team can make a plan for next iteration

Sprint Review:

True

- The ScrumMaster is the only one in charge
- The Product Owner is the only one in charge
- The Product Owner is the only one in charge
- The Product Owner is the only one in charge
- The Product Owner is the only one in charge
- The Product Owner is the only one in charge

False

- The ScrumMaster is the only one in charge
- The Product Owner is the only one in charge
- The Product Owner is the only one in charge
- The Product Owner is the only one in charge
- The Product Owner is the only one in charge
- The Product Owner is the only one in charge