

Agile Requirements Management with User Stories

Fran O'Hara

fran.ohara@inspireqs.ie

Inspire Quality Services

www.inspireqs.ie – Agile Training & Coaching

In association with AgileInnovation

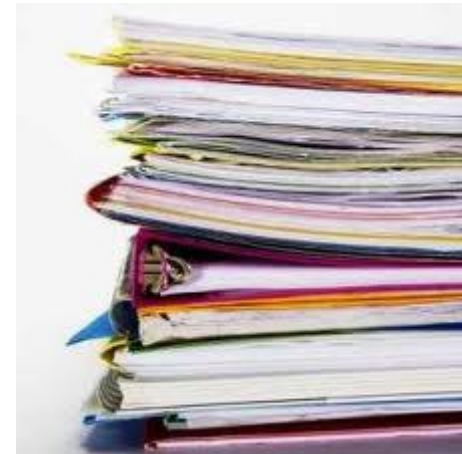
Agenda



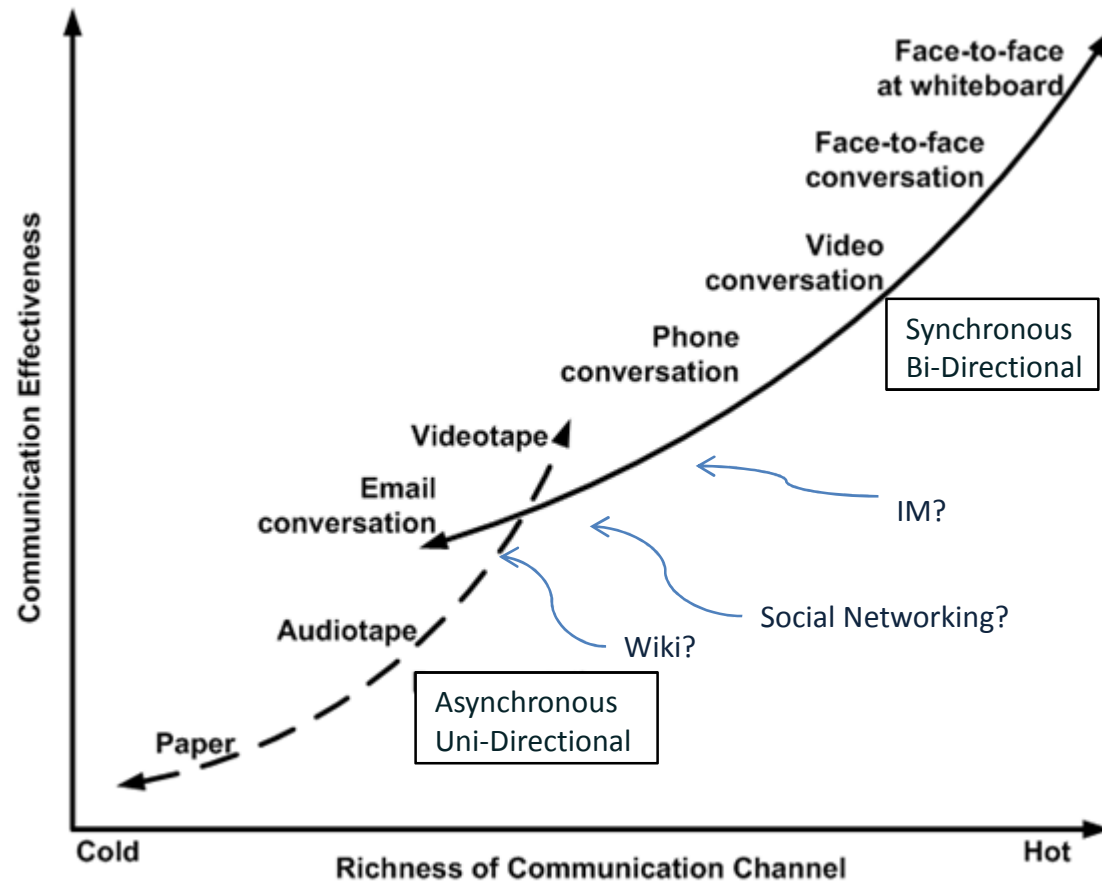
- Why user stories?
- Backlogs, Epics, Stories, Acceptance Criteria...
- Estimation and Planning with Stories

What's Wrong with 'Requirements'?

- Mandatory, Fixed, Hard to Change
- Feature Centric rather than Value Centric
- Specify the What, not the Why
- Critical bits hidden in the detail
- Expensive



Communication Modes



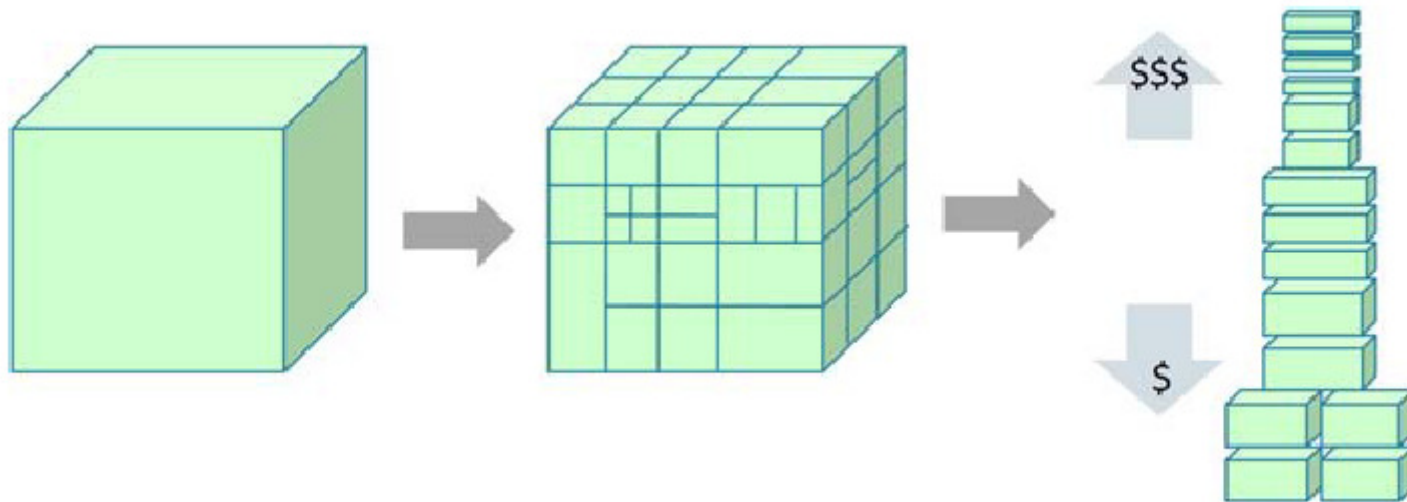
Copyright 2002-2005 Scott W. Ambler
Original Diagram Copyright 2002 Alistair Cockburn



<http://www.agilemodeling.com/essays/communication.htm>

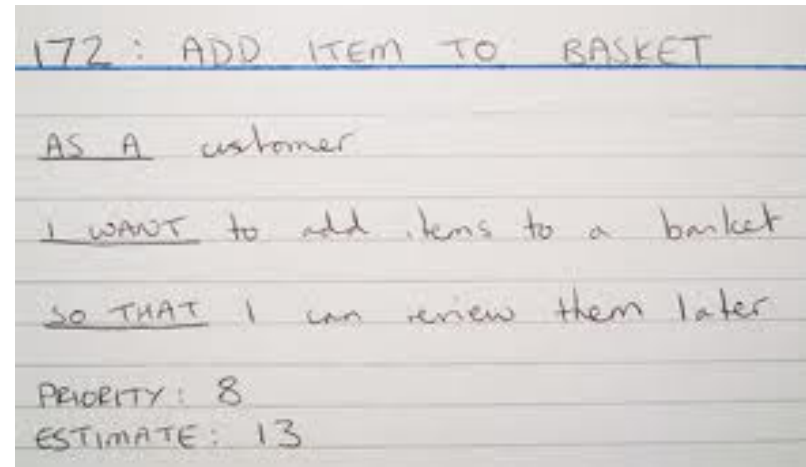
© 2012 Inspire Quality Services / AgileInnovation

From 'All at Once' to 'A Little at a Time'



What is a User Story?

- A Written Card
 - For Planning
 - A Promise to Talk
 - Spec by Example
 - Acceptance Criteria



- It Defines
 - The Actor/User/Persona
 - The Goal/Action/Task
 - The Benefit/Value
- What it's not:
 - A Use Case
 - Requirements Document
 - Scenarios

Why User Stories?



- User Centric – what's important to your customer
- Story – The Power of Narrative
 - We pay much more attention to stories than facts
 - Drives generation of tacit knowledge
 - A story paints a picture, and a picture tells a thousand words
- Focus on the benefit, the value, what's important
 - Define Acceptance Criteria BEFORE we implement
- Supports 'pull' of information as its needed
 - Iterative development

Specification

The product shall have a gasoline-powered engine.

The product shall

- a. The product shall have a spare tire mounted to each wheel.

The product shall have a spare tire mounted to each wheel.

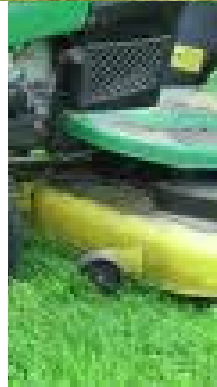
- a. The product shall have a horn on the steering wheel.

The product shall have a steel body.



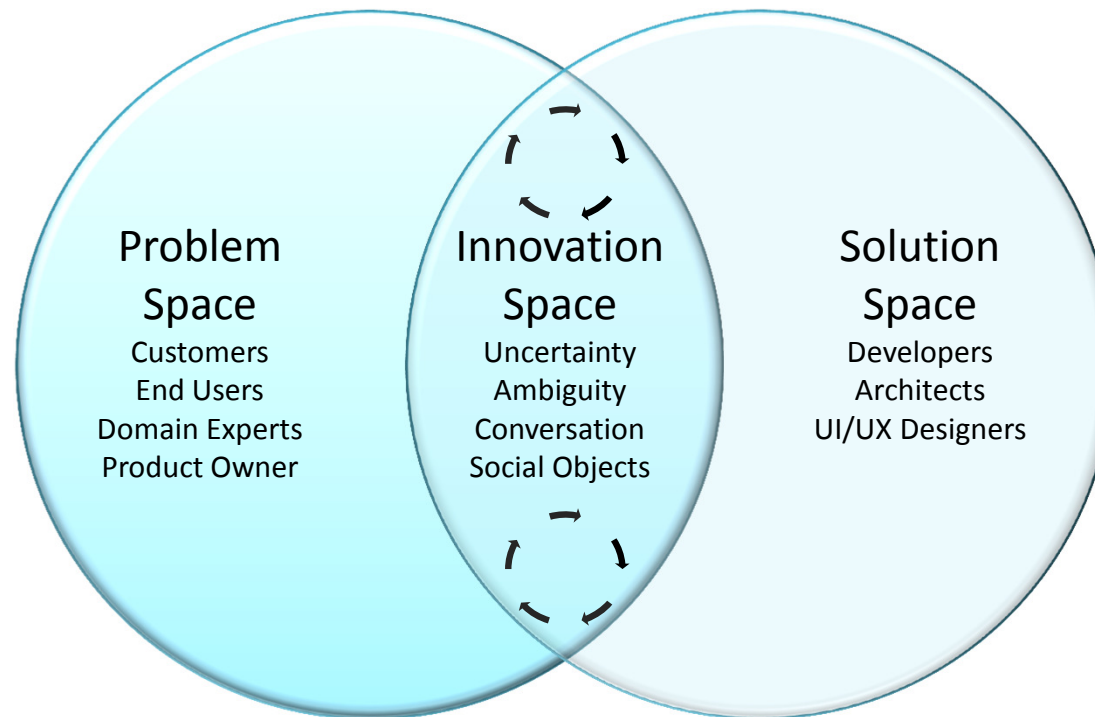
User Story

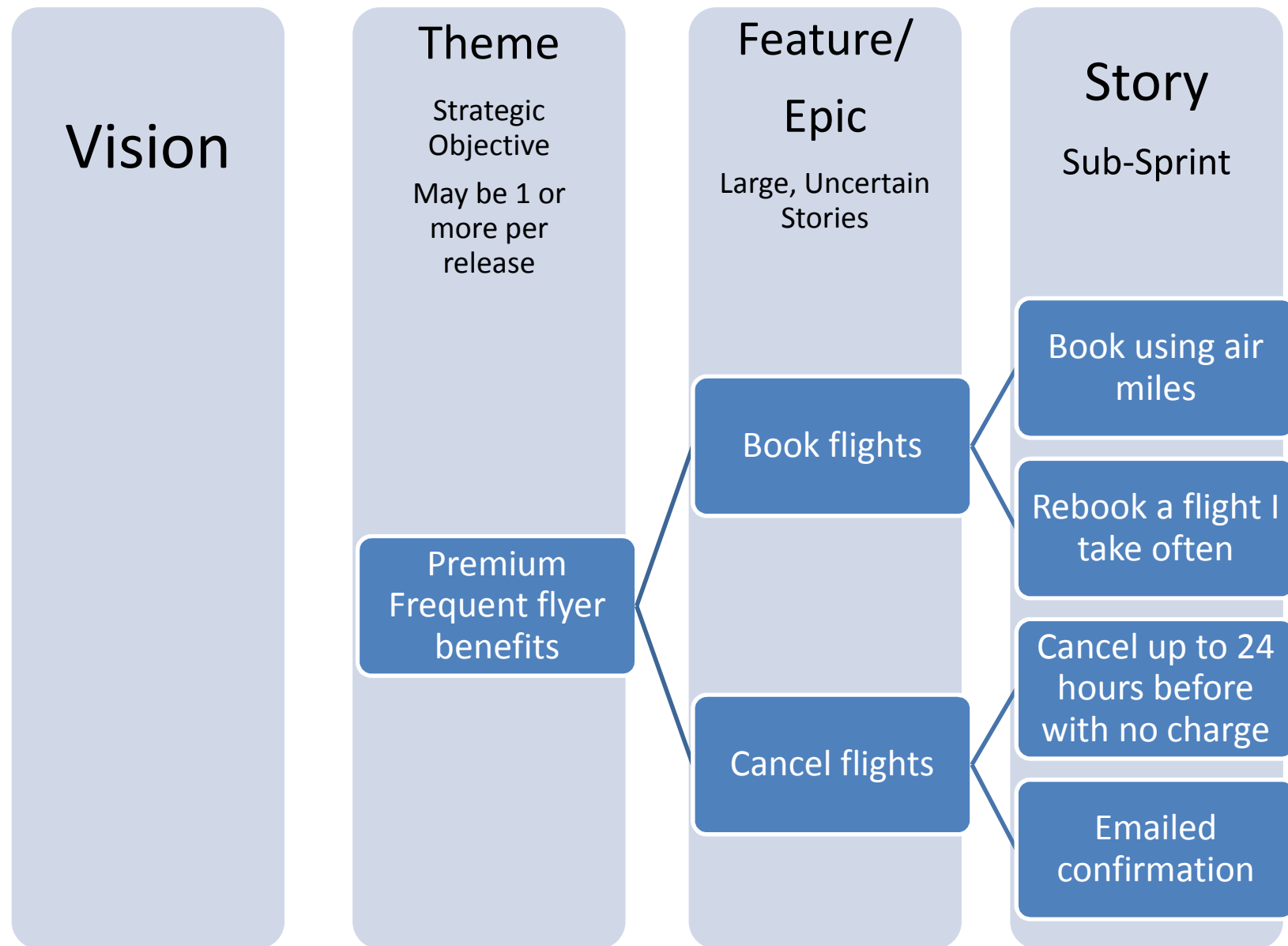
As a HomeOwner, I want to regularly trim my lawn so its neat and tidy.



ire Quality Serv

Co-Design

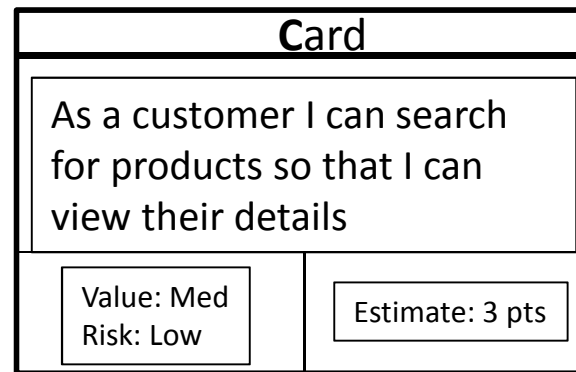




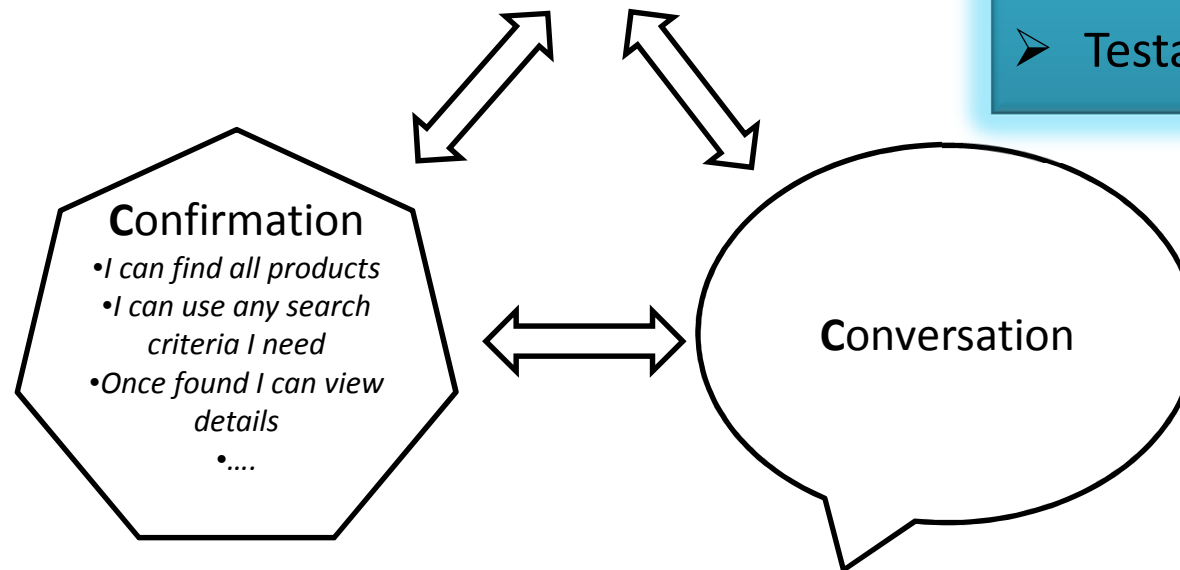
User Stories - CCC

INVEST

As a <role>
I need <action>
so that <result>



- Independent
- Negotiable
- Valuable
- Estimable
- Small
- Testable



User Story Example – Email Attachments

Email Attachments.

As a user I want emails with attachments to go faster so that I can work more efficiently

Confirmation:

- User notices emails with attachments go at least twice as fast
- Works with attachments up to 10MB
- Works with up to 50 attachments

CONVERSATION:

- What if attachment already compressed?
- What if it's a small file to start with?
- Should we store the compressed version?
- Should we allow user select compression options?
- Would up to 100 attachments be enough?
- Can each attachment be up to 10MB?

User Story Example – Hotel Reservation

Reservation Cancellation

As a user I want to cancel a reservation so that I avoid being charged full rate

Confirmation:

- Verify a premium member can cancel the same day without a fee
- Verify a non-premium member is charged 10% for same day cancellation but otherwise not charged
- Verify an email confirmation is sent to user with appropriate information
- Verify that the hotel is notified within 10 minutes of a cancellation

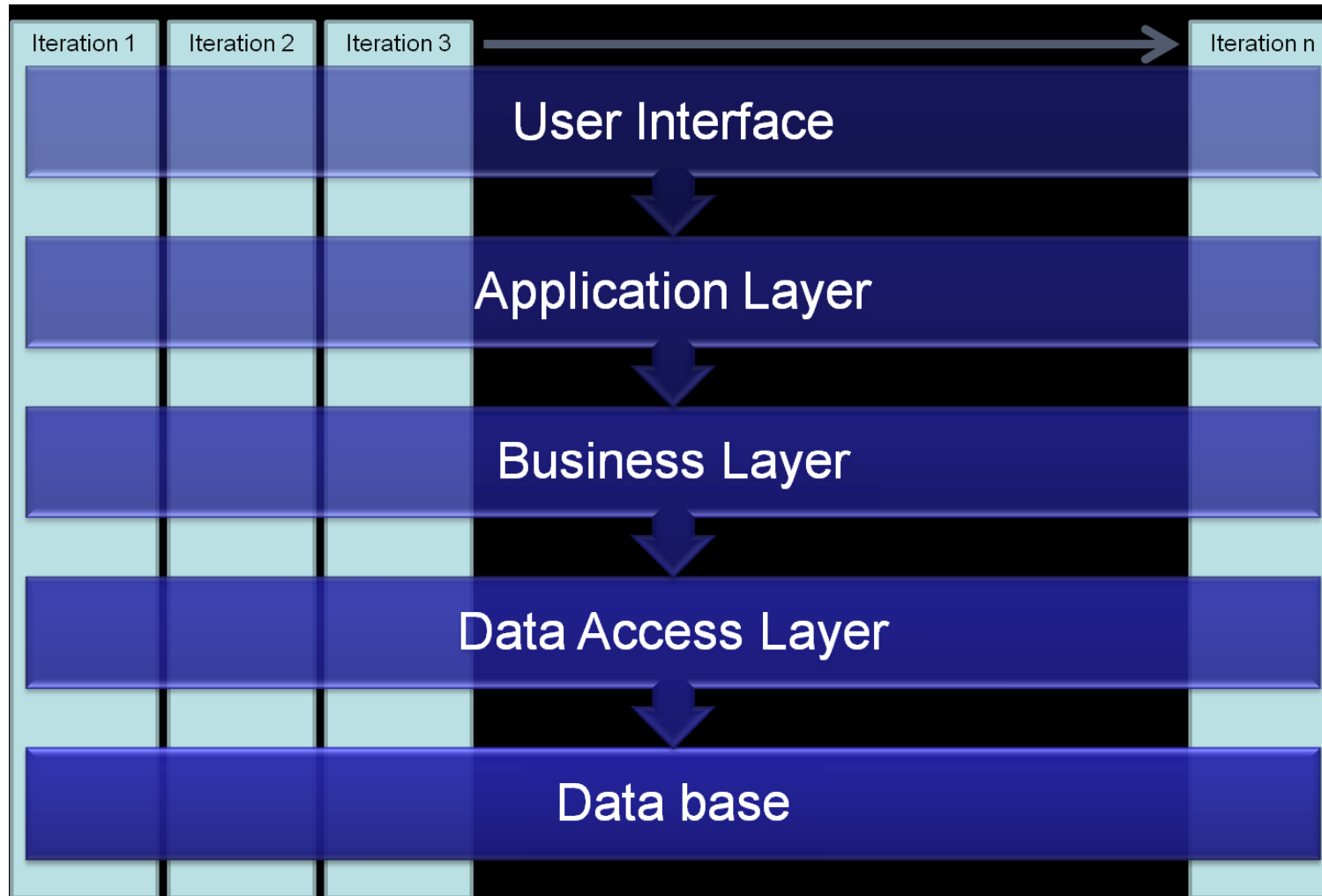
CONVERSATION:

- What if I am a premium member – do I have charges?
- When is a non-premium member charged and how much?
- How do these vary depending on when cancellation occurs?
- Do we need to send the user confirmation by email?
- When does the hotel need to be notified?
- What if the user has paid a deposit?

Purpose of confirmation/acceptance criteria

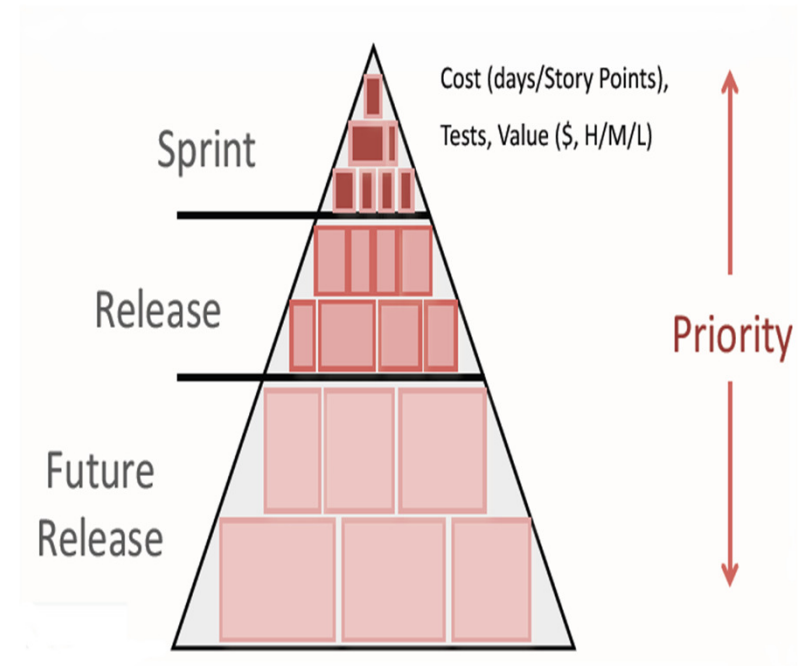
- define the boundaries for a user story/feature
- help the product owner answer what she needs in order for this feature to provide value (typically these are the minimum functional requirements)
- help the team gain a shared understanding of the story/feature
- help developers and testers to derive tests
- help developers know when to stop adding more functionality to a story

Vertical slices

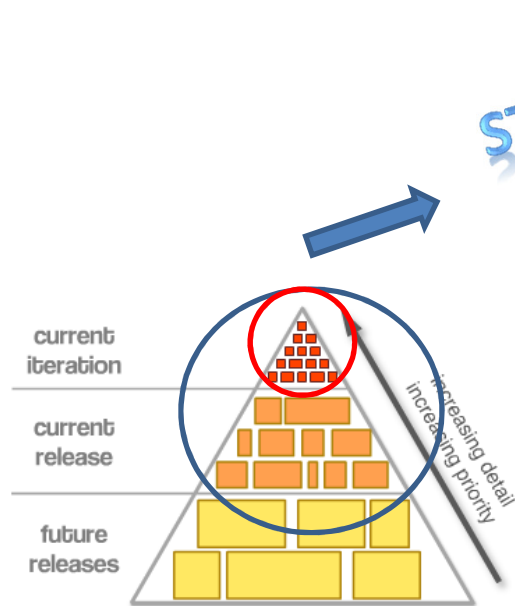


Product Backlog

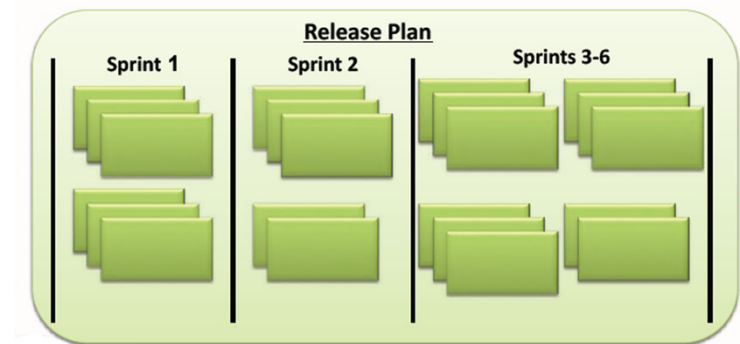
- A list of all desired work the project
- Product Backlog Items (PBIs)
 - Usually a combination of
 - Feature/story-based work (“let user search and replace”)
 - task-based work (“improve exception handling”)
 - constraints (“10,000 transactions a second”)
 - bugs
- List is prioritized by the Product Owner
 - “What’s the most important thing we could be doing now?”
 - Balance Business Value & Risk
- Initial backlog and ongoing backlog ‘grooming’



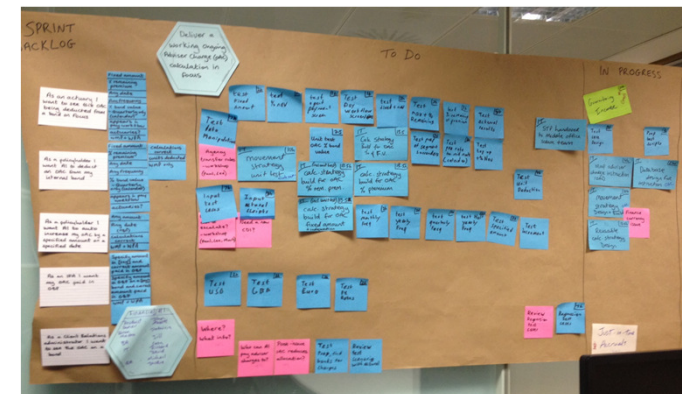
Estimation & Planning



STORY POINTS,
VELOCITY



Sprint Plan



Q&A



Fran O'Hara
InspireQS

www.inspireqs.ie
fran.ohara@inspireqs.ie

AgileIreland
LinkedIn Group

All-Island Community for
Agile & Lean Software
Development

Lean Competitiveness Fund
Enterprise Ireland

Funding Support for SMEs
Training, Consulting,
Coaching in Lean