Monday, 5 November 12

User Stories Overrated (Farlig) Lyntale

Tom@Gilb.com
www.Gilb.com
Smidig Oslo 2012

1

User Stories are Overrated:
why they might be too light

By Tom Gilb

Published Paper in AgileRecord.com

Gilb's Mythodology Column

User Stories: A Skeptical View

by Tom and Kai Gilb

The Skeptical View

We agree with the ideals of user stories, in the 'Myths' [1, Denning & Cohn] discussed below, but do not agree at all to Myth arguments given, that user stories are a good, sufficient or even

of our product clearly superior to all competitive products at all times.

Scale: average seconds needed for defined [Users] to Correctly Complete defined [Tasks] defined [Help]

Original Claims for User Stories attributes are here

The
LEADER'S GUIDE
to Radical
Management

REINVENTING the WORKPLACE
for the 21ST CENTURY

How to Inspire Continuous Innovation, Deep Job Satisfaction & Client Delight

STEPHEN
DENNING

http://stevedenning.typepad.com/

Denning's Claims are From Mike Cohn's User Stories books & papers

User Stories:

- *Structure*
  - Stakeholder A
  - Needs X
  - Because Y

Sample user stories

As an account holder, I want to check my savings account balance.

As an account holder, I am required to authenticate myself before using the system.

As the primary account holder, I can grant access to additional users so that they can see transactions.

© 2003–2008 Mountain Goat Software®

My General Assertion

- User Stories are good enough for small scale and non-critical projects

- But, they are not adequate for non-trivial projects

- The claims (= 'myths', in slides ahead) are not generally true,
  - especially _when we scale up_

Myth 1:
User stories and the conversations provoked by them comprise *verbal communication*, which is clearer than written communication.

- **Verbal communication is not clearer than written communication**
- ***The use of "Dialogue***
  - **to clear up *'bad written user stories' "***
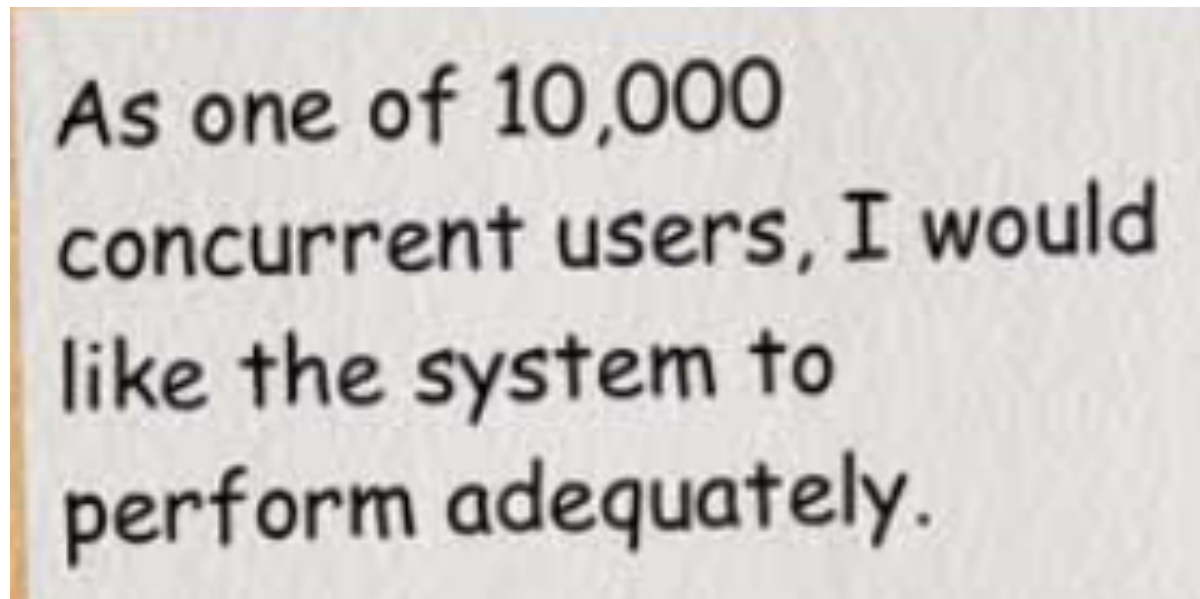 - **does not *prove* that there are no *superior* written formats**

- **I, as a user, want clearer interfaces to save time**

- **Usability:**
  - **Scale: Time for defined Users to Successfully complete defined Tasks**
  - **Goal [Users = Novices, Tasks = Inquiry] 20 Seconds.**
  - **Successfully: defined as: correct, no need to correct it later.**

Myth 2:
"User stories represent a common language.
They are intelligible to both users and developers."

As one of 10,000 concurrent users, I would like the system to perform adequately.

- What does 'perform' mean ?
- What does 'adequately' mean?
- What does it mean under higher or lower loads?

Myth 3:
"User stories are the *right size* for planning and prioritizing."

- **Right Size** [Requirement]: defined as:
- **The size that is sufficient for all requirements purposes,**
- **without any 'In project' supplements,**
- **at a cost that is lower than**
- **the costs of dealing with defects in the statement later.**

- Assertion
- User Stories are *rarely* <u>detailed</u> enough and <u>clear</u> enough to do intelligent planning (for example *estimation*)
- Or intelligent (dynamic) Prioritization

Myth 4:
User stories are *ideal for iterative development*,
which is the nature of most software development.

- User stories are a disaster for iterative development
  - because you cannot understand their incremental and final consequences;

- you cannot measure evolutionary *value* delivery progress toward such objectives.

- The nature of software development should not be to 'write use cases', stories, and functions,
  - as some seem to believe.

- The Agile ideal is to **deliver incremental *value* to *stakeholders*.**

Myth 5:
"User stories help *establish priorities* that make sense to both users and developers."

- Ambiguous unintelligible written stories are a logically bad basis for determining the priority of that story for *anyone*.
- Here is my idea of 'priority'.
- **A potential increment will be *prioritized* based on 'stakeholder value for costs', with 'respect to risk'.**
- Ambiguous written stories do not admit numeric evaluation of value for defined stakeholders, or of all cost aspects, or of all risk aspects.
- Also a **well-defined** requirement can be evaluated for potential **value** to **stakeholders**,
  - it *cannot* be evaluated for *cost*.
  - The **cost resides entirely in the design**,
  - and the design is in principle not chosen yet!
- Consequently **you cannot choose best value for money with user stories** alone.

- Try the story:
- ***"We want the most intuitive system possible"***
- What is the cost?
- You cannot have any useful idea of cost,
  - because the requirement is so vague
  - that you cannot even understand it fully,
  - let alone *choose* a best design at all;
  - and you cannot *cost* a design that is not *chosen*. It is illogical
- In addition, *until you know the specific design*,
  - you *cannot understand the risk of deviation* from your objectives and costs,
  - so you cannot prioritize iterations with regard to risk either.
- So, the prioritization argument for user stories **is logically unreasonable.**

Myth 6: "The process enables *transparency.* Everyone understands why."

- The arguments above, particularly the prioritization argument, say **no**, everybody does *not* understand why.
- They may *feel* they understand,
  - but since the user story is incomplete and ambiguous,
  - they cannot *really* understand *anything*;
  - for example anything about value, stakeholders, design, costs, and risks.
- There may be an illusion of understanding,
  - but there is no rationally defined understanding.

- However, there may be *social* comfort if teams misunderstand it together,
  - but in non-transparently *different* interpretations.
- *That* does not lead to value or system success,
  - even for those who *thought* they understood the consequences of the user story choice.

- MORE DETAIL ?
- Ask me for free digital copy of Book, Paper and Slides
  - Tom@Gilb.com

- Download Related Papers and Slides
- www.Gilb.com
- (Downloads tab)