


DEVELOP YOUR EXPERTISE

# Writing Effective User Stories for Agile Requirements

Mike Cohn  
September 26, 2005

SD BEST PRACTICES CONFERENCE & EXPO 2005

## Mike Cohn—background



- Programming for 20 years
- Author of
  - *User Stories Applied*
  - *Agile Estimating and Planning*
  - Java, C++, database programming books
- Founding member and director of the Agile Alliance and the Scrum Alliance
- Founder of Mountain Goat Software
  - Process and project management consulting and training

2 All slides copyright 2000-2005, Mountain Goat Software

## Today's agenda



- What user stories are
- Users and user roles
- Gathering stories
- INVEST in good stories
- Why user stories?

3

All slides copyright 2000-2005, Mountain Goat Software

## Ron Jeffries' Three Cs

Card

- Stories are traditionally written on note cards.
- Cards may be annotated with estimates, notes, etc.

Conversation

- Details behind the story come out during conversation with customer

Confirmation

- Acceptance tests confirm the story was coded correctly

4

All slides copyright 2000-2005, Mountain Goat Software

## Samples – Travel reservation system

---

As a user, I can reserve a hotel room.

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

As a user, I can cancel a reservation.

As a user, I can restrict searches so that I only see hotels with available rooms.

5

All slides copyright 2000-2005, Mountain Goat Software

## Where are the details?

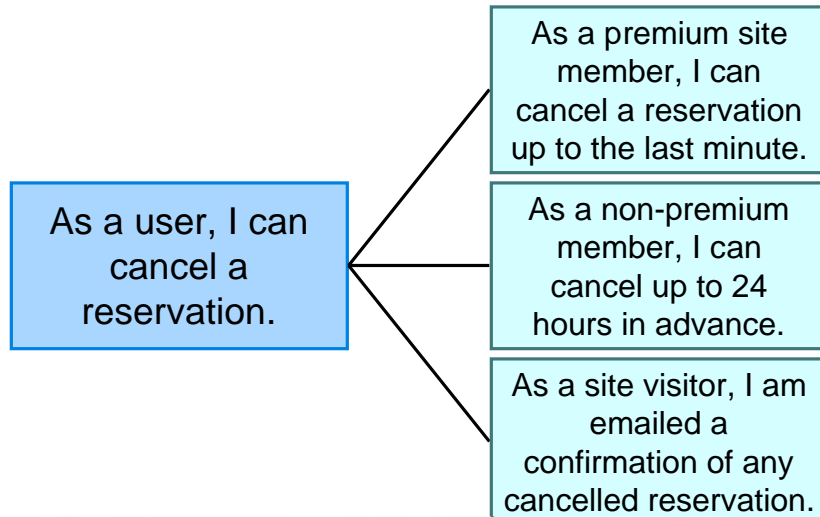
---

- As a user, I can cancel a reservation.
  - Does the user get a full or partial refund?
    - Is the refund to her credit card or is it site credit?
  - How far ahead must the reservation be cancelled?
    - Is that the same for all hotels?
    - For all site visitors? Can frequent travelers cancel later?
  - Is a confirmation provided to the user?
    - How?

6

All slides copyright 2000-2005, Mountain Goat Software

## Details added in smaller “sub-stories”



7

All slides copyright 2000-2005, Mountain Goat Software

## Details added as tests

- High level tests are added to the story
  - Can be used to express additional details and expectations

As a user, I can cancel a reservation.

- Verify that a premium member can cancel the same day without a fee.
- Verify that a non-premium member is charged 10% for a same-day cancellation.
- Verify that an email confirmation is sent.
- Verify that the hotel is notified of any cancellation.

8

All slides copyright 2000-2005, Mountain Goat Software

## Today's agenda



- What user stories are
- Users and user roles
- Gathering stories
- INVEST in good stories
- Why user stories?

9

All slides copyright 2000-2005, Mountain Goat Software

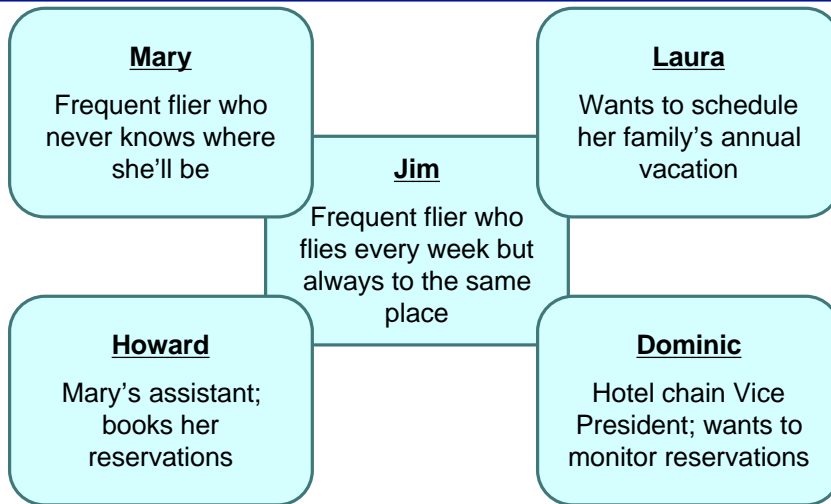
## "The User"

- Many projects mistakenly assume there's only one user:
  - "The user"
- Write all stories from one user's perspective
- Assume all users have the same goals
- Leads to missing stories

10

All slides copyright 2000-2005, Mountain Goat Software

## Travel Site—Who's the user?



11

All slides copyright 2000-2005, Mountain Goat Software

## User roles

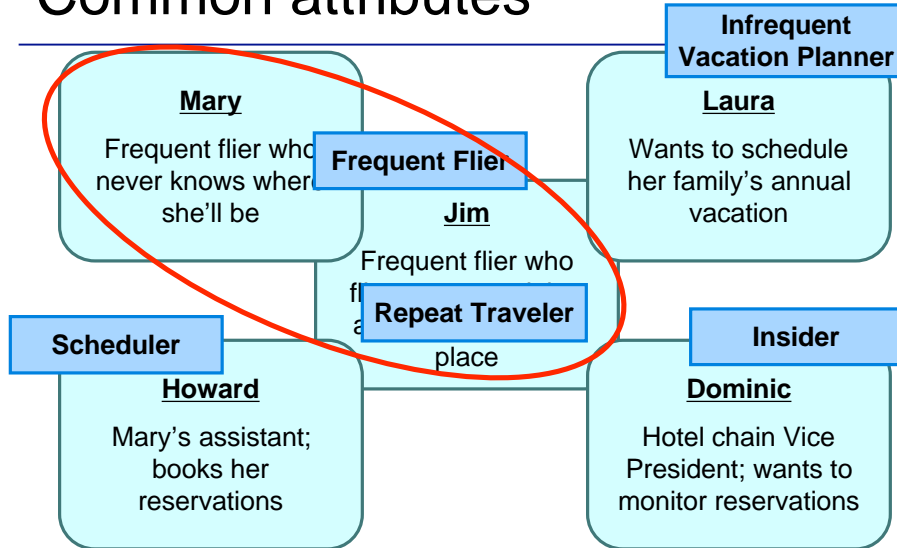
- Broaden the scope from looking at one user
- Allows users to vary by
  - What they use the software for
  - How they use the software
  - Background
  - Familiarity with the software / computers
- Used extensively in usage-centered design
- Definition
  - A user role is a collection of defining attributes that characterize a population of users and their intended interactions with the system.

Source: *Software for Use* by Constantine and Lockwood (1999).

12

All slides copyright 2000-2005, Mountain Goat Software

## Common attributes



13

All slides copyright 2000-2005, Mountain Goat Software

## User role brainstorming

- Brainstorming meeting
  - Customer, developers, anyone who understands a product's intended users
- Everyone grabs a stack of cards
- Write role names on cards
  - As fast as possible and with no judgment
    - No turns
  - Place card on table
  - Call out role name as you place it

14

All slides copyright 2000-2005, Mountain Goat Software

## User role brainstorming



We've been hired by fBay to create "the best new web auction site since eBay."

- Brainstorm the user roles who will interact with this site.

15

All slides copyright 2000-2005, Mountain Goat Software

## User role modeling steps

- Brainstorm an initial set of user roles
- Organize the initial set
- Consolidate roles
- Refine roles

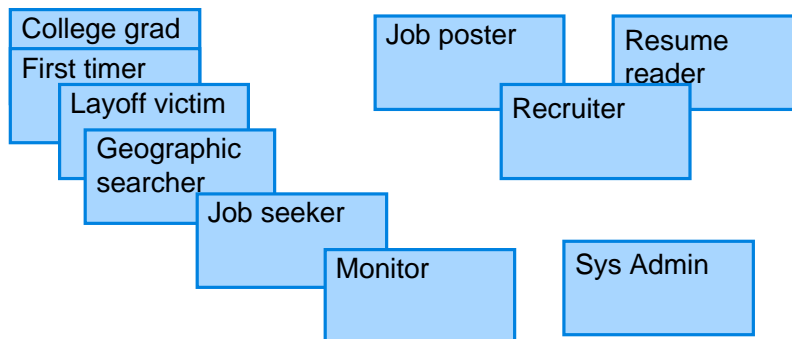
16

All slides copyright 2000-2005, Mountain Goat Software



## Organize the initial set

- Arrange cards spatially to indicate overlapping and similar roles
  - Use any arrangement rules you want



17

All slides copyright 2000-2005, Mountain Goat Software

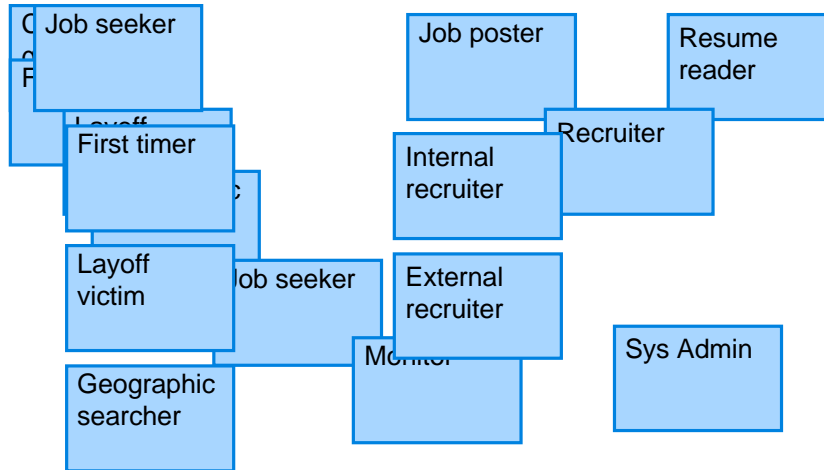
## Consolidate roles

- Discuss what is meant by each card
- Arrange cards spatially to indicate overlapping and similar roles
  - Use any arrangement rules you want
- Look for cards to
  - Combine
  - Replace with a more generic/different card
- Eliminate cards that are unimportant to the success of the product

18

All slides copyright 2000-2005, Mountain Goat Software

## Consolidating—an example



19

All slides copyright 2000-2005, Mountain Goat Software

## Organize and consolidate



- 1) Organize your initial set of user roles for fBay.
- 2) Consolidate the user roles.

20

All slides copyright 2000-2005, Mountain Goat Software

## Advantages of using roles

Users become tangible

Start thinking of software as solving needs of real people.

Avoid saying “the user”

Instead we talk about “a frequent flier” or “a repeat traveler”

Incorporate roles into stories

“As a <role>, I want <story> so that <benefit>.”

21

All slides copyright 2000-2005, Mountain Goat Software

## Today's agenda



- What user stories are
- Users and user roles
- Gathering stories
- INVEST in good stories
- Why user stories?

22

All slides copyright 2000-2005, Mountain Goat Software

## Gathering stories

- Common metaphors for requirements are wrong
  - “Eliciting requirements”
  - “Capturing requirements”
- These metaphors imply
  - Users know the requirements but don’t want to tell us
  - Requirements need to be locked up once “captured”

23

All slides copyright 2000-2005, Mountain Goat Software

## The proper metaphor

- Trawling<sup>†</sup> for requirements
  - Trawl: “sift through as part of a search” (OAD)
- Metaphor captures these aspects:
  - Requirements can be captured with different sized nets
  - Requirements change, mature, possibly die
  - Skill is a factor

<sup>†</sup>*Mastering the Requirements Process*  
by Suzanne and James Robertson,  
1999.



24

All slides copyright 2000-2005, Mountain Goat Software

## A little is enough, or is it?

---

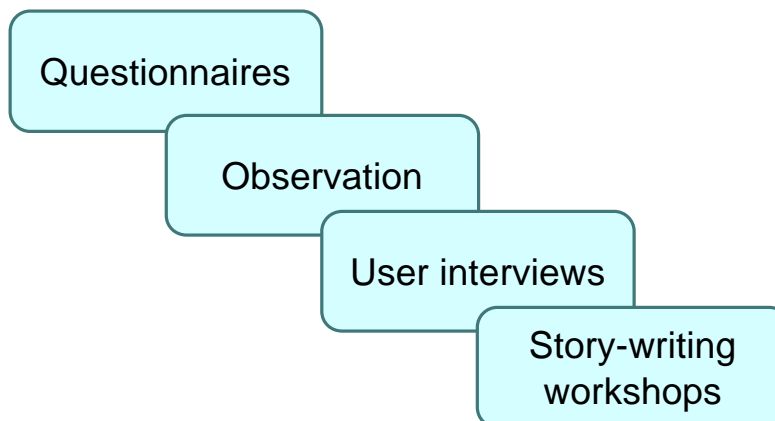
- Agile processes acknowledge that we cannot trawl with such a fine net that we can write all the user stories upfront
- However,
  - This doesn't mean we shouldn't write as many as we can

25

All slides copyright 2000-2005, Mountain Goat Software

## Techniques for trawling for stories

---



26

All slides copyright 2000-2005, Mountain Goat Software

## Questionnaires

---

- Good technique for learning more about stories you already have
- If you have a large user base, great way to get information to help prioritize stories
- Not effective as a primary means of trawling for new stories

27

All slides copyright 2000-2005, Mountain Goat Software

## Observation

---

- Great way to pick up insights
- Two approaches
  - Just observe, with or without user's knowledge
  - Have the user demonstrate to a group how she uses the software

28

All slides copyright 2000-2005, Mountain Goat Software

## Observation example

---

- Stated need:
  - “We need a large text field to summarize.”
- Observed need:
  - Have the system record the user’s choices



29

All slides copyright 2000-2005, Mountain Goat Software

## Interviews

---

- Default approach taken by many teams
- Selection of interviewees is critical
  - Try to interview as many user roles as possible
- Cannot just ask “So whaddaya want?”
  - Most users are not adept at understanding their true needs

30

All slides copyright 2000-2005, Mountain Goat Software

## Context matters

“My wife and I  
split up...”



“He’s no longer  
with us...”

31

All slides copyright 2000-2005, Mountain Goat Software

## My context isn't your context

“Dad,  
make it  
warmer.”



32

All slides copyright 2000-2005, Mountain Goat Software



## A horrible question

---

- This question sucked two years out of my life

“Would you like it in a browser?”

“Of course, now that you mention it!”

33

All slides copyright 2000-2005, Mountain Goat Software

## We can do better

---

“What would you think of having this app in a browser rather than as a native Windows application even if it means reduced performance, a poorer overall user experience, and less interactivity?”

- It's open
  - Full range of answers
- But it has too much context

34

All slides copyright 2000-2005, Mountain Goat Software

## The best way to ask

---

“What would you be willing to give up in order to have it in a browser?”

- We want to ask questions that are
  - Open-ended
  - Context-free

35

All slides copyright 2000-2005, Mountain Goat Software

## Beware of assumptions

---

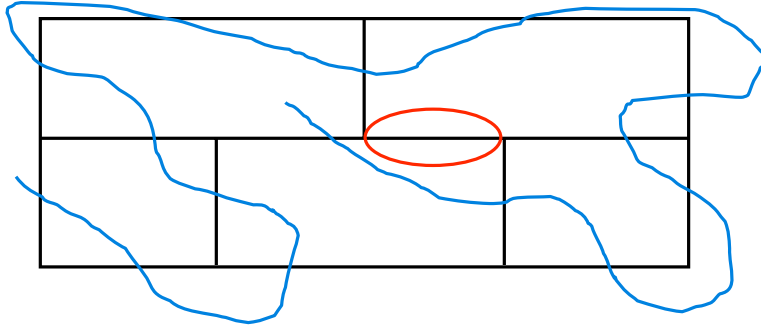
- What time did the sun rise in Boston, MA, on October 10, 1582?
- Draw a single line that crosses each line segment in this figure exactly once:



36

All slides copyright 2000-2005, Mountain Goat Software

## Seeking a solution



- “Dad, do you know the answer?”

37

All slides copyright 2000-2005, Mountain Goat Software

## It's my problem, I know the solution

- Having a problem does not uniquely qualify you to solve it
- “It hurts when I go like this...”



38

All slides copyright 2000-2005, Mountain Goat Software

## We need to stop asking users

- Since users don't know how to solve their problems, we need to stop *asking*
- We need to *involve* them instead

Empirical design

- Designers of the new system make decisions by studying prospective users in typical situations

Participatory design

- The users of the system become part of the team designing the behavior of the system

39

All slides copyright 2000-2005, Mountain Goat Software

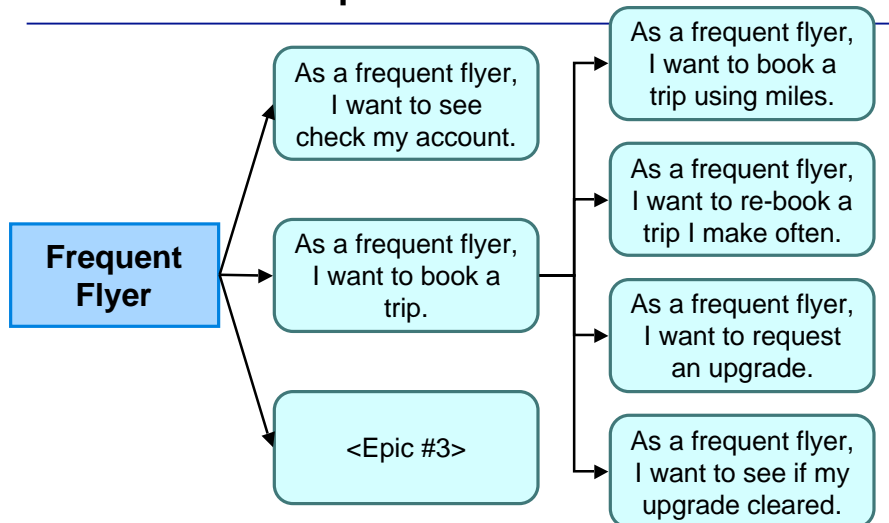
## Story-writing workshops

- Includes developers, users, customer, others
- Goal is to write as many stories as possible
  - No prioritization at this point
- Uses low-fidelity prototyping and brainstorming techniques

40

All slides copyright 2000-2005, Mountain Goat Software

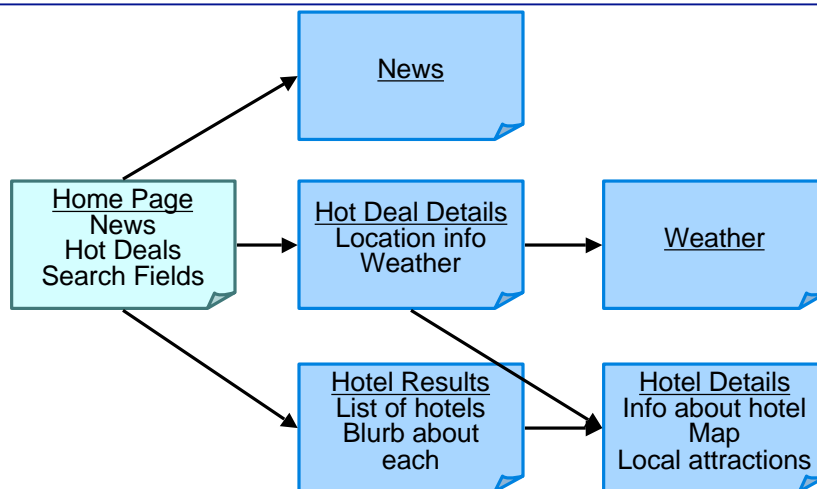
## Start with epics and iterate



41

All slides copyright 2000-2005, Mountain Goat Software

## A low-fidelity prototype



42

All slides copyright 2000-2005, Mountain Goat Software

## Low-fidelity prototyping

---

- Use paper, note cards, white board, big Post-its
- Prototype is of components or areas within the application, *not* of actual screens
  - Hotel Results could be on Home Page or be a separate page
- Doesn't require knowledge of how screens will look
- Throw it away a day or two later
- Works better to go depth-first

43

All slides copyright 2000-2005, Mountain Goat Software

## Creating the low-fidelity prototype

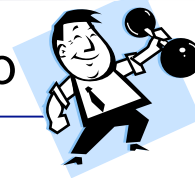
---

- Start with an empty box:
  - "Here's the main screen in the system"
- Ask open-ended, context-free questions as you go:
  - What will the users most likely want to do next?
  - What mistakes could the user make here?
  - What could confuse the user at this point?
  - What additional information could the user need?
- Consider these questions for each user role

44

All slides copyright 2000-2005, Mountain Goat Software

## A mini story-writing workshop



Write some user stories for fBay based on the roles you identified.

Tip: try this template:

“As a <role>, I want to <story> so that <benefit>.”

45

All slides copyright 2000-2005, Mountain Goat Software

## Today's agenda

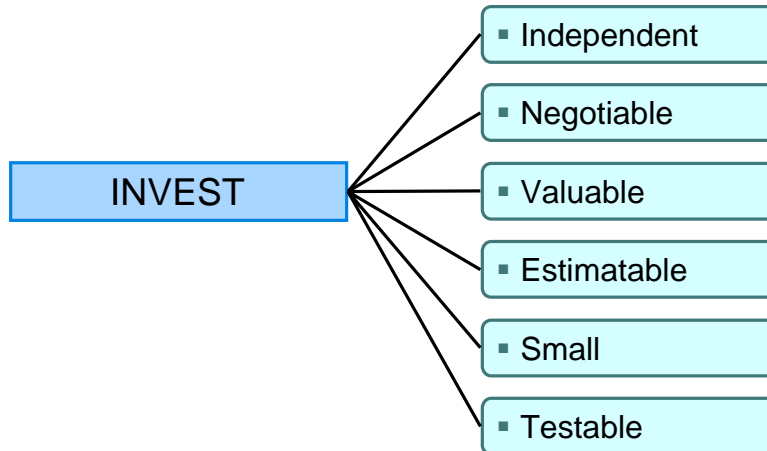


- What user stories are
- Users and user roles
- Gathering stories
- INVEST in good stories
- Why user stories?

46

All slides copyright 2000-2005, Mountain Goat Software

# What makes a good story?



Thanks to Bill Wake for the acronym. See [www.xp123.com](http://www.xp123.com).

47

All slides copyright 2000-2005, Mountain Goat Software

# Independent

- Avoid introducing dependencies
  - Leads to difficulty prioritizing and planning

A company can pay for a job posting with a Visa card.

?

A company can pay for a job posting with an AmEx card.

A company for a job po ?  
a MasterCard.

?

- The first of these stories will take 3 days to develop
  - It doesn't matter which is first
- The others will take 1 day

48

All slides copyright 2000-2005, Mountain Goat Software



## Making stories independent

Combine the stories

- A customer can pay with a credit card.

Split across a different dimension

- A customer can pay with one type of credit card.
- A customer can pay with two other types of credit cards.

Write two estimates and move on

- 3 days if first; 1 otherwise

49

All slides copyright 2000-2005, Mountain Goat Software

## Negotiable

- Stories are not
  - Written contracts
  - Requirements the software must fulfill
- Do not need to include all details
- Too many details give the impressions of
  - false precision or completeness
  - that there's no need to talk further
- Need some flexibility so that we can adjust how much of the story gets implemented
  - If the card is contract then it needs to be estimated like a contract

50

All slides copyright 2000-2005, Mountain Goat Software

## Is this story negotiable?

Print dialog allows the user to edit the printer list. The user can add or remove printers from the printer list. The user can add printers either by auto-search or manually specifying the printer DNS name or IP address. An advanced search option also allows the user to restrict his search within specified IP addresses and subnet range.

51

All slides copyright 2000-2005, Mountain Goat Software

## Valuable

- Stories must be valuable to either:

Users

- As a user, I can search for a job by title and salary range.

Purchasers

- Throughout the project, the team will produce documentation suitable for an ISO 9001 audit.
- The development team will produce the software in accordance with CMM level 3.
- All configuration information is read from a central location.

52

All slides copyright 2000-2005, Mountain Goat Software

## Stories valued by developers

- Should be rewritten to show the benefit

All connections to the database are through a connection pool.

Up to 50 users should be able to use the application with a five-user database license.

All error handling and logging is done through a set of common classes.

All errors are presented to the user and logged in a consistent manner.

53

All slides copyright 2000-2005, Mountain Goat Software

## Estimatable

- Because stories are used in planning
- A story may not be estimatable if:

Developers lack domain knowledge

■ As a new user, I am given a diabetic screening.

Developers lack technical knowledge

■ As a site visitor, I can elect to see all text in a larger font.

The story is too big

■ As a user, I can find a job.

54

All slides copyright 2000-2005, Mountain Goat Software

# Small

---

- Large stories (epics) are
  - hard to estimate
  - hard to plan
    - They don't fit well into single iterations
- Compound story
  - An epic that comprises multiple shorter stories
- Complex story
  - A story that is inherently large and cannot easily be disaggregated into constituent stories

55

All slides copyright 2000-2005, Mountain Goat Software

# Compound stories

---

- Often hide a great number of assumptions

As a user, I can post my resume.

- A resume includes separate sections for education, prior jobs, salary history, publications, etc.
- Users can mark resumes as inactive
- Users can have multiple resumes
- Users can edit resumes
- Users can delete resumes

56

All slides copyright 2000-2005, Mountain Goat Software

## Splitting a compound story

Split along operational boundaries (CRUD)

- As a user, I can create resumes, which include education, prior jobs, salary history, publications, presentations, community service, and an objective.
- As a user, I can edit a resume.
- As a user, I can delete a resume.
- As a user, I can have multiple resumes.
- As a user, I can activate and inactivate resumes.

57

All slides copyright 2000-2005, Mountain Goat Software

## Splitting a compound story, cont.

Split along data boundaries

- As a user, I can add and edit educational information on a resume.
- As a user, I can add and edit prior jobs on a resume.
- As a user, I can add and edit salary history on a resume.
- As a user, I can delete a resume.
- As a user, I can have multiple resumes.
- As a user, I can activate and inactivate resumes.

58

All slides copyright 2000-2005, Mountain Goat Software

## Other ways to split large stories

- Remove cross-cutting concerns
- Don't meet performance targets
- Avoid splitting stories into tasks
- Avoid the temptation of related changes

59

All slides copyright 2000-2005, Mountain Goat Software

## Testable

- Tests demonstrate that a story meets the customer's expectations
- Strive for 90+% automation

A user must find the software easy to use.

As a novice user, I am able to complete common workflows without training.

A user must never have to wait long for a screen to appear.

New screens appear within 2 seconds in 95% of all cases.

60

All slides copyright 2000-2005, Mountain Goat Software

## Fixing stories



- 1) Assess the stories you've written for fBay against the INVEST attributes.
- 2) Rewrite those that do not meet these criteria.
- 3) If you can't figure out how to rewrite a story, save it for class discussion.

<b>I</b> ndependent	<b>E</b> stimatable
<b>N</b> egotiable	<b>S</b> mall
<b>V</b> aluable	<b>T</b> estable

61

All slides copyright 2000-2005, Mountain Goat Software

## As an OEM procurement agent...



...I want the ability to search for suppliers based on criteria placed in an input interface. The results include the following functionality:

- I am able to select one or more suppliers from the list of suppliers in the database against which to perform the query.
- I input certain criteria into an interface, query the database, and receive results as to which suppliers meet the criteria.
- The results are shown in order from highest match to lowest match with a symbol to show complete match of required criteria and a bar to show overall match. The list should be paginated and have a certain discrete number of returns per page, with next/previous type navigation.
- The query should be performed on the Business Factors: Type of Business, Location, and Supplier Size.
- The query should be performed on technical factors: material, machine type, size, swing, axis.

62

All slides copyright 2000-2005, Mountain Goat Software

## Today's agenda



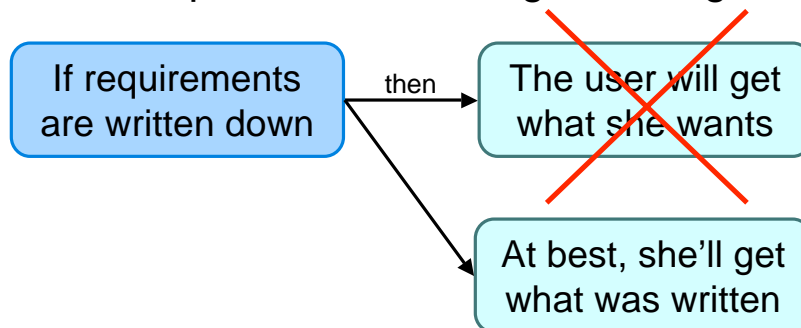
- ☑ What user stories are
- ☑ Users and user roles
- ☑ Gathering stories
- ☑ INVEST in good stories
- ☐ Why user stories?

66

All slides copyright 2000-2005, Mountain Goat Software

## So, why user stories?

- Shift emphasis from writing to talking



- “You built what I asked for, but it’s not what I need.”

67

All slides copyright 2000-2005, Mountain Goat Software



## Actual examples

The user can enter a name. It can be 127 characters.

- Must the user enter a name?
- Can it be other than 127 chars?

The system should prominently display a warning message whenever the user enters invalid data.

- What does *should* mean?
- What does *prominently display* mean?
- Is *invalid data* defined elsewhere?

68

All slides copyright 2000-2005, Mountain Goat Software

## Additional reasons

- Stories are comprehensible
  - Developers and customers understand them
  - People are better able to remember events if they are organized into stories<sup>†</sup>
- Stories are the right size for planning
- Support and encourage iterative development
  - Can easily start with epics and disaggregate closer to development time

<sup>†</sup>Bow er, Black, and Turner. 1979. *Scripts in Memory for Text*.

69

All slides copyright 2000-2005, Mountain Goat Software

## Yet more reasons

- Stories support opportunistic development
  - We design solutions by moving opportunistically between top-down and bottom-up approaches<sup>†</sup>
- Stories support participatory design
  - Participatory design
    - The users of the system become part of the team designing the behavior of the system
  - Empirical design
    - Designers of the new system make decisions by studying prospective users in typical situations

<sup>†</sup>Guindon. 1990. *Designing the Design Process*.

70

All slides copyright 2000-2005, Mountain Goat Software

## Most importantly...

Don't forget the purpose

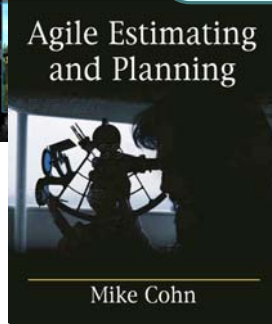
- The story text we write on cards is less important than the conversations we have.
- "Stories represent requirements, they do not document them."<sup>†</sup>

<sup>†</sup>Rachel Davies, "The Power of Stories," XP2001.

71

All slides copyright 2000-2005, Mountain Goat Software

## Mike Cohn contact information



- [mike@mountaingoatsoftware.com](mailto:mike@mountaingoatsoftware.com)
- (303) 810-2190 (mobile)
- (720) 890-6110 (office)
- [www.mountaingoatsoftware.com](http://www.mountaingoatsoftware.com)

