

# An Introduction to Agile Estimating & Planning

Mike Cohn  
February 10, 2010



1

## Mike Cohn - background



© Mountain Goat Software, LLC

2



# Imagine...

- That you're fed up with software development as a career
- And you decide to go into the landscaping business
- Your first job is moving this pile of rock from the front of my house to the back



© Mountain Goat Software, LLC



3

# How might you estimate this?

- One way:
  - Look at the pile of rock and estimate how many wheelbarrow loads it represents
- After an hour, see how many wheelbarrow loads you've moved then extrapolate the total duration



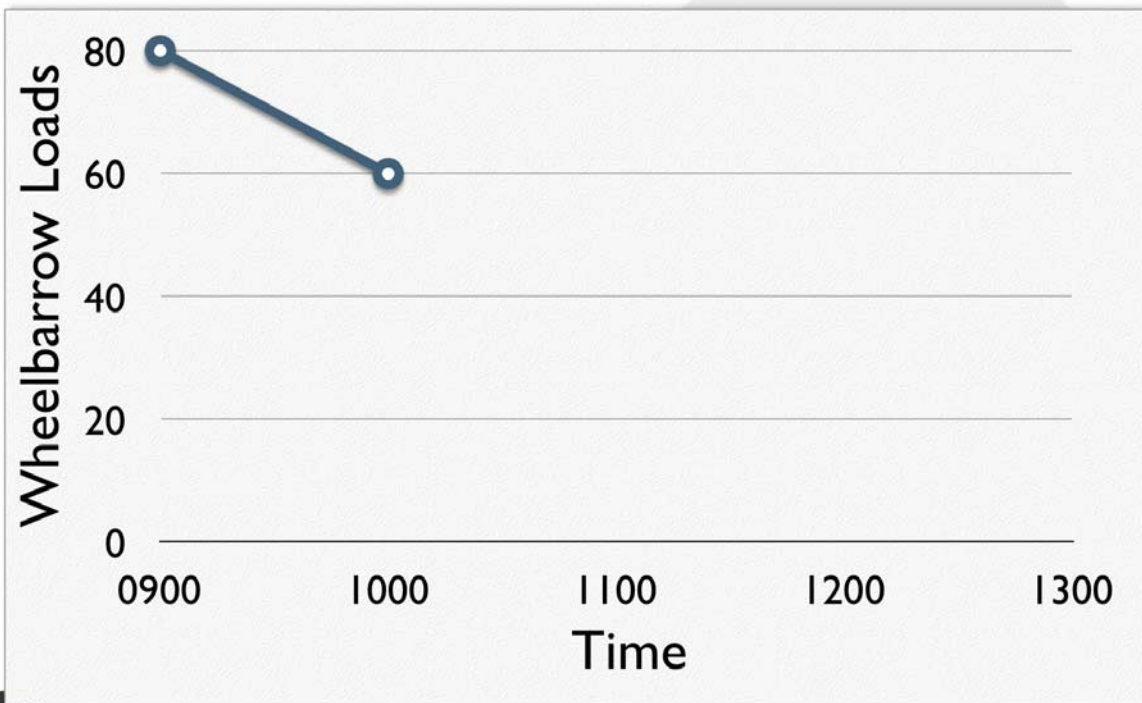
- I think that's 80 wheelbarrow loads
- After an hour I've moved 20 loads
- So, I'll be done in a total of 4 hours



© Mountain Goat Software, LLC

4

# My landscaping

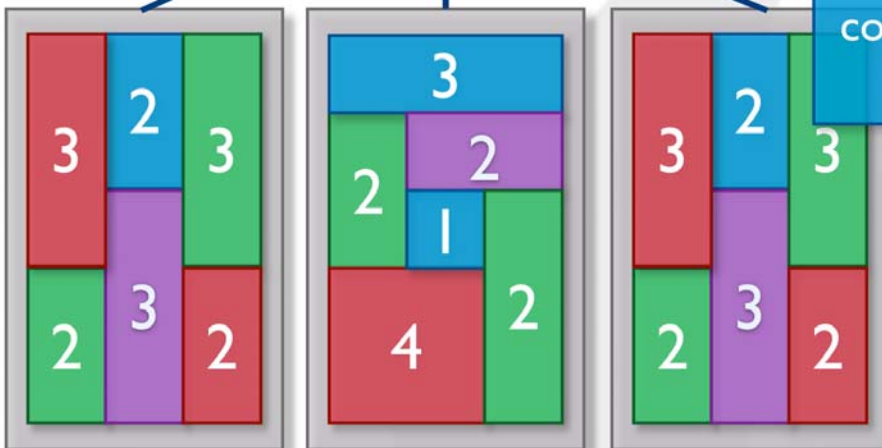


© Mountain Goat Software, LLC

5

- An iteration is a short, constrained period of time
- Typically 1-4 weeks

Velocity is the amount of work planned or completed in an iteration.



A release typically comprises more than one iteration



© Mountain Goat Software, LLC

6

# The planning onion



- Agile teams plan at the innermost three levels.
- Others (on the team in the company) plan at the outer levels.



# Relating the different planning levels

## Product Backlog

As a frequent flyer, I want to...	3
As a frequent flyer, I want to...	5
As a frequent flyer, I want to...	5
As a frequent flyer, I want to...	2
As a frequent flyer, I want to...	2

## Iteration Backlog

Code the UI	8
Write test fixture	6
Code middle tier	12
Write tests	5
Automate tests	4

“Yesterday I started on the UI; I should finish before the end of today.”

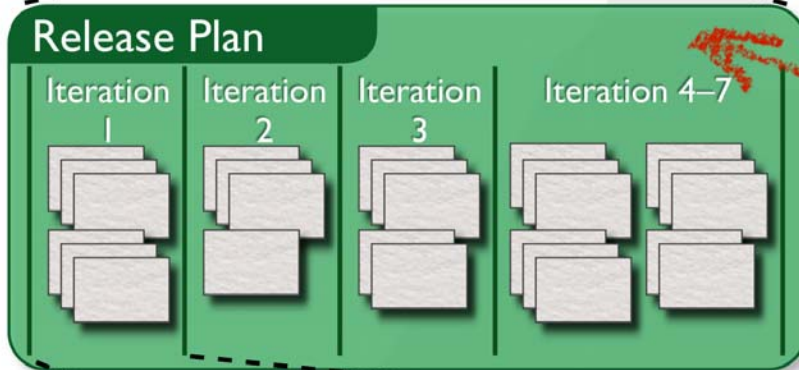


# Product, release, iteration planning

Release 1

Release 2

Release 3



We'll focus here today

Task A	8 hours
Task B	16 hours
Task C	5 hours
Task D	8 hours



© Mountain Goat Software, LLC

9

## Agenda



- Estimating
- Release planning



© Mountain Goat Software, LLC

10

# Story points

- Probably the most commonly used estimating unit among agile teams today
  - Name is derived from agile teams commonly expressing requirements as “user stories”
- Based on everything that influences the effort to develop a feature
- Unitless but numerically relevant estimates
  - A 10-point user story is expected to take twice as long as a 5-point user story



# Dog points



Assign “dog points” to the following dogs

Labrador retriever  
Dachshund  
Great Dane  
Terrier  
German Shepherd  
Poodle  
St. Bernard  
Bulldog



# Consider these two piles of work



What story point values might we put on these?



© Mountain Goat Software, LLC

13

## Three key advantages

- Estimating in story points:
  1. Forces the use of relative estimating
    - Studies have shown we're better at this<sup>†</sup>
  2. Focuses us on estimating the size, not the duration
    - We derive duration empirically by seeing how much we complete per iteration
  3. Puts estimates in units that we can add together
    - Time based estimates are not additive

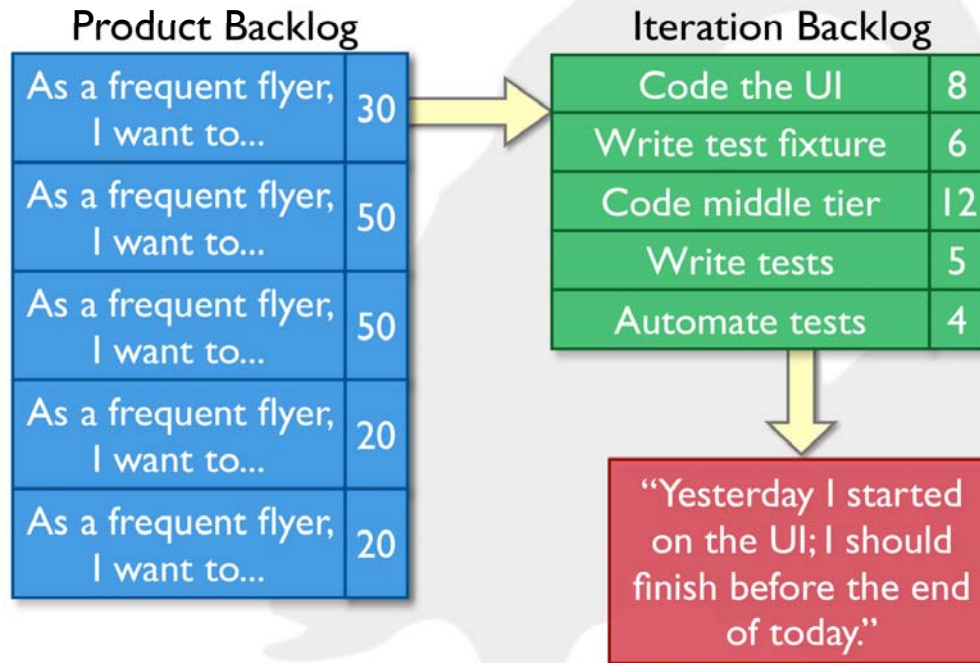
<sup>†</sup>Lederer and Prasad, 1998. *A Causal Model for Software Cost Estimating Error* and Vicinanza et al., 1991. *Software Effort Estimation: An Exploratory Study of Expert Performance*.



© Mountain Goat Software, LLC

14

# Comparing apples to apples



© Mountain Goat Software, LLC

15

# Planning Poker®

- An iterative approach to estimating
- Steps
  - Each estimator is given a deck of cards, each card has a valid estimate written on it
  - Customer/Product owner reads a story and it's discussed briefly
  - Each estimator selects a card that's his or her estimate
  - Cards are turned over so all can see them
  - Discuss differences (especially outliers)
  - Re-estimate until estimates converge



© Mountain Goat Software, LLC

16



# Planning Poker® - an example



Estimator	Round 1	Round 2
Susan	3	5
Vadim	8	5
Ann	2	5
Chris	5	8

© Mountain Goat Software, LLC

17

# Estimate these



Product backlog item	Estimate
Read a high-level, 10-page overview of agile software development in <i>People</i> magazine.	
Read a densely written 5-page research paper about agile software development in an academic journal.	
Write the product backlog for your uncle who owns a clock store and now wants an eCommerce site.	
Recruit, interview, and hire a new member for your team.	
Create a 60-minute presentation to introduce agile software development to your non-agile team.	
Wash and wax your boss' Porsche.	
Read a 150-page book on agile software development.	
Write an 8-page summary of that book for your boss.	

© Mountain Goat Software, LLC

18

# Why planning poker works

- Those who will do the work, estimate the work<sup>1</sup>
- Estimators are required to justify estimates<sup>2, 3</sup>
- Focuses most estimates within an approximate one order of magnitude<sup>4, 5</sup>

<sup>1</sup>Jørgensen, Magne. 2004. *A Review of Studies on Expert Estimation of Software Development Effort*.

<sup>2</sup>Hagafors, R., and B. Brehmer. 1983. *Does Having to Justify One's Decisions Change the Nature of the Decision Process?*

<sup>3</sup>Brenner, et al. 1996. *On the Evaluation of One-sided Evidence*.

<sup>4</sup>Miranda, Eduardo. 2001. *Improving Subjective Estimates Using Paired Comparisons*.

<sup>5</sup>Saaty, Thomas. 1996. *Multicriteria Decision Making: The Analytic Hierarchy Process*.



© Mountain Goat Software, LLC

19

# Why planning poker works

- Combining of individual estimates<sup>6</sup> through group discussion<sup>7</sup> leads to better estimates
- Emphasizes relative rather than absolute estimating
- Estimates are constrained to a set of values so we don't waste time in meaningless arguments
- Everyone's opinion is heard
- It's quick and fun

<sup>6</sup>Hoest, Martin, and Claes Wohlin. 1998. *An Experimental Study of Individual Subjective Effort Estimations and Combinations of the Estimates*.

<sup>7</sup>Jørgensen, Magne, and Kjetil Moløkken. 2002. *Combination of Software Development Effort Prediction Intervals: Why, When and How?*



© Mountain Goat Software, LLC

20

# Reduces impact of irrelevant information

## Group A

- Given project spec.

• 20 hours

## Group B

- Given same spec but with estimation-irrelevant details added:
  - end users' desktop applications
  - user passwords,
  - etc.

• 39 hours



Source: *How to avoid impact from irrelevant and misleading information on your cost estimates*, Magne Jørgensen and Stein Grimstad, Simula Research Laboratory, Simula Research Labs Estimation Seminar, Oslo, Norway 2006.

© Mountain Goat Software, LLC

21

# Specification length

## Group A

- Given a one-project spec.

• 117 hours

## Group B

- Given a spec with exactly the same text but was 7 pages long
- Increased length achieved through
  - double line space
  - wide margins
  - larger font size
  - more space between paragraphs

• 173 hours

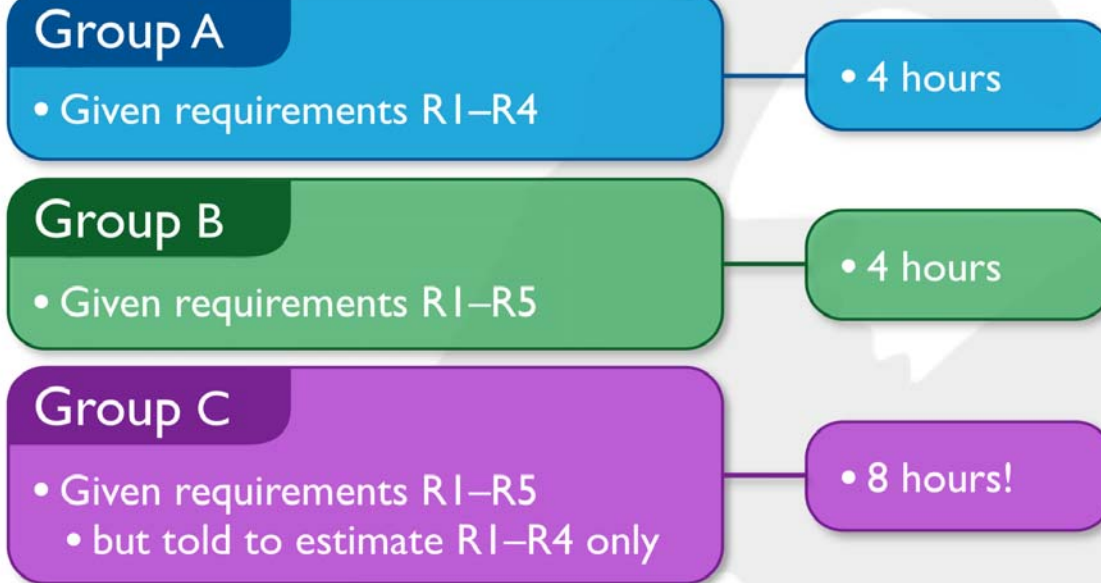


Source: *How to avoid impact from irrelevant and misleading information on your cost estimates*, Magne Jørgensen and Stein Grimstad, Simula Research Laboratory, Simula Research Labs Estimation Seminar, Oslo, Norway 2006.

© Mountain Goat Software, LLC

22

# Extra requirements



Source: *How to avoid impact from irrelevant and misleading information on your cost estimates*,  
Magne Jørgensen and Stein Grimstad, Simula Research Laboratory,  
Simula Research Labs Estimation Seminar, Oslo, Norway 2006.



© Mountain Goat Software, LLC

23

# Reduces likelihood of anchoring



Source: *How to avoid impact from irrelevant and misleading information on your cost estimates*,  
Magne Jørgensen and Stein Grimstad, Simula Research Laboratory,  
Simula Research Labs Estimation Seminar, Oslo, Norway 2006.



© Mountain Goat Software, LLC

24

The screenshot shows a web browser window with the URL <http://rthijs.planningpoker.com/games/26>. The page title is "Payroll system replacement". The main content area contains three estimation tasks:

- Task 1: "Write a list of definitions." with an estimate of 3.
- Task 2: "As a/an unauthenticated user I would like to log in so that I can start using the application." with an estimate of 3.
- Task 3: "As a/an authenticated user I would like to change my password." with an estimate of 2.

Below these tasks, a question asks "As a/an admin I would like to add new users so that they can log in". It then asks "How are they going to get their username and password?". The answers are shown in a grid of cards:

3	3	5	13	20
Thijs V.	Manfred S.	Mike C.	Giel N.	Angie

Below this, another row of cards shows estimates for the same task:

5	5	5	5
Manfred S.	Thijs V.	Giel N.	Mike C.

A "Complete" button is at the bottom, with a note: "(Note: Completes automatically when all estimates are in)".

On the right sidebar, there is a "Start timer" button, a "Complete game" button, and a list of participants: Angie, Giel de Nijs, Manfred Stienstra, Mike Cohn, and Thijs van der Vossen (moderator).

Free, or I wouldn't mention it



## Agenda



Estimating

Release planning



# Release planning

## Purpose

To answer questions such as:

- How much will be done by 30 June?
- When can we ship with this set of features?
- How many people or teams should be on this project?

## Inputs

- Velocity
- The length of the project
- Prioritized product backlog



# An example with velocity=14

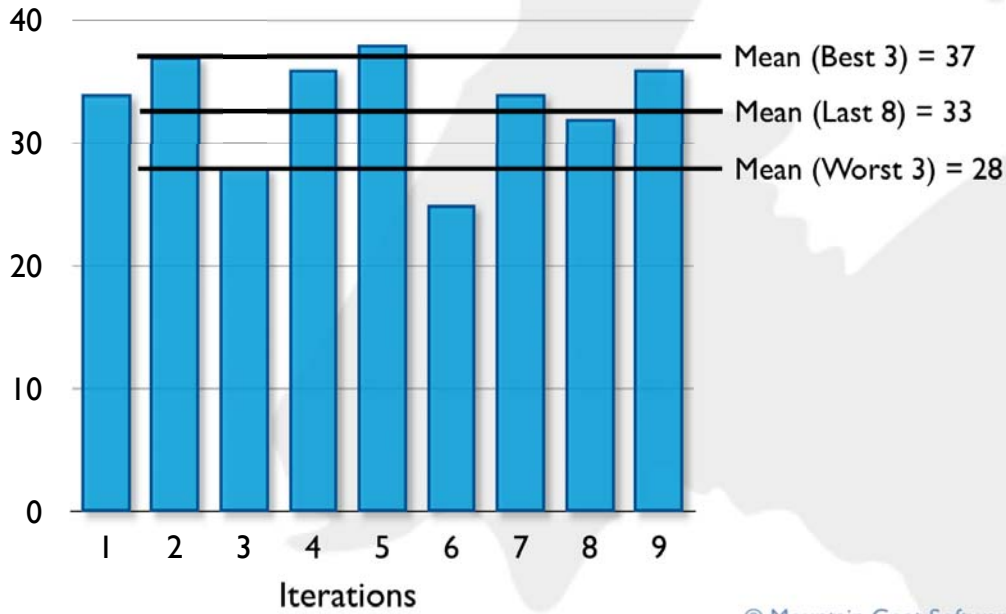
Iteration 1

Iteration 2

Story A 5	Story F 5
Story B 8	Story G 1
Story C 3	Story H 13
Story D 5	Story I 5
Story E 1	Story J 8



# Updating the release plan



# Extrapolate from velocity



Assume:  
There are five  
iterations left.

← At our slowest velocity we'll finish here ( $5 \times 28$ )

← At our long-term average we'll finish here ( $5 \times 33$ )

← At our best velocity we'll finish here ( $5 \times 37$ )



# Fixed-date planning

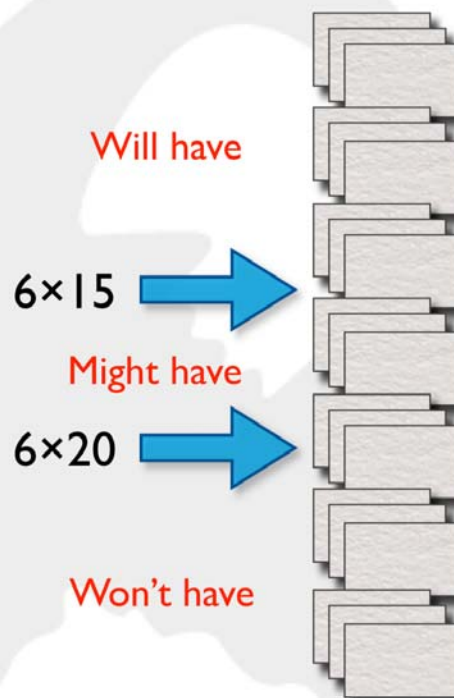
How much can I get by <date>?

1. Determine how many iterations you have
2. Estimate velocity as a range
3. Multiply low velocity  $\times$  number of iterations
  - Count off that many points
  - These are “Will Have” items
4. Multiply high velocity  $\times$  number of iterations
  - Count off that many more points
  - These are “Might Have items”



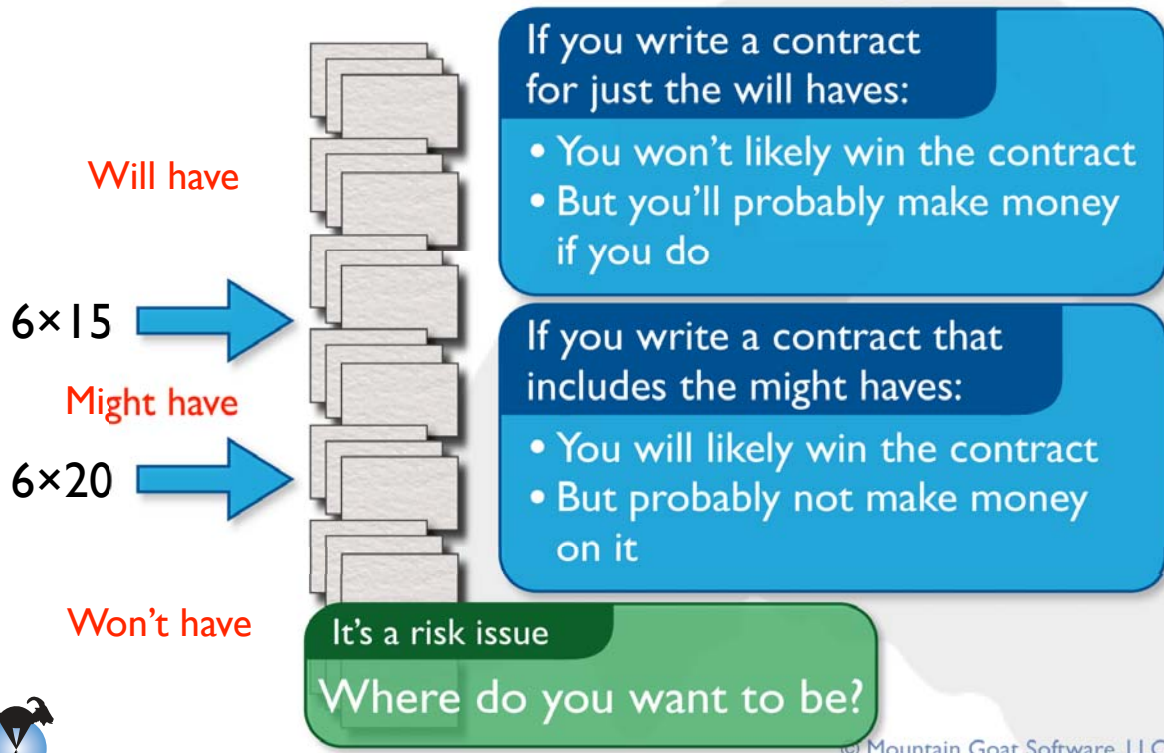
## Fixed-date planning: an example

Desired release date	30 June
Today's Date	1 January
Number of iterations	6 (monthly)
Low velocity	15
High velocity	20





# Fixed-date contracting



33

# Upcoming public classes

Date	What	Where
March 1–2 March 3–4	Certified ScrumMaster Succeeding with Agile	Boulder
April 12 April 13–14 April 15	Effective User Stories Certified ScrumMaster Agile Estimating and Planning	La Jolla
May 17–18 May 19–20	Certified ScrumMaster Succeeding with Agile	Sunnyvale
July 19–20 July 21–22	Certified ScrumMaster Certified Scrum Product Owner	Orlando

Information and registration at  
[www.mountaingoatsoftware.com](http://www.mountaingoatsoftware.com)

© Mountain Goat Software, LLC

34

# Mike Cohn contact info

Mike Cohn

mike@mountaingoatsoftware.com

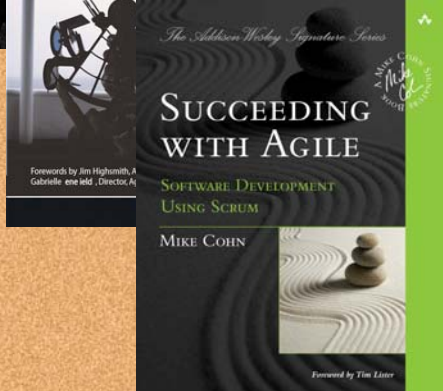
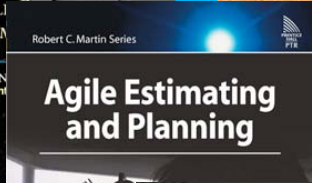
www.mountaingoatsoftware.com

twitter: mikewcohn

(720) 890-6110



MOUNTAIN GOAT  
SOFTWARE



© 2003-2008 Mountain Goat Software®