Agile Estimating and Planning
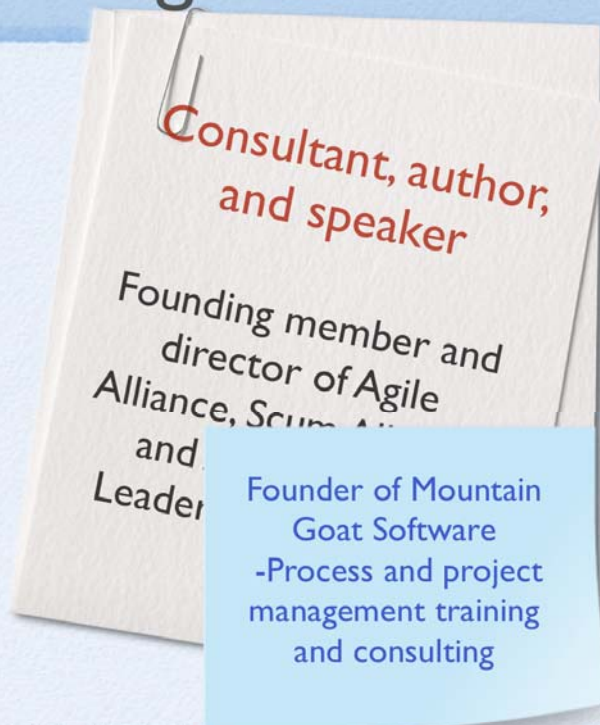
Mike Cohn

Founder

Mountain Goat Software

Boulder, Colorado

mike@mountaingoatsoftware.com

Mike Cohn - background

USER STORIES APPLIED
FOR AGILE SOFTWARE DEVELOPMENT

The Addison-Wesley Signature Series
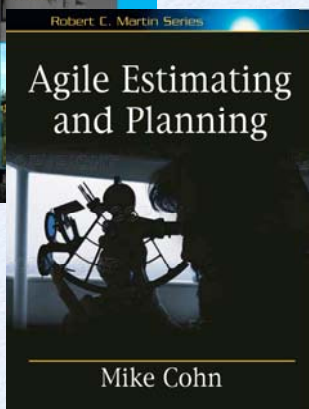
MIKE COHN
Foreword by Kent Beck

Robert C. Martin Series

Agile Estimating and Planning

Mike Cohn

Consultant, author, and speaker

Founding member and director of Agile Alliance, Scum A... and... Leade...

Founder of Mountain Goat Software
-Process and project management training and consulting
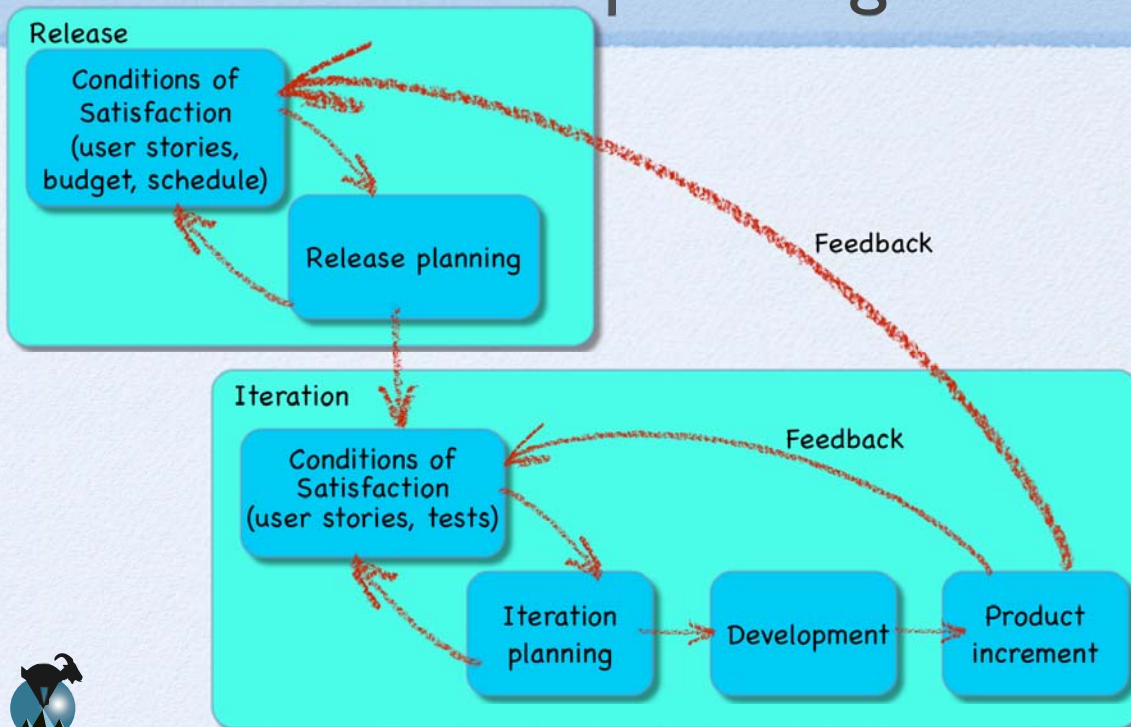
2

Today's agenda

- ☐ Overview
- ☐ Estimating
  - ☐ Story points
  - ☐ Ideal time
- ☐ Techniques for estimating
- ☐ Iteration planning
- ☐ Release planning
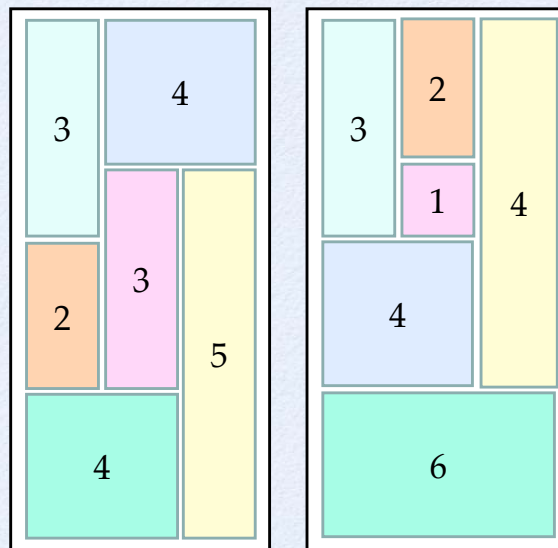- ☐ Estimating velocity
- ☐ Havannah

An overview of planning
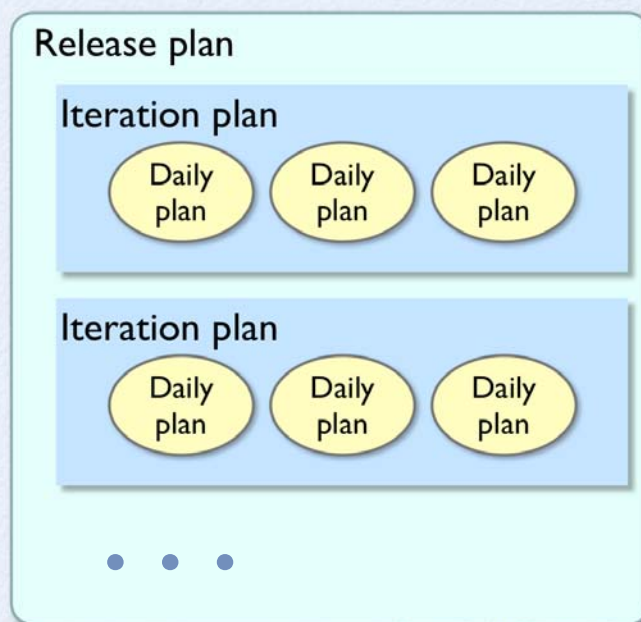
Release, iteration, & velocity

- A release comprises multiple iterations

- Each iteration can be thought of as a same-sized box

- Stories are put into each box until it's full

- The size of the box is the planned velocity

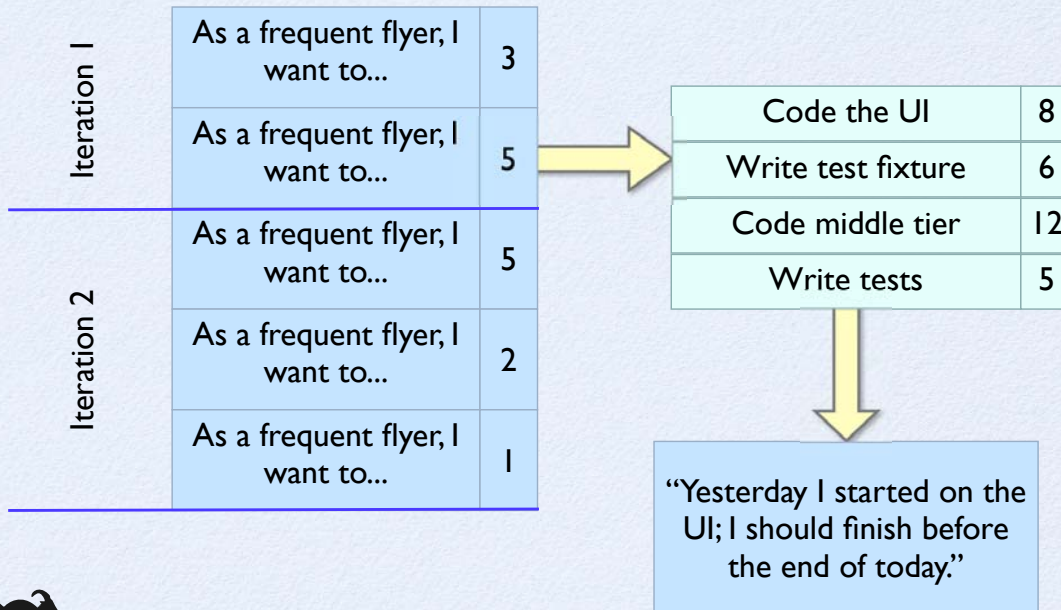|	
3	4
2	3
4	5

3	2	4
	1	
4		
6		

5

Three levels of planning...

Release plan

Iteration plan

Daily plan   Daily plan   Daily plan

Iteration plan

Daily plan   Daily plan   Daily plan

• • •

6

...three levels of precision

	Story	Points
Iteration 1	As a frequent flyer, I want to...	3
	As a frequent flyer, I want to...	5
Iteration 2	As a frequent flyer, I want to...	5
	As a frequent flyer, I want to...	2
	As a frequent flyer, I want to...	1

Task	Estimate
Code the UI	8
Write test fixture	6
Code middle tier	12
Write tests	5

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

7

Today's agenda

- ☑ Overview
- ☐ Estimating
  - ☐ Story points
  - ☐ Ideal time
- ☐ Techniques for estimating
- ☐ Iteration planning
- ☐ Release planning
- ☐ Estimating velocity
- ☐ Havannah

8

How long will it take...

- ...to read the latest Harry Potter book?

- ...to drive to Seattle?

Estimate size; derive duration
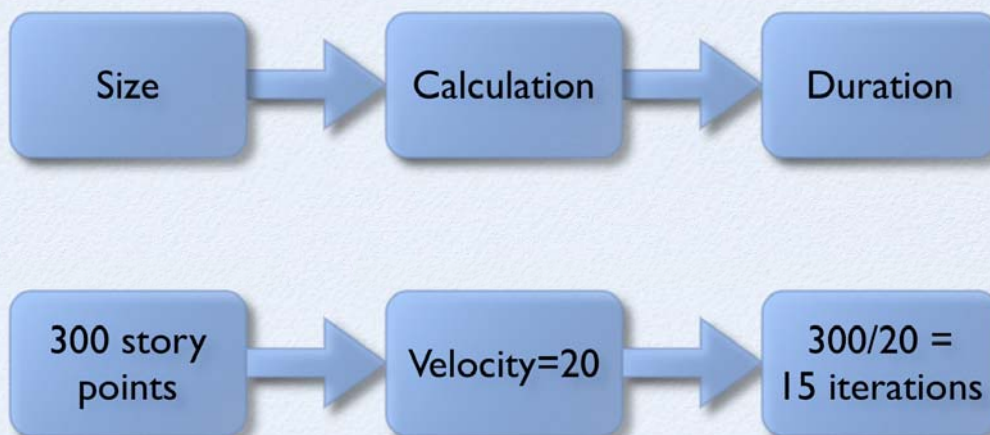
Measures of size

- Traditional and agile measure size differently

Traditional measures of size

Lines of Code
Function Points

Agile measures of size

Story points
Ideal days
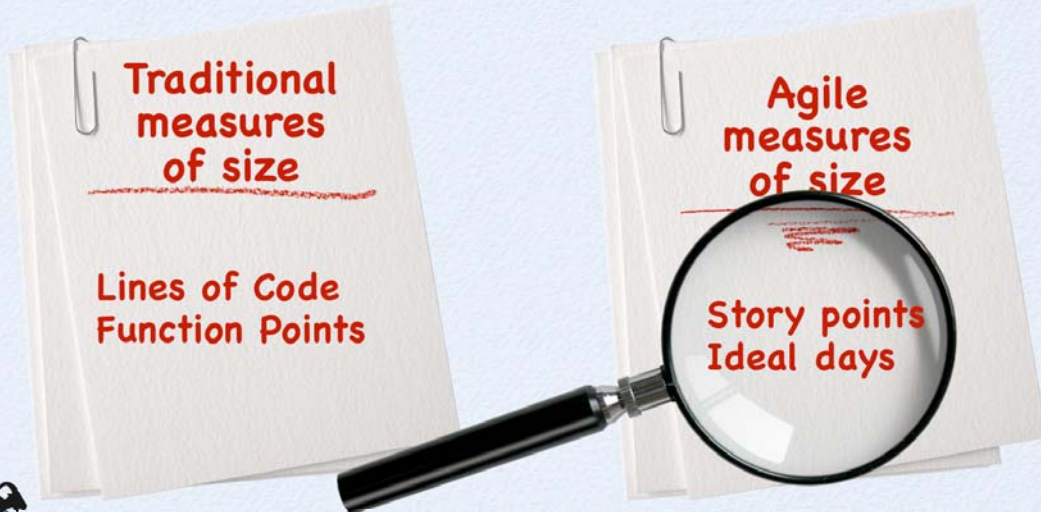
Copyright Mountain Goat Software, LLC

Story points

- The "bigness" of a task
- Influenced by
  - How hard it is
  - How much of it there is
- Relative values are what is important:
  - A login screen is a 2.
  - A search feature is an 8.
- Points are unit-less

As a user, I want to be able to have some but not all items in my card gift wrapped.

5

Copyright Mountain Goat Software, LLC

Dog points

Assign "dog points" to the following breeds

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

Ideal days

- How long something would take if

  - it's all you worked on

  - you had no interruptions

  - and everything you need is available

- The ideal time of a football game is 60 minutes

  - The elapsed time is much longer (3½ hours?)

Elapsed time vs. ideal time

Ideal time vs. elapsed time

- It's easier to estimate in ideal time

- It's too hard to estimate directly in elapsed time

    - Need to consider all the factors that affect elapsed time at the same time you're estimating
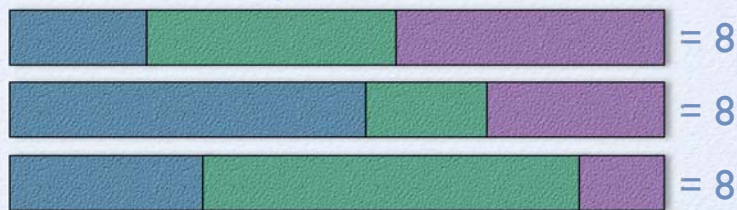
Specialization

- First, don't worry about it too much
  - We're usually better off with fairly rapid, imprecise estimates than spending more time
- Second
  - Just add up the components and report one total estimate of ideal days

= 8

= 8

= 8

Which do you prefer?

1) Do you prefer story points or ideal time?
2) Why?

Advantages to story points

- Story points help drive cross-functional behavior

- Story point estimates do not decay

- Story points are a pure measure of size

- Estimating in story points is typically faster

- My ideal days are not your ideal days

- There are studies that show we are better at relative estimating

Advantages to ideal days

- Ideal days are easier to explain outside the team

- Ideal days are easier to estimate at first

What I usually do

- Try to use story points if the team is willing
- If not
  - Start with ideal time
    - Gives the team a nice foundation for the initial stories
    - Helps team get started
  - Define "1 story point = 1 ideal day"
- Then
  - Gradually convert team to thinking in unit-less story points
  - "This story is like that story."

Today's agenda

- ☑ Overview
- ☑ Estimating size
  - ☑ Story points
  - ☑ Ideal time
- ☐ Techniques for estimating
- ☐ Iteration planning
- ☐ Release planning
- ☐ Estimating velocity
- ☐ Havannah

Estimate by analogy

- Comparing a user story to others

  - "This story is like that story, so its estimate is what that story's estimate was."

- Don't use a single gold standard

- Triangulate instead

  - Compare the story being estimated to multiple other stories

Triangulation

- Confirm estimates by comparing the story to multiple other stories.
- Group like-sized stories on table or whiteboard
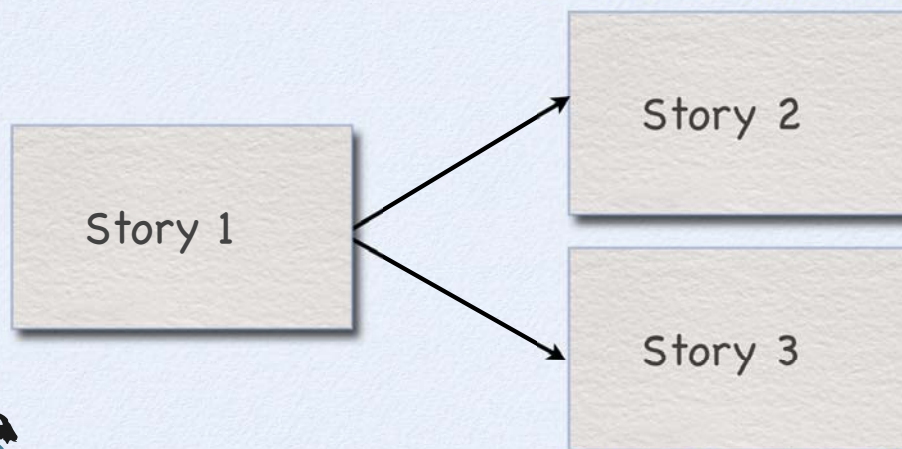
How much effort?

- A little efforts helps a lot

- A lot of effort only helps a little more
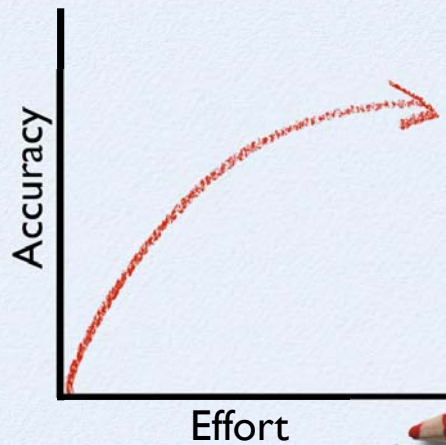
Use the right units

- Can you distinguish a 1-point story from a 2?

- Can you distinguish a 17 from an 18?

- Use units that make sense, such as

  - 1, 2, 3, 5, 8

  - 1, 2, 4, 8

- Stay mostly in a 1-10 range

Include 0 and ½ if you want

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

Planning poker - an example
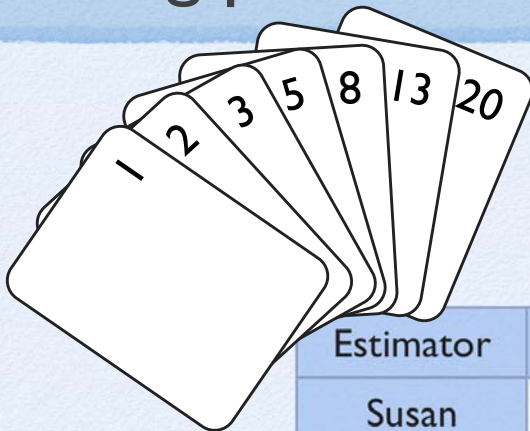
Estimator	Round 1	Round 2
Susan	3	5
Vadim	8	5
Ann	2	5
Chris	5	8

Remodeling my kitchen

1. Install new hardwood floor
2. Refinish (remove, sand, repaint) the cabinets
3. Install granite countertop instead of tile
4. Repaint entire kitchen
5. Lay shelf paper
6. Install recessed lighting
7. Replace electric stove with gas stove
8. Install built-in refrigerator
9. Install a new oven
10. Plumb the island and add sink
11. Replace simple window with a bay window

Today's agenda

☑ Overview
☑ Estimating size
  ☑ Story points
  ☑ Ideal time
☑ Techniques for estimating
☐ Iteration planning
☐ Release planning
☐ Estimating velocity
☐ Havannah

Iteration planning

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	

Iteration 1 / Iteration 2

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

Creating this is iteration planning

"Yesterday I started on the UI; I should finish before the end of today."
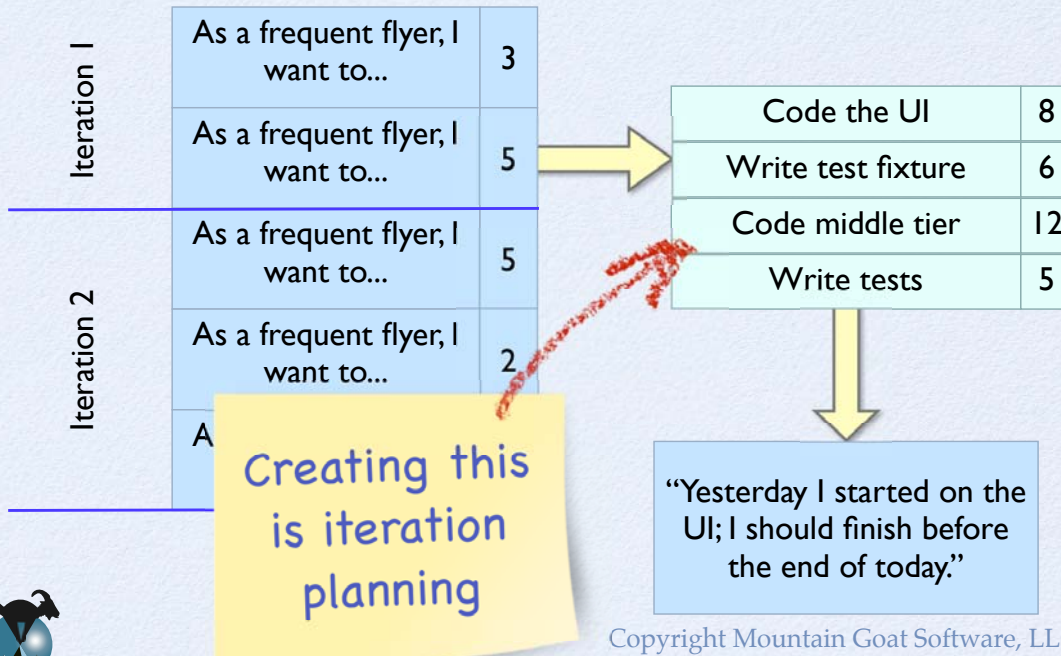
Two approaches

- Velocity-driven iteration planning

- Commitment-driven iteration planning

Commitment-driven iteration planning

- Discuss the highest priority item on the product backlog
- Decompose it into tasks
- Estimate each task
  - Whole team estimates each task
- Ask ourselves, "Can we commit to this?"
  - If yes, see if we can add another backlog item
  - If not, remove this item but see if we can add another smaller one
- No one signs up for specific tasks yet

Start by thinking about availability

- Estimating available hours

Person	Hours per Day	Hours per Iteration
Sergey	4-6	40-60
Yuri	5-7	50-70
Carina	2-3	20-30
Total		110-160

Then commit to what fits

Code the ...	12
Do...	8
Docur 48 e...	8
Test the...	8
Do the...	4
Document the...	8

20	20

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

Code the UI	8
Write ... ture	6
Code 31 tier	12
Write tests	5

50	50

Is this a problem?

3 4
2 3
   5
4

--Code the abc class (8 hours)
--Code the user interface (4)
--Write test fixtures (4)
--Code the xyz class (6)
--Update performance tests (4)

--Prototype the UI (8 hours)
--Demo UI to 3 outside users (3)
--Code new UI (12)
--Update documentation (3)
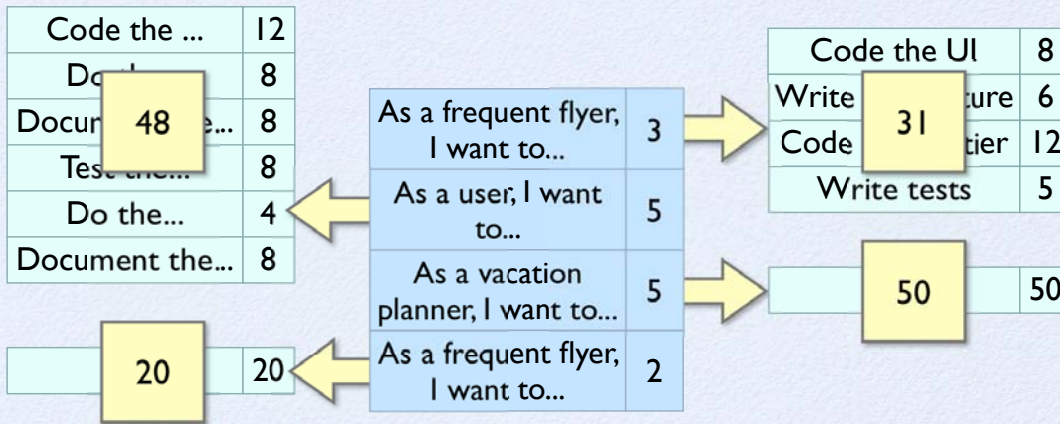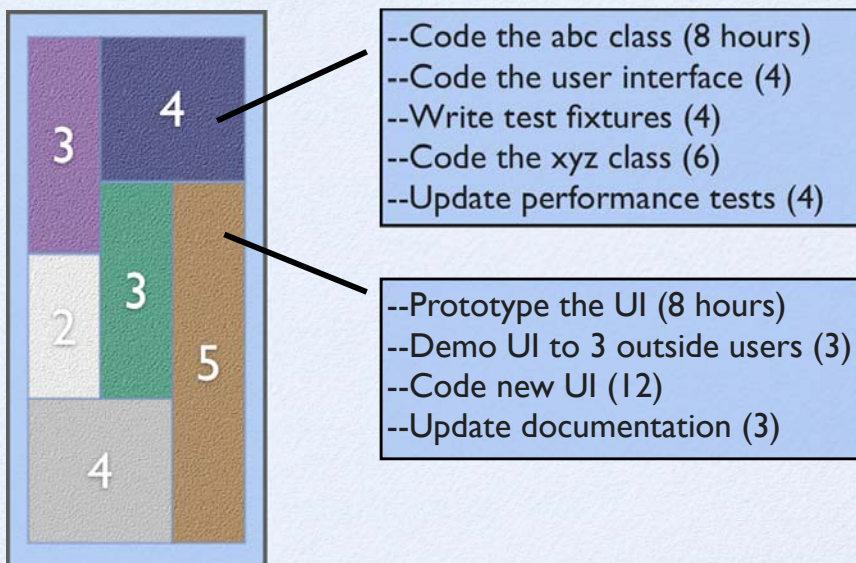
Take items in priority order (mostly)

- Take items based on the order defined by the product owner

- But:

  - Pay attention to possible synergies with (slightly) lower priority items

- Typical iteration may work on items 1, 2, 3, and 8

37

Today's agenda

- ☑ Overview
- ☑ Estimating size
  - ☑ Story points
  - ☑ Ideal time
- ☑ Techniques for estimating
- ☑ Iteration planning
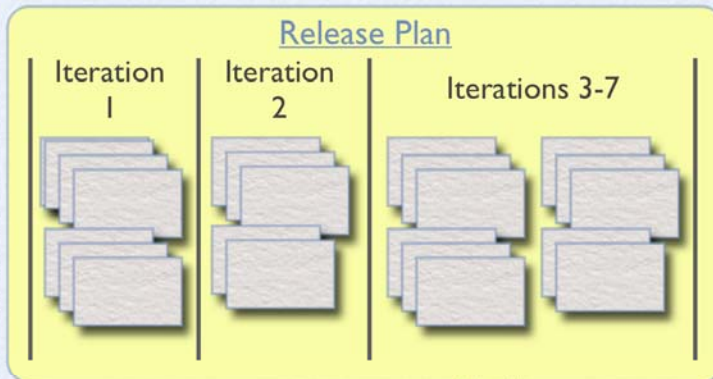- ☐ Release planning
- ☐ Estimating velocity
- ☐ Havannah

38

The release plan

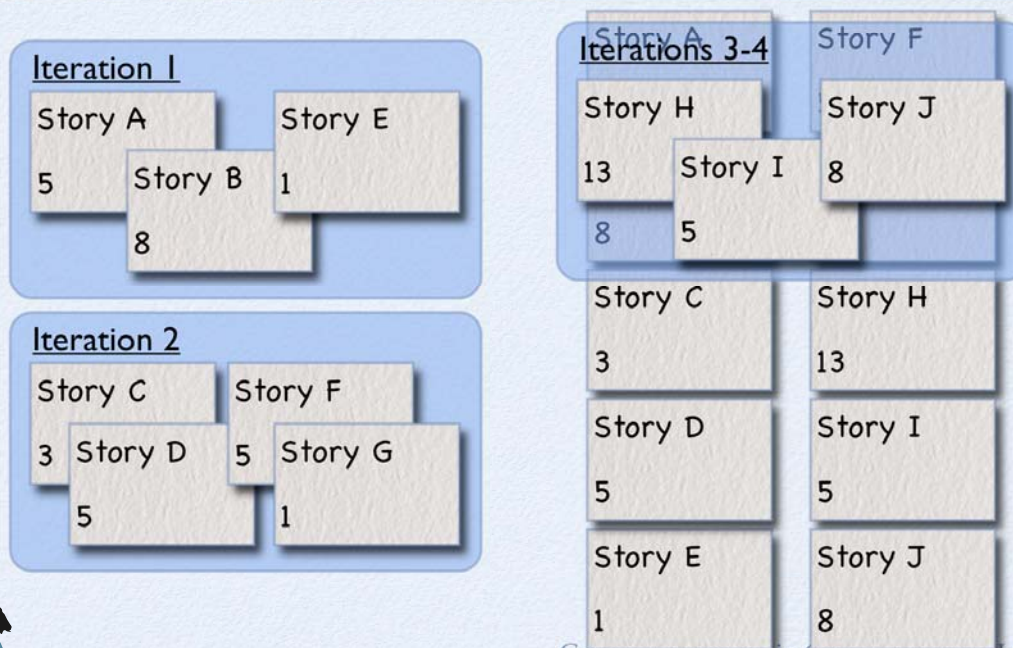- We can create a release plan from
  - the size estimate given to each story
  - the team's velocity
- Shows what will be worked on in each iteration

Release Plan

Iteration 1	Iteration 2	Iterations 3-7

39

An example with velocity=14

Iteration 1

Story A
5
Story B
8
Story E
1

Iteration 2

Story C
3
Story D
5
Story F
5
Story G
1

Iterations 3-4

Story A

Story F

Story H
13
Story I
8
Story J
8

Story C
3
Story H
13

Story D
5
Story I
5

Story E
1
Story J
8

40

Changing the release plan

Story A	5
Story B	3
Story C	5
Story F	3
Story D	5
Story E	5
Story G	3
Story I	3
Story H	5
Story J	2
Story K	5
Story L	3

	Story A	5
✓	Story A	5
✓	Story B	3
✓	Story C	5
	Story F	3
	Story D	5
	Story E	5
	Story G	3
	Story I	3
	Story H	5
	Story J	2
	Story K	5
	Story L	3

41

Today's agenda

- ☑ Overview
- ☑ Estimating size
  - ☑ Story points
  - ☑ Ideal time
- ☑ Techniques for estimating
- ☑ Iteration planning
- ☑ Release planning
- ☐ Estimating velocity
- ☐ Havannah

42

How to estimate velocity

Use historical values

Don't, until you've run 1-3 iterations

Forecast it

43

Forecasting velocity

- Just like commitment-driven iteration planning

- Plan an iteration like normal

  - Remember: velocity plays no role in commitment-driven iteration planning

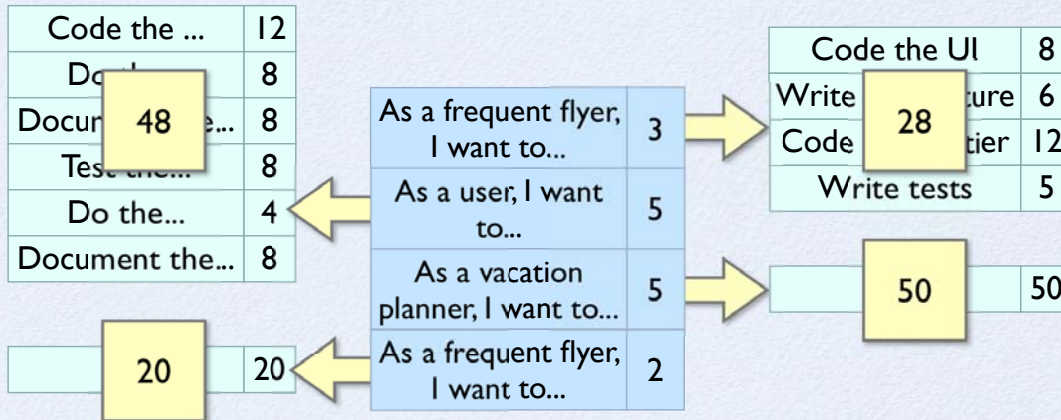- When done, add up the velocity of the stories committed to for the iteration

Ideally, "plan" more than one iteration

44

What's their velocity?

Code the ...	12
Do...	8
Docu 48 e...	8
Test...	8
Do the...	4
Document the...	8

Code the UI	8
Write 28 ture	6
Code tier	12
Write tests	5

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

20	20

50	50

If they committed to the first three stories, what's their planned velocity?

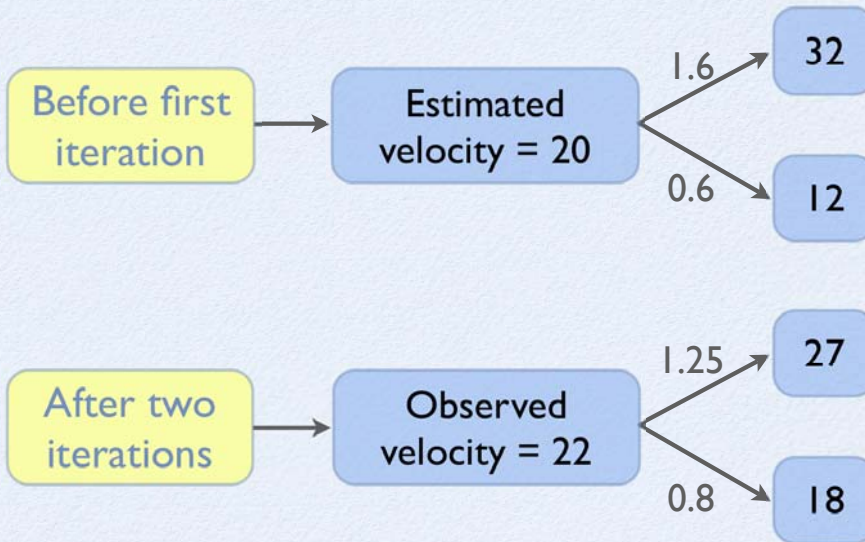Copyright Mountain Goat Software, LLC

45

Put a range around it

- You're unlikely to have precisely forecasted the exact velocity the team will average

- So, put a range around your estimate

  - I suggest plus and minus from 10% to 60%

Copyright Mountain Goat Software, LLC

46

Expressing velocity as a range

Before first iteration → Estimated velocity = 20 — 1.6 → 32 — 0.6 → 12

After two iterations → Observed velocity = 22 — 1.25 → 27 — 0.8 → 18

Today's agenda

☑ Overview
☑ Estimating size
  ☑ Story points
  ☑ Ideal time
☑ Techniques for estimating
☑ Iteration planning
☑ Release planning
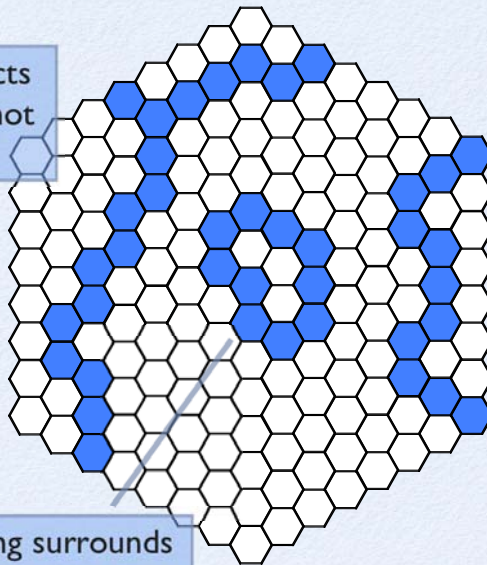☑ Estimating velocity
☐ Havannah

Havannah

A fork connects three edges (not corners)

A bridge connects any two corners

A ring surrounds at least one space (occupied or not)

User stories

1. As a new player, I can play against a medium-strength computer opponent.
2. As an experienced player, I can play against a strong computer opponent.
3. As a player, I can save and restore a game.
4. As a player, I can use the program to play against another human on my computer.
5. As a player, I'd like to be able to choose between a wooden board and pieces and a metal board and pieces.
6. As a player, I'd like to ask for a hint.
7. As a player, I want to place a piece on the board using either my keyboard or my mouse.
8. As a player, I'd like to undo and redo moves.
9. As a new player, I want access to an online help system.
10. As a player, I want all pieces of the winning shape to blink or glow so that I can see the winning shape.
11. As a new player, I'd like to be warned after making a horrible move and be given the chance to take it back.

Planning Havannah
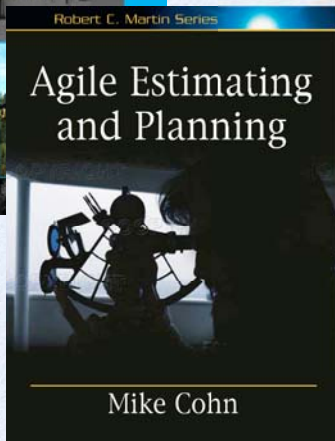
1) Estimate all of the Havannah stories
2) Assume a small team of 3-4 people
3) Plan their first iteration
      (Stories #3, 4, and 7)
4) Forecast their velocity
      Remember to make it a range
5) Develop the release plan

51

Mike Cohn contact info

USER STORIES APPLIED
FOR AGILE SOFTWARE DEVELOPMENT
MIKE COHN
Foreword by Kent Beck

Robert C. Martin Series
Agile Estimating and Planning
Mike Cohn

mike@mountaingoatsoftware.com

www.mountaingoatsoftware.com

(720) 890-6110 (office)

(303) 810-2190 (mobile)

MOUNTAIN GOAT
SOFTWARE

52