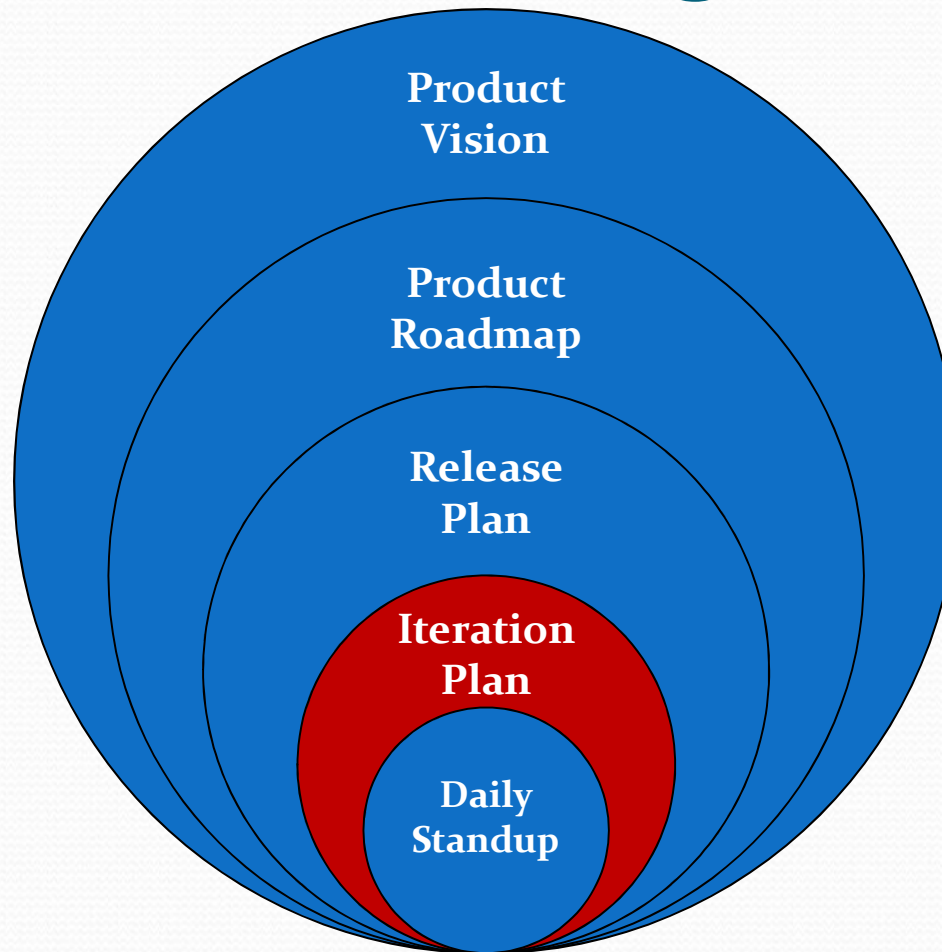


Iteration Planning



5 Levels of Planning

Adapted from "5 Levels of Agile Planning" by Hubert Smits



Iteration Planning

- Define scope as a team
- Define a clear understanding of “done”
- Plan just enough that you can commit



Roles

- Product Owner
- Scrum Master
- Team Member



Product Owner

- Prioritizes the backlog
- Communicates what is important ... and what is not
- Is a proxy for the customer and other stakeholders



Scrum Master

- Responsible for the process
- Facilitates the meeting



Team Member

- Asks questions
- Collaborates with others
- Signs up for work



The Backlog

- A ranked list of stories
- What is a story?
 - A scenario that we must do work to implement which results in business value
 - Typically in the form of: “As a <type of user>, I want <feature> so that <business value>”
 - Good stories meet the INVEST criteria



Before you Start



- Well Groomed Product Backlog
 - Prioritized
 - Estimated
- Iteration Theme/Goal

1.0 Iteration 9 All Owners Not Done Add Existing Create Update Delete View Surveys

Refresh Log out

Velocity Used: ▶ 3.5 of 3.7 (95%) - Planigle

Name	Owner	Effort	Status	Public	Priority	User Priority	
User accepts license agreement	Walter Bodwell	1.5	In Progress	true	1	4	
User maps iterations to release	Walter Bodwell	2	Created	true	2	6,333	

Estimated Prioritized

Exercise 1



- Create a prioritized backlog
 - As a <user> I want <feature> so that <business value>
 - Estimate relative size
- At least enough for one iteration
- Choose any domain you like
- We'll use the results in a future exercise

What's your goal for the iteration?

A Typical Iteration Planning Session

- Discuss Logistics
- Review Iteration Goals
- Understand the Stories
- Task Out the Stories
- Commit

Typical Duration: 3-4 hours

Attendees:

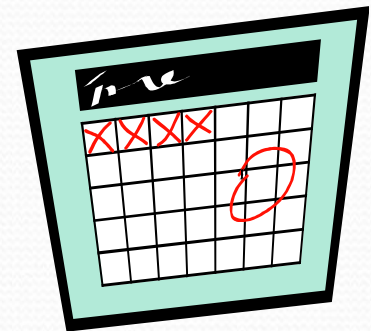
- Product owner
- Scrum master
- Delivery team

Materials:

- Stories (cards or online)
- Task planning material (cards, whiteboard, online)
- Planning/estimation materials (e.g. planning poker cards)

Discuss Logistics

- Review Historical Velocity
- Review Team Availability
 - Holidays / Vacations
 - Meetings
 - L3 Support, outside commitment, etc
- Review the Definition of Done



Definition of Done

- You need to define for your environment
- Definition will evolve over time
- Example:
 - Unit tests written and passed
 - Acceptance tests automated and passed
 - User facing documentation written
 - Checked in to the build
 - No defects introduced



Review Iteration Goal(s)



- Product Owner
 - Explain the Goal (theme)
 - Make priority adjustments based on feedback from delivery team
- Team Members
 - ASK QUESTIONS
 - Understand the Goal, not just the desired features

Understand the Story



- Product Owner
 - Explain the Story
 - Explain the “Why” (“as a <role> I <what> so that <WHY>”)
 - Break down as needed
 - Elaborate on acceptance criteria/tests
 - Make priority adjustments based on feedback from team
- Team Members
 - Understand the story
 - Understand and question the acceptance criteria (how will you build a test for each? What about...)
 - Validate the size/implementability

Acceptance Criteria

- What is required for the success of this story?
- Typically determined at iteration planning jointly between product owner, dev, QA, writers, etc.



Task out the Story

- Define tasks
- Estimate the work involved
- Validate capacity again



The Product Owner can help
in avoiding less valuable work

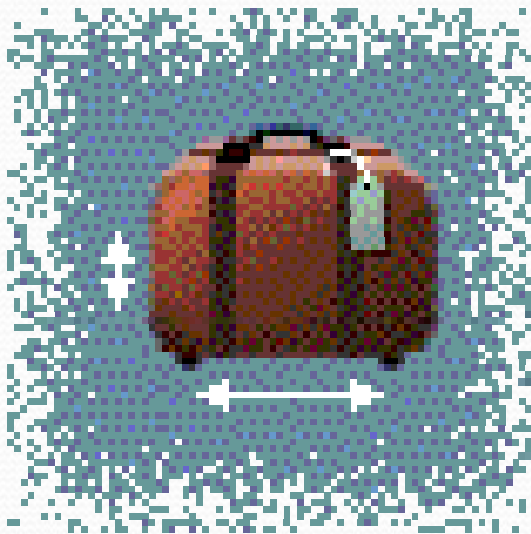
Hold Off On Names

- Keeps everyone focused on all the tasks, not just theirs
- Encourages team commitment
- Within the iteration, encourages focus on priorities
- And teamwork



Repeat

- Until the team cannot take on more
- Split stories as necessary








Splitting a Story

- The closer to the present a story is, the smaller it will become
- Those for this iteration need to fit within the iteration
- When splitting a story, each “slice” should add incremental user value



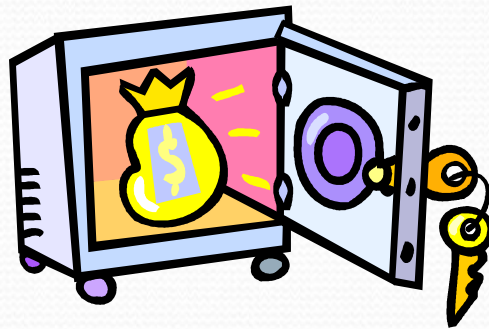
Commit



- Everyone agrees the iteration is doable
- Use disagreement and uneasiness in team members to drive out hidden risks, tasks, and issues
- Drive agreement with a fist of five
 -  Absolutely, no question
 -  I think this is good and will make it happen
 -  I can support this
 -  I'm uneasy about this and think we need to talk about it more
 -  Let's continue discussing this idea in the parking lot

Effective Meetings

- Everyone should be focused on the task at hand
 - No working on laptops
- Every minute should be valuable
- If not, figure out how to make it so



Tools

Stories									
1.1	Iteration 4	Team A	All Owners	All Statuses					
Number of Stories: 5		Velocity Allocation: 24 of 19.33 (124%) - Team A		Utilization: 23 of 19 (121%) - Team A					
Name	Owner	Size	Time	Status	Public	Rank	User Rank		
- User searches for books by author, title or ISBN number	Sue Tester	8	7	In Progress	true	1	1		
- Search by title showing just titles	Bob Developer	2		In Progress					
- Add more details to results	Bob Developer	2		Not Started					
- Add search by author or ISBN	Bob Developer	1		Not Started					
- Test search	Sue Tester	2		Not Started					
+ User views detailed information on a book	Sue Tester	5	5	In Progress	true	2	4		
+ Administrator adds new books to site	Sue Tester	5	5	Not Started	true	6	5		
+ Administrator deletes book	Sue Tester	3	3	Not Started	true	7	6		
+ Administrator edits existing book info	Sue Tester	3	3	Not Started	true	8	7		

Tasks	Mon	Tues	Wed	Thurs	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	4	
Test the middle tier	8	16	16	11	8
Write online help	12				
Write the foo class	8	8	8	8	8
Add error logging			8	4	

I1
Pebbles

Sprint 3 for MultiVue Install			IT Team Tasks				
Task description	Commit	Status	days remaining				
			9/27	9/28	9/29	9/30	10/1
Requirements Component			256	0	0	0	0
Project Requirements Gathering	Campbell	Not started	16				
Formal Requirements Documentation	Campbell	Not started	8				
MultiVue Configuration Component			16				
Append Additional Demographics	Campbell	Not started	16				
SAP database Component			16				
Design SAP Database	Campbell	Not started	16				
Creation of the SAP Database	Campbell	Not started	4				
Create stored procedures on SAP database	Campbell	Not started	12				
SAP Code Component			16				
Creation of SAP .NET Component	Jan	Not started	16				
Creation of SAP Web Application	Jan	Not started	16				
SAP Security			24				
Creation of Security Administration Site	Campbell	Not started	24				
Secure Messaging	Campbell	Not started	12				
Security Integration	Jan	Not started	12				
SAP system testing	Campbell	Not started	8				
SAP System Verification	Campbell	Not started	8				
SAP Hardening			16				
Bug Fixing / Cosmetic Changes	Mark	Not started	16				
Install in Live Environment	Campbell	Not started	16				
BizTalk 2004 Component			4				
Extend ePEX-3 Adaptor	Campbell	Not started	4				
Extend Sw ift Adaptor	Mark	Not started	8				
Extend Upstream Schemas	Mark	Not started	8				
Create AIC schema	Campbell	Not started	4				
Create Mappings	Campbell	Not started	16				
Create SAPAIC	Mark	Not started	16				

Exercise 2

- Do iteration planning
- Go through stories in priority order
- Create acceptance criteria
- Task out
- Stop when you can't do more
- Commit



Do you believe in your result?

Estimating

- Identify a medium sized story that is well understood; call it a 5
- Now estimate other stories relative to that
- Is it about the same, $\frac{1}{2}$ as difficult, twice as difficult?
- Use Fibonacci numbers: 1, 2, 3, 5, 8, 13, 21
- If bigger than that or if too hard to estimate, split the story
- Tackle as a team; Planning poker can help (www.planningpoker.com)

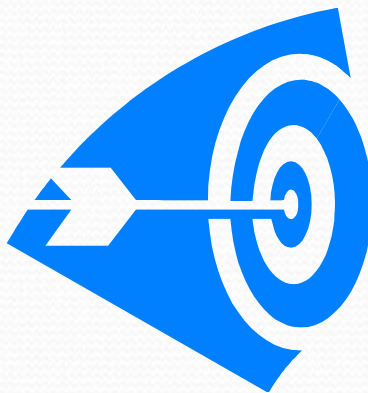
Why Story Points?

- Time estimates
 - Vary by person
 - Encourage padding
 - Tend to grow stale
- Story points
 - More consistent from person to person
 - Not a commitment to time frame
 - Don't change as much
 - Easier to estimate relative size



Velocity

- Now that stories have sizes, you can track how many points you typically get done in an iteration
- You can now use this to predict future completion rates



Release Planning Deliverables

- Plan for each Iteration
- Assumptions
- Dependencies
- Risks



- Are things synched up across teams?
- Are you attacking the most important stories?
- Does the team believe in the results?

Coordinating Teams

- Simplest if one team has the skills to take on an item by themselves
- If not, try to minimize the gap
 - Within the same iteration is ideal
- Touch base before and after iteration planning
- Daily scrum or scrum meetings can help



Kanban

- Instead of planning it all up front, you can pull things in as you go
- Keep iterations (Scrumban) or not (pure Kanban)
- Advantages
 - More flexibility (great for start ups and support)
- Disadvantages
 - Less predictability
 - Harder to coordinate

Questions?

Walter Bodwell
Planigle

wbodwell@planigle.com

Twitter: @wbodwell

www.planigle.com

www.walterbodwell.com

