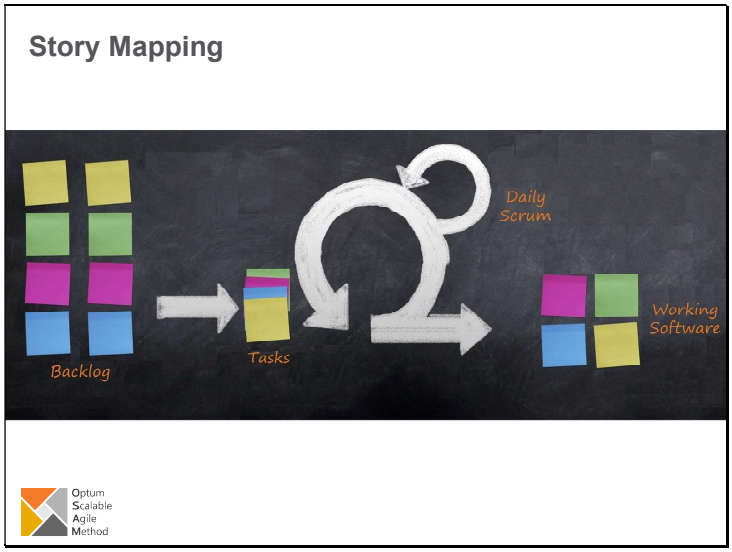
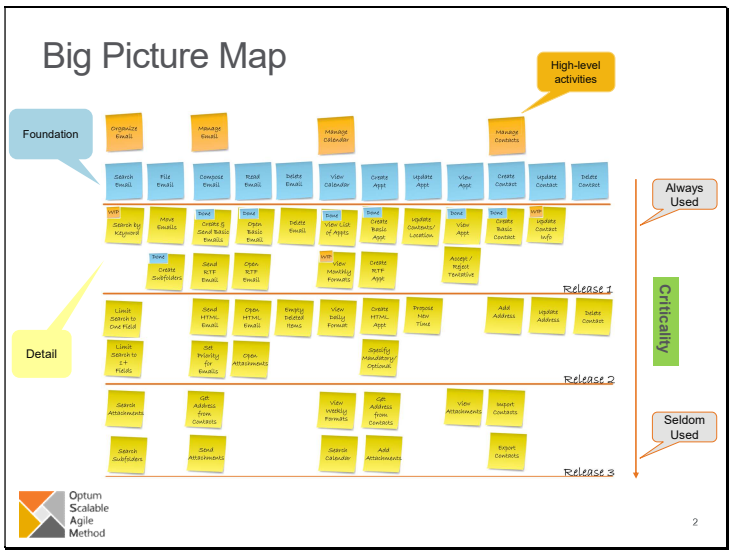
Slide 1



Welcome to the self-study Story Mapping module

Slide 2



The above graphic is an example of a completed story map.

Rows:
Each row, from left to right, shows the narrative flow. It visualizes the sequence a user may follow when interacting with an email system. Of course, any specific user might choose to do different things in a different order. You must have extended conversations with stakeholders to flesh-out details and variations.

Columns:
Each column, from top to bottom, shows the specific backlog hierarchy and the prioritization for each goal. It displays the organization of larger goals into smaller work items.

Criticality:
As seen on the right, the criticality, or priority, is depicted from the top of the map to the bottom of the map. Typically, we would see user actions that are always or frequently used toward the top of the map and user actions that we project will be infrequently used toward the bottom of the map.

Foundation:
The blue cards show your initial higher-level goals and establish the basis of your solution

Detail:

The smaller yellow cards shown on the map show smaller pieces of each foundational goal.

Releases:
When organizations decide to release an increment of value to customers is highly subjective. Some solutions gain maximize value by deploying to market frequently, (e.g., multiple times per day, daily, weekly, monthly, etc.). Some solutions satisfy customer expectations best with less frequent releases (e.g., quarterly, annually, etc.). Regardless of your release cadence, a story map will help you to decide *what* to release in each increment.

Slide 3

Story Map Characteristics

- Is dynamic
- Exposes entire system's intention
- Helps visualize product's progress
- Displays the relationship of higher-level goals to lower-level goals
- Facilitates value driven releases

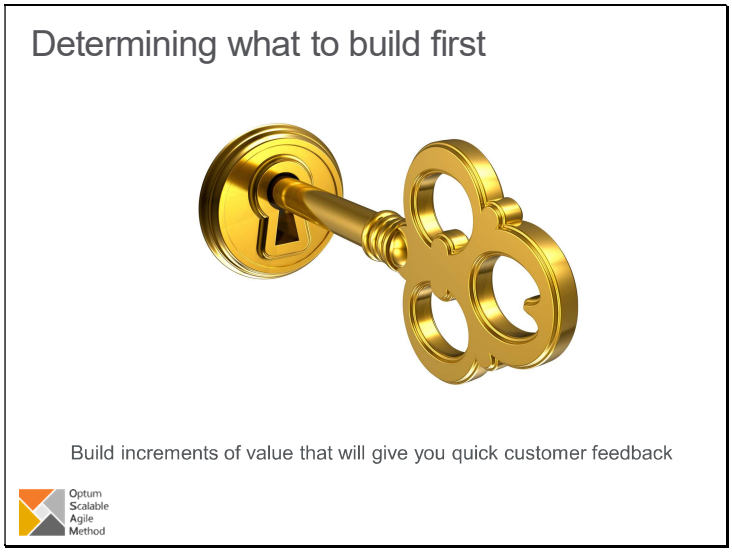
Optum Scalable Agile Method

3

A story map:

- Is dynamic, meaning that it will evolve and change over time as we learn more about the users of the system.
- Evolves with your understanding of your users and your product solution
- Exposes the entire system's intention. It illustrates what your system will do
- Shows us goals arranged in the order they're needed by people and business processes
- Facilitates value driven releases with an understanding of what to build first

Slide 4



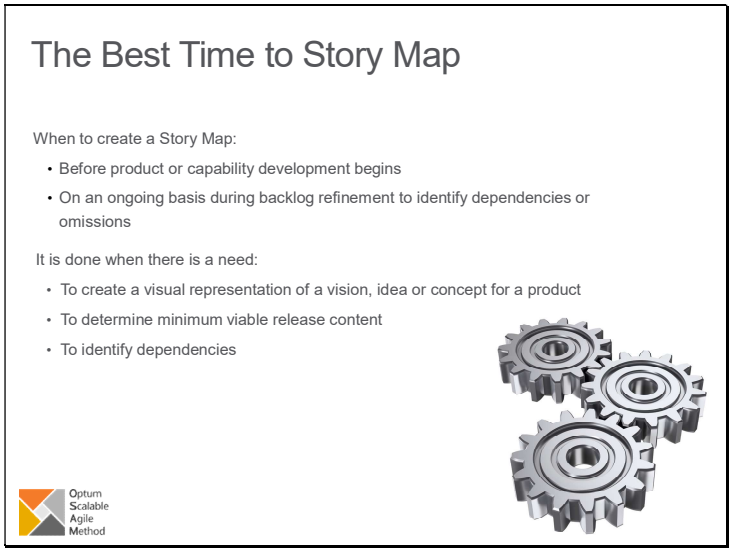
How do we determine what to build first, then second, then third, and so on?

A traditional product development approach would be to build out all the intended functionality for a given idea, and then release the entire solution to market. This could take the implementation team anywhere from six months to several years to complete. This delays valuable feedback, and often leads to teams building the wrong thing, or building the thing wrong. As a result, this type of development practice has a high risk of failure. The preferred approach to building everything in a single release, is to build just the bare minimum amount of functionality, incrementally. With each increment, you will be able to gather value customer feedback, then apply that feedback to future development. This ensures that development efforts continually align to customer needs and wants.

To minimize the cost of change, the core agile practice of building *increments* of value, in order to gather fast feedback and quick return on investment, should be the backbone of your specific build and deployment plan.

To do that, you will take your idea, do the necessary market research and cost-benefit analysis, and then create your vision, and then your backlog. To maximize value, you will need to slice up your idea into small increments of value, starting with the most basic, key functionality required (maybe even just a working prototype), and then build upon it based on feedback from actual customers.

Slide 5



There is no perfect time to do story mapping – that's why it's ongoing. The circumstance around when to do it are when there is a need for elaborating on a vision, idea or concept for a product or system to determine the MVP, MMF, etc. and in identify gaps and dependencies.

IMPORTANT!

To ensure agility, any backlog items that are created in a Story Mapping session should be very HIGH LEVEL. We don't want to spend too much time elaborating our user stories, until we are sure they will actually be implemented.

Slide 6

Story Mapping participation is subjective based on what you're mapping. In some cases, like when you are doing it to elaborate on the product vision – very few participants are needed; the PM, PO, and Stakeholders may be the only people necessary. When more detail is needed, people with other skill sets should be included.

Small teams are better. While it is generally necessary to have a cross-section of people who have different skills and focuses, be sure to keep the overall number of participants small. The more people, the more communication channels. When you add more people, the effectiveness and efficiency starts to decrease.

Slide 7

Before holding a Story Mapping session, make sure you have the following:

Pre-requisites:
A written definition, or vision, of the product or capability you would like to create, that includes:

- The overarching goal you are aiming for; the problem you are trying to solve.
- The expected Business benefits
- Demographic you are targeting
- A clear projection of the value you expect to get from the features in your solution

These items are critical components that the chief stakeholders should review before you move forward with a Story Mapping session.

Face-to-Face Supplies:
When you are ready for a Story Mapping session, if your key members are co-located, you will want to prepare with the following:

- A large room that will fit all the participants comfortably
- Markers
- Painters tape
- Index cards/sticky notes
- Large empty walls, whiteboards, or tables
- Snacks and beverages

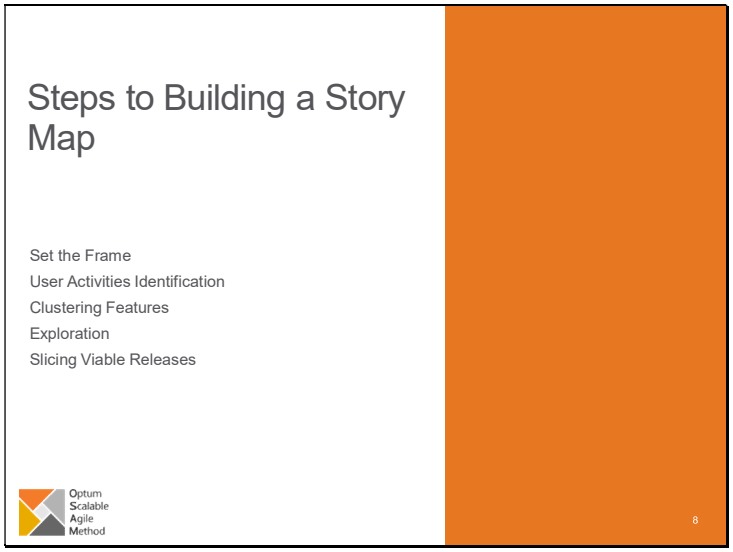
Virtual Supplies:
For teams that are unable to be in person:

- An electronic map can be created using tools such as CA Agile, Visio, or Excel.
- Webex invitations, video and visual virtual mediums and means of communication

If mapping virtually, consider making small groups of teams that work on different areas of the map, and then come together as a whole to review.

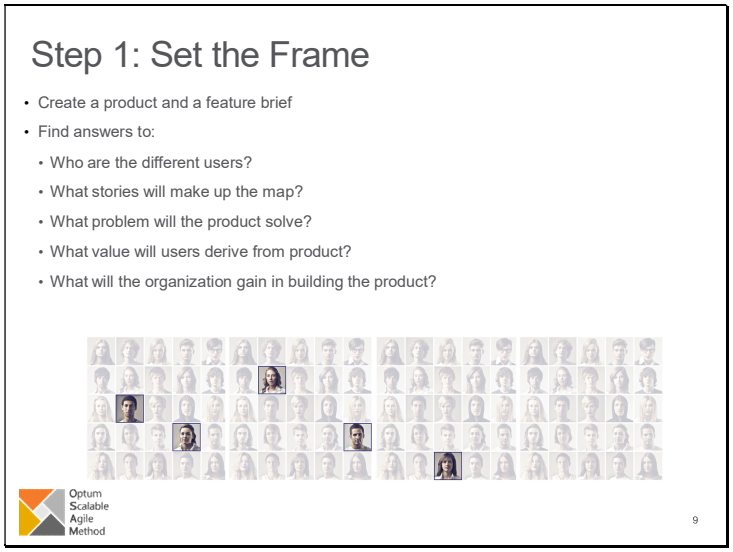
Preservation:
As mentioned previously, a story map is dynamic and should evolve over time. Once your Story Mapping session is complete, the Story Map itself must not set in stone. Therefore, you must preserve your story map in a format that can be easily altered to accommodate valuable change as new information surfaces as work progresses and as the market needs change.

Slide 8



The next section explains the suggested steps to creating a story map

Slide 9



The first step in creating a story map is Framing.

Know your various users.

The following video is an example from UHG of how we can gather and apply user personas and the user journey to offer depth and insight that can improve our work. This type of customer feedback is integral to developing successful capabilities, and must be assessed before and after Story Mapping.

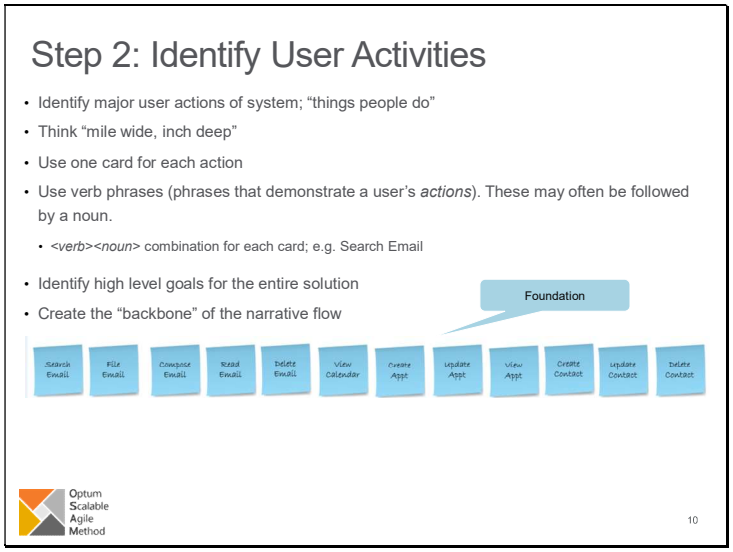
Add this URL to the address bar of your browser to watch and listen to the video: https://vod01.uhc.com/html5/html5lib/v1.7.0/mwEmbedFrame.php/wid/0_9cgugsiv/uiconf_id/14969400/entry_id/0_3ot0scnf (1.5 min)

Setting the Frame: Create a brief of the product and/or the feature(s); this helps constrain the scope of the mapping and sets the vision

Document and get stakeholder feedback on the answers to several key questions such as:

- What problem will your solution solve? Or What features/functions needs to be added to the product
- Who are the different users?
- What value or benefit will users derive from product?
- Why will the organization gain in building the product?

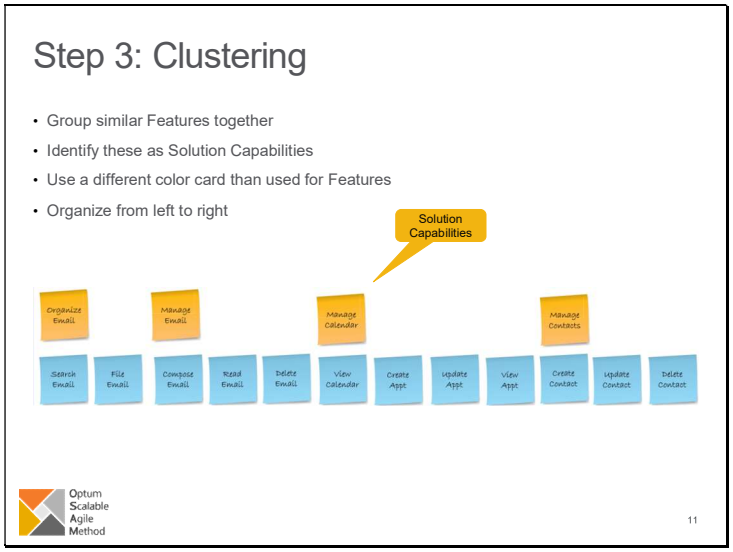
Slide 10



When creating a story map, it is generally easiest to start with the core user goals. You should capture these core goals on "foundation" cards. These cards represent user activities, or tasks, at a higher goal level. These goals create a foundation that you can then decompose into smaller goals, and also potentially group into higher categories. The foundation cards give your map its initial structure.

The foundation is arranged in a narrative flow that is sequential (when it make sense). The foundation shows how users will interact with your solution from beginning to end, depicted on the map. When creating the foundation, discuss high-level things users will do with your solution. Arrange and re-arrange the activities as needed based on your discussion.

Slide 11

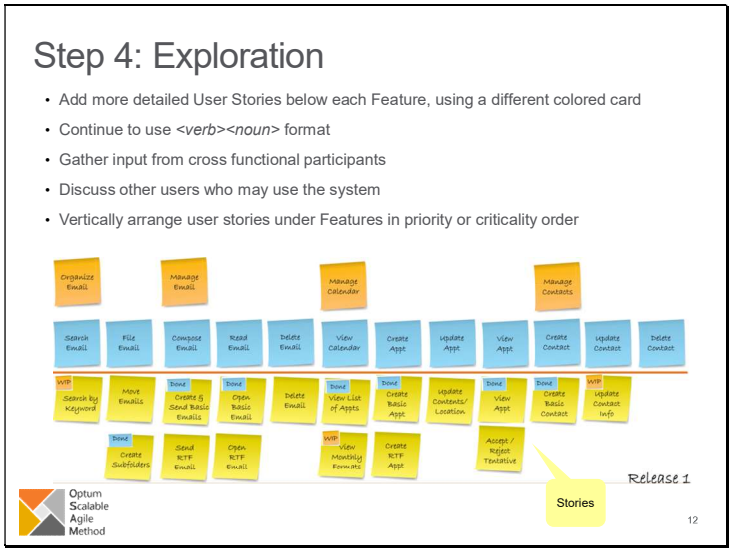


The third step in creating a story map is “Clustering”.
This will help you to group similar Features together. This grouping exercise will flesh-out your map and help you to visualize how each piece fits together in the larger context. This will also help see various release possibilities, so you can discuss and make the most value-based decisions on what to build when..

Clustering involves the following:

- Group similar features together - move things that are similar to each other closer to each other and things that are dissimilar should be moved farther apart
- Depending on your planning horizon, these may become “solution capabilities”
- Use a different color card for the higher-level backlog items
- Organize the cards from left to right in the order a user would typically complete the actions

Slide 12



The fourth step in creating a story map is to perform exploration.
During exploration, you will

- Add high-level user stories below each Feature
  - Continue to use *<verb><noun>* - use high level descriptions for the user stories (does not need to be in the classic user story format of "As a user…). These <verb><noun> phrases can later be used as good user story titles in your backlog.
- Gather input from cross-functional participants
- Discuss other users who may use the system
- Vertically arrange user stories under Features in priority or criticality order

Slide 13

The fifth step to story mapping is to do a "Slicing of Viable Release."

This includes:
Slicing the map horizontally into releases to users
Prioritize releases in a top down fashion
- Often used user stories should be on top (Release 1) as shown in picture
- Seldom used user stories at the bottom (Release 3) as shown in picture

Use a line to identify slices of tasks that users might use your software for to reach their goals. The smallest number of tasks that allow your specific target users to reach their goal compose a viable product release.

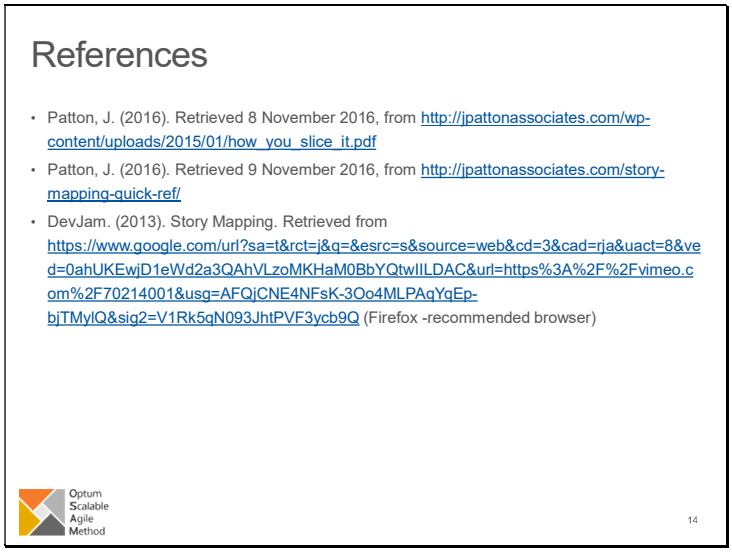
When slicing your releases this way, you are also slicing features into small chunks. For example, taking the Search Email feature. In Release 1, the feature title could be Basic Search Email Function and in Release 2, the feature title could be Enhanced Search Email Function.

The team that build the map did not consider the "Delete Contact" user story necessary to be included in the first Release. They could get a MVP or MMP without it and still have a fully functioning system.

This "horizontal" slicing concept shouldn't be confused with the "vertical" slicing concept we talk about in implementing user stories vertically to account for the User

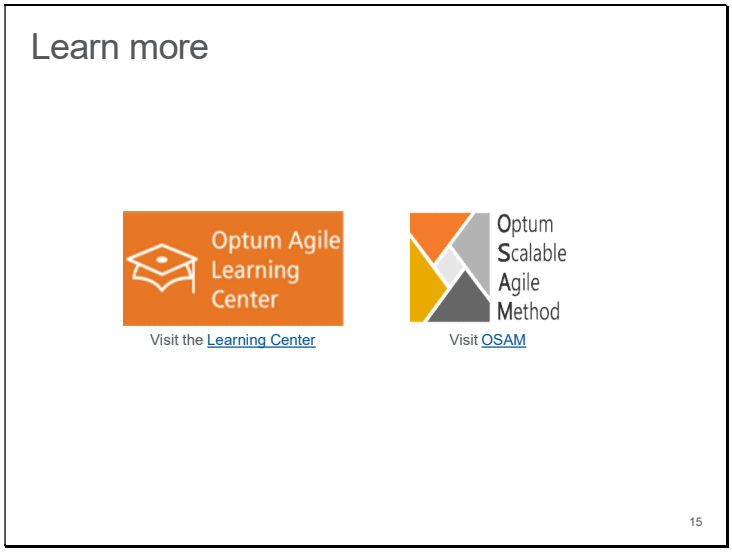
Experience, the Business Rules and the Infrastructure. This is creating a full slice of a working system that meets the minimal user needs.

Slide 14



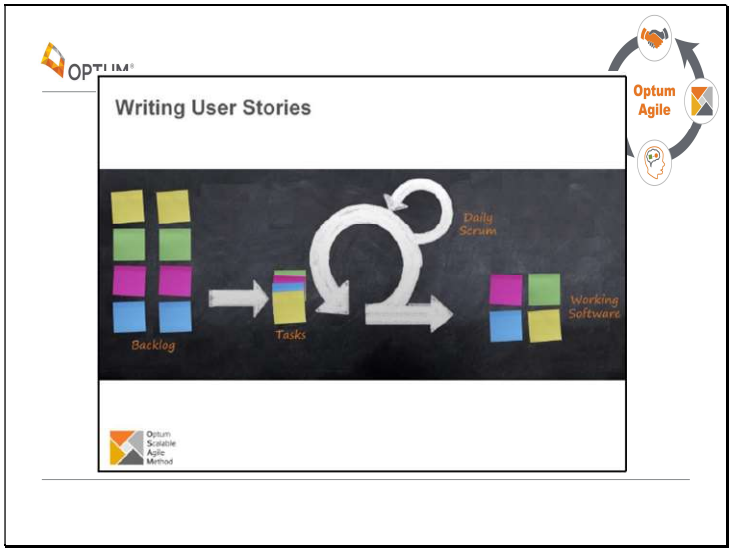
Patton, J. (2016). Retrieved 8 November 2016, from http://jpattonassociates.com/wp-content/uploads/2015/01/how_you_slice_it.pdf
Patton, J. (2016). Retrieved 9 November 2016, from http://jpattonassociates.com/story-mapping-quick-ref/
DevJam,. (2013). Story Mapping. Retrieved from https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&cad=rja&uact=8&ved=0ahUKEwjD1eWd2a3QAhVLzoMKHaM0BbYQtwIILDAC&url=https%3A%2F%2Fvimeo.com%2F70214001&usg=AFQjCNE4NFsK-3Oo4MLPAqYqEp-bjTMylQ&sig2=V1Rk5qN093JhtPVF3ycb9Q (Firefox -recommended browser)

Slide 15

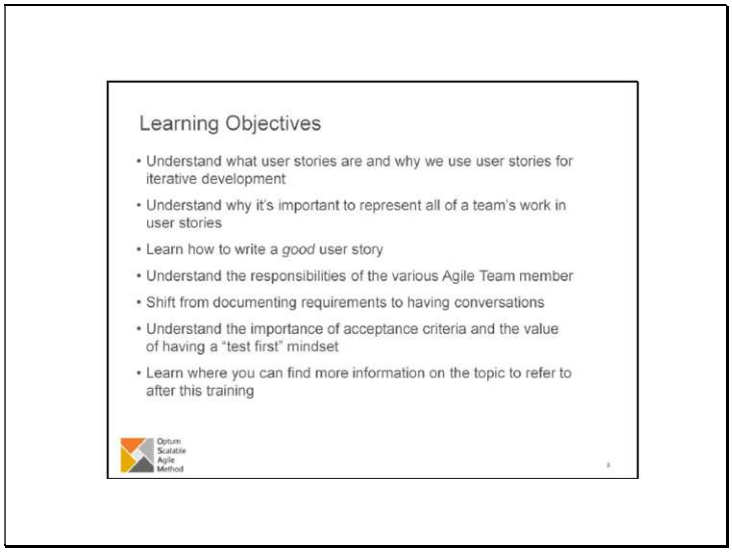


Learning Center: http://it500.optum.com/sites/DFP/EDPLearning/Pages/Home.aspx
OSAM: https://hubconnect.uhg.com/groups/optum-scalable-agile-method-osam-community-edition-in-technology

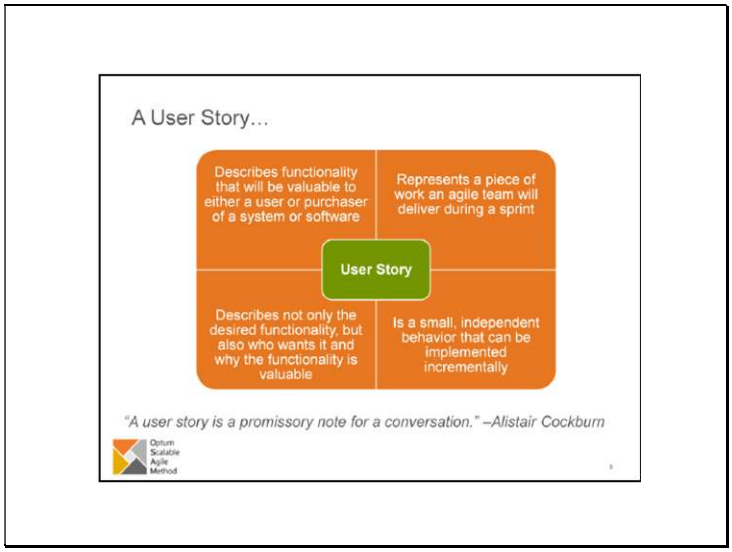
Slide 16


Writing User Stories
Optum Agile
Daily Scrum
Backlog
Tasks
Working Software

Slide 17


Learning Objectives
• Understand what user stories are and why we use user stories for iterative development
• Understand why it's important to represent all of a team's work in user stories
• Learn how to write a *good* user story
• Understand the responsibilities of the various Agile Team member
• Shift from documenting requirements to having conversations
• Understand the importance of acceptance criteria and the value of having a "test first" mindset
• Learn where you can find more information on the topic to refer to after this training
Optum Scalable Agile Method

Slide 18


A User Story...
Describes functionality that will be valuable to either a user or purchaser of a system or software
Represents a piece of work an agile team will deliver during a sprint
User Story
Describes not only the desired functionality, but also who wants it and why the functionality is valuable
Is a small, independent behavior that can be implemented incrementally
"A user story is a promissory note for a conversation." –Alistair Cockburn
Optum Scalable Agile Method

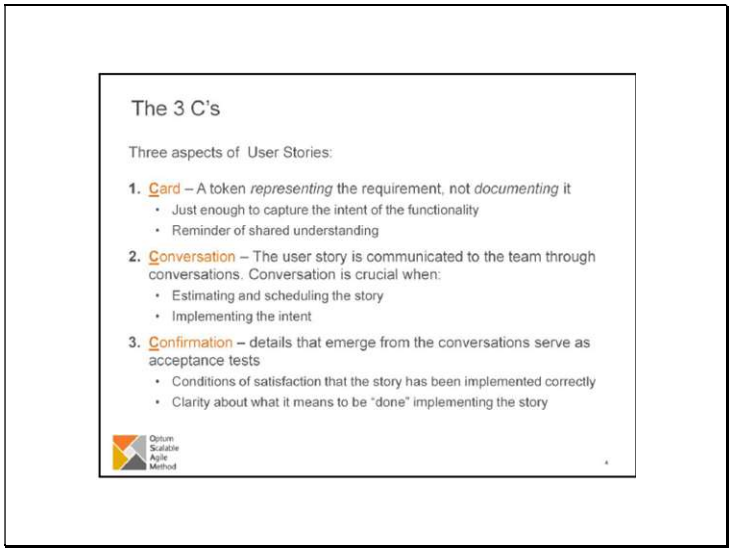
Slide 19


I'm glad the fire isn't on our end!
Optimizing for the way that we work for the greater good

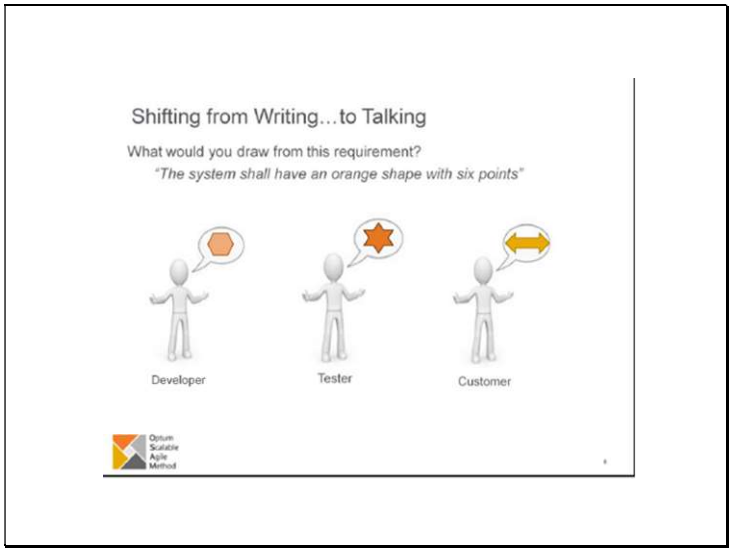
Slide 20

0

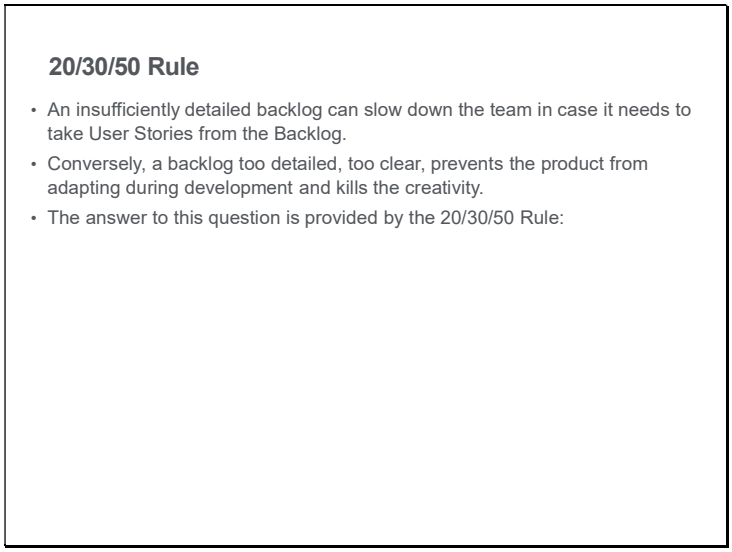
Slide 21


The 3 C's
Three aspects of User Stories:
1. Card – A token representing the requirement, not documenting it
• Just enough to capture the intent of the functionality
• Reminder of shared understanding
2. Conversation – The user story is communicated to the team through conversations. Conversation is crucial when:
• Estimating and scheduling the story
• Implementing the intent
3. Confirmation – details that emerge from the conversations serve as acceptance tests
• Conditions of satisfaction that the story has been implemented correctly
• Clarity about what it means to be "done" implementing the story
Optum Scalable Agile Method

Slide 22


Shifting from Writing…to Talking
What would you draw from this requirement?
"The system shall have an orange shape with six points"
Developer
Tester
Customer
Optum Scalable Agile Method

Slide 23



This rule was introduced by Bob Galen in his book Scrum Product Ownership.

Slide 24

This rule was introduced by Bob Galen in his book Scrum Product Ownership.

Slide 25

The conversations happen during backlog refinement and iteration planning to solidify the details

Slide 26

Writing User Stories - Participants

While anyone can write a user story, the Product Owner is responsible for maximizing the value of the product by:

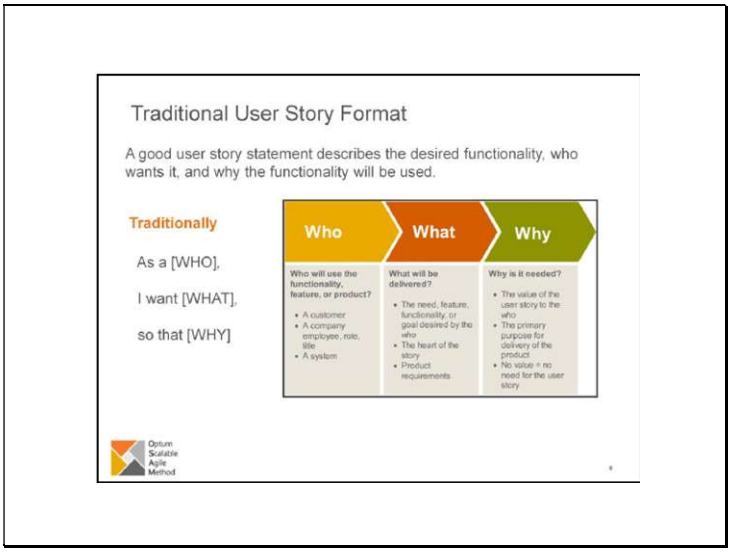
- Clearly expressing user stories
- Prioritizing the user stories to best achieve product goals and vision
- Accepting completed stories into the system baseline

The whole agile team participates in the team backlog refinement process to:

- Review and revise upcoming backlog items to get them ready for implementation
- Ask questions and gain clarity, to achieve a shared understanding of the work
- Estimate the size the stories

Optum Scalable Agile Method

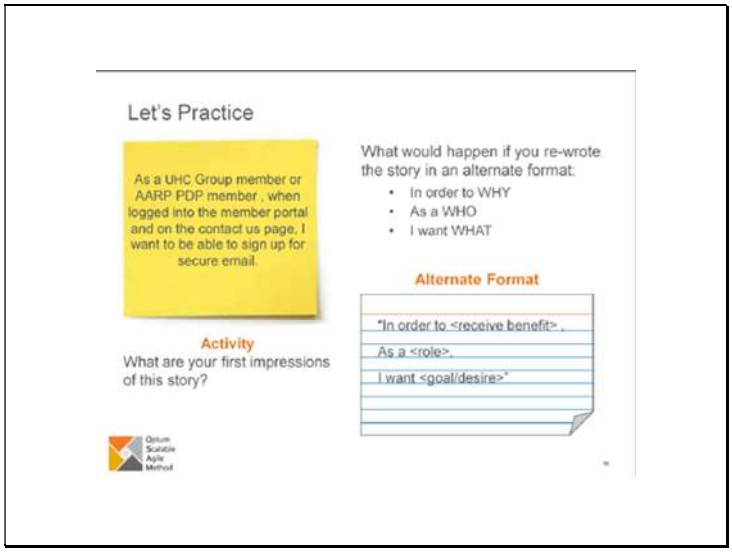
Slide 27


Traditional User Story Format
A good user story statement describes the desired functionality, who wants it, and why the functionality will be used.
Traditionally
As a [WHO],
I want [WHAT],
so that [WHY]
Who
What
Why
Who will use the functionality, feature, or product?
• A customer
• A company employee, role, title
• A system
What will be delivered?
• The need, feature, functionality, or goal desired by the who
• The heart of the story
• Product requirements
Why is it needed?
• The value of the user story to the who
• The primary purpose for delivery of the product
• No value = no need for the user story
Optum Scalable Agile Method

Slide 28


User Story Examples
DESCRIPTION
As a PC Rep, when the SAM user make changes to the following fields
Sales Rep/Sales Producer
Bill Type
Medical Employer Contribution
I want the updated information reflected in PRIME so I can processed the customer renewal.
DESCRIPTION
As a PC Rep, I want customers who complete their renewal process via the SAM portal to be notified when the renewal process has been accepted in PRIME, so that the brokers know the status of the renewal process is complete.
DESCRIPTION
As a PC Rep, I want a Standard HRA terminated via the SAM tool to be updated in PRIME while keeping the other coverages active, so that I can process the customer's renewal request.
Optum Scalable Agile Method

Slide 29


Let's Practice
As a UHC Group member or AARP PDP member , when logged into the member portal and on the contact us page, I want to be able to sign up for secure email.
Activity
What are your first impressions of this story?
What would happen if you re-wrote the story in an alternate format:
In order to WHY
As a WHO
I want WHAT
Alternate Format
"In order to <receive benefit> ,
As a <role>,
I want <goal/desire>"

Slide 30

Acceptance Criteria

- Describes what must be done for a user story to be acceptable to the Product Owner.
  - The Product Owner will include the acceptance criteria from the business perspective
  - During refinement, additional acceptance criteria may be identified and added
- The Product Owner will accept the story when:
  - The development team demonstrates the story is working as expected
  - The acceptance criteria has been met

Optum Scalable Agile Method

Slide 31

Writing Acceptance Criteria - Participants

- Agile Product Owner captures initial acceptance criteria and scenarios; then
- Reviews the user story with the agile team and together determines if there are more scenarios to be considered/captured

Optum Scalable Agile Method

Slide 32

Writing Specifications By Example

Acceptance Criteria Format

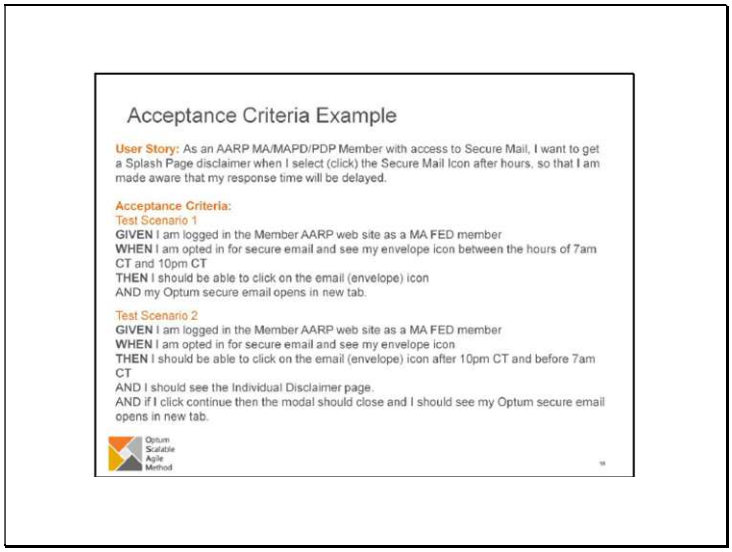
GIVEN [some context]
WHEN [the action that is carried out]
THEN [the particular set of observable consequences that should be obtained]

An example

GIVEN my bank account is in credit and no transactions are pending
WHEN I attempt to withdraw an amount of money less than my limit
THEN the transaction should complete without issues or errors

Optum Scalable Agile Method

Slide 33


Acceptance Criteria Example
User Story: As an AARP MA/MAPD/PDP Member with access to Secure Mail, I want to get a Splash Page disclaimer when I select (click) the Secure Mail Icon after hours, so that I am made aware that my response time will be delayed.
Acceptance Criteria:
Test Scenario 1
GIVEN I am logged in the Member AARP web site as a MA FED member
WHEN I am opted in for secure email and see my envelope icon between the hours of 7am CT and 10pm CT
THEN I should be able to click on the email (envelope) icon
AND my Optum secure email opens in new tab.
Test Scenario 2
GIVEN I am logged in the Member AARP web site as a MA FED member
WHEN I am opted in for secure email and see my envelope icon
THEN I should be able to click on the email (envelope) icon after 10pm CT and before 7am CT
AND I should see the Individual Disclaimer page.
AND if I click continue then the modal should close and I should see my Optum secure email opens in new tab.
Optum
Scalable
Agile
Method

Slide 34

A Good User Story

A well-written user story follows the INVEST model:

I	Independent - of all others
N	Negotiable - a flexible statement of intent, not a contract
V	Valuable - addresses the user's needs
E	Estimable - to a good approximation
S	Small - so as to fit within one sprint
T	Testable - understood enough to know how to test it

Optum Scalable Agile Method

Slide 35

"What" vs. "How"

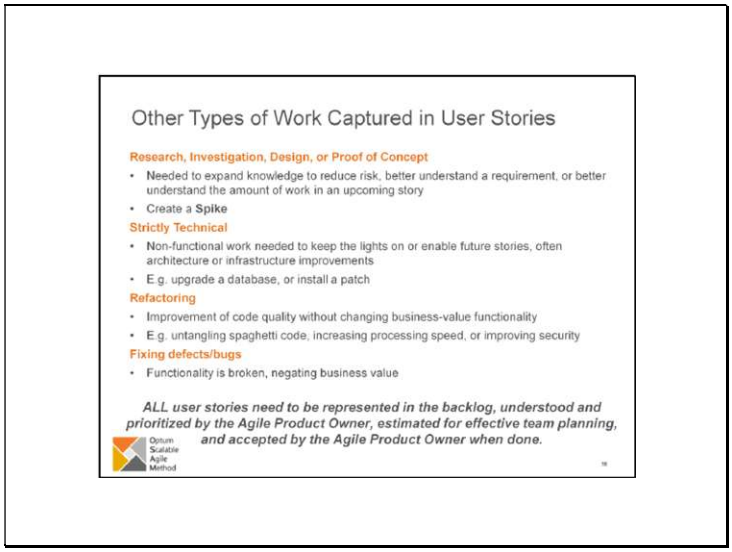
The *User Story* represents the WHAT, the *Tasks* represent the HOW

- The dev team determines HOW the work will be done
- Stories are broken into tasks during sprint planning
- Create tasks that are:
  - Individual units of work that help a story reach completion
  - Estimated in hours, and small enough to be completed within a day (typically 2-8 hours)
- Tasks are deliberately small to make:
  - Estimating easy
  - Progress toward done every day

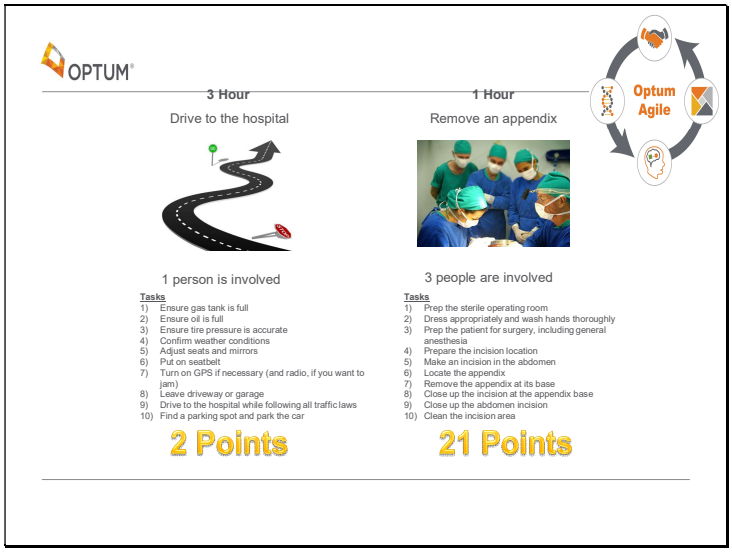
ID	NAME	ESTIMATE (H)
TA2645224	DEV: Code developments for Term Standard HRA	4.00
TA2645225	DEV: Code developments for Term Standard HRA - Remarks	4.00
TA2645226	DEV: Unit Test Term Standard HRA	4.00
TA2645227	DEV: Unit Test Term Standard HRA - Remarks	4.00
TA2645675	Host code artifacts and review	4.00
TA2715231	QA-Step Definition/ Create Test scripts	3.00
TA2715233	QA-Test Data setup	4.00
TA2715234	QA-Module Coding: Test Execution(Term Standard HRA)	8.00
TA2715235	Create	
TA2715236	Coding	
TA2715738	Techni update	
TA2715815	Implen Coding frame	

Optum Scalable Agile Method

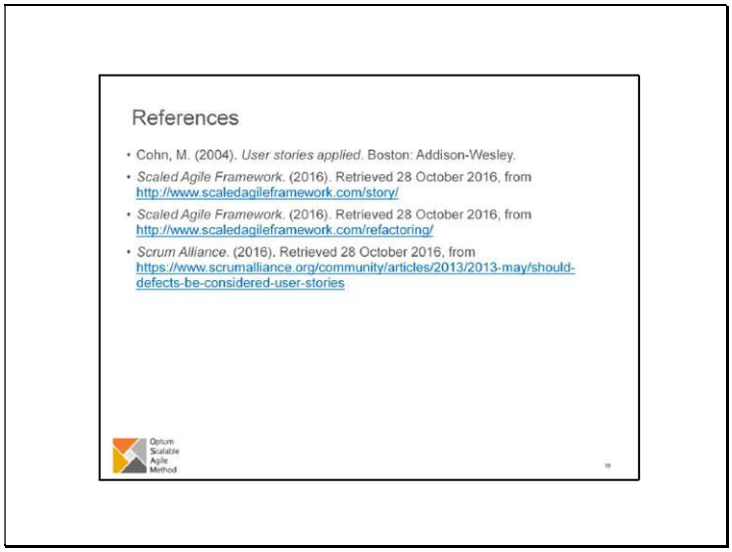
Slide 36


Other Types of Work Captured in User Stories
Research, Investigation, Design, or Proof of Concept
• Needed to expand knowledge to reduce risk, better understand a requirement, or better understand the amount of work in an upcoming story
• Create a Spike
Strictly Technical
• Non-functional work needed to keep the lights on or enable future stories, often architecture or infrastructure improvements
• E.g. upgrade a database, or install a patch
Refactoring
• Improvement of code quality without changing business-value functionality
• E.g. untangling spaghetti code, increasing processing speed, or improving security
Fixing defects/bugs
• Functionality is broken, negating business value
ALL user stories need to be represented in the backlog, understood and prioritized by the Agile Product Owner, estimated for effective team planning, and accepted by the Agile Product Owner when done.
Optum Scalable Agile Method

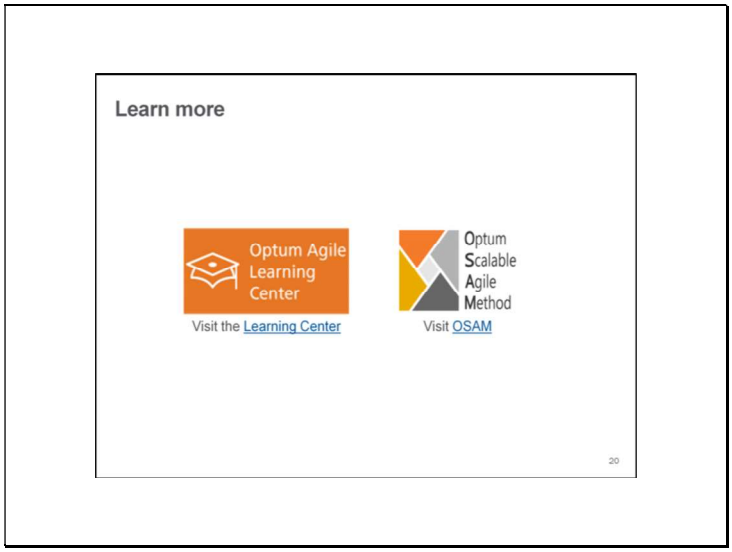
Slide 37


OPTUM
Optum Agile
3 Hour
Drive to the hospital
1 person is involved
Tasks
1) Ensure gas tank is full
2) Ensure oil is full
3) Ensure tire pressure is accurate
4) Confirm weather conditions
5) Adjust seats and mirrors
6) Put on seatbelt
7) Turn on GPS if necessary (and radio, if you want to jam)
8) Leave driveway or garage
9) Drive to the hospital while following all traffic laws
10) Find a parking spot and park the car
2 Points
1 Hour
Remove an appendix
3 people are involved
Tasks
1) Prep the sterile operating room
2) Dress appropriately and wash hands thoroughly
3) Prep the patient for surgery, including general anesthesia
4) Prepare the incision location
5) Make an incision in the abdomen
6) Locate the appendix
7) Remove the appendix at its base
8) Close up the incision at the appendix base
9) Close up the abdomen incision
10) Clean the incision area
21 Points

Slide 38

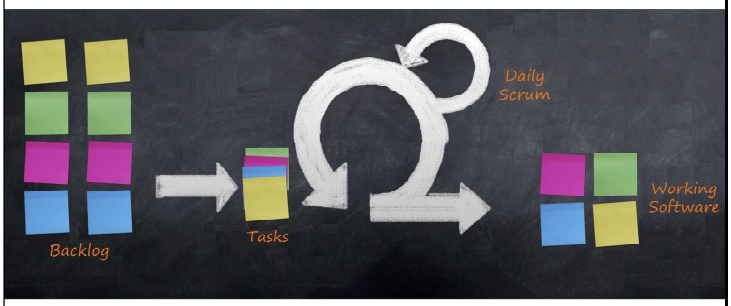

References
• Cohn, M. (2004). User stories applied. Boston: Addison-Wesley.
• Scaled Agile Framework. (2016). Retrieved 28 October 2016, from http://www.scaledagileframework.com/story/
• Scaled Agile Framework. (2016). Retrieved 28 October 2016, from http://www.scaledagileframework.com/refactoring/
• Scrum Alliance. (2016). Retrieved 28 October 2016, from https://www.scrumalliance.org/community/articles/2013/2013-may/should-defects-be-considered-user-stories
Optum Scalable Agile Method

Slide 39


Learn more
Optum Agile Learning Center
Visit the Learning Center
Optum Scalable Agile Method
Visit OSAM
20

Slide 40

Story Sizing



Optum
Scalable
Agile
Method

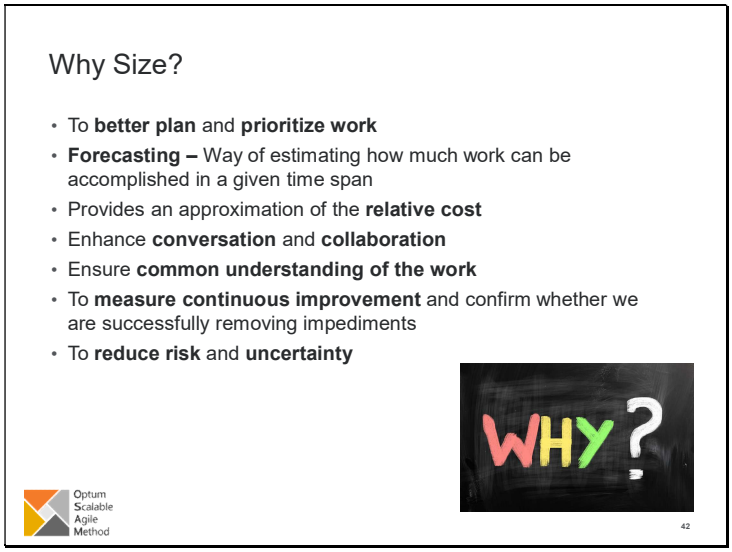
Revised: 9/6/18

Slide 41

Learning Objectives

- Understand the benefits of agile estimation
- Learn how to size features & user stories
- Understand relative sizing
- Know the roles involved in sizing user stories
- Understand how planning poker is used to size user stories

Slide 42



Objective: Participants will understand the purpose of estimates and how they are used

Good estimation can give the product owner new insight into the level of effort for each work item, which then feeds back into their assessment of each item's relative priority.

The estimate has an indirect link to Relative cost. The larger an estimate, the larger the cost---because generally, a larger story will take longer to build. This knowledge is crucial for a Product Owner when prioritizing work ---- because to be effective he/she needs to figure out what features to build largely based on which features will generate the most profit---Predicted revenue, minus the cost of building (see WSJF)

Is a way of understanding the backlog size. Once we understand backlog size, along with the development team's rate of work completion (velocity), we have a much more accurate forecast of work completion than using traditional estimation methods.

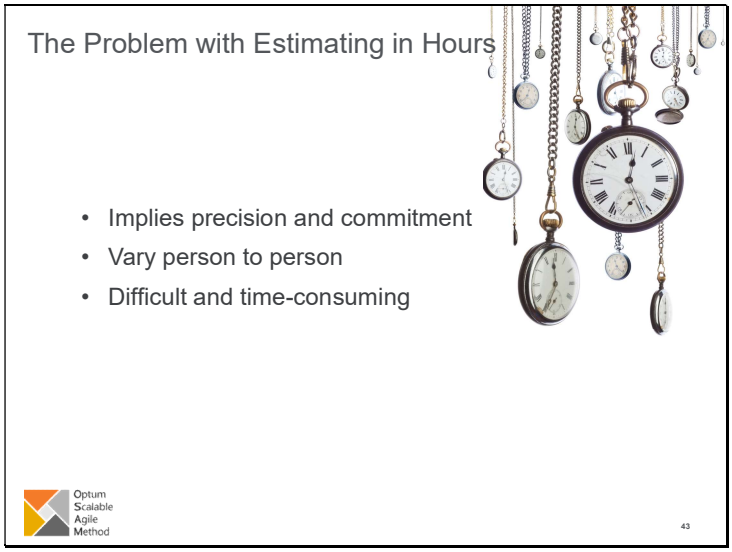
Enhance conversation and common understanding of the work
Measure output in each iteration and accurately forecast feature release
To measure continuous improvement and confirm whether or not, we successfully remove impediments

Sizing is a collaborative activity and involves team discussion, therefore it helps the agile team to have a better understanding of the user stories prior to the start of development.
This leads to:

- Reducing risks, by identifying them up front, and accounting for them in the size of the work item.
- Reducing uncertainties & doubts by having a discussion with the Product Owner and the rest of the agile team, & collecting information that would lead to completely remove or reduce the doubts.

Also, sizing activity by nature promotes collaboration as the team works together to discuss & size each item. Sizing could also be a fun & energetic activity!
Finally, as the team continues working together & becomes a “sticky team”, their estimates become more accurate & their delivery becomes more reliable.

Slide 43



OBJECTIVE: Class will understand why we don't estimate in hours. Size the WORK not the EFFORT

Estimation is a fundamental building block in Scrum. Without it Product Owners and Scrum Masters will struggle with securing a release date and showing velocity improvement. When adopting Scrum the tendency is to continue approximating in time. Unfortunately, reams of research shows that humans are inherently horrible at estimating in time. It turns out when we estimate jobs by how long they will take us to complete we have an error rate of 400%. (According to Scrum Inc. / Jeff Sutherland)

Points are…
1. A measure of team OUTPUT – more accurate than hours
• Correlated to but not necessarily the same as effort

•Implies precision and commitment

Hours have an implied precision and tend to be looked at as 100% accurate, all humans understand what an hour is so if you say 10 hours it must be ten hours. To compensate for this the person estimating will build in some "extra time" to compensate for the unknowns. It is just human nature they don't want to be held accountable for an estimate when they didn't have all the data needed.

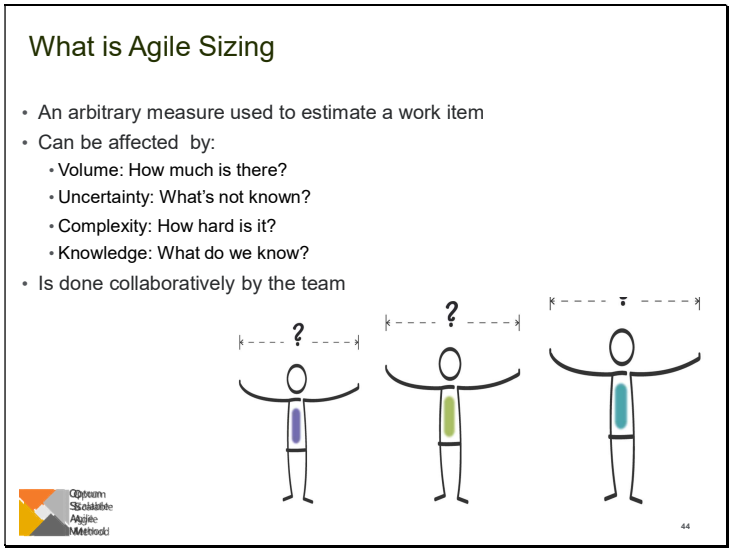
•Vary person to person

Depending on who does the work, the timespan for completion will vary, based on skills, experience etc. The person doing the estimating is not the person who 6 weeks later does the work. Then we are surprised or disappointed when a work item, estimated in hours, isn't finished on time or takes longer than the false precise estimate.

The problem with this is the larger the task/story the greater the complexity and the more time it would take to develop a truly accurate estimate. At some point it is just not worth the effort. Especially when you are looking at work that will not be done for several months.
Points on the other hand have an ‘‘acceptable’’ implied imprecision since they are relative only to each other.

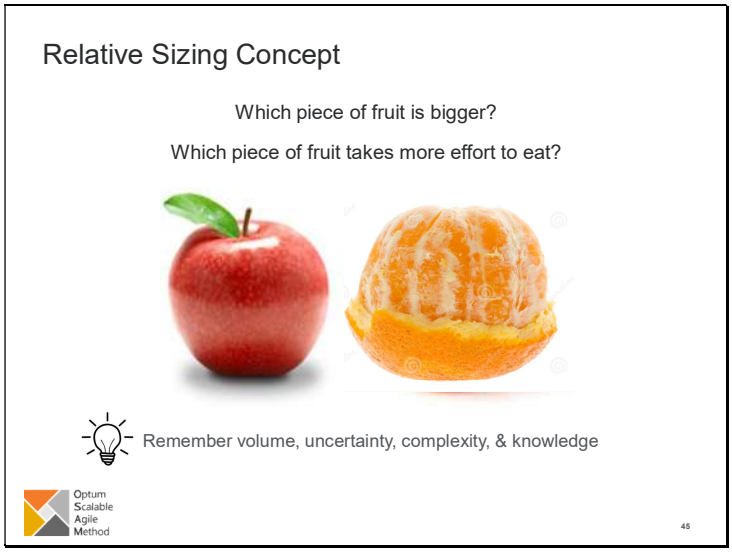
Estimates don't need to be precise, they need to be consistent. Our velocity corrects for the inaccuracy of our estimates. Estimates on stories are meant to be relative to other estimates on stories. When engineers think of an estimate, they need to be comparing the story to other stories in the past in order to evaluate consistently.

Slide 44



- In agile, sizing is a quick way of estimating the effort required to implement a user story or a feature.
- Sizing does not directly correlate to hours but rather it is an arbitrary measure (we learn more about this in upcoming slides).
- The size of a work Item (feature, user story) may be affected by different elements such as volume, uncertainty, complexity, & knowledge. For example, factors such as the *volume* & amount of work, the *uncertainty* & doubt, and *complexity* & difficulty have direct affect on the size of a work item (meaning that the higher these factors are, the larger the size could possibly be). On the other hand, having a higher level of domain *knowledge* may reduce the size of a work item.
- Sizing is not done by one individual, rather it is a collaborative work effort done by the agile team.

Slide 45

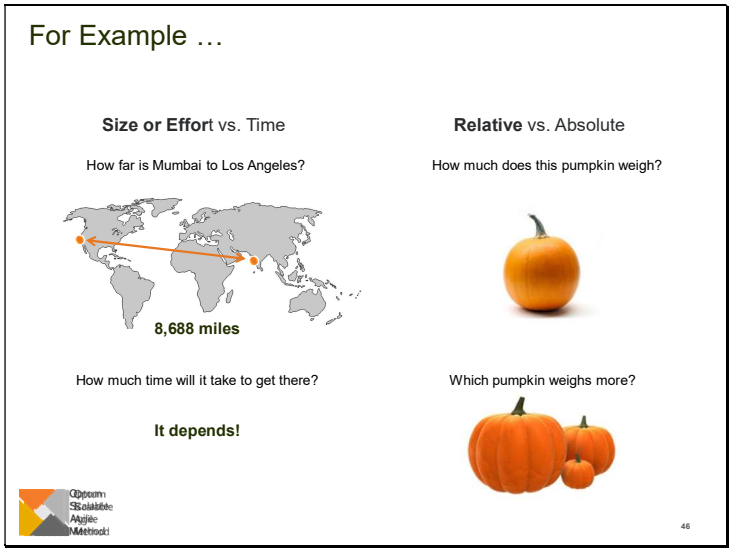


- Story points are not the same as hours & are Not absolute values.
- After working together for few sprints, agile teams establish an average velocity; they become more predictable on how many points they can deliver in a sprint.

Concept of relative sizing is used in OSAM for both Features and User Story sizing:

By looking at these two pictures:
- Which piece of fruit is bigger?
- Which piece of fruit take more effort to eat?
- it is hard to say which one is bigger because they are about the same size. However, when we consider the effort, we can right away say that orange requires more effort as it should be peeled first.
- So, we must pay to other factors such as effort, size, complexity, and doubt when estimating a user story.

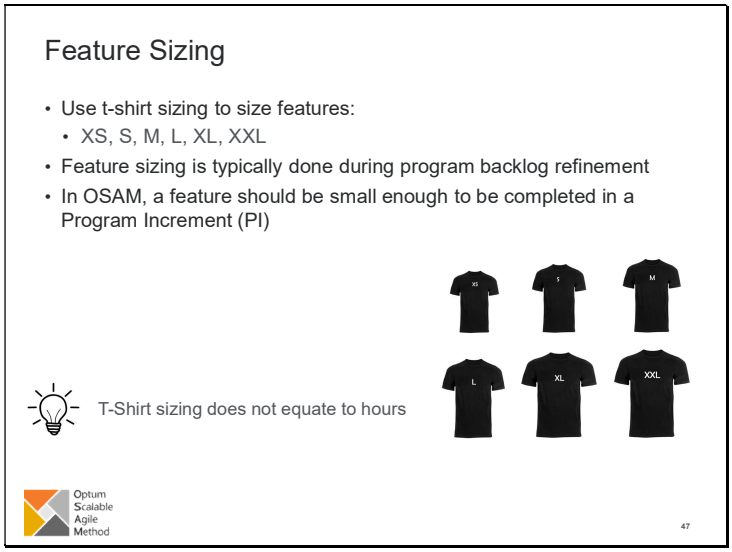
Slide 46



The first part of this example:
Illustrates how the "size/effort" differs from "time". The distance from Mumbai to Los Angeles has a fix value, but *time* needed to travel this distance may vary based on how we are going to travel.
Second part of this example:
Illustrates how "relative sizing" differ from "absolute" sizing. It is hard to calculate the absolute size of an object such as weigh of a pumpkin, but it is way easier to compare multiple pumpkins and estimate their size relative to each other.

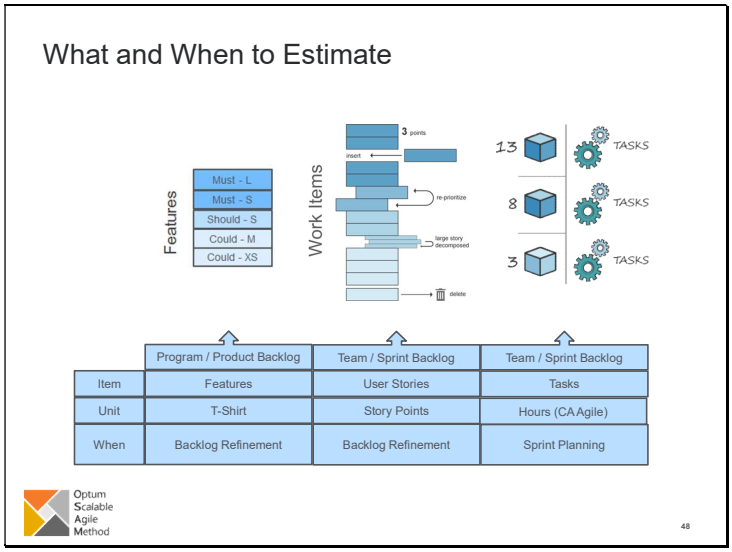
- In agile we use relative sizing by comparing stories, and the concept of size does not directly correlate to time.

Slide 47



- Features are larger user needs that will eventually decompose into user stories.
- Features are usually sized during program backlog refinement, and are sized using t-shirt sizing (XS, S, M, L, XL, XXL)
- Features should be small enough so that they can be done within a fixed time-box which is usually about 5-6 two-week sprints. In OSAM, they should be small enough to be completed in 1 Program Increment (PI).
- Remember that t-shirt sizing does not equate to hours (does not measure using hours).

Slide 48



Feature sizing usually happens during program backlog refinement; PMC, or SoS depending on your planning horizon

•A Feature should be small enough to be delivered in an agreed upon number of sprints

•In OSAM, a feature should be small enough to be completed in a Program Increment (PI)

- Story sizing usually happens during team backlog refinement
- Stories should be small enough to be delivered within a sprint
- Story should be sized prior to entering a sprint

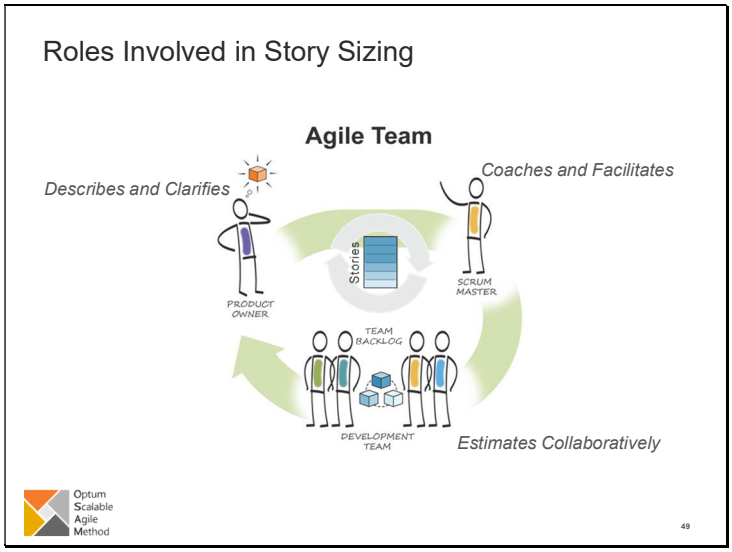
- Flow teams can assign points in Refinement, or decide not to use points, and only rank order the work (features or stories) during Refinement.
  Sprint and PI planning horizon teams need to assign points to backlog items and this is done during Refinement.

- At the Team level, the team or sprint backlog is refined. The team backlog includes user stories that represent all the work that needs to be done (including new feature development, defects, spikes, and technical debt). The Agile Product Owner prioritizes the user stories in the team backlog.

  At the Train level, the product backlog is refined. The product backlog typically includes features and the Product Manager (PM) prioritizes the features in the product backlog.
  Product backlog refinement

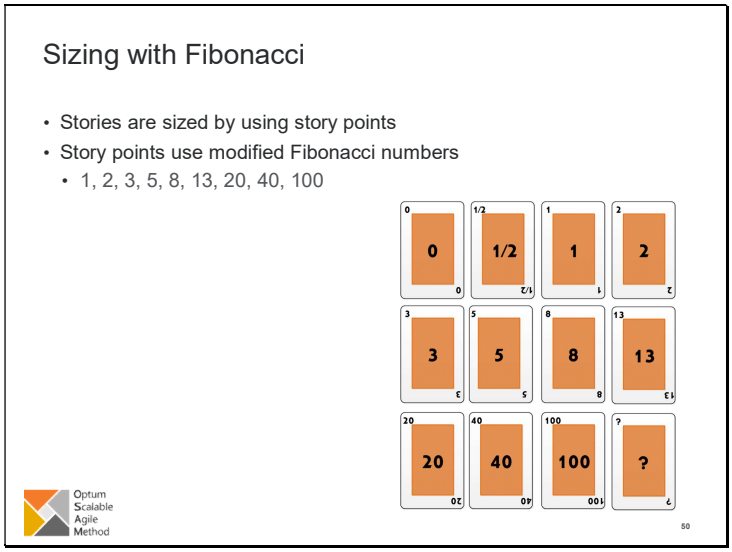
- Product Backlog Refinement is a combined activity at the sprint train level. It is an on-going activity throughout the project to decide which teams implement which items, and in what sequence over upcoming Sprints.
  The product backlog typically includes features. The Product Manager prioritizes the features in the product backlog.
  It is also a chance to increase alignment with the Product Owner and all teams
  Attendees include Product Managers, Agile Product Owners and Agile team representatives

Slide 49



- The entire agile team participates in sizing activity; however ONLY the Development team actually sizes---the APO and SM do not.
- The Scrum Master facilitates the activity and coaches the team.
- The Product Owner provide insight on each user story, and makes sure the development team has a good understanding of the story.
- The development team collectively estimates each user story; they may ask clarifying questions from the from the Product Owner to figure out what it takes to deliver the item.
- No one can force the development team to change their estimates.

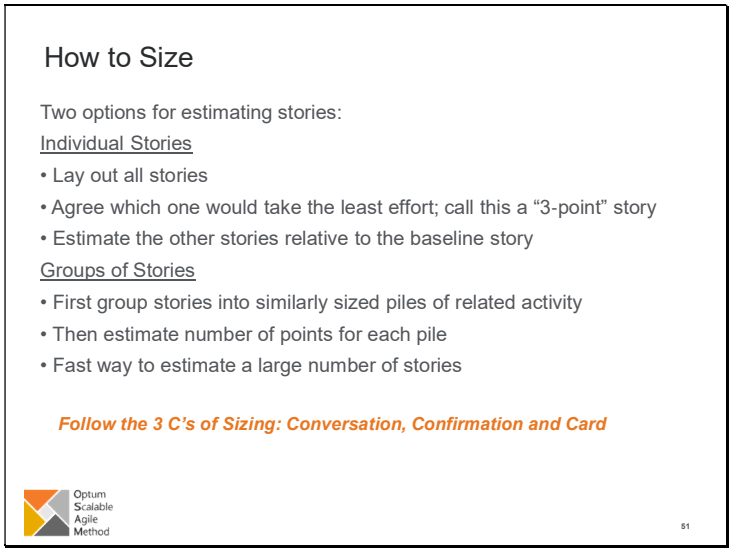
Slide 50



Story Points:

- is the unit of measure for sizing user stories and other types of work items such as defects and spike.
- We use modified Fibonacci numbers for story points (0, ½, 1, 2, 3, 5, 8, 13, 20, 40, 100)
- You may want to consider including 0 as a valid number within your estimation range. Although it is unlikely that a team will encounter many user stories that truly take no work, but you may find a story that it closer to 0 than ½ or you may prefer to not use 0 or ½ at all.

Slide 51



Estimation in Agile:

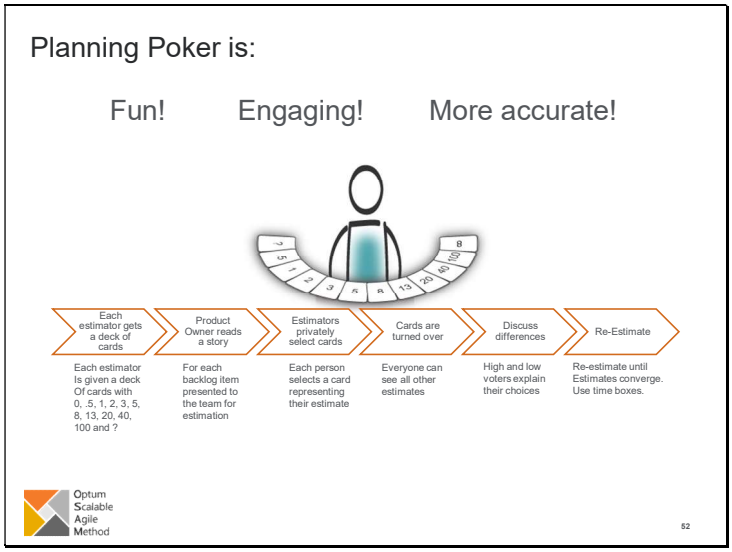
- The development team who will be doing the work will estimate the user stories collaboratively
- In agile we use relative sizing (comparing stories with the baseline story) and size the stories relative to the baseline story
- Agile suggests just-in-time estimation when the story is broken down into multiple smaller stories.
- To arrive at an estimate, we rely on expert opinion, analogy, and disaggregation. A fun and effective way of combining these is planning poker (more to come on that)

For Relative sizing:

- Select a small story as a baseline (so that other stories can be compared with this story); give 3 points to this baseline story.
- Then compare other stories with this baseline, and size other stories relative to it.
- Avoid Starting the baseline with a One-Point Story. It doesn't leave room for anything smaller items without resorting to fractions, and those are harder to work with later.

Example: an 8 point story might be approximately 4 times bigger (in terms of effort, complexity, and/or doubt) than a 2 point story

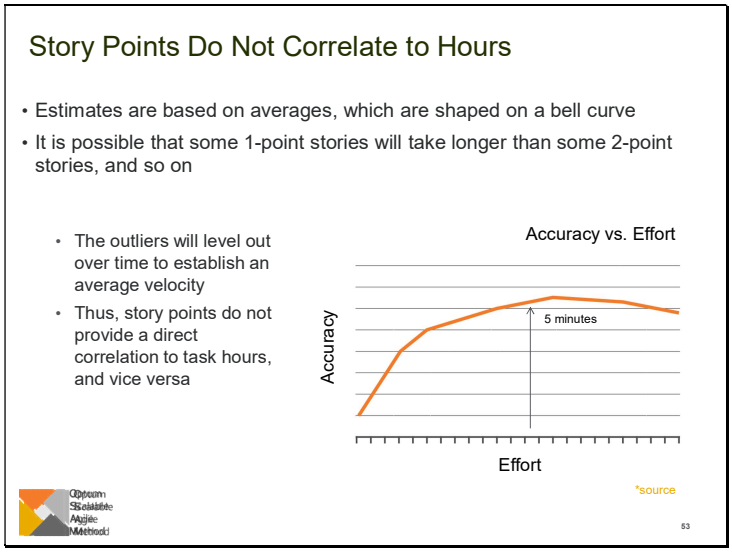
Slide 52



- To size a story, the development team should consider all activities that should happen in order to bring a story to completion (i.e.: analysis, development, testing …); not just development work, or just testing!
  - Story Point is a unit of measure for expressing the overall size of a user story.
  - Estimates are not created by a single individual on the team, the entire development team should participate!

- It engages each member of the Development Team.
- It is a fun activity and supports team collaboration.
- It results in more accurate sizing.

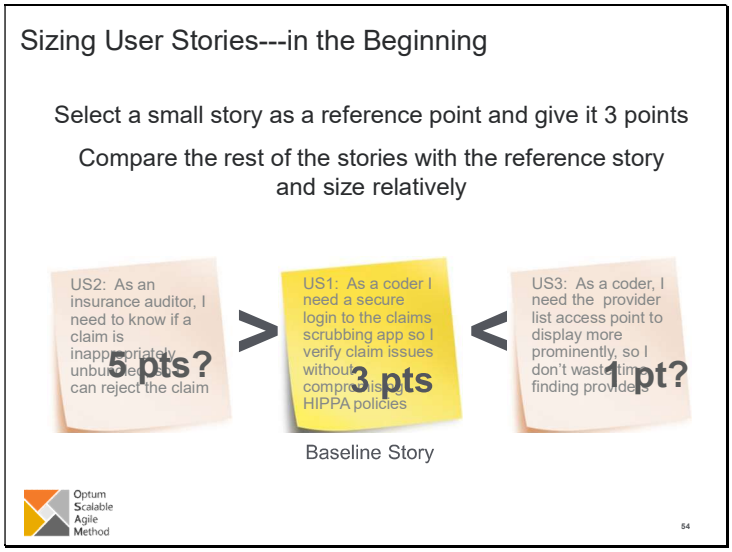
Slide 53



- Story points are not the same as hours & are Not absolute values.
- In relative sizing, it is possible for a 1-point story to take longer than a 2-point story.
- As illustrated in this image, points may slightly overlap. Estimates have bell curve shape.
- After working together for few sprints, agile teams establish an average velocity; they become more predictable on how many points they can deliver in a sprint.

- Mike Cohn (2006)

Slide 54



For Relative sizing:

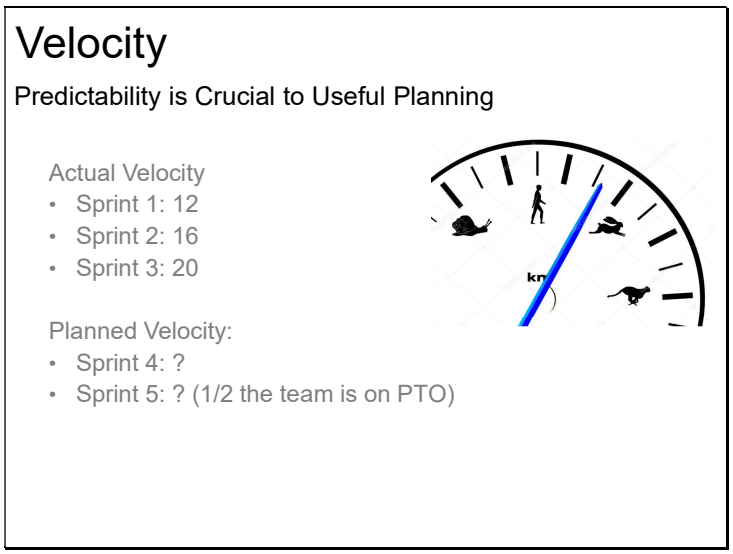
- Select a small story as a baseline (so that other stories can be compared with this story); give 3 points to this baseline story.
- Then compare other stories with this baseline, and size other stories relative to it.
- Avoid Starting the baseline with a One-Point Story. It doesn't leave room for anything smaller items without resorting to fractions, and those are harder to work with later.

Example: an 8 point story should take 4X longer than a 2 point story

Examples:

- A story slightly smaller than the baseline could be a 2 point story
- A story significantly smaller than the baseline could be a 1 point story
- A story larger than the baseline could be a 5 point story
- A story approximately 3 times as large as the baseline could be an 8 point story

Slide 55



Objective: Class will understand what velocity is, and the purpose of a consistent, known velocity.

Velocity is the average number of story points that can be completed in an iteration by a team. It is a tool that measures the pace at which a team gets work done.

The team uses this number to determine how much work they will most likely be able to finish within an iteration.

Your goal should be optimal velocity, not maximum velocity.

If your team is pushed into maximum velocity, often the product suffers. Don't ignore defects, testing, and other quality factors just to show maximum velocity.

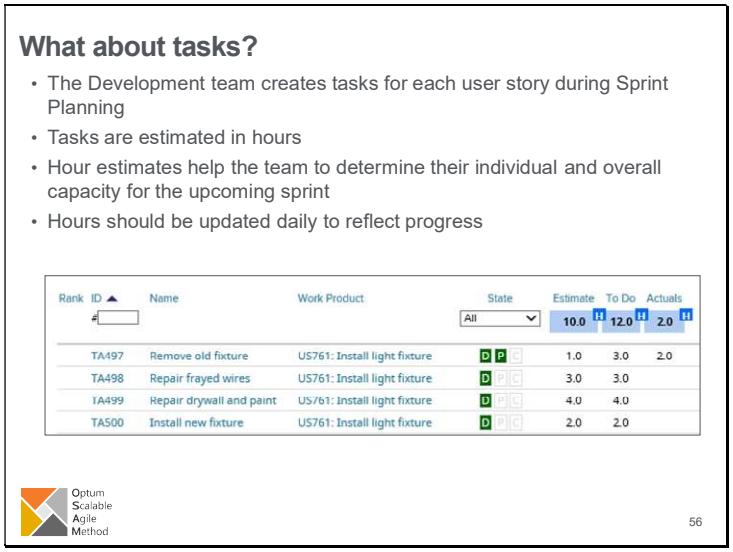
Velocity generally stabilizes after 3 – 6 iterations

Exercise:

Ask the class what the planned velocity would be for Sprint 4 and Sprint 5

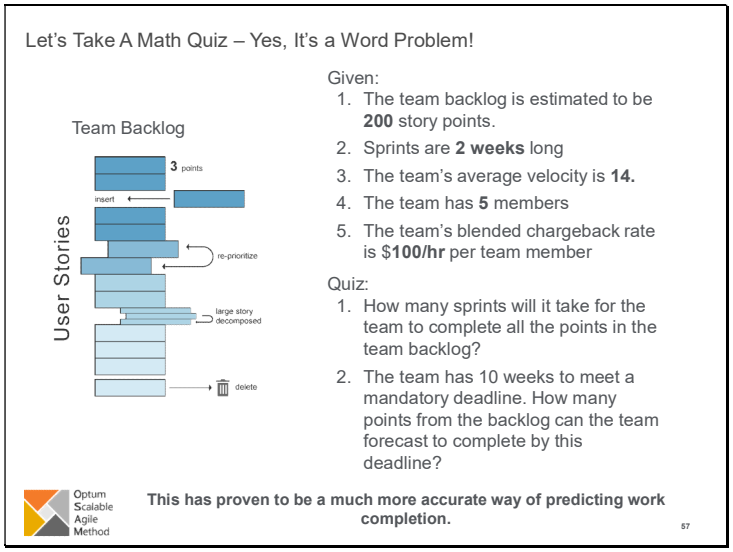
Sprint 4: **14** (12*16*20/3)=
Sprint 5: around **7**

Slide 56



The Development team creates tasks for each user story during Sprint Planning. These tasks describe the HOW and the WHO. The PO does not create tasks for the team.
Tasks are estimated in hours
Hour estimates help the team to determine their individual and overall capacity for the upcoming sprint
The “To Do” and “Actuals” hours should be updated daily to show progress.

Slide 57



Question 1 – answer = 12.5 sprints =(200/14) --14 === how many weeks would that be? 28
Question 2 – answer = 140 points =(10*14)

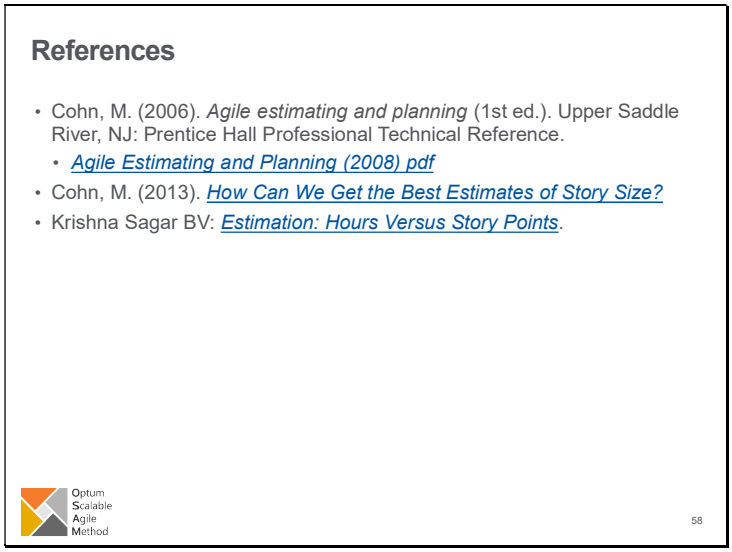
Instructor: Could also pose the following question:

What would be the *approximate* cost of developing the entire backlog?

Weekly cost of team: Number of team members (5) * rate per hour (100) * days in the week (5) * hours per day (8) = $20000

Total cost for entire backlog: Number of weeks (28) * Weekly cost of team (2500) = $560,000

Slide 58



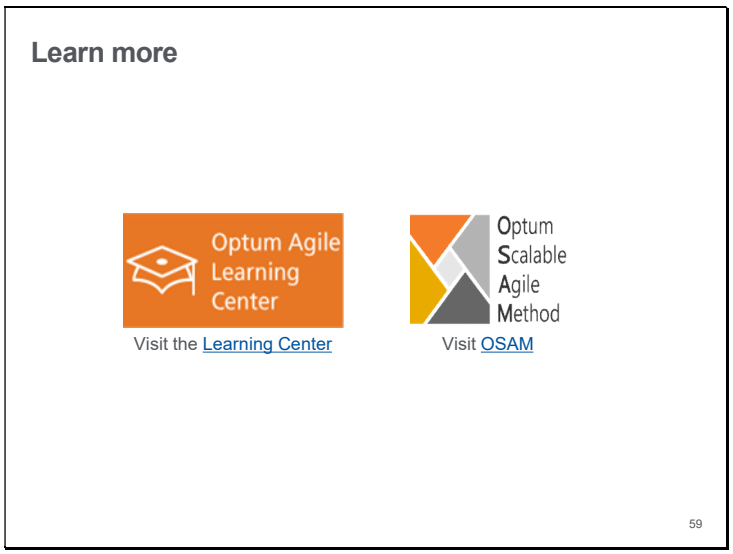
Cohn, M. (2006). *Agile estimating and planning* (1st ed.). Upper Saddle River, NJ: Prentice Hall Professional Technical Reference.

- *Agile Estimating and Planning (2008) pdf*

Cohn, M. (2013). *How Can We Get the Best Estimates of Story Size?*

Krishna Sagar BV: *Estimation: Hours Versus Story Points*.

Slide 59



Learning Center: http://it500.optum.com/sites/DFP/EDPLearning/Pages/Home.aspx
OSAM: https://hubconnect.uhg.com/groups/optum-scalable-agile-method-osam-community-edition-in-technology

The left navigation of the Optum Agile Learning Center contains the following links:
Training – shows the learning path with the recommended order for completing the Product Owner courses; course links take you to registration in LearnSource
Calendar – shows the format of the classes and recommend order to complete the entire learning path within 45 days
Course Materials – provides the most recent version of the training decks in PDF format

OSAM is the agile method used in the enterprise. There is a link to Optum Agile Learning Center from the right side navigation in OSAM.

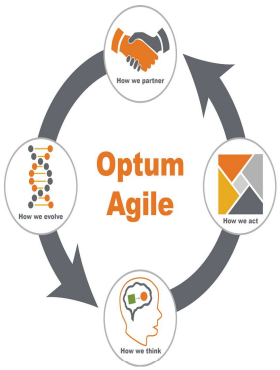
Slide 60

OPTUM

User Story Definition

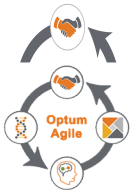
Slicing User Stories

July 2018



Slide 61

Slicing User Stories: *Agenda*



By the end of this training you should be able to

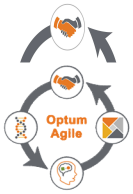
Appropriately slice a User Story

by…

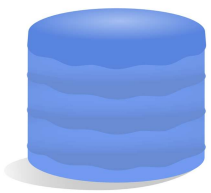
- *Identifying User Stories*
- *Understanding what makes a complete a User Story*
- *Ideating the approach so you can estimate the size*
- *Finding ways to segment deliverable business value*
- *Refining a User Story using the INVEST criteria*

Slide 62

Slicing User Stories: *Identifying User Stories*



A User Story is like a cake



As a Client Administrator (CA)
I need to associate one or more business offices with a client
so I can let them know where to find local support

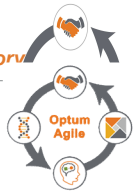
GET CLIENT DATA
GET BUSINESS OFFICES LISTS
OPEN RECORDS
CREATE ASSOCIATIONS
SAVE RECORDS
SEND EMAILS
ETC!

This particular story is quite large!

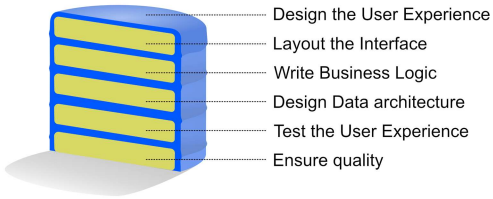
Let's think about what makes it large.

Slide 63

Slicing User Stories: *Understand what makes a User Story*



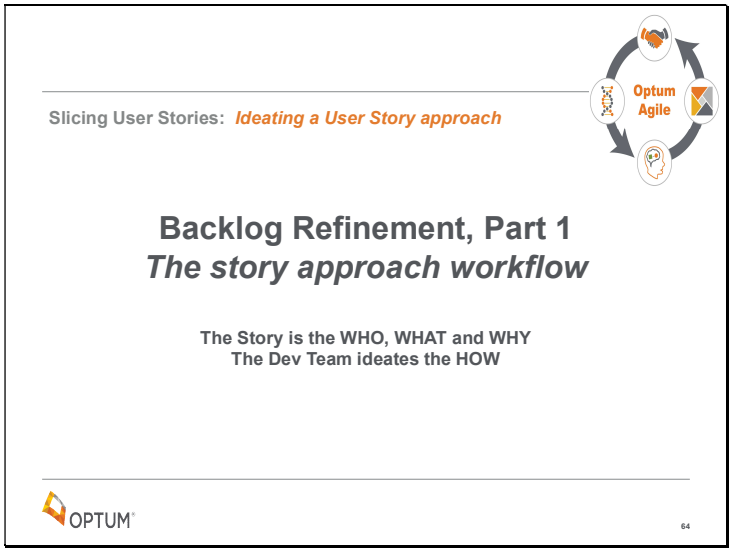
Stories always have to deliver customer business value.
That means we perform tasks like…



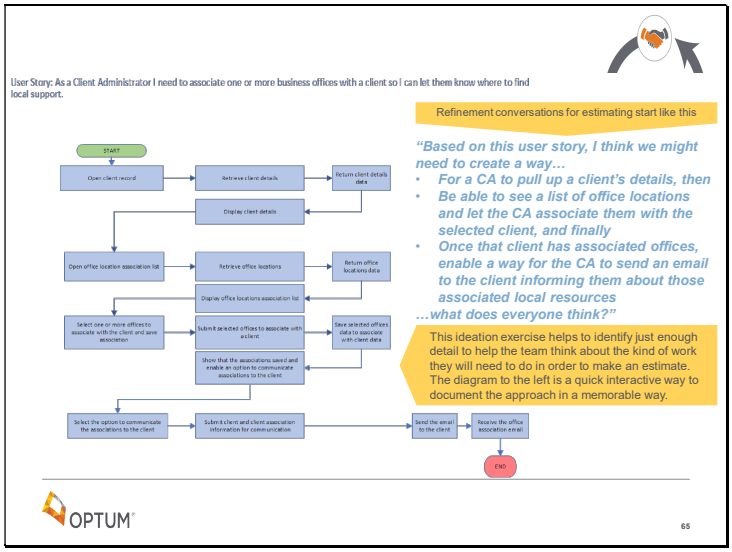
Stories need to address everything it takes to achieve the business value.

OPTUM

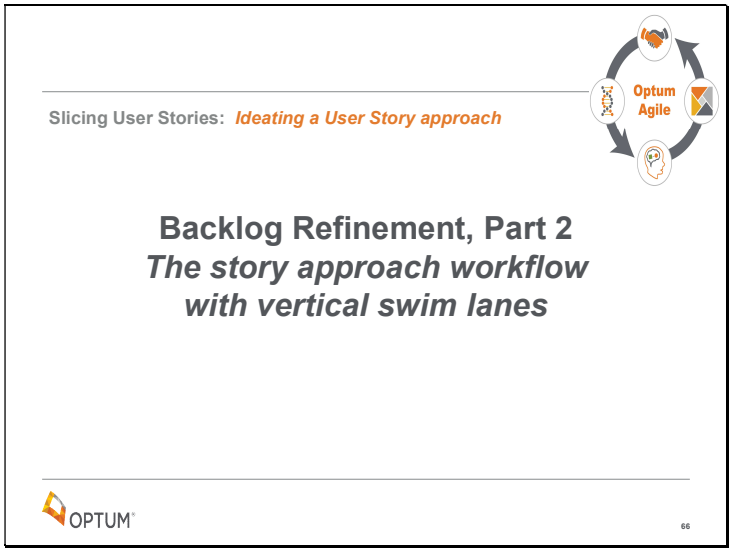
Slide 64


Slicing User Stories: Ideating a User Story approach
Optum Agile
Backlog Refinement, Part 1
The story approach workflow
The Story is the WHO, WHAT and WHY
The Dev Team ideates the HOW
OPTUM
64

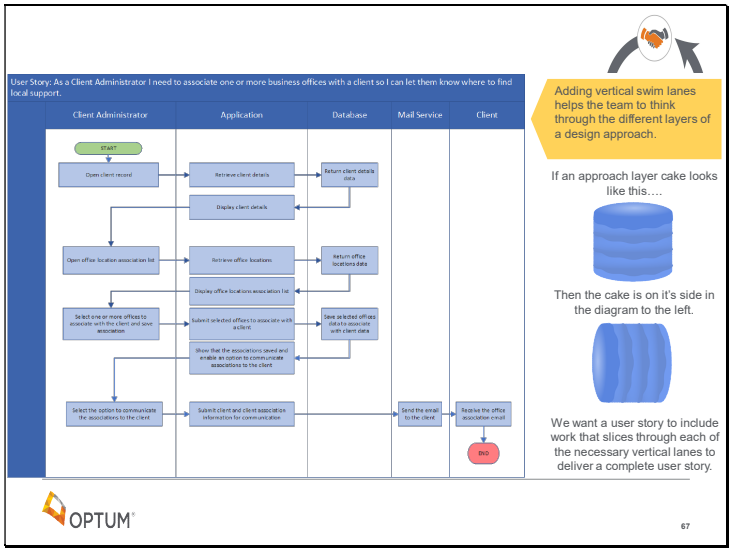
Slide 65


User Story: As a Client Administrator I need to associate one or more business offices with a client so I can let them know where to find local support.
Refinement conversations for estimating start like this
"Based on this user story, I think we might need to create a way…
For a CA to pull up a client's details, then
Be able to see a list of office locations and let the CA associate them with the selected client, and finally
Once that client has associated offices, enable a way for the CA to send an email to the client informing them about those associated local resources
…what does everyone think?"
This ideation exercise helps to identify just enough detail to help the team think about the kind of work they will need to do in order to make an estimate. The diagram to the left is a quick interactive way to document the approach in a memorable way.
START
Open client record
Retrieve client details
Return client details data
Display client details
Open office location association list
Retrieve office locations
Return office locations data
Display office locations association list
Select one or more offices to associate with the client and save association
Submit selected offices to associate with a client
Save selected offices data to associate with client data
Show that the associations saved and enable an option to communicate associations to the client
Select the option to communicate the associations to the client
Submit client and client association information for communication
Send the email to the client
Receive the office association email
END
OPTUM
65

Slide 66

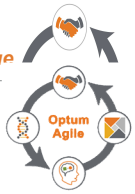

Slicing User Stories: Ideating a User Story approach
Optum Agile
Backlog Refinement, Part 2
The story approach workflow with vertical swim lanes
OPTUM
66

Slide 67

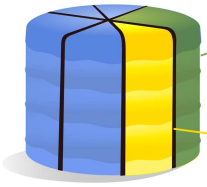

User Story: As a Client Administrator I need to associate one or more business offices with a client so I can let them know where to find local support.
Client Administrator
Application
Database
Mail Service
Client
START
Open client record
Retrieve client details
Return client details data
Display client details
Open office location association list
Retrieve office locations
Return office locations data
Display office locations association list
Select one or more offices to associate with the client and save association
Submit selected offices to associate with a client
Save selected offices data to associate with client data
Show that the associations saved and enable an option to communicate associations to the client
Select the option to communicate the associations to the client
Submit client and client association information for communication
Send the email to the client
Receive the office association e-mail
END
Adding vertical swim lanes helps the team to think through the different layers of a design approach.
If an approach layer cake looks like this….
Then the cake is on it's side in the diagram to the left.
We want a user story to include work that slices through each of the necessary vertical lanes to deliver a complete user story.
OPTUM
67

Slide 68

Slicing User Stories: *Segment deliverable business value*



If a Story is too large, we need to break it up!
Each segment becomes it's own complete User Story that can be released as deliverable business value.



Segmented Story One:
As an {A} I need to {B} so I can {C}

Segmented Story Two:
As an {A} I need to {B} so I can {C}

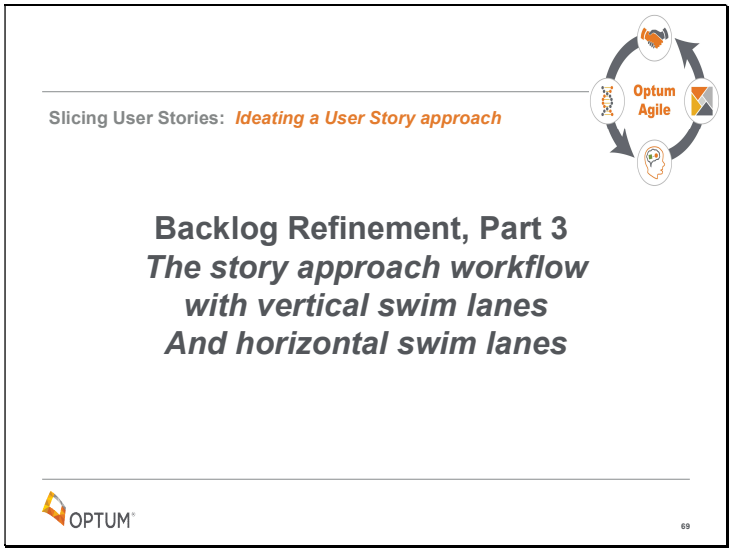
Together, the segmented stories equal the original large story!

Who only wants a slice of frosting?

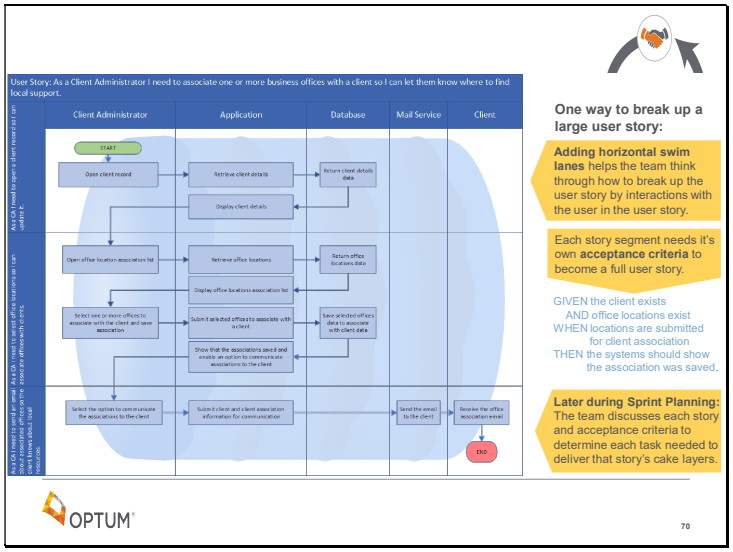
To deliver appropriately, we slice through all of the layers!

OPTUM

Slide 69

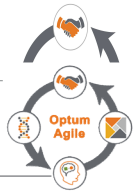

Slicing User Stories: Ideating a User Story approach
Optum Agile
Backlog Refinement, Part 3
The story approach workflow
with vertical swim lanes
And horizontal swim lanes
OPTUM
69

Slide 70


User Story: As a Client Administrator I need to associate one or more business offices with a client so I can let them know where to find local support.
Client Administrator
Application
Database
Mail Service
Client
START
Open client record
Retrieve client details
Return client details data
Display client details
Open office location association list
Retrieve office locations
Return office locations data
Display office locations association list
Select one or more offices to associate with the client and save association
Submit selected offices to associate with a client
Save selected offices data to associate with client data
Show that the associations saved and enable an option to communicate associations to the client
Select the option to communicate the associations to the client
Submit client and client association information for communication
Send the email to the client
Receive the office association email
END
One way to break up a large user story:
Adding horizontal swim lanes helps the team think through how to break up the user story by interactions with the user in the user story.
Each story segment needs it's own acceptance criteria to become a full user story.
GIVEN the client exists
AND office locations exist
WHEN locations are submitted for client association
THEN the systems should show the association was saved.
Later during Sprint Planning: The team discusses each story and acceptance criteria to determine each task needed to deliver that story's cake layers.
OPTUM
70

Slide 71

Slicing User Stories: *Additional ways identify slices*



Good Ways

- By users
  - Do multiple users relate differently to the story?
- By acceptance criteria
  - Types/groups
  - Positive vs Negative
- By story interactions
  - Are there multiple interaction moments with the target user?
- By story event timing
  - Is there a wait moment in the process?

Bad Ways

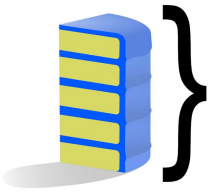
- By task
  - Don't turn tasks into stories
- By development role
  - Don't create stories based on different roles doing work
- By effort hours
  - If you are breaking up stories by grouping hours of effort, it is a good sign that the story still needs to break up more
- By task groupings
  - Same as by effort hours

OPTUM

Slide 72

Slicing User Stories: *Refine each User Story Slice using INVEST*

To have a true User Story we need to INVEST.



INVEST in each Story:

Independent
Negotiable
Valuable
Estimable
Small
Testable

Learn to slice through the layers in a manner that allows us to INVEST.