

User Story Mapping

A guide to USM and Agile Product Management



Product Owner

Responsible for optimizing value delivered by the Team & keeping the backlog transparent. Owns the product backlog & the vision (the WHAT) for the product & has the mandate to make business decisions. It is the Product Owner (PO) who is responsible for realistic expectation management, which means saying NO to non-value-adding ideas & deadlines - and it is the PO who sets the goal for each sprint to enable continuous value delivery. Responsible for facilitating stakeholder and customer collaboration.

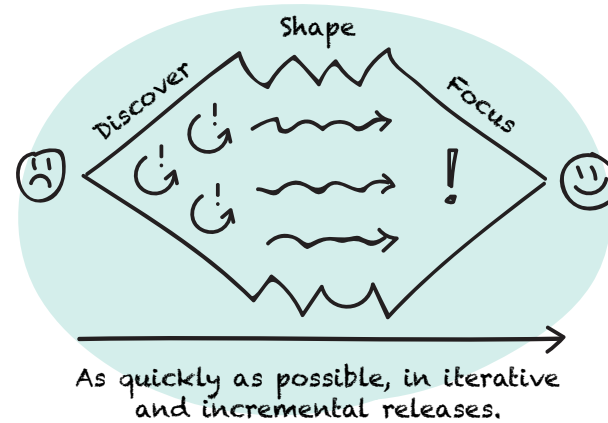
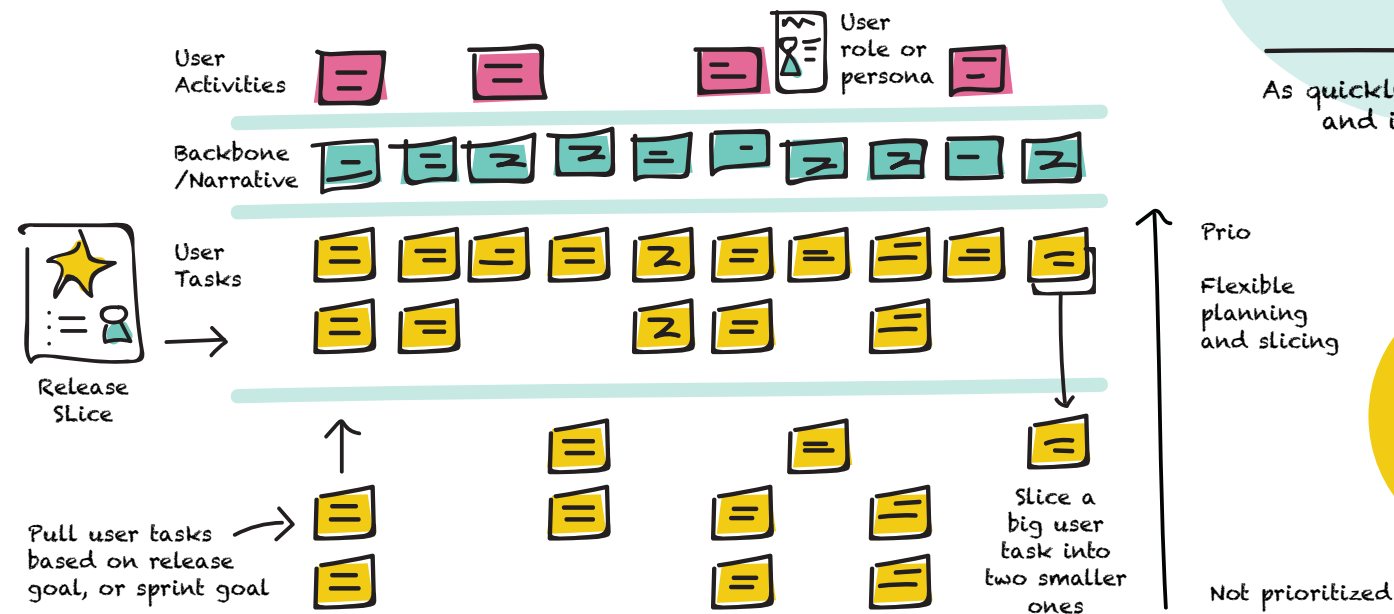
DANDY
PEOPLE

User Story Mapping is a method created by Jeff Patton.

This is part of the Dandy People Product Management training materials.

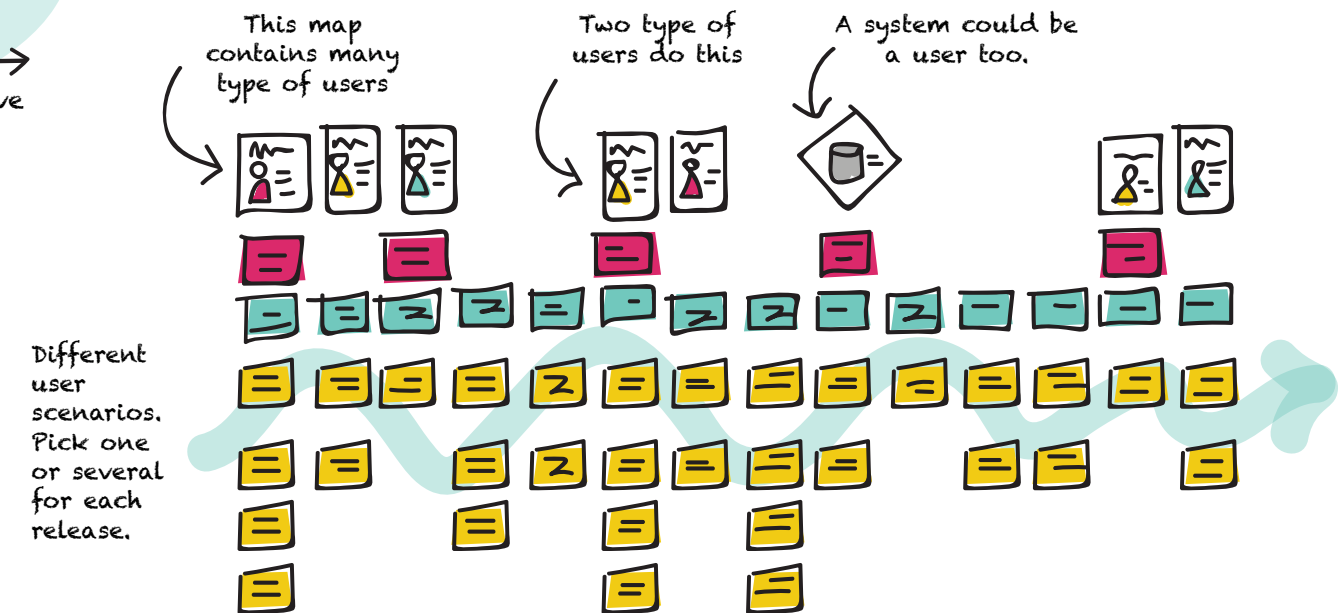
The User Story Maps is a simple and yet powerful way to visualize the story about how the users are using your product or service.

It is simple because it offers great support to move quickly from understanding the user and their problems - to building and shipping the product, and it can be done just with sticky notes on a wall, or digitized in storiesonboard.com, or other tools, and imported slice by slice (sprint) to Trello or Jira if needed still keeping the big picture intact.



It is a living, transparent, and value-based backlog that support the Product Owners and teams to find thin slices to release that create real value based on user scenarios, and not features. If you are looking to become a value and product-driven organization, this tool offers a lot of support.

It is powerful because it tells a story, it gives context to the user story and it gives a clear overview of the backlog and what we need to build to be able to support the user scenarios over all relevant touchpoints. It also supports collaboration and both horizontal and vertical slicing.



Creating the Map

1. Before mapping

Set the stage on what customer journey, product, service, and target users you most likely will have in your map. Remember this might be updated once you uncover the full story. This might tell you who to involve in creating it, and what customer journeys, user transaction maps, or personas that might need to be done before you move over to the user story mapping.

2. Map the big picture

Focus on getting the whole story. Think "mile- wide, inch deep" The activities and high-level user tasks that tell the whole story form the backbone of your story map. Look at the customer journey end to end, what they do before, during and after using your product.

Start with the user type most critical to your product's success. Imagine a typical day in your user's life with your new product, or use information from your user research and map it directly from your customer journey or user transaction map. Map the steps they take as user tasks left to right. It is ok to make assumptions at this phase, but all risky assumptions would have to be validated to make sure you deliver the right product and meet real user goals.

Identify user activities – groups of tasks that work together to support a common goal. Activities often emerge after you see more of the story. Make sure the user activity end with the user reaching its user goal in that activity.

Add in additional users. As you follow the typical use of your product, you may find other types of users enter your story. Continue modeling their story left to right.

3. Explore

Fill the body of your story map by breaking down larger user tasks into smaller subtasks. During this phase, you'll add cards, split one card into two, rewrite cards, and reorganize them. Which you also will do in, or before each sprint. It might be a good idea to timebox this phase. Keep in mind that "good enough" actually will take you longer than having all the details in now.



Facilitates
valuable
discussions
with users and
the teams

4. Slice out viable releases

Slice your map into holistic product releases that span the users and use of the product. These slices form an incremental product release roadmap where each release is a minimal viable product release.

For each release name the target outcomes and impact. Outcomes and impact say how this release contributes to the overall goal in the "big why" that motivates building the product, and how users will behave in a way that helps us reach the goal.

For each release, identify product success metrics. Answer the question: "what would we measure to determine if this product was successful?" Ideally, you'll find specific changes in user's behavior as they use the product the way your story map imagines.

5. Slice out a development strategy

Slice the first release of your map into three or more delivery phases that allow you and your team to learn fast and avoid risk. Think of the opening, mid, and end-game phases of a chess game.

This development strategy will help you release the best product possible in the time constraints you have.

Opening Game builds a "functional walking skeleton" – the simplest possible functional version of the product. As you finish "Opening game" vet the product with users and other stakeholders. Begin validating wanted performance and scalability.

Mid Game completes all major functionality and makes existing functionality richer and more complete. Continue user testing and leverage feedback to adjust the product. Continue testing performance and scalability.

End Game refines the product in preparation for release. Continuously assess release readiness based on your release level product goals. Count on unforeseen work to emerge during this last stretch of development.

User Story Mapping

A guide to USM and Agile Product Management

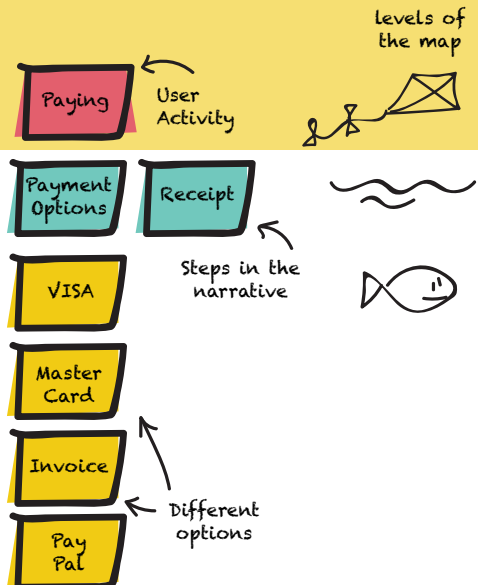


The User Story Map as the Product Backlog - with one, or several teams

A lot of times when using a User Story Map for a big project, you end up not doing about 50% of the stories. This is due to that those are not needed to reach our goals and satisfy our users. This of course saves a lot of time and money, that we can spend on the next prioritized goal.

When working Agile we want to keep our teams stable over time and they will release incrementally, and iteratively, and the teams always maintain what they built.

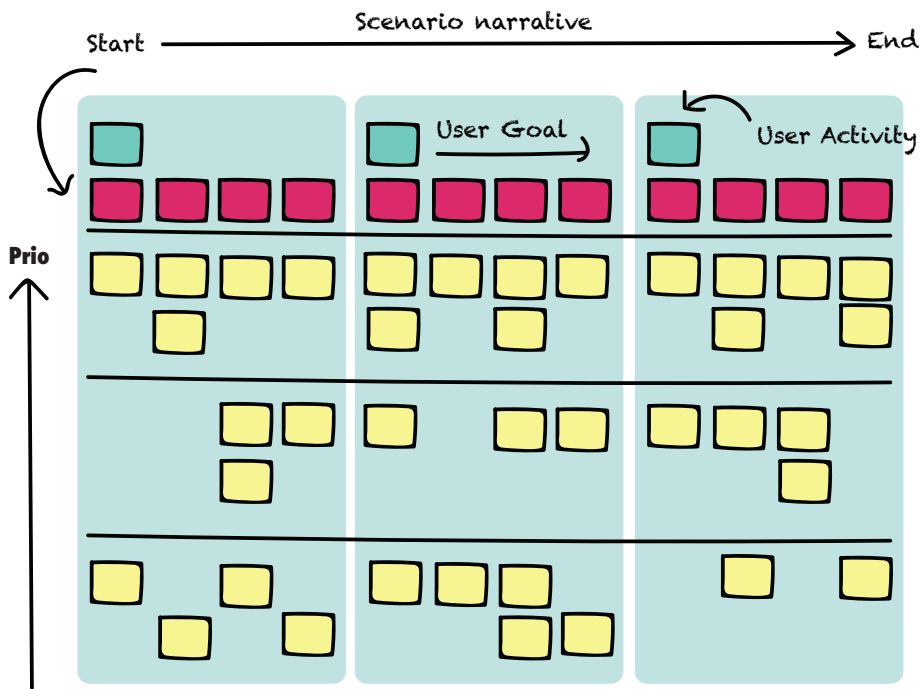
The user story map is a great way for the product owner to facilitate that ownership over time since the backbone of a process always stays the same they can add in new stuff when they learn about it, and re-plan their sprints and release goals. The map is also a great way to keep stakeholders in the loop on what the teams are working on, and what not. It is super transparent.



User Story Mapping is a method created by Jeff Patton.
This is part of the Dandy People Product Management training materials.



From product development work done for procurement of a new planning system for the Royal Swedish Opera.



Colour coding for different personas in the map



Release #1 enabling the user scenario "Start an account, put in money and use it online" for the persona "Millennial Millan"

Release #2 enabling the user scenario "Send and receive money from friends" for the persona "Millennial Millan"

Not prioritized
We just pull down what's not needed now. Usually we learn that lot's of ideas that we had just wasn't needed to create a great user experience and to enable the user goals we had in our customer journey.



The full end-to-end customer journey and experience done together with the development team before taking on a big new project together.

With one team...

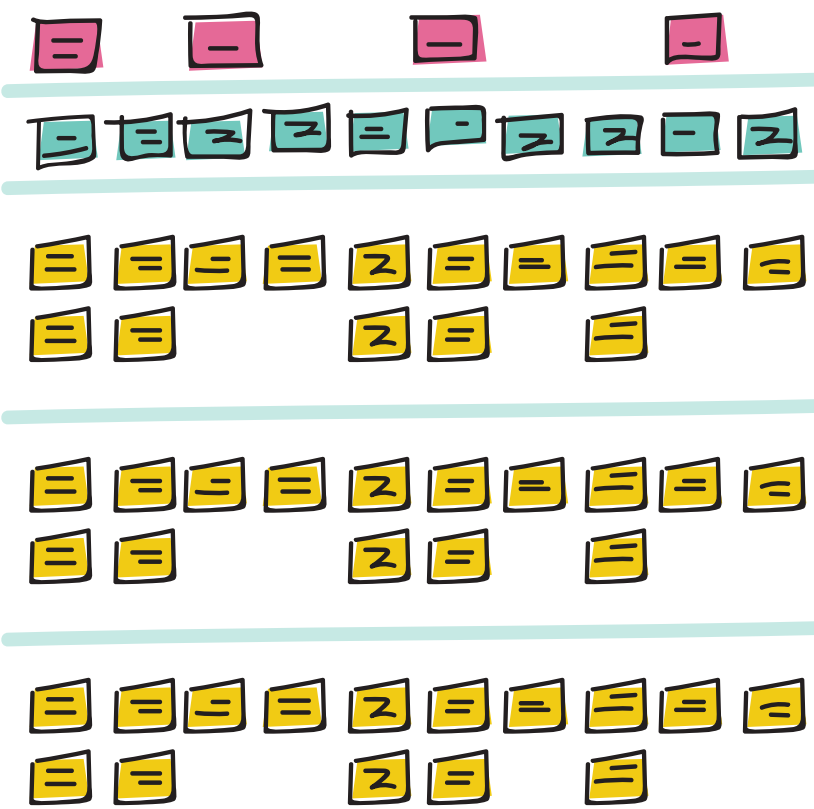


With one team - The full user story map can be delivered by one team. It most likely is a small product, or a part of a product that the team is focusing on, or you keep the same user story map over a long period of time, as the main product backlog, and just add in new findings and opportunities in to the user story map as they occur.

Several teams in a big product or service several teams can easily collaborate and use one user story map as their shared backlog. Based on the release goal they each sprint pull from the user story map into the sprint backlog - making it easy to co-ordinate and deliver valuable thin slices also together. The team can then more easily self-organize around shared goals and get transparency between the teams.

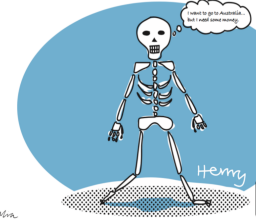


... or with several teams as a shared program backlog.



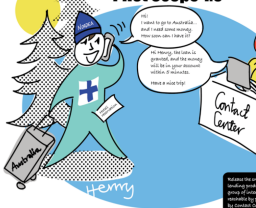
Release planning, visualizing release goals with graphics to get an aligned view over +10 teams.

Walking Skeleton Scope



Release #1

Contact Center by Phone



Release #2

Mobile Bank



Release #3