



USER STORY MAPS: OVERVIEW AND EXERCISE

Jeff Bitner and Paige English
October 2, 2018

TOPICS

- **What are User Story Maps?**
- **Benefits of User Story Maps**
- **User Story Map Overview and Process**
- **Resources**
- **Group Exercise**

WHAT ARE USER STORY MAPS

- A popular visual and collaborative technique invented by Jeff Patton for discovering User Stories and Planning Releases and Sprints
- An approach to collaboratively creating a “walking skeleton” of the System’s major activities and tasks
- A means to discover and prioritize User Stories based off of the “Walking Skeleton”

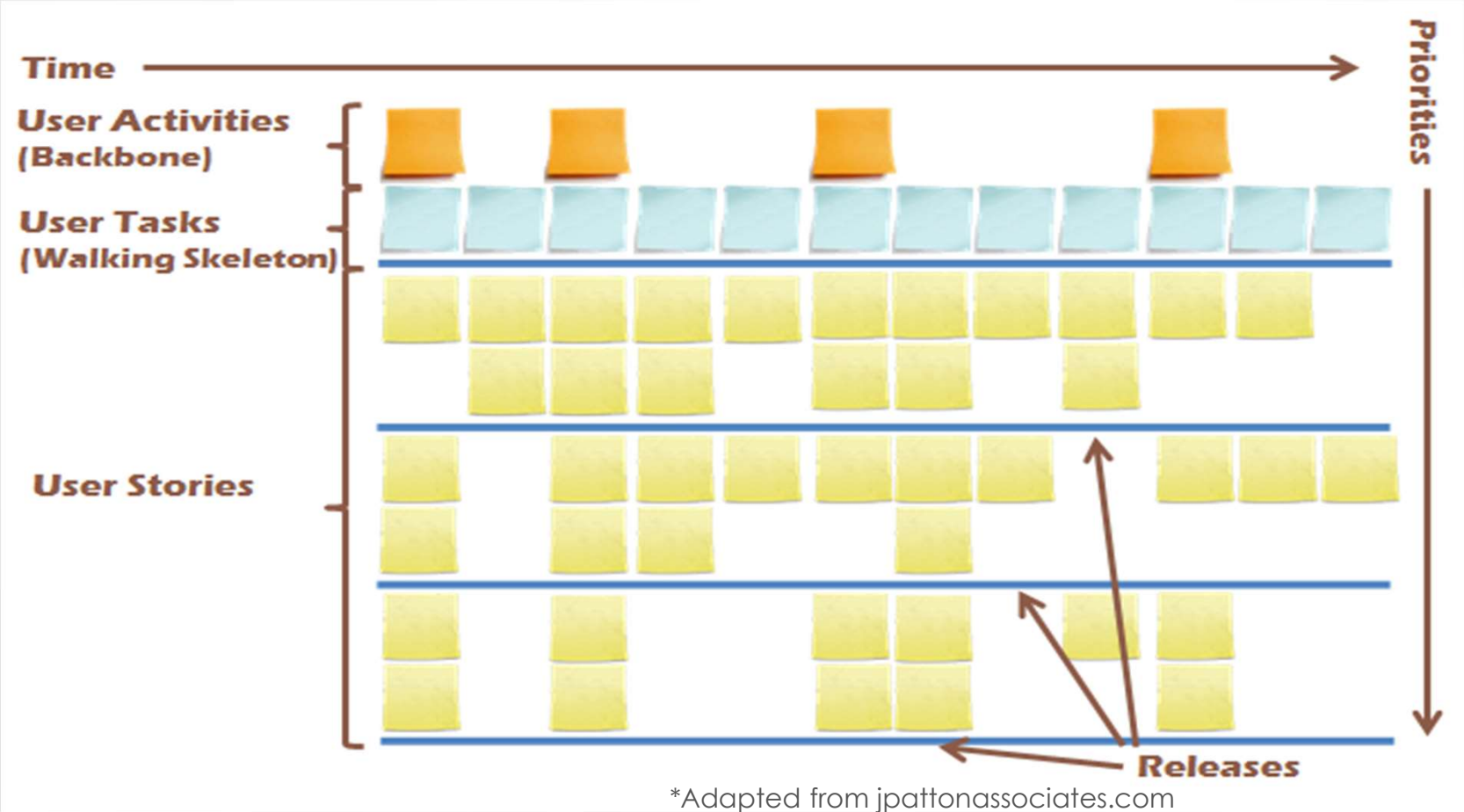


BENEFITS OF USER STORY MAPS

- Allows **teams** to discover stories in context of a larger, consistent vision (i.e. avoid a “Frankensystem”)
- Allows teams to see the big picture in their backlog
- Provides a useful visual model and tool for discussing, prioritizing, and managing scope
- Promotes collaboration and brainstorming in discovering and grooming user stories
- Fosters looking at a product or feature from the users’ perspective (much like Use Cases).



BIRDS-EYE GRAPHIC



*Adapted from jpattonassociates.com

GRAPHIC WITH ADDITIONAL DETAILS

Users

A map tells a story about a type of person doing something to reach a goal. Make sure to include them in your map along with a little information about them.

Try using lightweight persona sketches to describe your users.

User Tasks

User's tasks are short verb phrases that are the basic building block of a map. If I ask you what you did earlier today when using email, you'll likely respond with tasks like:

- Read an email message
- Respond to a message
- Mark a message as spam

Goal-Level

The actions that users take in order to reach their larger goals have a goal level themselves that's tied to user behavior.

Summary: lots of tasks done in support of a bigger goal.

Functional: I'd expect to complete this task before taking a break.

Sub-Functional: smaller things done in support of a bigger task.

As you read across tasks in the backbone, check to make sure that tasks are of a similar goal level.

Activities

Activities organize tasks done by similar people at similar times to reach a goal. For your email software activities might include:

- Going through my inbox
- Configuring my email client
- Organizing messages into folders

Backbone

Activities and tasks at a higher goal level give the story map its structure. The backbone is arranged in a narrative flow. Smaller sub-tasks, details and variations hang down to form the ribs connected to the backbone.

Narrative Flow

The left to right axis in a story map is organized in the order you'd tell the story about your user to someone else.

Of course any specific user might choose to do different things in a different order. Use conversation to explain the details and variations.

If you're looking for the precision of a workflow model, flow chart, or UML model, then a story map isn't your best choice.

A story map will take lots of conversation to use effectively. But then that's the purpose of stories.

Release Slice

Use a tape 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.

Use release slices to identify small experiments, minimal viable product releases, or a "walking skeleton" version of your product.

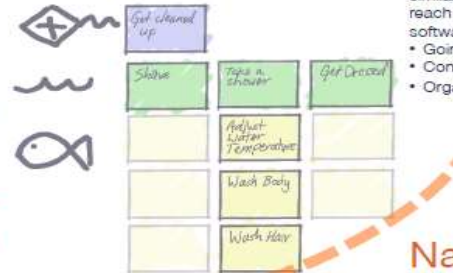
Identify the target outcomes of your slice in a sticky note or card to the left of the slice.

Details, Details...

Break down high goal level tasks into:

- Sub-tasks
- Alternative tasks
- Exceptions
- Details

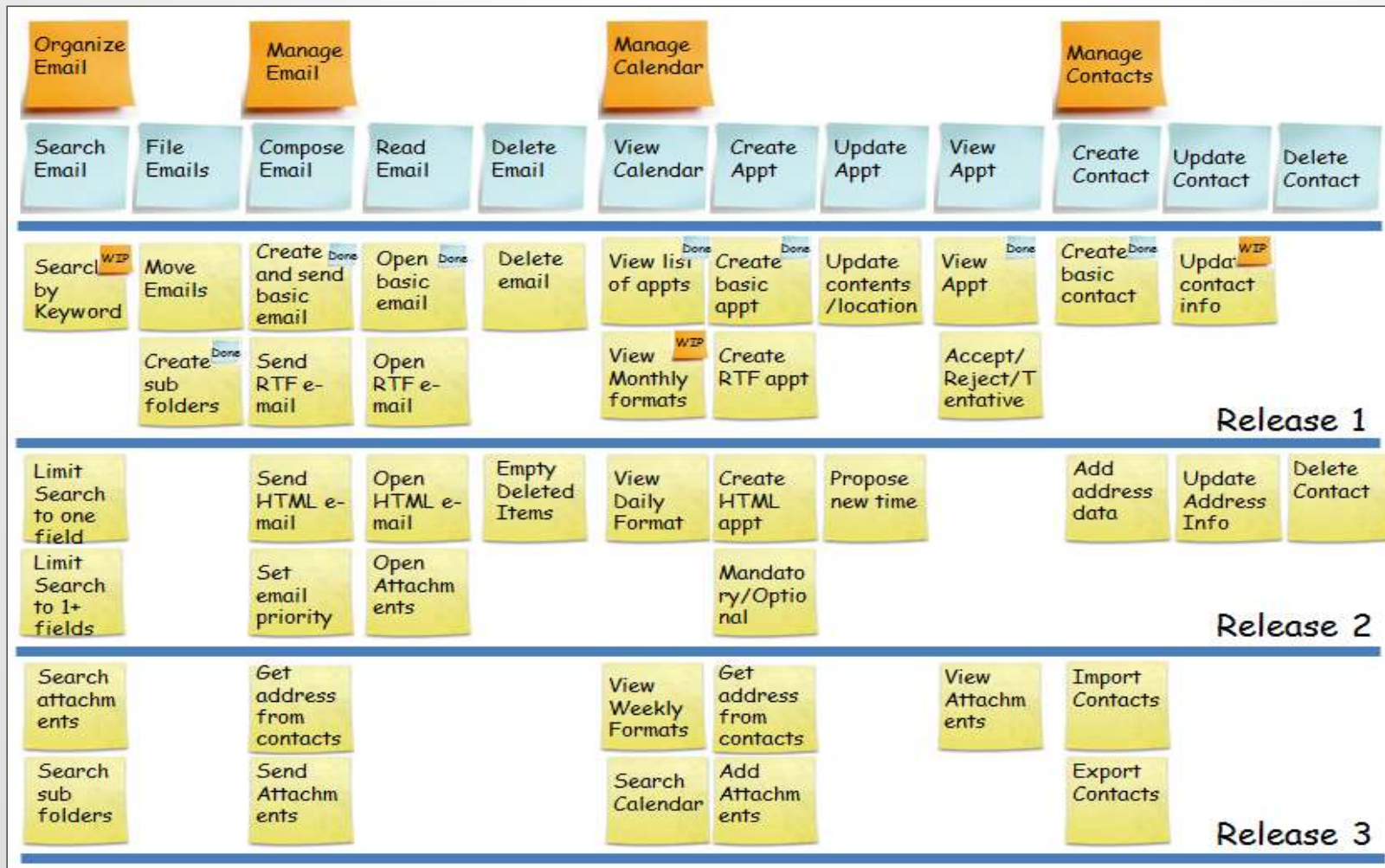
Down in the details of the map, it's OK to include details about what UI might look like or what the system might do in the background.



User Tasks make great story titles!
Write short verb phrases on cards or stickies. Use them later as your story titles. If you use the story template to write descriptions, the task fits nicely right after "I want to," the activity fits nicely right after "so that..."

*Adapted from jpattonassociates.com

SIMPLE EXAMPLE – EMAIL SYSTEM



*Adapted from jpattonassociates.com

NOTIONAL USER STORY MAP “PROCESS”

| | |
|---------------------------|---|
| Frame | Create (or review) a short product or feature overview (purpose of the product/feature, different types of users/customers using it, and major benefits) |
| Map the Big Picture | Identify Activities and High-Level Tasks for each user (mile-wide, inch deep). (This may lead to additional discoveries of users, activities) |
| Explore | Fill out the body w/ candidate story details, (Look for variations, exceptions, rules, UI, data). You may (again) discover new tasks/activities/users or combine/split stories. |
| Slice out Viable Releases | Slice the map into product releases that make sense. Describe the target outcome and impacts. |

ALTERNATIVES TO USER STORY MAPS

- **Some alternatives to User Story Maps include:**
 - Use Cases
 - Process Diagrams
 - Work Breakdown Structure
- **Since none of these techniques were specifically invented for Agile, some issues with their use include:**
 - Dogmatic adherence to full-blown technique vs. adaptation to suit current purpose. (i.e. increased time, heavyweight)
 - Non-Acceptance by team members that are “Agile Purists”
 - The dreaded “A-Word” (Audit) against traditional checklists
 - Risk of non-collaboration

RESOURCES

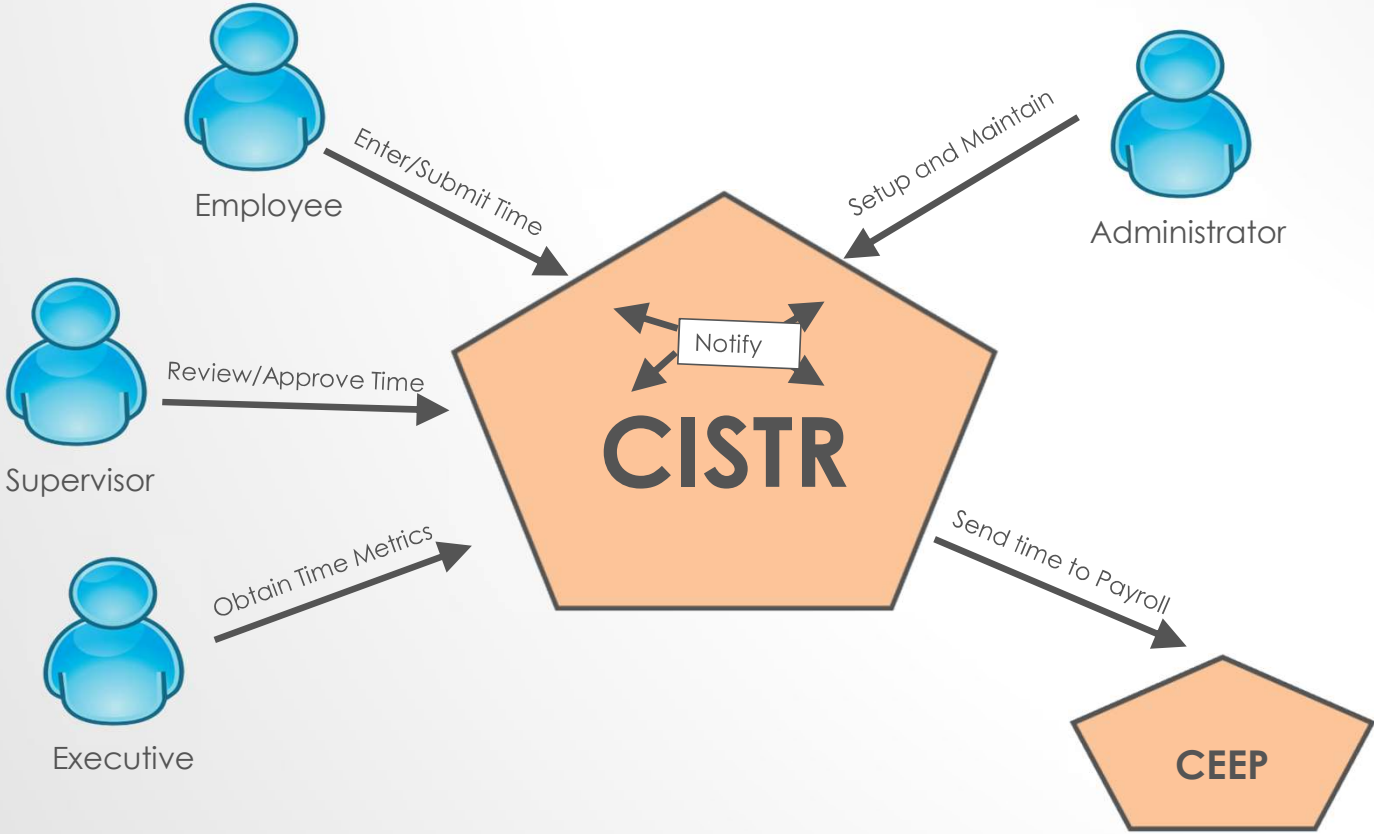
- [Jeff Patton Web site](#)
- [Amazon User Story Maps Book](#)
- [Winnipeg Agilist](#)

GROUP EXERCISE OVERVIEW - **TIMESHEET SYSTEM**

Overview: The Corporate Integrated System for Time Reporting (CISTR) is being developed to replace the existing, paper-based timesheet system. It is intended to provide the following benefits:

- A streamlined, intuitive, and efficient way for employees to enter, submit, and correct timesheets. Efficiencies include allowing an employee to create a shell of their previous timesheet to use in the current pay period or set up timesheet templates to use for time periods.
- A streamlined, intuitive, and efficient way for supervisors to approve/reject timesheets.
- Integrated notification and workflow of timesheet submission and subsequent approvals/rejections improving timesheet turnaround time.
- Increased quality of timesheet data resulting from data validation by the system by only allowing employees to record time to project charge codes (including time for holidays, vacation, sick days, etc.) for which they are authorized.
- An efficient and on-line ability for administrators to setup and maintain timesheet reference information (e.g. time periods, charge codes, employees)
- Automated integration with the Corporate Electronic Enterprise Payroll System (CEEPS)
- Better Visibility into Metrics and Reporting for Executives

GROUP EXERCISE - VISUAL



TIMESHEET EXAMPLE

Set up and
administration

Initial system
set up

Ongoing
Maintenance

Establish
Charge
Codes

Enter users
and
permissions

Set up Time
Collection
Periods

Map users to
charge codes

R1

GROUP EXERCISE INSTRUCTIONS

| Activity | Time (mins) |
|---|-------------|
| Identify several High Level Activities as a group. | 5 |
| Split into teams. Each team identifies the High-Level and some user tasks for a selected activity | 10-15 |
| Half of each team switch to the other team and discuss, question, elaborate on your stories. | 10 |
| As a group, try to break out into R1 (MVP) and Non-MVP. | 10 |

Any Questions?? Ask the Product Owner