

# WHAT'S YOUR SKATEBOARD?

## EMILY STAMEY

© PHOTO BY KENNY LOUIE





I'm a PHP developer

I've often worked directly  
with non-technical product  
owners and users

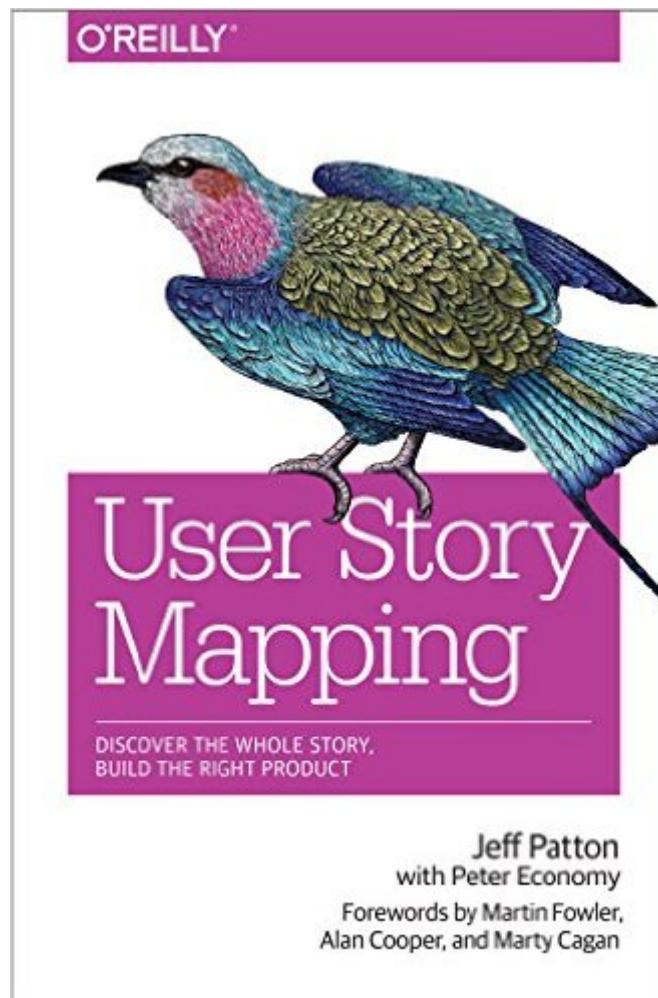
I workED at a University

INQUEST

WOMEN WHO  
**CODE**



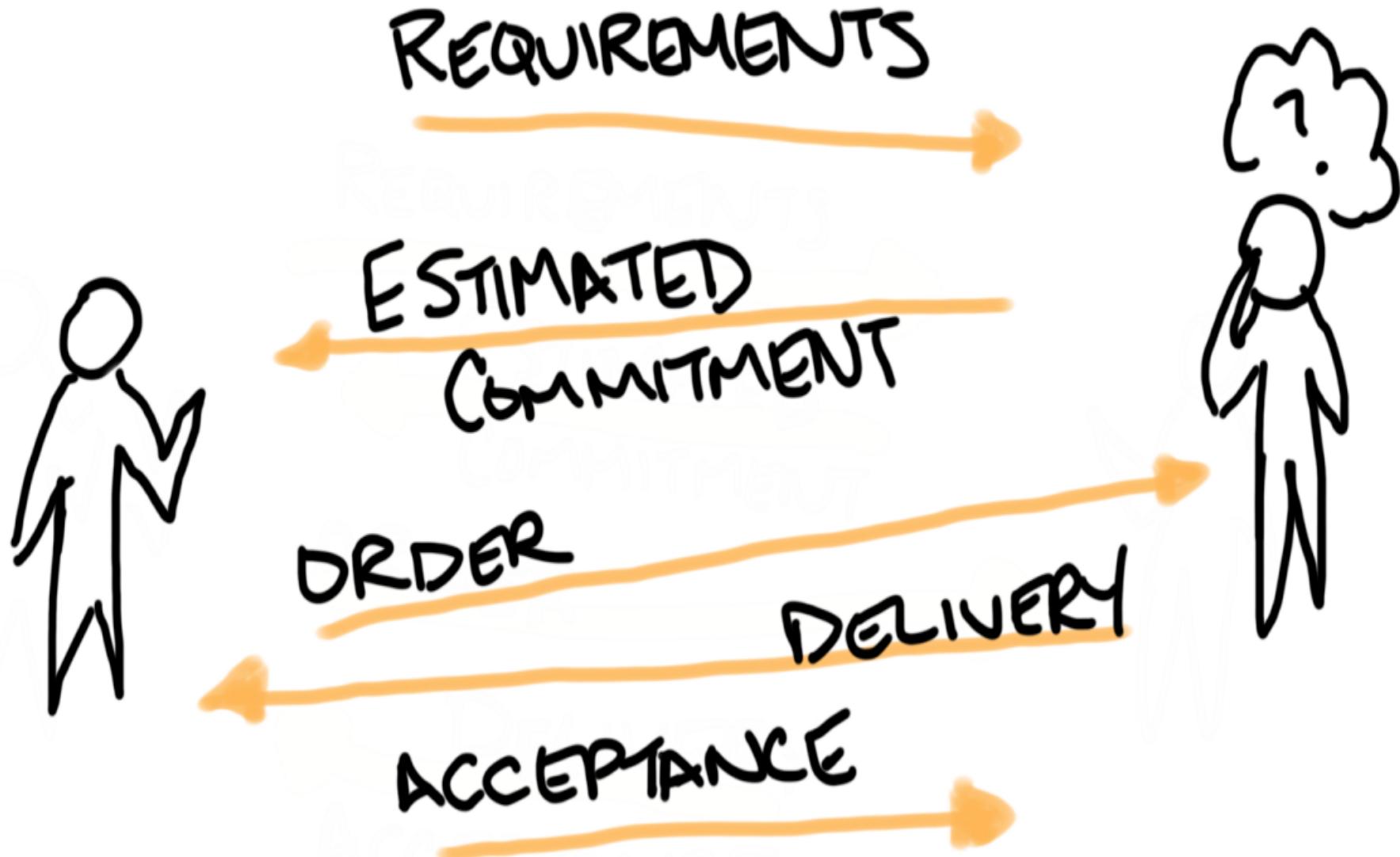
# INSPIRED BY



# A PROBLEMATIC PROJECT WORKFLOW

1. Research
2. Develop over several months
3. Deliver something
4. Learn if the big guess was correct

# THE CLIENT-VENDOR ANTI-PATTERN



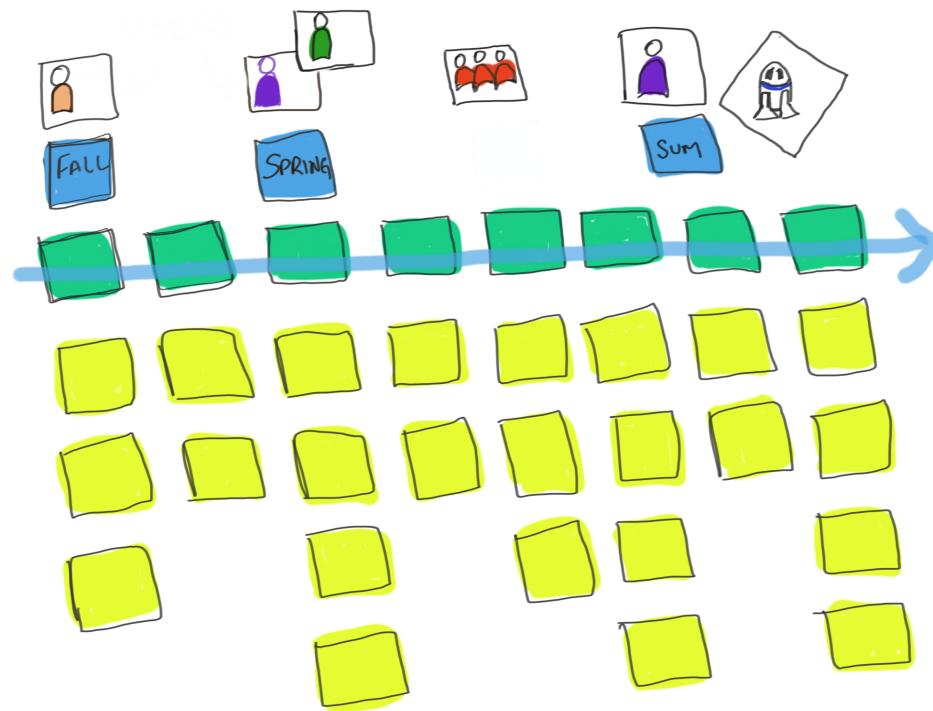
# WHAT IS A STORY MAP?

A diagram of a project that tells the story of the people and systems involved in a process.

Detail is added as we learn more about the project

The map can be built for an existing application or a new application.

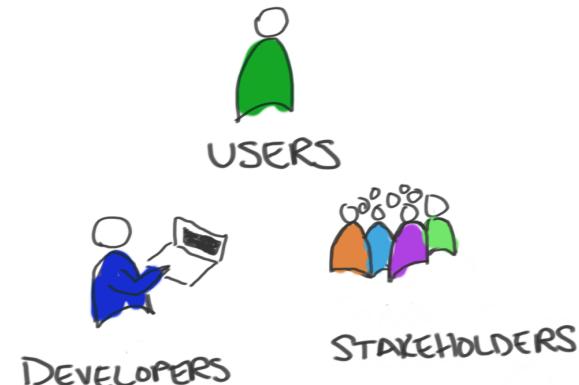
# THE STORY MAP



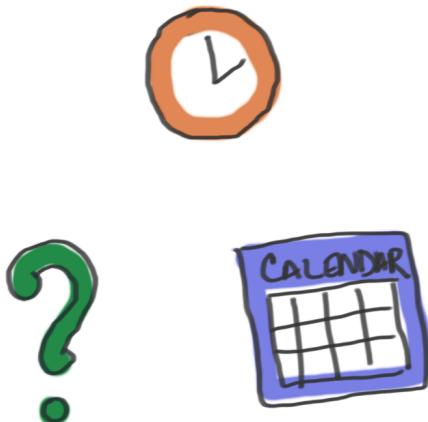
# WHO SHOULD STORY MAP?

Anyone who knows the  
process ... Not Just  
Developers

At least one knowledgeable  
person from each group of  
stakeholders



# WHEN TO STORY MAP?



- When you have questions
- Before You begin development

# WHY STORY MAP?

- *Build shared understanding*
- Encourage full discovery before prototyping
- Prioritize work to be done as a group

\*Lowers problems with estimates and feature creep

# THE BIG PICTURE



© PHOTO BY BARNYZ

# FOCUS DURING DEVELOPMENT



© PHOTO BY THEILR

# WHERE?

A large, clear wall or whiteboard.

A place central to the team, at least in the beginning.

# YOU'LL NEED

- painter's tape
- markers
- post-its (many colors & sizes)

# A SIMPLE STORY

List five tasks you do in order to get to work. Put each one on a post-it provided.



# MY MORNING AT WORK (FIRST DRAFT)

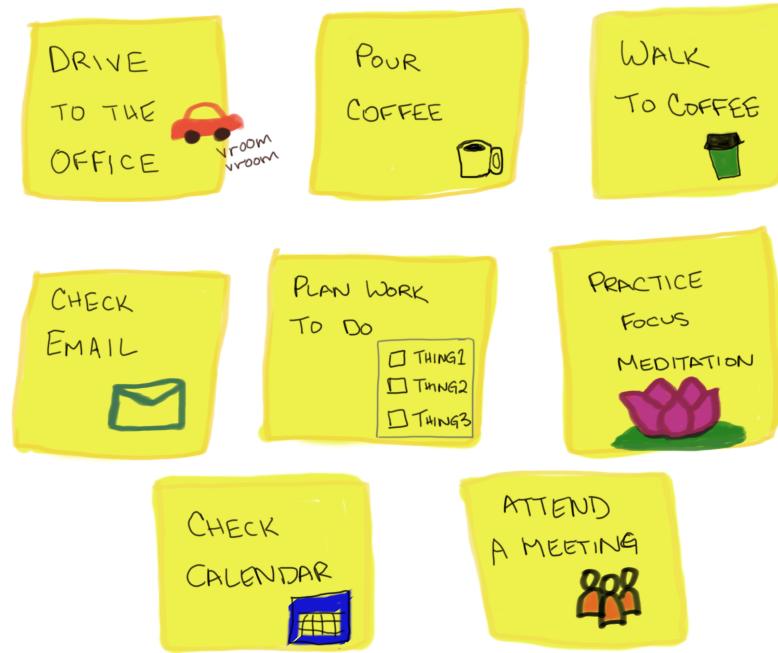


# FIND VARIATIONS

Pick 3 things that you did today that are different from your normal work routine.

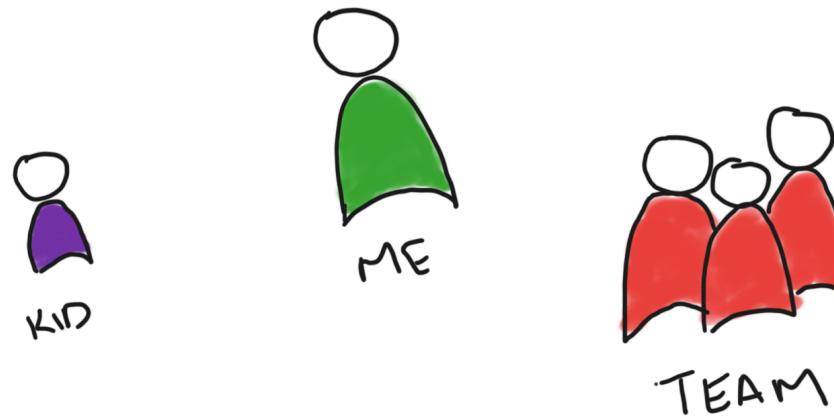


# MY MORNING AT WORK (REVISED)



There are variations between people, days of week, at-home and out-of-town work days

# BUILDING THE NARRATIVE: USERS



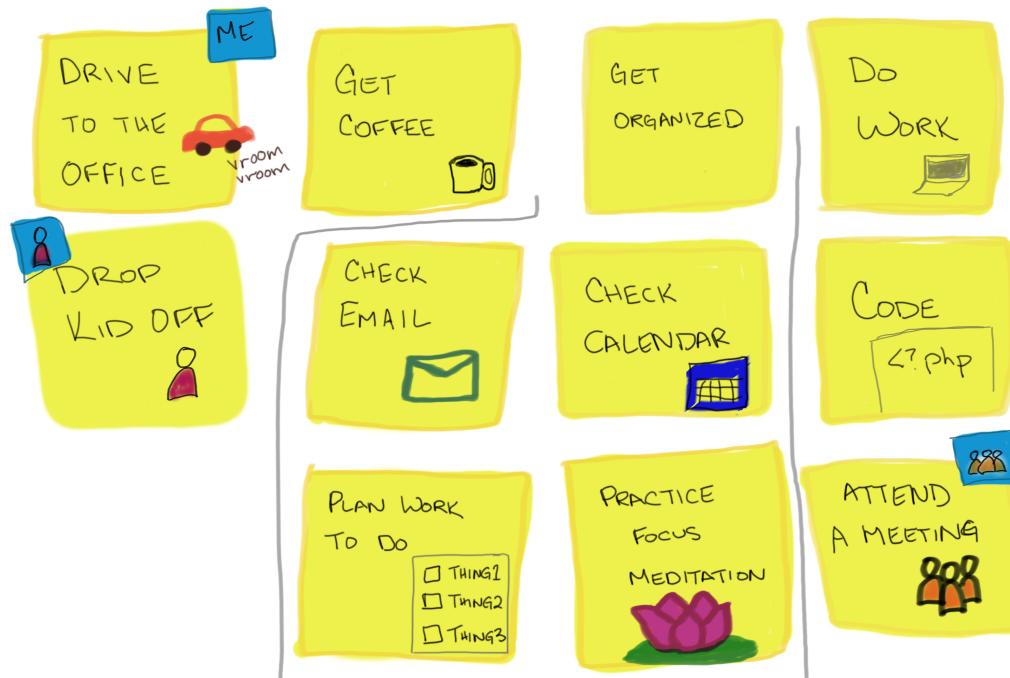
Who are the Actors/Users in your morning routine?

# BUILDING THE NARRATIVE: ACTIVITY GROUPS



Sometimes several tasks can be grouped together

# BUILD THE STORY



Let's put them together to build a story of your getting to work!

# LEARN MORE

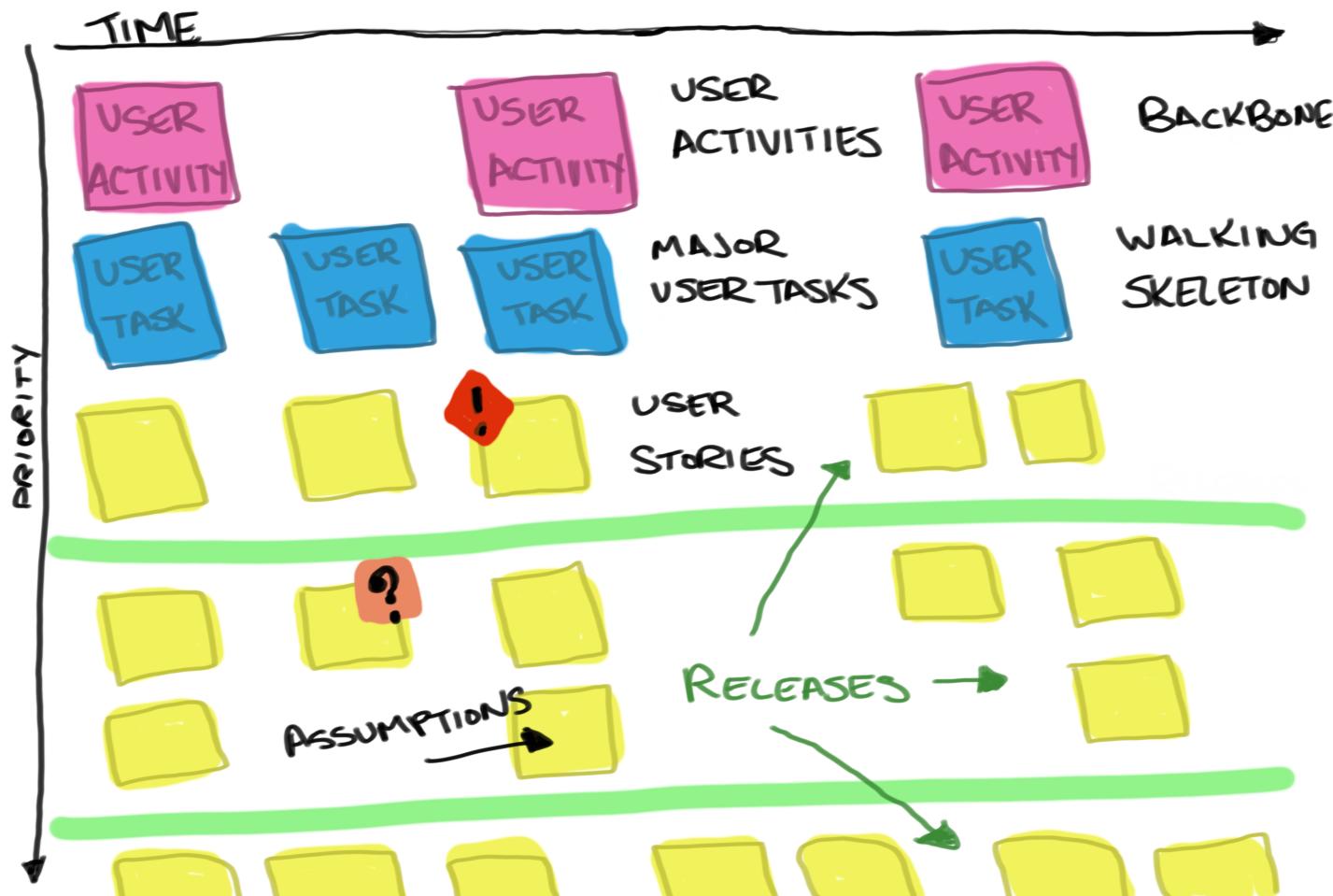
- Risk
- Assumptions
- Uncertainty

If you cannot elaborate, mark it and revisit

# CLARIFYING QUESTIONS

- What could we learn to replace **risk** with **REAL** information?
- Do we really know what has been mapped, or did we fill in **assumptions**?
- Are you sure about the story you're telling?

# LABEL THESE

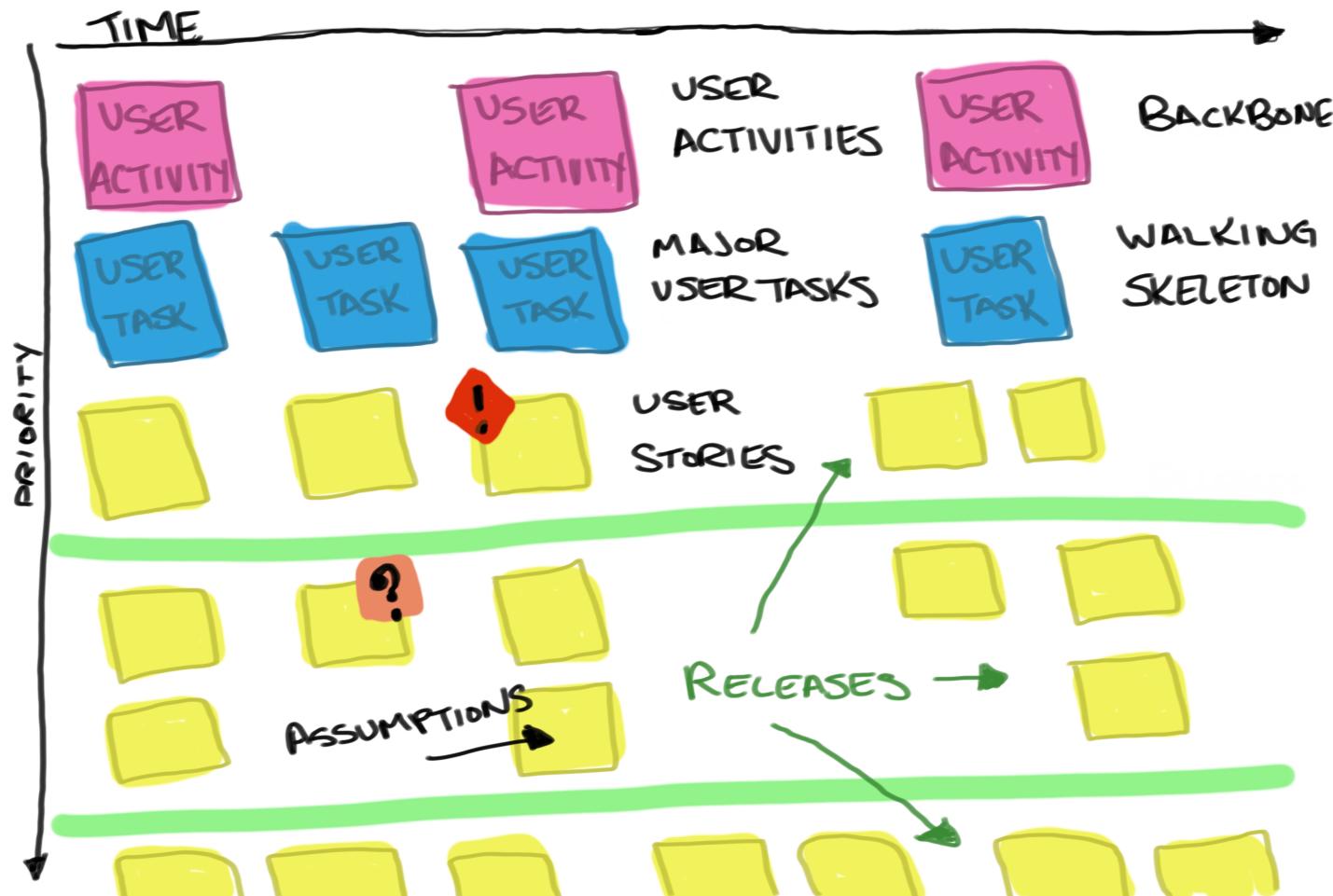


# THOROUGH DISCOVERY

- Understand the full process
  - Understand "why" steps are needed in the process
  - Talk about things inside and outside of the app

# THOROUGH DISCOVERY

- Simplify and lower risk at implementation
  - Lowers the questions at the phase of implementation
  - Limits Feature Creep (beginning implementation w/o understanding, new features come in)
  - Better estimates



# MORE THAN JUST ADDING STORIES

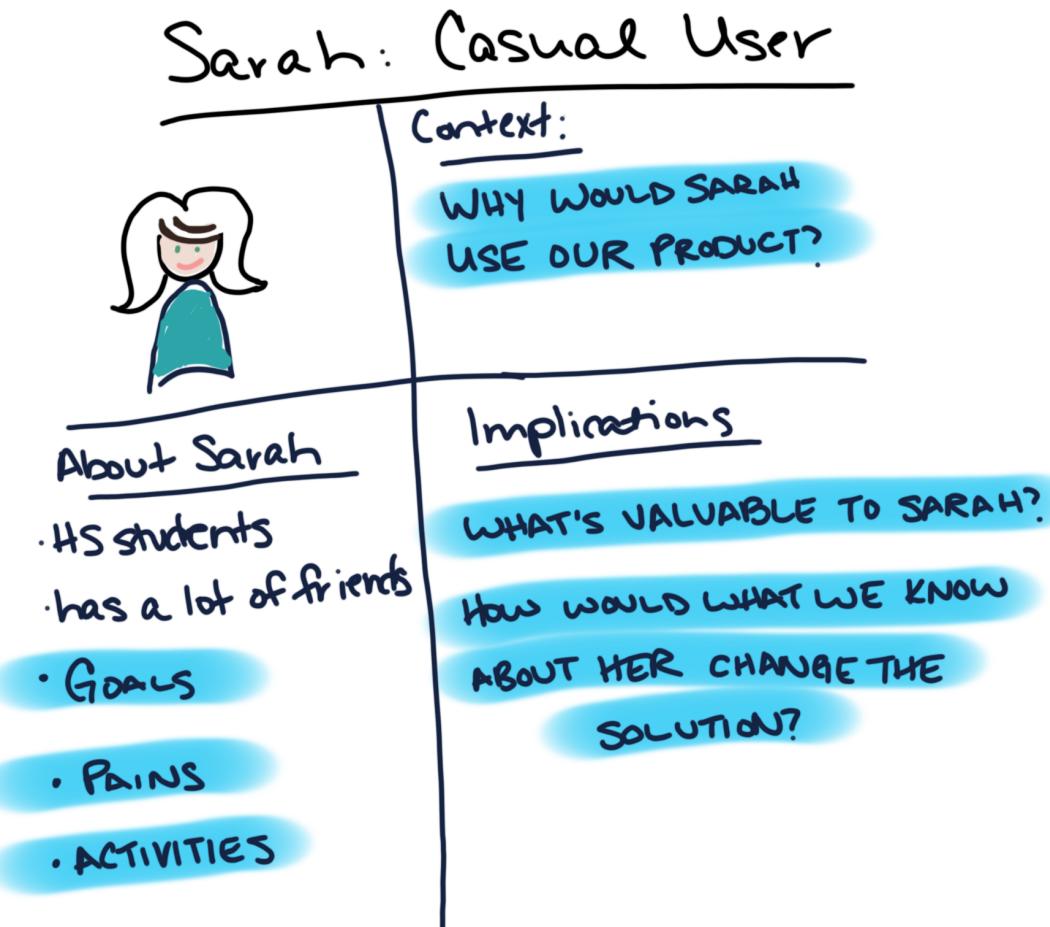
“ If the only thing you create while making sense of a big opportunity is more, small stories, **you're doing it wrong.**

- Jeff Patton

# FOUR STEPS TO DISCOVERY

1. Frame the idea
2. Understand Customers and Users

# SIMPLE PERSONA



# FOUR STEPS TO DISCOVERY

1. Frame the idea
2. Understand Customers and Users
3. Envision your solution
4. Minimize the plan

# PRIORITIES

# PRIORITIZING THE PROJECT

- Who will use this product?
- What steps must they accomplish for success?
- Remove/postpone the rest

# PRIORITIZING FEATURES

- **Differentiator** - feature sets you apart from competition
- **Spoiler** - moves in on someone else's differentiator
- **Cost reducer** - reduces organizational costs
- **Table stakes** - feature necessary to compete

# FOCUS ON OUTCOMES

- What are you hoping to do with your application?
- Prioritize features based on the problem they solve
- Implement only what solves the problem or meets the objective

# PROTOTYPING

# PROTOTYPING

- What is the smallest thing you could build to prove/disprove an assumption?
- Sketch & prototype to test viability of the solution
- Aim for less than minimum, get feedback, and iterate often
- When you give prototype to **development partners** you can include metrics to see what they actually do

# MVP VS. MOST VALUABLE FEATURES

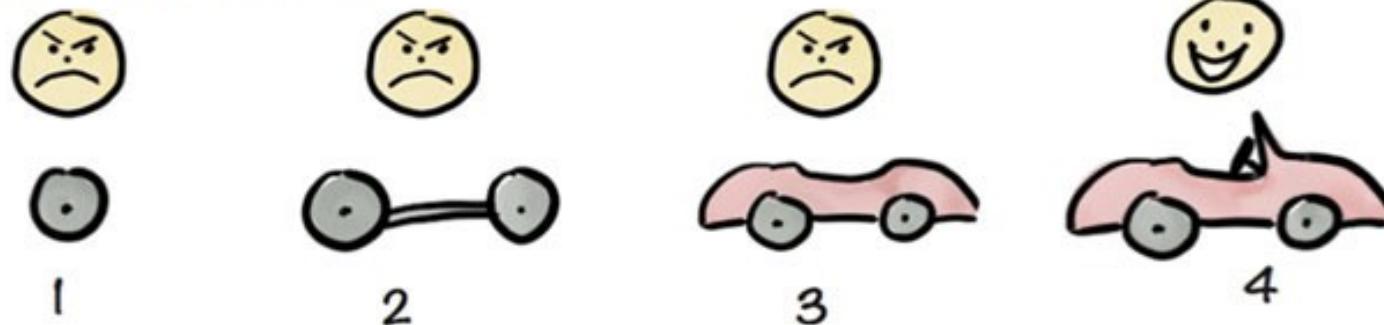
Focus on releasing valuable features every time.

Sometimes we plan features in a chronological order

Or we divide the project into components

# WHEN WE BUILD IN PIECES

Not like this....

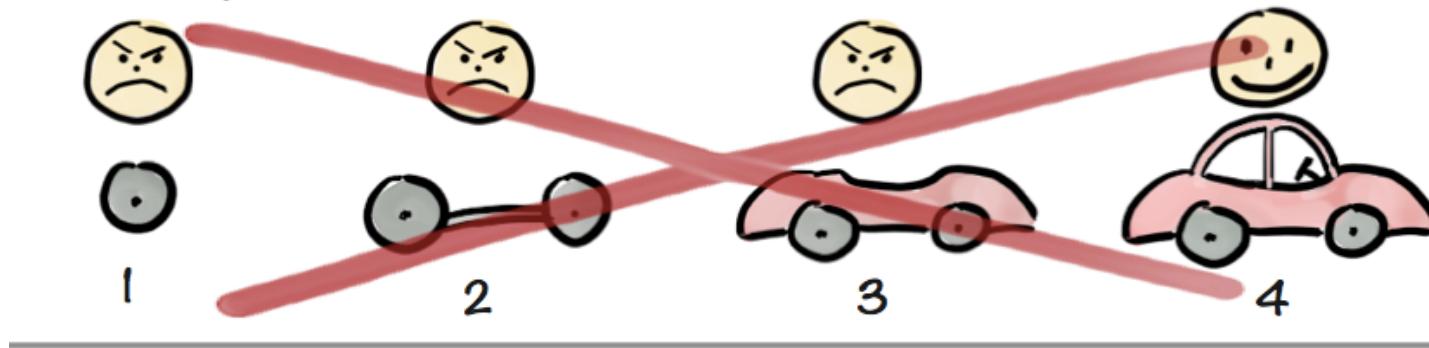


# WHEN WE BUILD IN PARTS

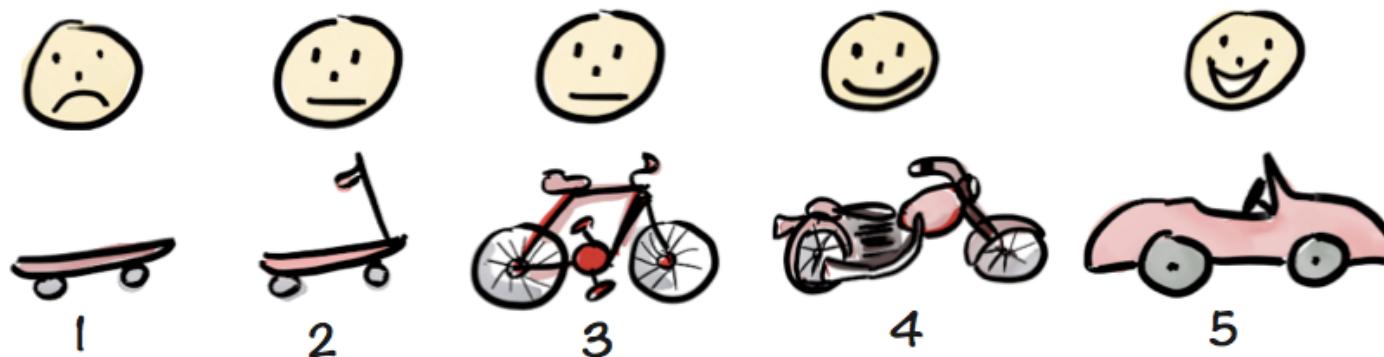


# INSTEAD WE WANT TO ITERATE!

Not like this....



Like this!



Henrik Kniberg



1



2



3



4

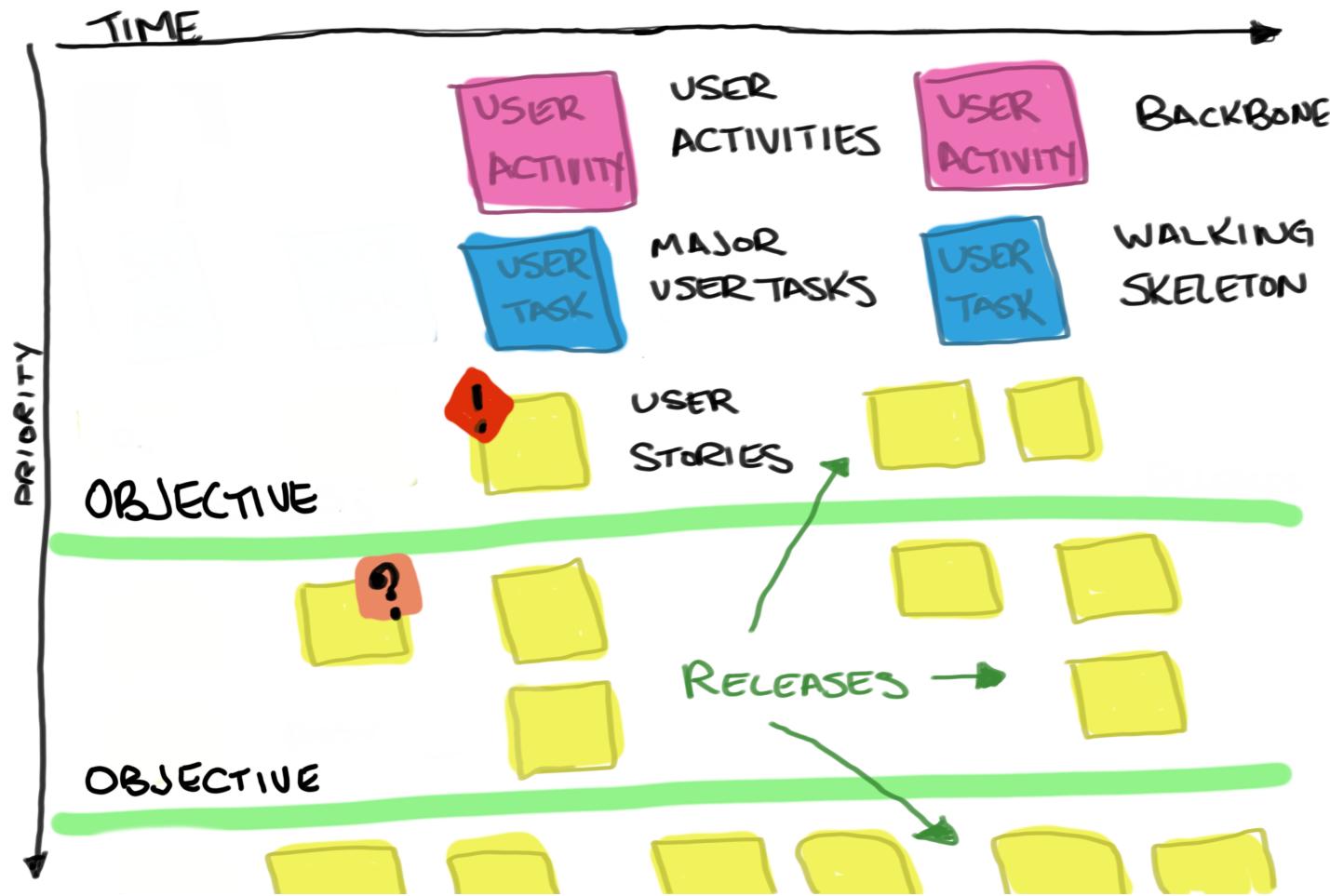


5

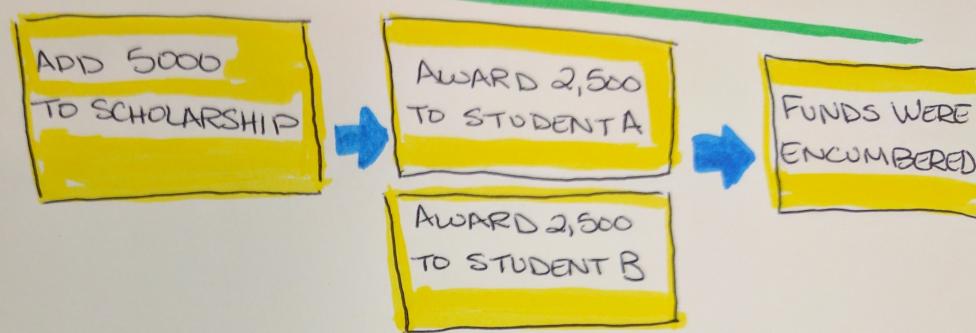


# STRATEGIES



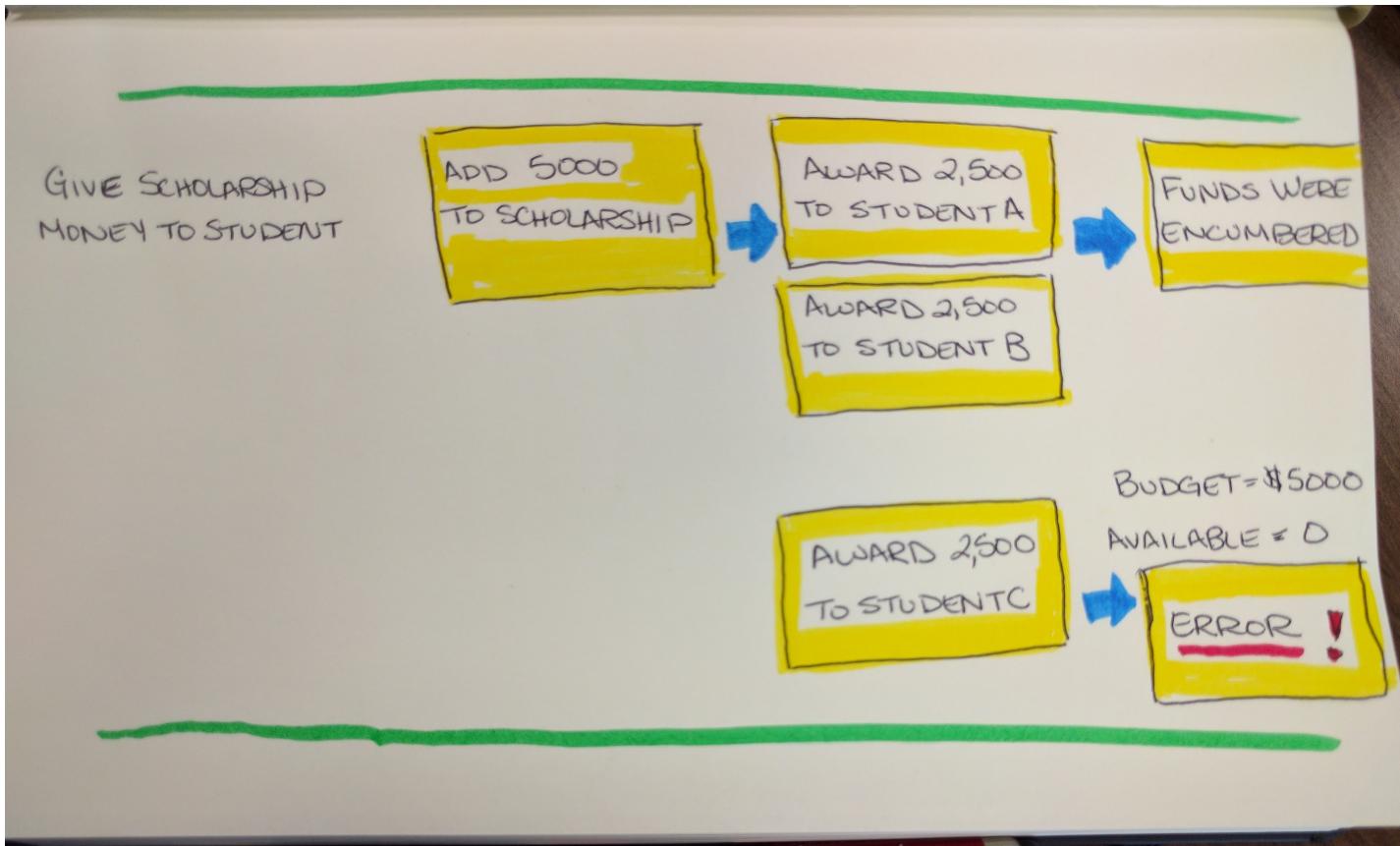


GIVE SCHOLARSHIP  
MONEY TO STUDENT



BUDGET = \$5000

AVAILABLE = 0



# SUMMARY

- A tool you can use with non-technical subject matter experts, customers, etc
- A visual guide for managing your workload
- Focus on objectives when you prioritize
- Plan to deliver a usable product at each deliverable

sooo...

# WHAT'S YOUR SKATEBOARD?



# THANK YOU!



# EMILY STAMEY

@ELSTAMEY

[HTTP://ELSTAMEY.COM](http://ELSTAMEY.COM)