



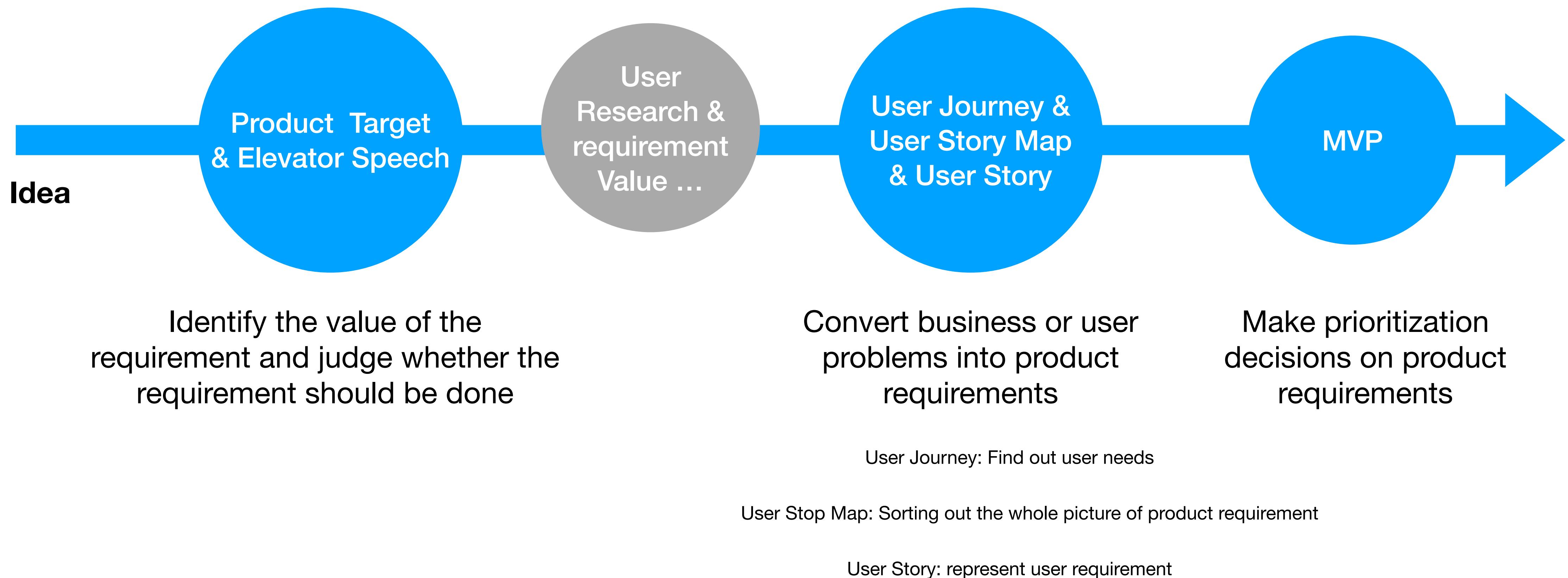
AFS

Agile Full Stack  
Developer Bootcamp  
Thoughtworks ©

# User Journey & User Story Mapping

/thoughtworks

# Product Value



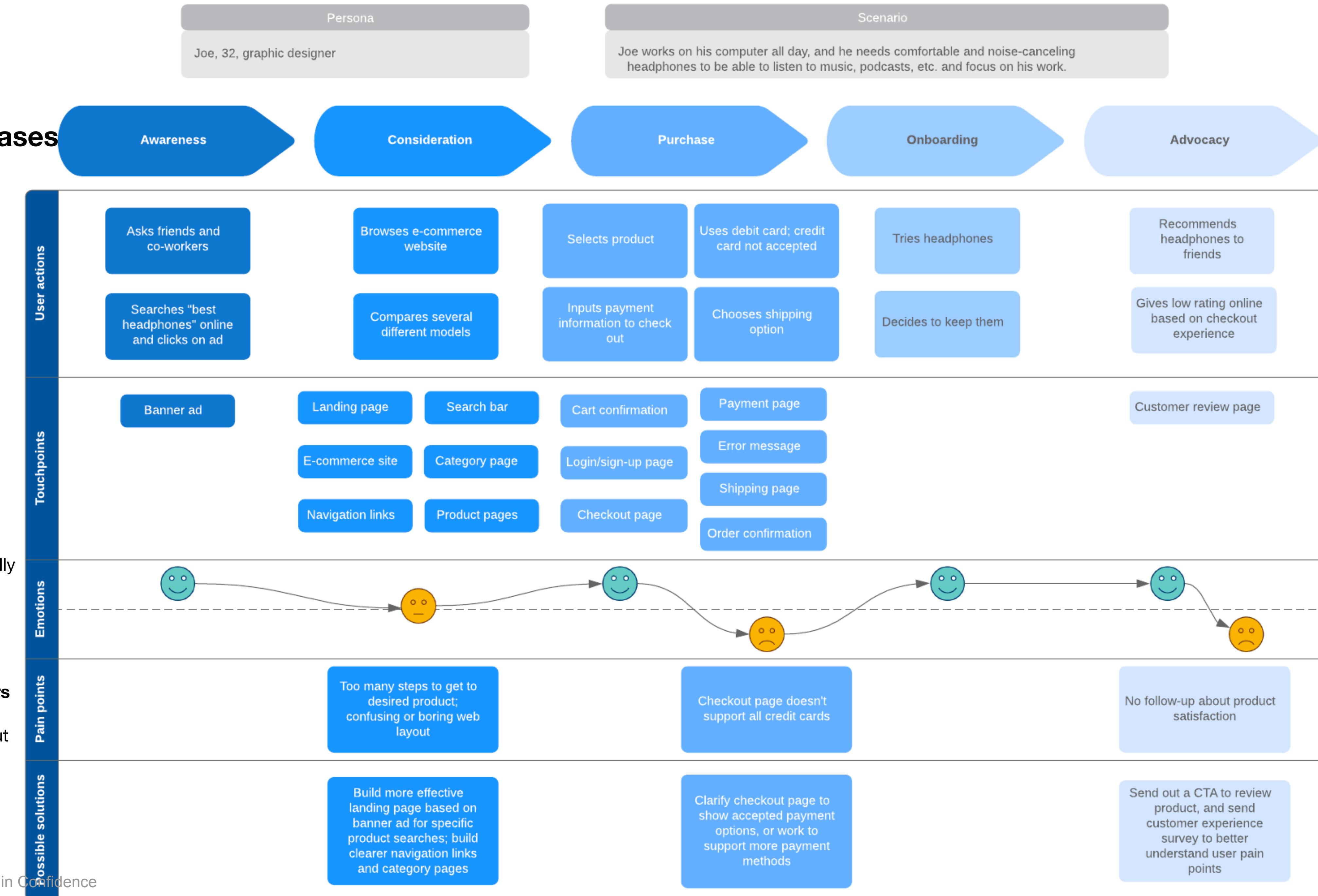
# User Journey

## **A User Journey is:**

- A representation of how a user works through a system to accomplish something they need to achieve (get some value).
- The user journey helps us understand the usage behavior of the client through the time dimension.

# What does it look like?

## Phases

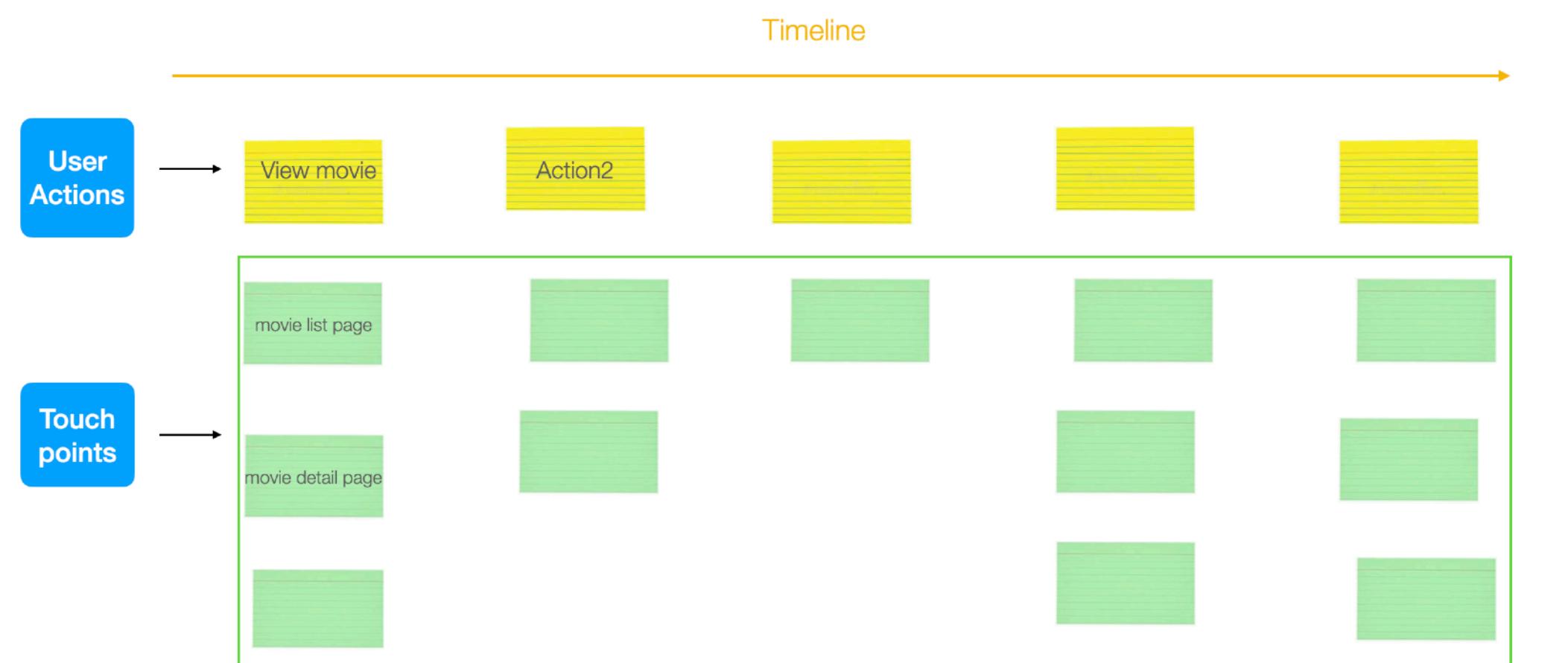


As-Is User Journey		Persona	Target		
Action		Bob	Send a gift to make girlfriend happy		
<b>Choose a gift</b>			<b>Buy a gift</b>		
Action	Girlfriend told me today is her birthday	Go to the mall	Randomly go to a store	Listen to the shopping guide recommendation	impulsive decision
TouchPoint	Mobilephone	Mall	shopping guide		
PainPoint	Forget	Time is too short, so I can only go to the mall in a hurry after get off work to buy	I don't understand at all, the gift is random	Checkout running around, hassle. gift wrap to line up	traffic jam
Possible Solution	personalized reminder	Expert recommendation	Intelligent recommendation	Cards	personalized packaging
	Portable operation of WeChat				Delivery
<b>To-Be User Journey</b>					
Action	Get personalized reminder	Go to the app on WeChat	Select age range	Clothes... Preference	Price range
					Get suggestion
					Select and order
					Select card
					Select package
					Address and time Pay
					Get notification once delivered

# Practice

## 1. User journey: 40mins

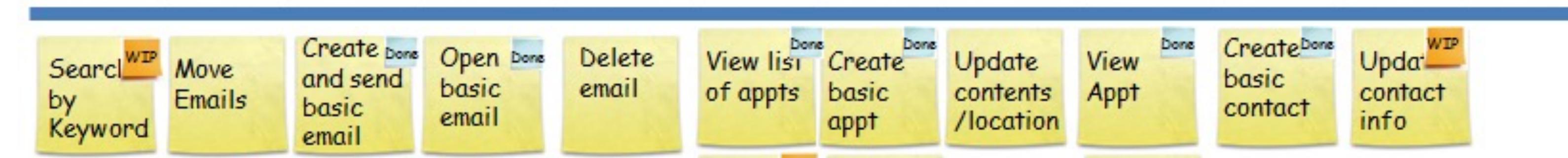
### User Actions



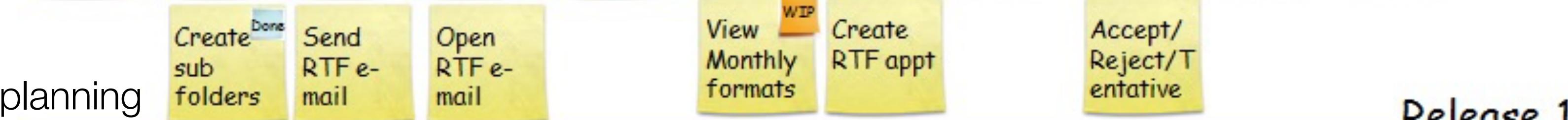
## To-Be User Journey

Action	Get personalized reminder	Go to the app on WeChat	Select age range	Clothes... Preference	Price range	Get suggestion	Select and order	Select card	Address and time
								Select package	Pay
© 2022 Thoughtworks Commercial in Confidence									Get notification once delivered

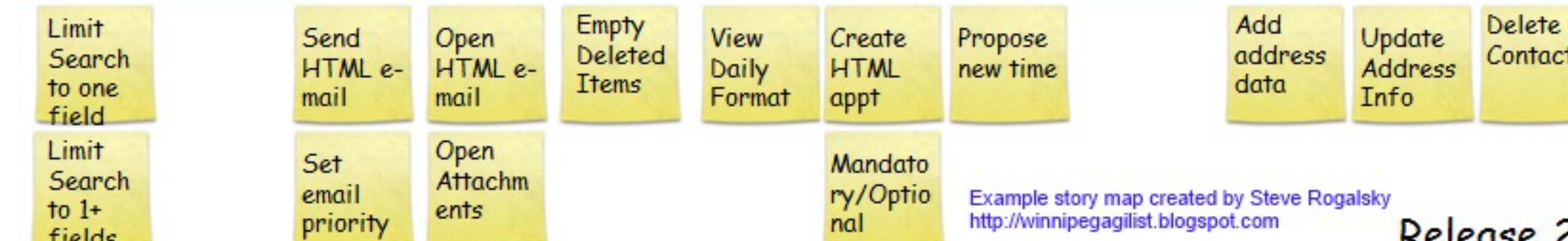
User Story



A release for iteration planning



Release 1

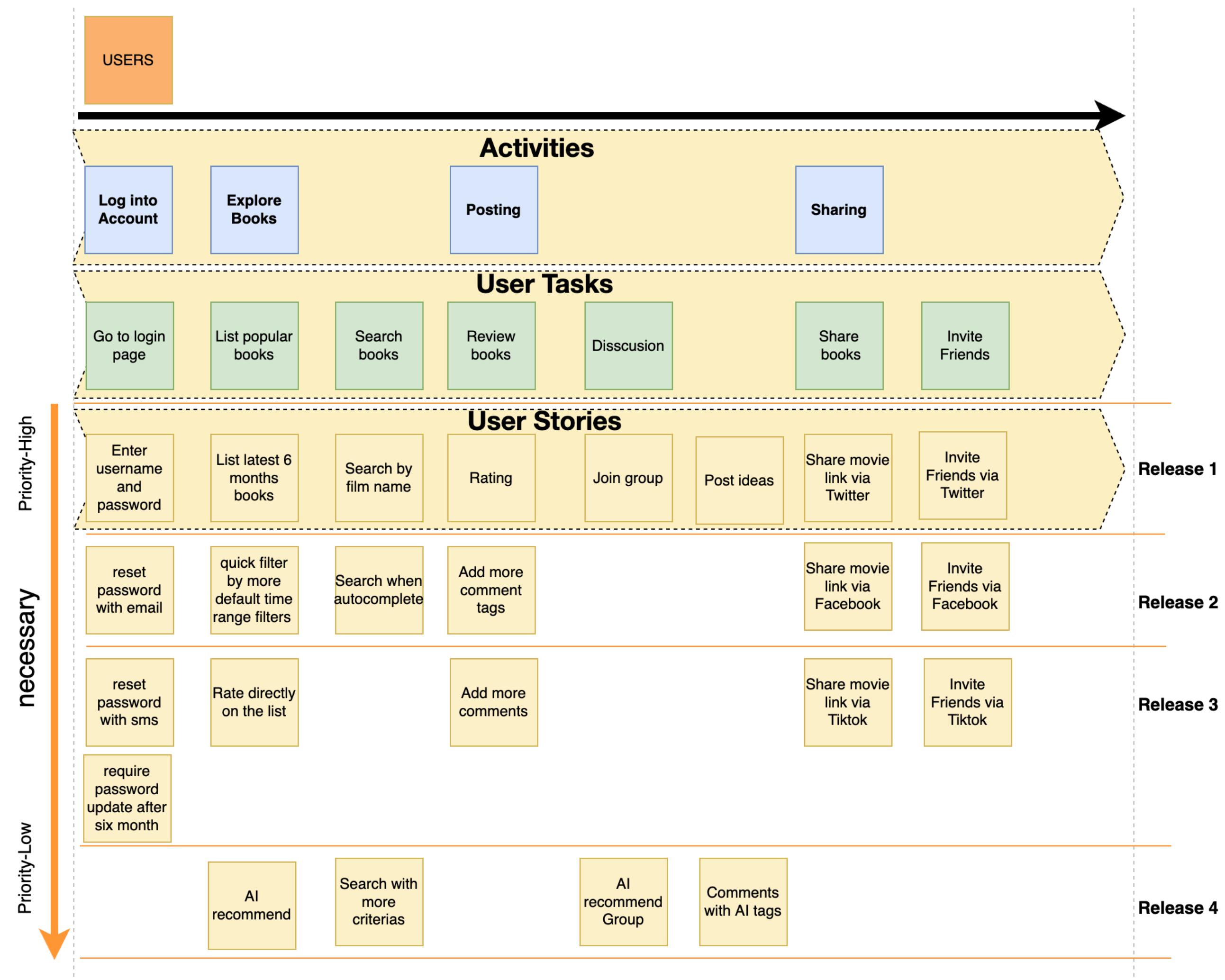


Example story map created by Steve Rogalsky  
<http://winnipegagilelist.blogspot.com>

Release 2

- Prepare the user journey for core users(to be)
- Starting from the core scenario, identify the functional requirements that support each user task
- Other secondary scenarios, identifying the functional requirements that underpin each user task
- Identify supporting requirements that underpin each functional requirement

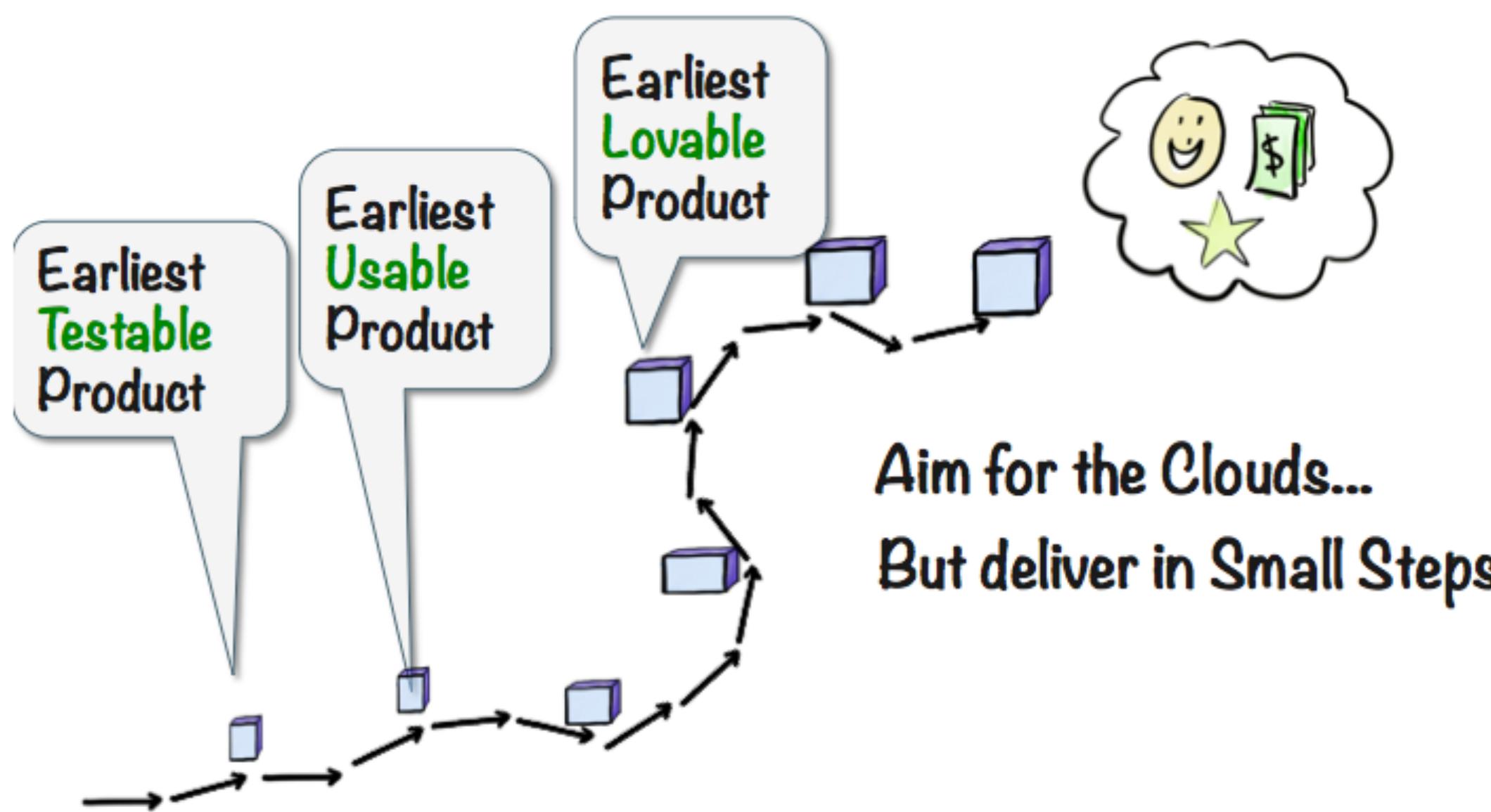
# Example - user story map



# MVP

A **M**inimum **V**iable **P**roduct (MVP) is a concept from Lean Startup that stresses the impact of learning in new product development. **Eric Ries**, defined an MVP as **that version of a new product which allows a team to collect the maximum amount of validated learning about customers with the least effort**. This validated learning comes in the form of whether your customers will actually purchase your product.

**Minimum viable**  $\Rightarrow$  **Earliest testable/usable/lovable**

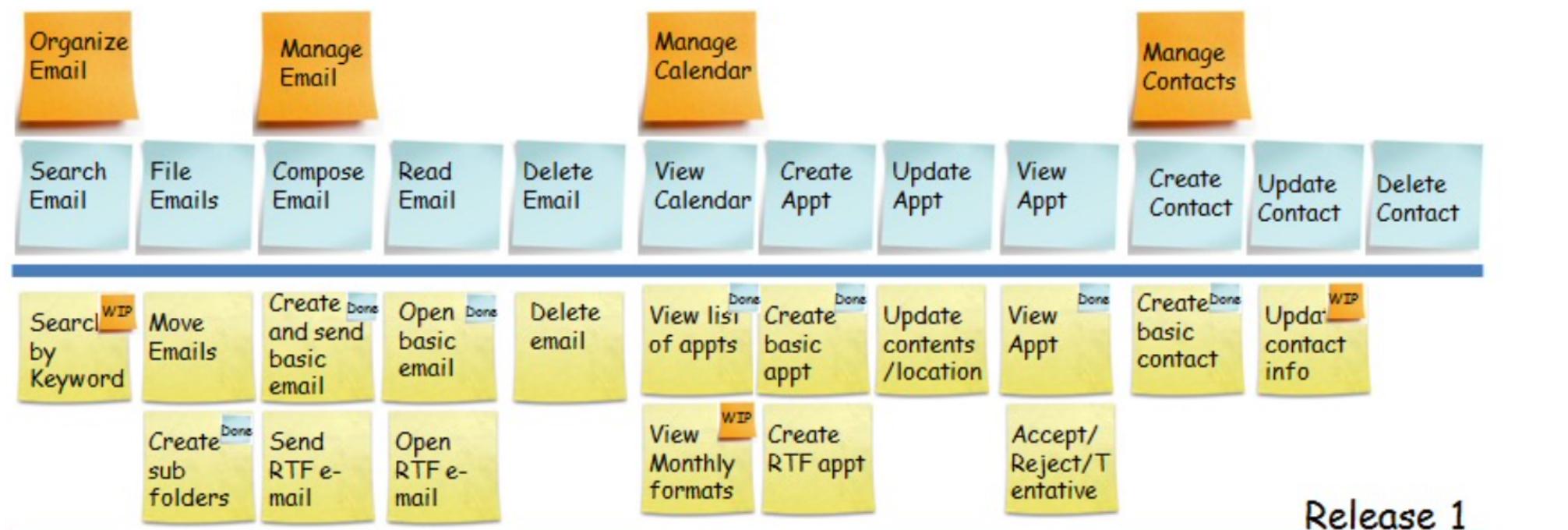


An MVP it is not necessarily the smallest product imaginable; it is simply the fastest way to get through the Build-Measure-Learn feedback loop with the minimum amount of effort.



# Practice

## 2. Stories and Priority: 60mins



Release 1





**AFS**

Agile Full Stack  
Developer Bootcamp  
Thoughtworks ©

# /thoughtworks

## User Story Card

# User Story

## Principle

I  
N  
V  
E  
S  
T

Ask the questions in this order

- V Valuable
- T Testable
- S Small
- I Independent
- N Negotiable
- E Estimable



# User Story Card

## Contents

- Story Type: feature/bug/spike
- Story Name
- Description
- Story Size
- Acceptance Criteria
- UI Design (for frontend)



# User Story Card Description

How to ...

- As a ...
- I'd like to ...
- so that ....



# User Story Card Description

For example

- As a customer
- I'd like to search items in the web
- so that I can find the item I want to buy



# Practice

**As a ... I want .... So that ...**

1

Log in to  
shopping site

2

Check my bank  
account balance

3

View my orders

# Acceptance Criteria

How to ...

- GIVEN ...
- WHEN ...
- THEN ....



# Story Size

How to ...

- The size of the requirement
- Iteration capacity (rate)
- S/M/L/XL/XXL
- 1/2/3/5/8 (Fibonacci sequence)
- 1 point means 1 hour in this project



# Story Card Sample

**Story Name:** User can park and fetch a car

**Description:**

As a parking lot manager, I would like to provide a parking lot, so that the customer can park his/her car and fetch it.

**AC1:**

GIVEN a car and a parking lot with available space

WHEN customer park a car into the parking lot

THEN the customer can park successfully and receive a parking ticket.

**AC2:**

GIVEN a parking ticket and a parking lot that parked the car

WHEN customer fetch car

THEN the customer can get his/her car

**AC3:**

GIVEN a wrong ticket (the ticket not provided by parking lot/ ticket has been used) or without a ticket

WHEN customer fetch car

THEN can't fetch car

**AC4:**

GIVEN a car and a parking lot without available space

WHEN customer park a car into the parking lot

THEN the customer can't park the car

**Story Size:** FE 3 points + BE 3 points

**Note:**

1. The ticket information is a random unique string

Attach the  
UI Design





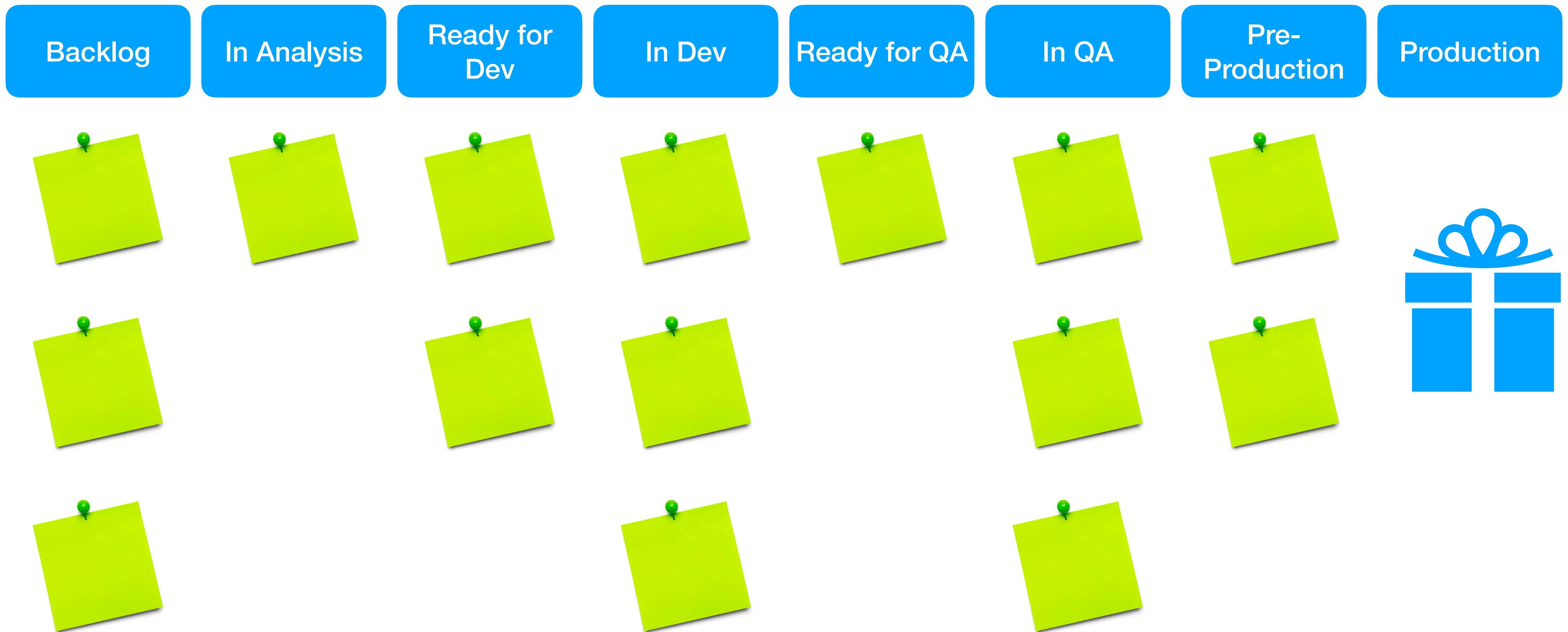
AFS

Agile Full Stack  
Developer Bootcamp  
Thoughtworks ©

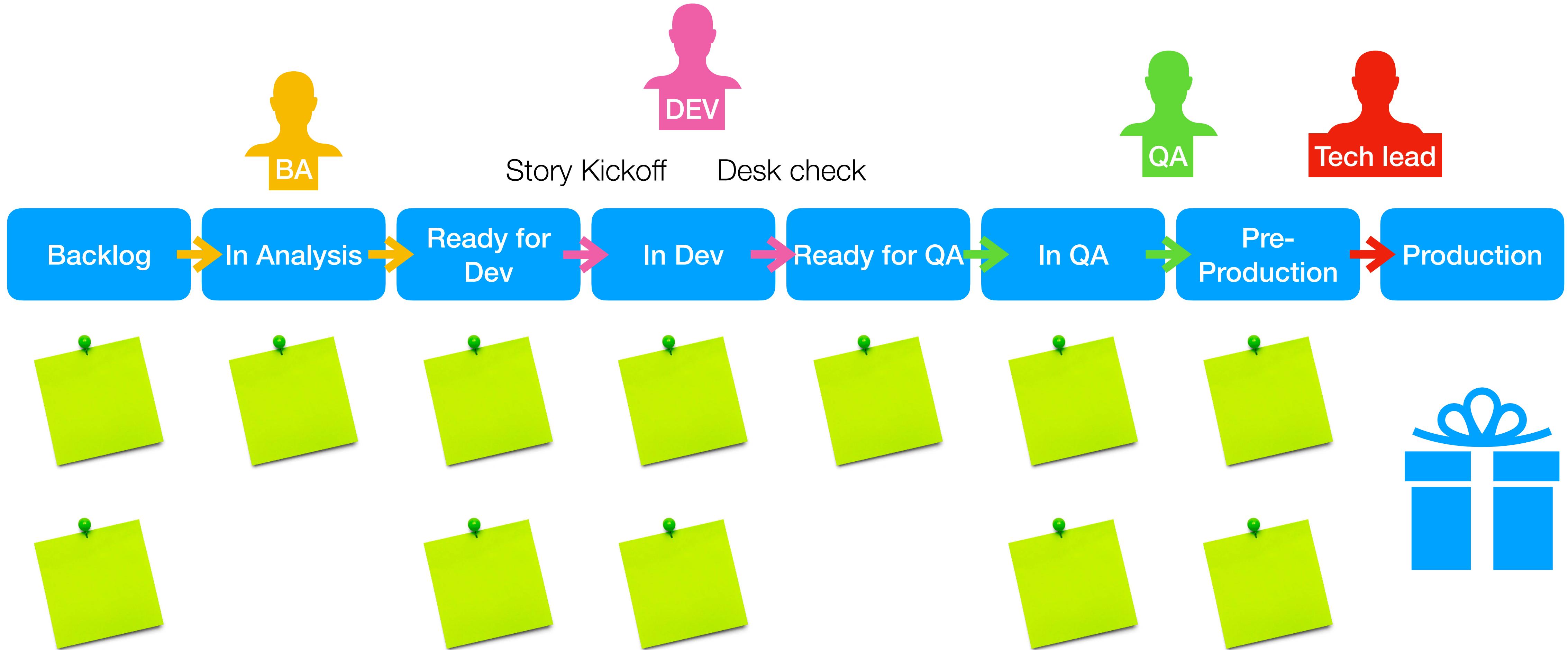
# Kanban Board

/thoughtworks

# Kanban board



# Kanban board





# Create your Kanban board

- Create board like the template below
- Invite all team members
- Invite POs

**Time box: 15min**

Board AFS Trello Template | AFS-SimulationProject | Workspace visible | LN | Invite

Backlog

#2 [BE]User can park and fetch a car  
#2 2 LN

In Analysis

+ Add a card

Ready for Dev

+ Add a card

In Dev

+ Add a card

Backlog

In Analysis

Ready for Dev

In Dev

Ready for QA

In QA

Pre-Production

Production



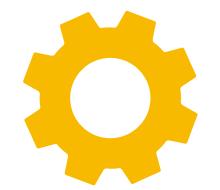
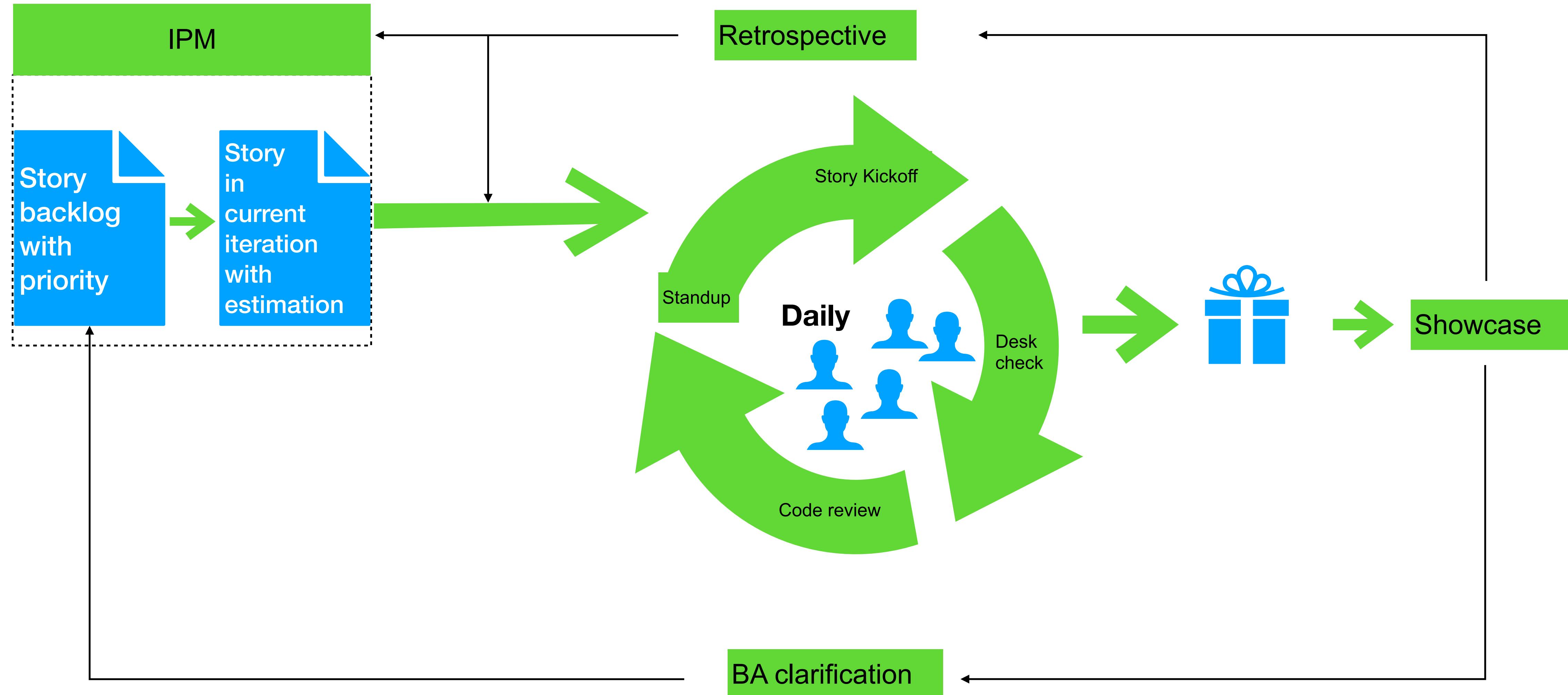
AFS

Agile Full Stack  
Developer Bootcamp  
Thoughtworks ©

# Agile activities

/thoughtworks

# Agile Activities



# Standup

**Owner:** Tech Lead

**Participants:** All team members

- Go through Kanban board
- Clarify everyone's work
- Call out any blockers



# BA Clarify Requirements

Owner: BA

- Communicate with PO
- Communicate with team
- Ensure everyone has a consistent understanding



# IPM

**Owner:** BA

**Participants:** All team members

An **Iteration Planning Meeting** is a key ceremony for any Agile team during an IT project development phase.

It's where your team comes together to talk about **work in the upcoming Iteration** and is designed to help deliver a **clear focus and direction** for the work ahead.



# IPM

Owner: BA

Participants: All team members

- After requirement clarification
- BA to facilitate
- Go through kanban board
- Decide which cards to do in this iteration



# Story Kickoff

**Owner:** Dev

**Participants:** BA+QA+Dev

- Make sure the story card contains all the information
- Make sure the story card has clear description and ACs
- Make sure everyone has a consistent understanding of the card



# Desk check

**Owner:** Dev

**Participants:** BA+QA+Dev

- Ensure that all ACs are implemented
- Ensure that all functions are working properly.



# Code Review

**Owner:** Tech Lead

**Participants:** All team members

- Learning by Challenge
- Learning by Sharing
- Focus on the code and commits
- Ask more questions
- Set time box per repo
- Summary and Record the Good Points and Questions



# Retrospective

**Owner:** Tech Lead

**Participants:** All team members

The Retrospective is an opportunity for the Team to inspect itself and create a plan for improvements to be enacted during the next iteration.



# Retrospective

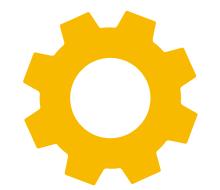
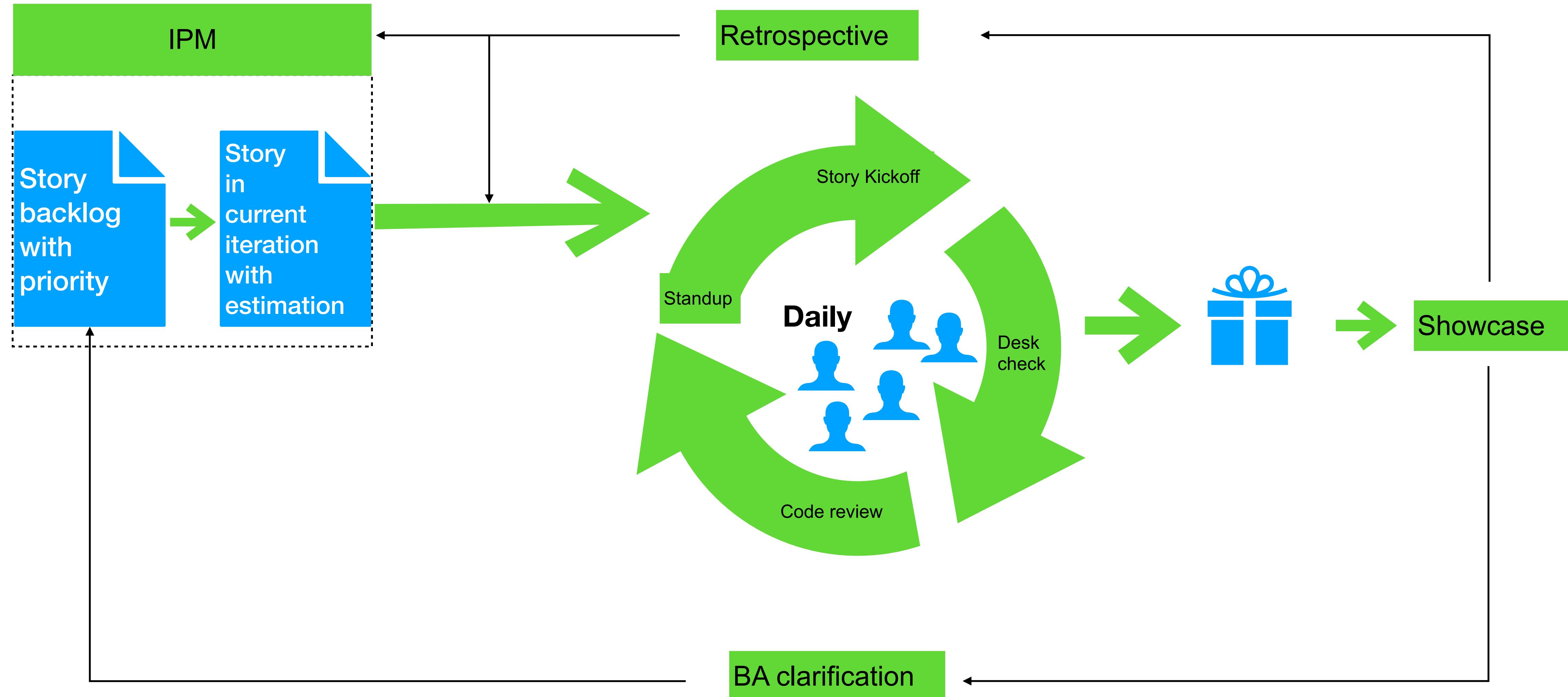
**Owner:** Tech Lead

**Participants:** All team members

- Action must have owner
- Tools e.g. Ideaboardz, <https://excalidraw.com/>



# Agile Activities





AFS

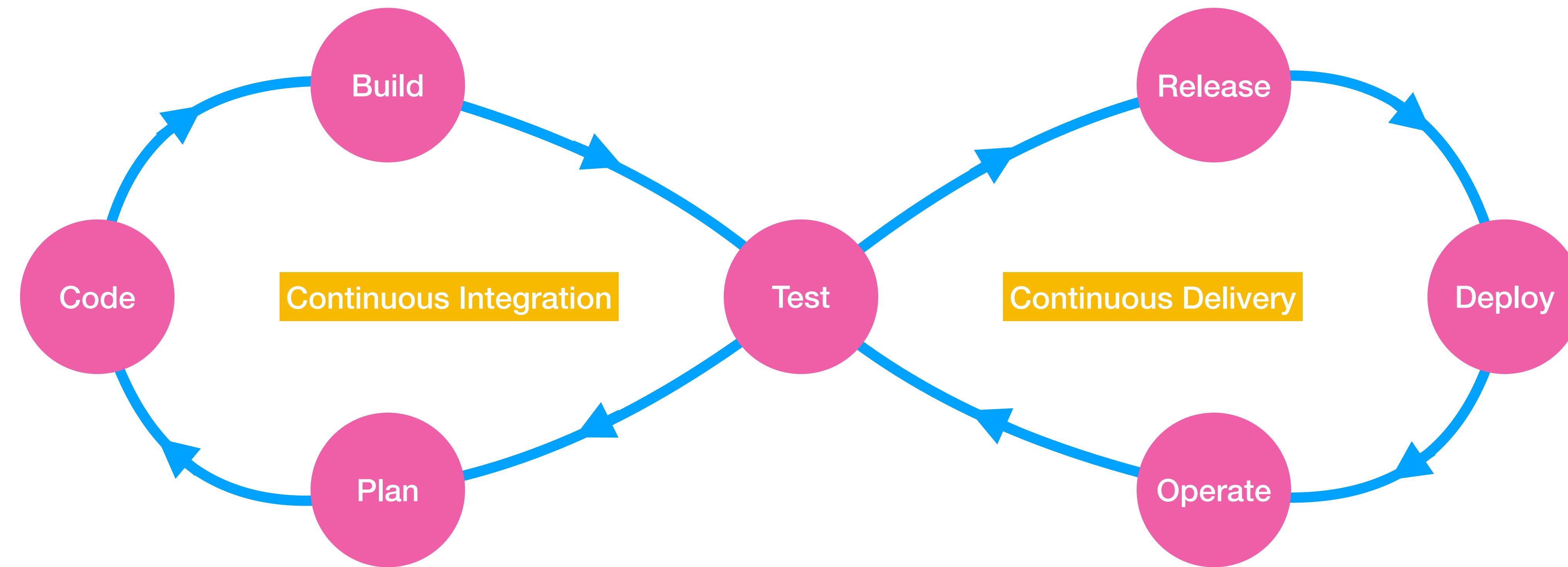
Agile Full Stack  
Developer Bootcamp  
Thoughtworks ©



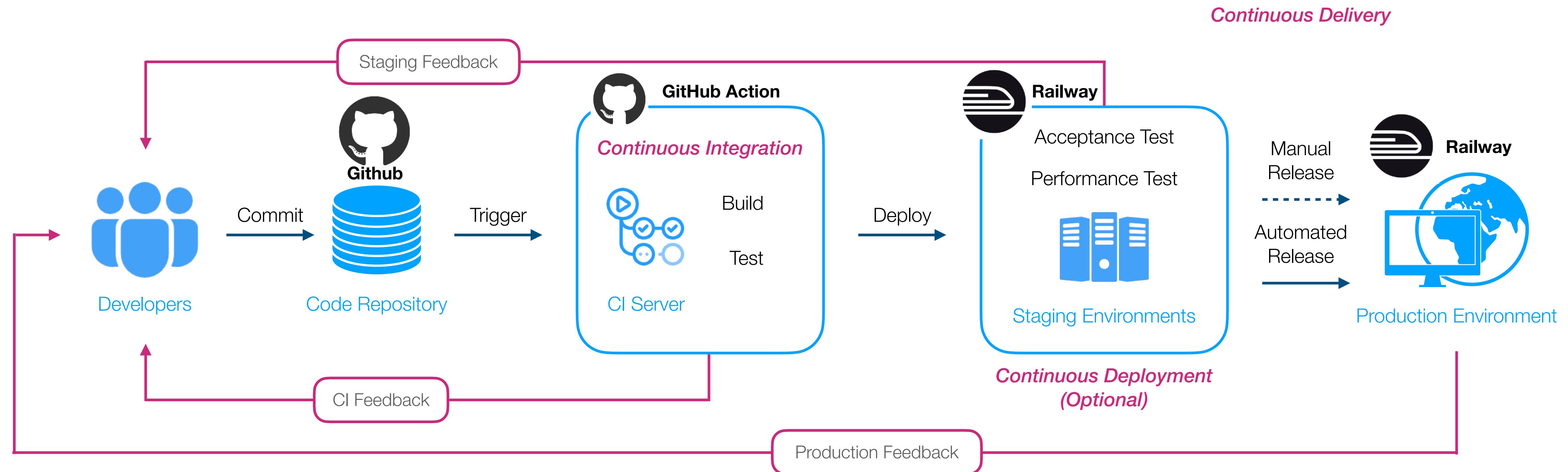
CI/CD

/thoughtworks

# Continuous Integration/Continuous Delivery



# Continuous Integration/Continuous Delivery



<https://github.com/afs-public-202211/railway-spring-mongodb/blob/main/.github/workflows/ci.yml>



# Create code with CI/CD

- Create GitHub organization
  - Invite all team members
  - Invite POs
- Setup CI/CD for Frontend and Backend



AFS

Agile Full Stack  
Developer Bootcamp  
Thoughtworks ©

# Simulation Project

/thoughtworks

# Simulation Project

- Project Output:
  - **Online Movie Ticket Reservation Application** based on training learnings, for example: Spring Boot, RESTful API, Integration Test and Service Test, React, Redux
  - **Team Showcase**(presentation with slides, details will be given later)
- Four days for the agile project implementation and one day for showcase
- One day one iteration, each iteration should follow the agile project process
- You'd better rotate roles to experience the responsibility of each role to find your advantage and disadvantage





**AFS**

Agile Full Stack  
Developer Bootcamp  
Thoughtworks ©

# Day 1-5 Activities

/thoughtworks

# Day 1-4 Activities

**One day one iteration**

TIME	CONTENT
<b>9:00- 9:15</b>	Standup
<b>9:15 - 9:50</b>	Code Review in Team
<b>10:00 - 10:40</b>	Team 1: Regular Showcase to PO inside team Team 2: Regular Showcase to PO inside team
<b>10:50 - 11:30</b>	IPM
<b>11:30 - 12:00</b>	BA Clarify Requirements / KickOff/ Coding/ DeskCheck
<b>13:30 - 17:00</b>	BA Clarify Requirements / KickOff/ Coding/ DeskCheck
<b>17:00 - 17:30</b>	Retro (Tuesday & Thursday)

# Day 5 Activities

TIME	CONTENT
<b>9:00- 9:15</b>	Standup
<b>9:15 - 12:00</b>	Final Showcase Preparation
<b>13:30 - 15:00</b>	Final Showcase Preparation
<b>15:00 - 17:00</b>	Learnings Review Showcase Introduction Final Showcase(Team1, Team2)
<b>17:00 - 17:30</b>	Certification Ceremony



AFS

Agile Full Stack  
Developer Bootcamp  
Thoughtworks ©

# Showcase

/thoughtworks

# Showcase Content

Start preparing from today, continuous improvement on showcase

- **Team introduction**
  - For example, role, responsibility, earnings
- **Product introduction**
  - For example, advantages, core features
- **Software architecture**
  - For example, tech stack, backend & frontend architecture (you can refer C4 Model), CI/CD flow
- **Development process & Agile practice & Technical practice**
  - For example, story card lifecycle, dev's daily work, git collaboration
  - For example, code review, standup, pair programming, IPM, retro
  - For example, TDD, refactoring, swagger or others(Lombok, Oauth)
- **Product Demo**
  - Showing by user scenarios on production environment or staging environment
- **Future planning of product**
  - For example, next step, business planning



# Showcase Content

Start preparing from today, continuous improvement on showcase

- Team introduction
- Product introduction **Day 1 showcase**
- Software architecture
- Development process & Agile practice & Technical practice
- Product Demo
- Future planning of product

**Day 2~4 & Final showcase**



# Investment Reference Dimensions

Investors will refer these dimensions to ask questions and determine the invest amount

- Presentation
  - online presentation ability
- Product
  - product functional integrity, product quality
- Technical practice
  - such as the use of technical architecture, tdd, refactoring, etc.
- Agile practice
  - such as code review, stand-up meetings, pair programming, etc.
- Teamwork
  - collaboration in agile projects, showcase collaboration



# Regular Showcase(day1-day4)

**Owner:** BA+QA to facilitate the activity

**Participants:** All team members

- 30 mins
- Each team member present a part
- Teamwork to complete the showcase



# Final Showcase(day5)

- 40 minutes per group(including the presentation & product demo, etc.)
- Teamwork to complete the showcase
- There are N investors
- Q & A after showcase
- Each investor has 25 million
- Your team can get up to 25 million investment from one investor
- Try your best to get the highest investment amount



# Today's Homework

1. Prepare day1's(iteration 1) story cards for IPM on Trello.  
(BA&QA)
2. Finish UI design for day1's(iteration 1) story cards.(UX)
3. Setup development environment(for example, code base, database configuration), and make sure you can start working on the frontend and backend project development. (Tech Lead & Dev)
  - Create code base under one organization and invite all members and trainers
  - Setup CI/CD for Frontend and Backend
4. Prepare showcase presentation and PPT(Team introduction, Product introduction, Software architecture)

