PROJECT MANAGEMENT

GO BACK 50 YEARS

1. Plan
2. Design
3. Build
4. Test
6. Review
7. Maintain

WHY IS THIS A BAD THINGS?

BUT CUSTOMERS GET FUSSY...

INTRODUCING AGILE

WATERFALL

1. Plan
2. Design
3. Build
4. Test
6. Review
7. Maintain

AGILE

WATERFALL

1. Add posting
2. View postings
3. Search for posting
4. User login

AGILE

View postings   Add posting   Homepage

HOW DO YOU GET STARTED

1. Define Problem
2. Document Stories
3. Sprint Planning
4. Track sprint as it goes
5. Retrospective

WHAT IS A USER STORY?

FORM:
As <PERSONA> I want to be able to <ACTION> because <JUSTIFICATION>

EXAMPLE:
As Kerry I want to be able to upload a photo to share with my friends and win prizes

PERSONAS

DEFINITION:
A user of your application

EXAMPLE:
Kerry is an 18 year old girl and loves using her iPhone. She is an avid Facebook user and regularly shares pictures with her friends.

WHAT TO PUT IN:
Relevant facts to the project that app that you are developing

WHAT NOT TO PUT IN:
Unrelated facts, e.g. hair colour, geographic location, things she likes, etc.

WHAT GOES IN A STORY?

Title: User upload photo
Epic: Sharing Photos

User Story:
As Kerry I want to be able to upload a photo to share with my friends and win prizes

Deliverables:
Kerry can upload a photo once she is logged in

Exceptions and Notes:
Kerry cannot upload a photo when she is not logged in
Kerry can only upload JPEG, PNG or GIF
Kerry should be given feedback during the upload process, e.g. upload bar
Kerry should be able to upload a photo from her phone

SPRINT PLANNING

Which stories go into the sprint?
Do the stories make sense together?
How hard is this going to be to complete?
Who can work on what?

TRACKING THE SPRINT

DAILY STANDUPS

What did you do yesterday?
What are you going to do today?
Is there anything in your way?

SPRINT REVIEW / RETROSPECTIVE

**Demo your work and get feedback from the team
What worked well?
What didn't work well? How can we improve things?
Do we have something that works?**

Product owner
- Define the features of the product
- Makes scope vs. schedule decisions
- Responsible for achieving financial goals of the project
- Prioritize the product backlog
- Adjust features and priority every sprint, as needed
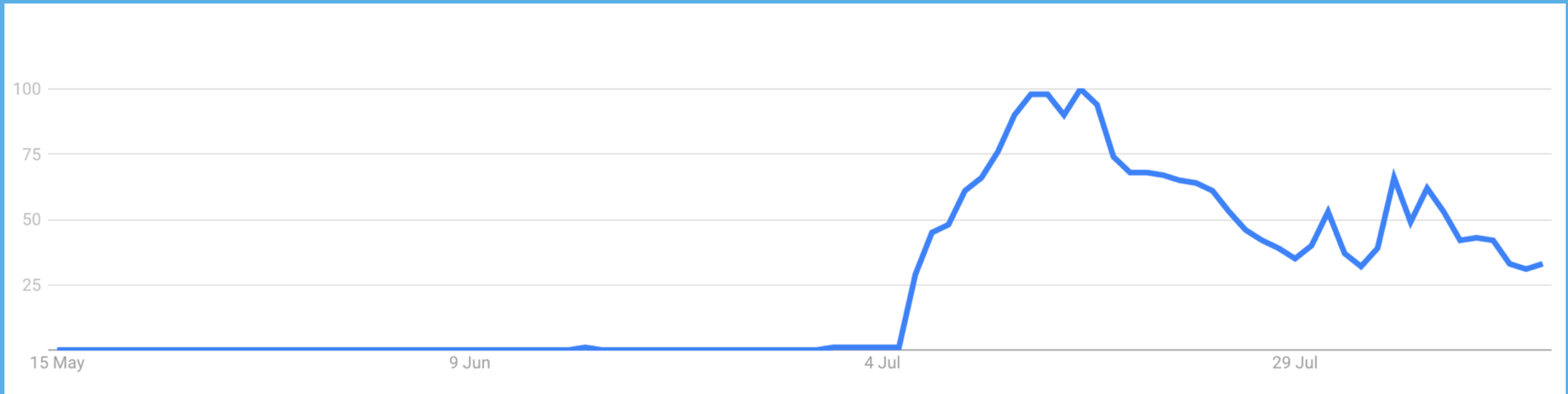- Accept or reject work results

The ScrumMaster

- Responsible for enacting Scrum values and practices
- Removes impediments
- Coaches the team to their best possible performance
- Helps improve team productivity in any way possible
- Enable close cooperation across all roles and functions
- Shield the team from external interference

The team

- Typically 5-9 people
- Cross-functional: Programmers, testers, user experience designers, etc.
- Teams are self-organizing
- Ideally, no titles but rarely a possibility
- Membership should change only between sprints

MINIMUM VIABLE PRODUCT (MVP)

100				
75				
50				
25				
	15 May	9 Jun	4 Jul	29 Jul

QUESTIONS