Software Development Process

Lecture 5

User Stories and Planning Poker

Involving Users in Backlog Creation

- Agile approach emphasizes on user involvement
  - They are the ones who know their domain best
  - They know what they want and they know what they want first
  - **The user should involve thorough the project**
- So, the users (and other stakeholders) should involve in system requirement elicitation
  - But they do not know modeling (e.g. use cases)
  - They do know how to come up with functional / non-functional requirements
- Then, what should we do to let the users express their needs, requirements and goal?
  - **User stories**

User Stories

- **User story**: A short, simple description of a feature told from perspective of a stakeholder
- It is usually written on an index card or sticky note:
  - As a [*type of user*], I want [*some goal*] so that [*some reason*]
  - As a ***astronaut***, I want a ***align my ship*** so that I can ***park at the space station***

User Stories Example

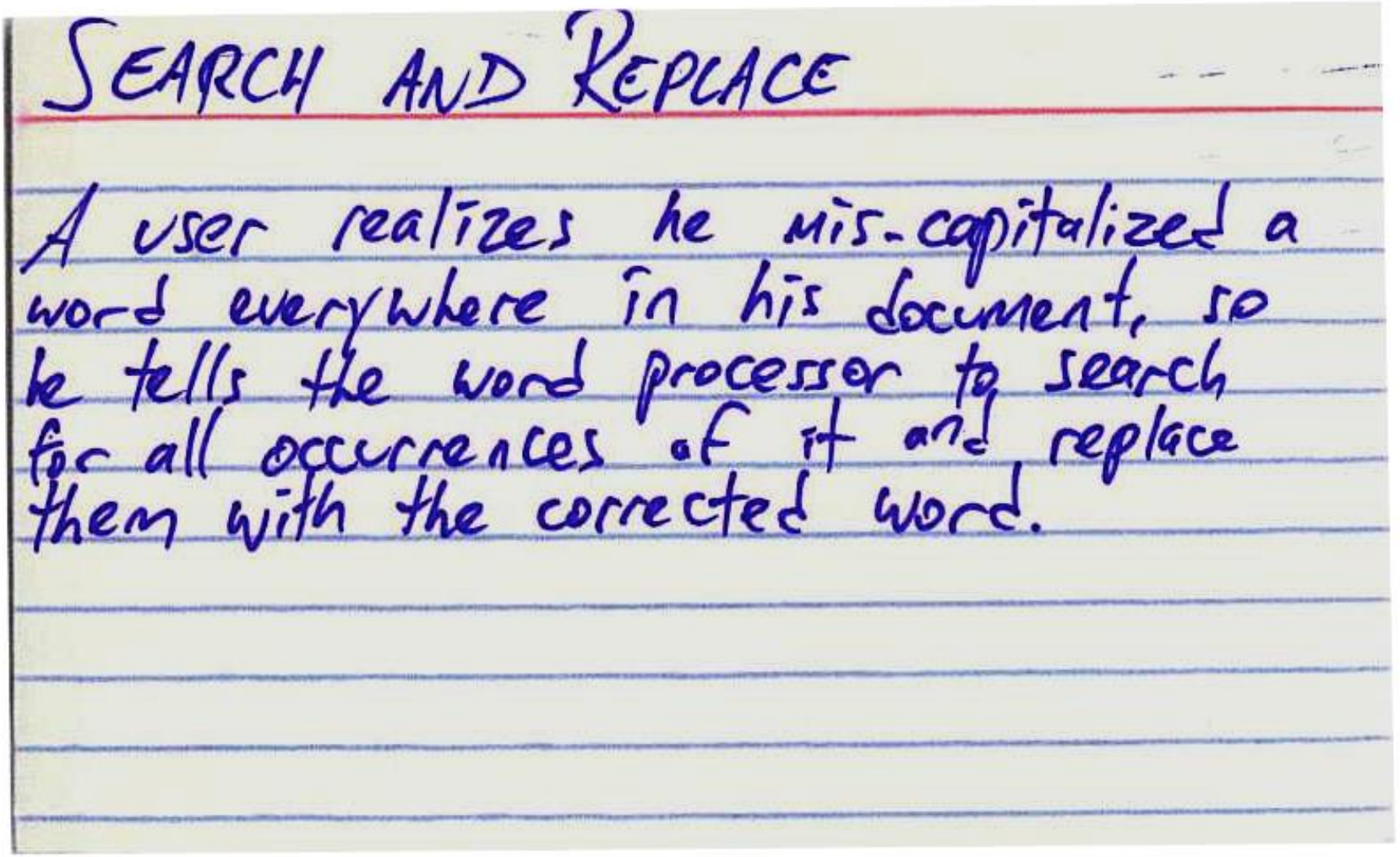

Image: http://www.stellman-greene.com/2009/05/03/requirements-101-user-stories-vs-use-cases/

User Stories Example (2)

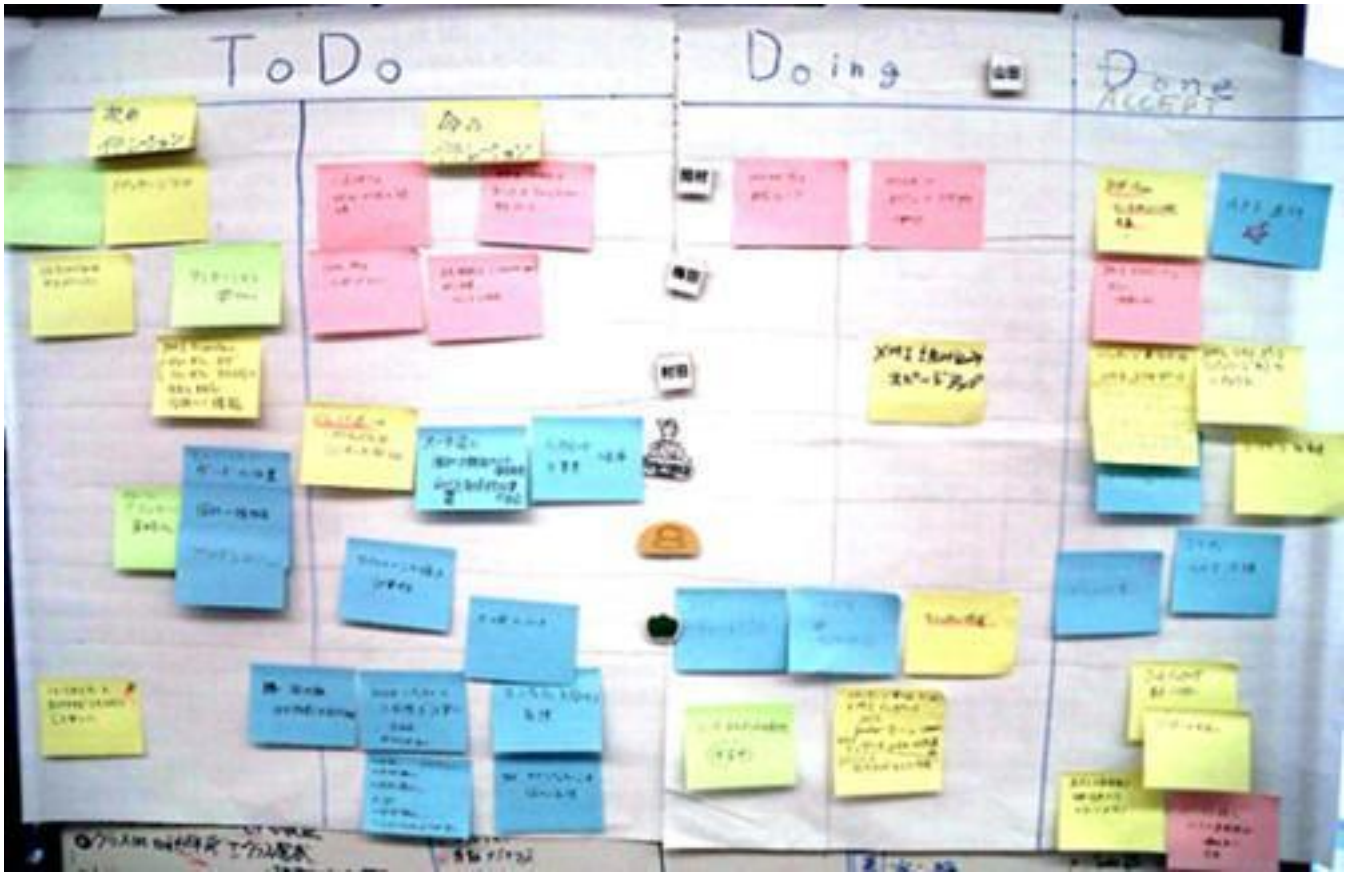

Image: http://www.agilebuddha.com/agile/story-mapping-andvs-process-maps/

User Stories vs Use Cases vs Scenarios

- A user story is an informal, user-oriented description of use case
  - Both user story and use case describe interaction between the actors and the system
  - But many stories may correspond to only one use case and one story may span into many use cases
  - User story is minimal, telling the user's needs
  - Use case is complete, explaining what the system must do to serve the user's needs
- Scenario is a story of actor(s) interacting with a system
  - The interactions are laid out in sequence
  - One scenario may consists of many use cases

Emergent Requirements

- **Emergent Requirements**: Features or functionalities that are not anticipated before
  - **Problems** that are discovered during the project
  - **New** ideas
  - Required features that everyone has **overlooked**
  - These issues can be seen only after you use your software
- Emergent requirements can always exist
  - Non trivial projects: You have no clear idea in the application domain or problem
  - Research and innovative projects
- Such requirements make it impossible to plan the project
- Agile approach emphasizes on working code after each iteration to detect emergent requirements early

Epics

- To deal with emergent requirements:
  - Assume that they exist and we cannot think of everything
  - Do not try come up with backlog that contains all user stories that may happen
- Instead, think about big, **epic** stories first
- **Epics**: A large user story that must be broken down into small ones before putting into an iteration
  - Example: “Land on the moon” “Build an elevator”
- We sort the epics first then break it down into smaller stories or functionalities when we actually implementing them

User stories can be grouped and prioritized

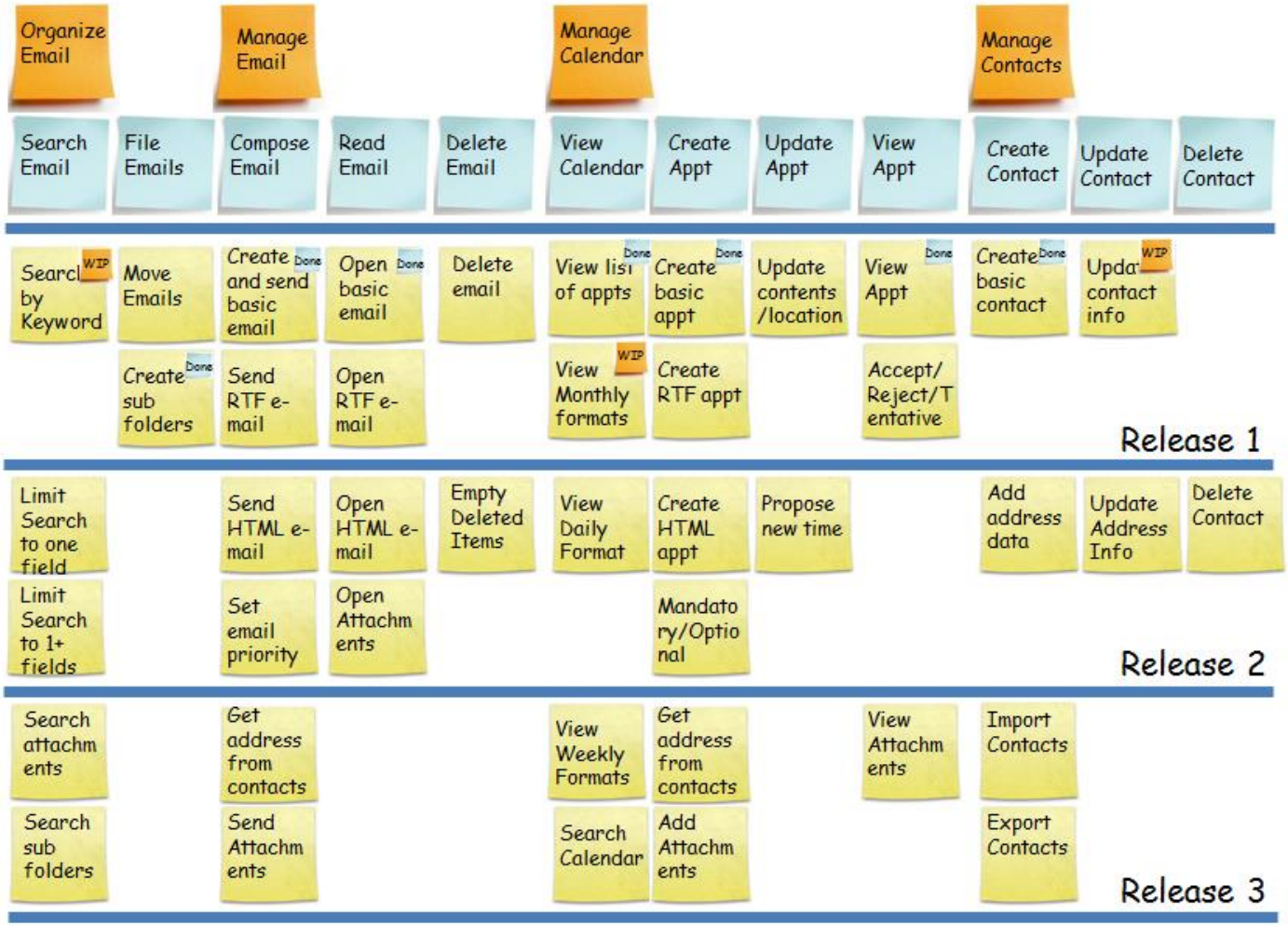

Image: http://www.agilebuddha.com/agile/story-mapping-andvs-process-maps/

Planning Poker

- **Planning Poker**: A process to estimate effort to complete a user story
  - Consensus based: The team member must estimate together and agree upon the end result
  - In the estimation process, no one is allowed to speak to avoid **anchoring**
- **Anchoring**: Human nature that tends to bias toward the first piece of received information
  - For example: The first developer says, “I think this story takes 30 days.” The other team members would immediately estimate relating to 30 days instead of having a unbiased estimation.

Planning Poker Cards

- A set of cards, usually showing a Fibonacci sequence
  - The larger the estimation, the larger the uncertainty
  - That is, you cannot say “17 days”, it is too precise for such large estimation. You can only guess widely.
- The number can represent any work units e.g. days, hours, half a day

Images: http://en.wikipedia.org/wiki/Planning_poker http://www.it-zynergy.com/scrum-planning-pokers

The Rules

- To estimate a story, the moderator selects a story and let the team members (may include PO and other stakeholders too) ask and discuss about the tasks
- Then, each team member select a playing card, **face down**, based on estimated effort. The player must not speak the number out loud
- Once everyone has selected a card, all member reveal their cards
- The players with the highest and lowest numbers are allowed to state their reasons. The others can later join the discussion
- Repeat the process until a consensus is reached
- All discussions must be time-boxed e.g. 3 minutes