Specifying Requirements

Some Teamwork Notes

- I'm hearing that some students are not participating appropriately.
- We expect:
  - At least one **in person** meeting per week
  - At least three meetings total
- This week, you will meet with your assigned TA. Slots will be released later today.
- Weekly team contribution reports will be due Sunday nights starting this coming Sunday

Requirements, User Stories

- Question: how to express requirements?
- Answer (user stories): "As a <stakeholder>, I want <something> so that <need>."
- Example: "As a student, I want to filter recipes by cost so I can keep dinner under $5 per person."

https://www.agilealliance.org/glossary/invest/

User Story Criteria: "INVEST"

- Independent
- Negotiable
- Valuable
- Estimable
- Small
- Testable

Independent

- Ideally: want to implement requirements in any order
  - In practice, there may be dependencies

Negotiable

- Details to be negotiated during development (allows team to choose a less-expensive design)
- Good Story captures the essence, not the details
- A non-negotiable user story (bad):
  - As a product purchaser, I want to be able to rate a product out of five stars so shoppers can assess product quality
- A negotiable user story (good):
  - As a product purchaser, I want to be able to rate how much I liked a product so shoppers can assess product quality

Valuable

- This story needs to have value to someone (hopefully the customer)
- Especially relevant to splitting up issues
- As a developer, I want to use React so that I can be up to date on the latest trends.
    - Does not have value for the user

Estimable

- Helps keep the size small
- Need to complete each user story in 1-2 weeks (or less)
- As a dermatologist, I want to use the AI system to distinguish between macules and papules
  - How am I supposed to estimate that if I don't even know what those are?

Small

- Fit on 3x5 card
- At most two person-weeks of work
- Too big == unable to estimate
- Too big == may not finish in time for delivery

Testable

- Ensures clarity
- If not testable, when do we say the task is done?
- "The login page should display an error message when incorrect credentials are entered": ✅
- "The login page should check credentials": ❌

Summary

- Write open-ended, high-quality questions to elicit requirements
- Use INVEST criteria to write good user stories

Are User Stories Enough?

- User stories capture what the system should do (functional requirements)
- What about *how well it does it*?
- More importantly, of all the designs that meet the requirements, which should we build?

Functional Requirements vs. Quality Attributes

- Functional requirements: things the system must do
  - "The system shall support receiving email from an IMAP server."
- Quality attributes: requirements concerning how the system meets its functional requirements
  - "The system should fetch 1 GB of email in under 1 minute."
  - Distinguish between different implementations that meet the functional requirements
  - Should be testable

Quality Attribute Examples

- Informal goal: "the system should be easy to use by experienced controllers, and should be organized such that user errors are minimized."
  - Verifiable usability quality requirement: "Experienced controllers shall be able to use all the system functions after a total of two hours training. After this training, the average number of errors made by experienced users shall not exceed two per day, on average."
- Availability: The system shall be available 99.99% of the time.

A Common Mistake

- Avoid writing requirements in terms of implementation details
- Design is a separate process; don't do it now
- "The database server shall handle at least 4M records."
  - "Oh, there's a database server? Just one?
  - → "The system shall support at least 4M users."

Exercise

- Write a quality requirement pertaining to an entertainment system for airplane passengers.
  - Careful: what assumptions are you making?

User Stories

- Succinctly express functional requirements (and sometimes quality attributes)
- Facilitate conversations with stakeholders
- Define acceptance criteria (so you know when you've succeeded)