Software Engineering 2022-2023, Prof. Nuo Li

TASKS IN WEEK 5

1. BEFORE NEXT LECTURE

A. Design classes for your project

- Focus on what your team will implement
- If necessary, use CRC card approach [i] to moderate team brainstorming
- Apply at least one of the SOLID design principles

B. Update your sequence diagrams. If necessary, create new sequence diagrams based on the identified classes.

C. Program the definitions of classes in your application. (Or directly import the class diagrams into your IDE, it your IDE supports)

D. Report progress in your blog

- Your class diagrams (either drawn by hand, or generated from your IDE) and updated sequence diagrams
- Explain your considerations when modeling the classes
- Highlight which design principle is applied in your project, why and how it is applied.
- Any other tasks you have achieved for your project, if there is any. E.g., data collection, setup virtual machine, etc.

E. Version Control your drawings in your GIT

F. Review two other blogs

- Checklist:
  - *Logical?*
  - *Will changing one class require changes to another?*
  - *Can re-work a class inside, without changing its interface, for outside interactions?*
  - *Are all of the methods in a class related to a single abstraction?*
  - *Are naming convention consistent?*

2. KEEP IN MIND

Check Discord ePortfolio channel to avoid double booking.

Reserve a date for your presentation.

If you'd like to present a topic not in the ePortfolio list, please send me an email.

---

[i] Class Responsibilities and Collaborators (CRC) method

Step 1: Find classes

(1) Write down all nouns in your requirements specification.

For instance:

- Player
- Game
- Internet
- Table
- Cell
- Start Button
- Dot
- Interval
- Color
- Mouse
- Point

(2) Analyze list of nouns to see if they meet the characteristics of a class.

- Retains information
- Provides services
- Has multiple attributes
- Has common attributes
- Has common operations
- Essential requirements

Step 2: Find responsibilities

A Responsibility is anything that the class knows or does. These responsibilities are things that the class has knowledge about itself, or things the class can do with the knowledge it has.

To compliment the 'Noun Extraction' technique above is verb extraction. Verb extraction identifies all of the verbs in a problem statement and/or use-case scenario.

These are usually good indicators of actions that must be performed by the classes of the system.

Other techniques included asking what the class knows, and what information must be stored about the class to make it unique. Ask what the class does, and then flesh out how the class might do it.

Step 3: Find collaborations

Collaboration occurs when a class needs information that it doesn't have.

Classes know specific things about themselves. Very often to perform a task a class needs information that it doesn't have. Often it's necessary to get this information from another class, in the form of collaboration.

Collaboration can also occur if a class needs to modify information that it doesn’t have. One property of information that a class knows about itself is the ability to update the information. Often a class will want to update information that it doesn't have. When this happens, the class will often ask another class, in the form of a collaboration, to update the information for it.

Generally for a collaboration to occur, one class is the initiator. In other words, there has to be a starting point for collaboration. Often times the initiating class is doing very little work beyond initialing the collaboration itself. However, be careful that a class doesn't just 'pass the buck', just to have it passed again. For example, if class A collaborates with class B, just to have class B pass the work to class C, consider eliminating the collaboration with class B.

Step 4: Review scenarios and merge classes

Identify synergies overlaps, and then merge classes.

Bad scenario example:
Jeff double clicks on the Character Studio, which opens a drop down menu that displays various character options.

Improved version:
Jeff goes to the Character Studio and creates his character John with blond hair, red hat, and white skin color.

CRC Cards Example:

CRC Cards for ATM Example