OOAD
UCSC Extension
Instructor: Edwin Mach

Project X

Note: This project will take some time to complete and will be due at the last class. There will be 1 CHECK POINT (see CHECK POINT section below) and a short presentation.

Delivery: Software code (JAVA, C++, Python, C#, .NET, etc)
Documentation
UML diagrams (show static and dynamic)
Use cases (write a simple and an expanded one)
code documentation
architectural diagrams
interaction diagrams

Requirements:

A software application (game, webpage, mobile phone app, etc,...) with a purpose. It does not have to be a new or novel idea. It can be something that has been done before and does not have to be a "full blown" software application.

You write the requirements, design, etc – I'm the boss and customer :-) so changing requirements and getting approval must go through me!

You must get approval from me for your project idea. I would like to see different projects and how you tackle them.

Examples of projects you can turn in are:
1. a Board game – but you don't need to implement all the backend engine or every single game detail.
2. music webpage – play a small database of music through a web interface. This interface must be dynamic in some way and accomodoate search, etc. On the server end, use a OO language to process the requests..
3. mobile iphone app – browse a cd library on the internet...

There must be one or more actors, actions, and be in line with GRASP principles. If any GRASP principles is broken, you'd need to explain why you had to break it.

These projects are banned: anything that uses CLI (command line interface) as the "graphical user" interface, BlackJack or 21 card game

If there is doubt or questions regarding any of the requirements, please ask!

I know this project sounds "open-ended" and it is. Treat this like a real project at work. How did you tie up loose ends? How did you get closure on open issues? Most importantly, how did you define the scope?

Programming languages:
Feel free to use ANY Object oriented programming language that you know. You must use an OO language if there exists one for what you are doing. So for the webpage example, for example, use Java or C++ for the backend, but not C.

Checkpoint

CHECK POINT #1 (DUE FOUR WEEKS BEFORE CLASS ENDS)

You need to turn in/email me:

Use cases = 80% completed.
Requirements = 80% done.
Architectural diagram
Progress on the “core” part

Last day of class

Show and Tell your project!!

- Give a 15-20 min presentation. The presentation must include walking through your design, and why you designed it that way. It's probably best to show us at a high level what you've done, and then talk about the design.
- It's important to not just show us UML and/or use cases, but also the ability to explain them.
- It’s also important to demo a live working product.
- Show a BEFORE and AFTER if your project specification or diagram changed. We want to see how the interative design process worked for you.

So, How will you show me?
OPTION #1: You can send me a link to a video posted online or tell me where I can download it.
OPTION #2: We can have a webex/goto meeting if you have access to that.

Finally, email me all code, documents, etc.