

Quin Thames
CMSI 402
Assignment 2
2/25/14

Problem 6.4

Do the following for the Study Abroad Management System (SAMS) presented in Chapter 4:

a. Identify the type of system and briefly justify your answer.

The Study Abroad management System would be an interactive system because it requires users/actors to directly interact with the system in Students searching for programs, submitting applications and staff members adding/deleting programs and reviewing applications.

b. Identify an architectural style and produce an architectural design for the system.

An architectural style that could be used for this would be the N-tier architecture system:

User --requests--> GUI --requests--> Controller Layer --requests--> Database Layer
--requests--> Network Communication Layer

c. Specify the functionality and interface for each of the subsystems and components in the architectural design.

The GUI layer is responsible for displaying information and interface options (buttons, menus etc) to the user. GUI layer sends requests to the Controller Layer which handles the specific events of the GUI Layer. The Controller layer determines what requests need to be made to the Database layer to serve up for the GUI Layer. The Database layer supplies the needed information to the Controller requests changes the Database's contents according to the request and then Network Communication layer allows the Database to communicate with the rest of the network to remain persistent and in sync.

d. Discuss which software design principles are applied in the design, how they are applied, and the benefits of applying each of the principles. Also point out the potential problems associated with the application of the design principles, if any.

The separation of concerns principle is applied in the N-tier architecture design in that each layer can be separated and worked on individually. The Keep It Simple and Stupid principle can also be applied. The system requires a few straightforward use cases that can be handled with relatively "stupid objects" without over complicating the architecture.

Problem 7.4

Do the following for the Car Rental System (CRS) given in Appendix D.1:

a. Derive use cases from the requirements you produced in exercise 4.8 [Instructor note: you are not required to actually do exercise 4.8; only turn in the use cases.]

Use cases:

1. Customer reserves a car
2. Employee checks out car
3. Employee Checks a car back in after returned
4. Employee logs damages and done to a car
5. Employee logs repairs and done on a car
6. Employee adds/deletes a car to/from the system

b. Construct a requirements-use case traceability matrix.

Requirements:

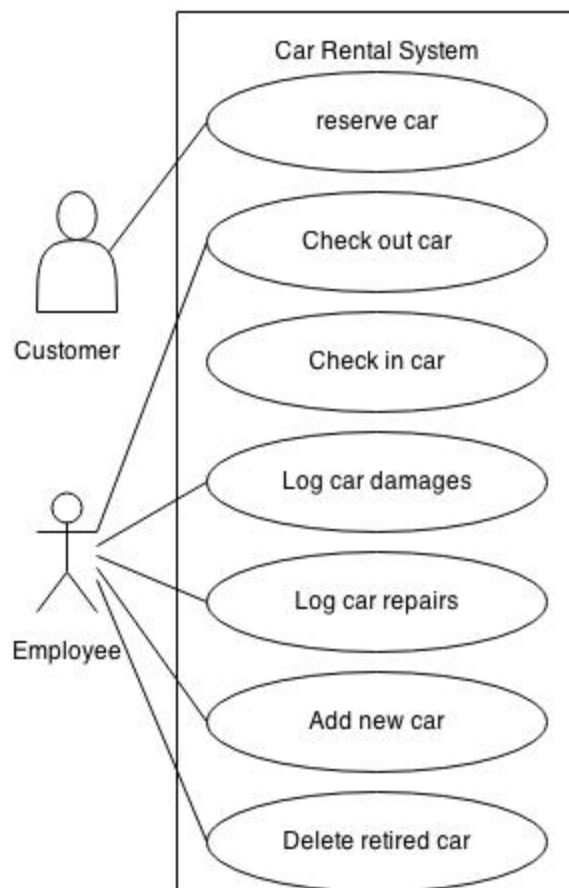
1. Employees shall be able to add and remove cars
2. Employee shall be able to log damages to cars
3. Employee shall be able to log repairs for a car
4. Customers must be able to reserve a car
5. Employees must be able to check in/out a car

Requirement	Priority	1	2	3	4	5	6
1	5						X
2	2				X		
3	2					X	
4	4	X					
5	5		X	X			
UC Priority		4	5	5	2	2	5

c. Specify the high-level use cases.

High level use cases are adding and deleting cars, and employees being able to check cars out and in.

d. Draw use case diagrams.



Problem 23.2

Apply the COCOMO II Application Composite Model described in Section 23.2.2 to estimate the effort required to develop the Add Program use case for a Study Abroad web application for the Office of International Education (OIE) of a university. Assume that the object point productivity is nominal and no reuse of existing components is expected. The Add Program use case allows an OIE staff to add an overseas exchange program, such as French Language and Culture. This use case needs to display three web pages and a message as follows:

- a. Initially, it displays a Welcome page, which consists of three frames. The main frame shows a description of the Study Abroad program. The top frame shows the OIE logo along with a row of buttons, such as OIE Home, About Study Abroad program, Contact Us, etc. The left frame shows a list of three menu items: Program Management, User Management, and System Settings. The Welcome page is displayed after an OIE staff member successfully logs into the system.
- b. To add a program, the OIE staff clicks the Program Management item, which expands the item and shows a list of four menu items under the expanded item — Add Program, Update Program, Delete Program, and Import Programs.
- c. The OIE staff clicks the Add Program item. The system displays an Add Program Form in the main frame of the page. The Add Program Form lets the OIE staff enter program information into the various fields. The OIE staff clicks the Submit button to save the program.
- d. When the Submit button is clicked, the system saves the program into a database and displays a "program is successfully saved" message.

Steps:

1. Count number of screens: 3
2. Complexity Levels of each:
 - Welcome page: medium
 - Program Management: simple
 - Program Form: simple
3. Complexity weight: 2,1,1
4. Add weights for each of the screens for Object Point Count: 4
5. No reuse: NOP is still 4
6. Object point productivity: Nominal-13
7. Person-month: $PM = NOP / PROD = 4 / 13 = 0.3077$

Problem 23.8

Suppose that a company is at level 1 of the CMMI. Propose a process improvement plan for the company to improve its process to level 5. Make necessary assumptions about the level 1 company.

For a company at a level of CMMI 1, one way to progress to a level 5 CMMI is to adapt a Test Driven Development (TDD) process of development. TDD provides an easy way to avoid defect prevention. If there are adequate unit tests for everything, then you can be sure that new code does not introduce a defect in the pre-existing code. Furthermore, TDD automatically uses data to identify weaknesses in the code and are easily documented in the list of passing and failing tests. The test driven method of agile development also helps improve the feedback loop because there is always a working minimal viable product at its current stage available that can be released/shown to users for feedback and improvement.

Problem 10.8

Describe three realistic, not dummy or hypothetical, situations to apply the controller, expert, and creator patterns.

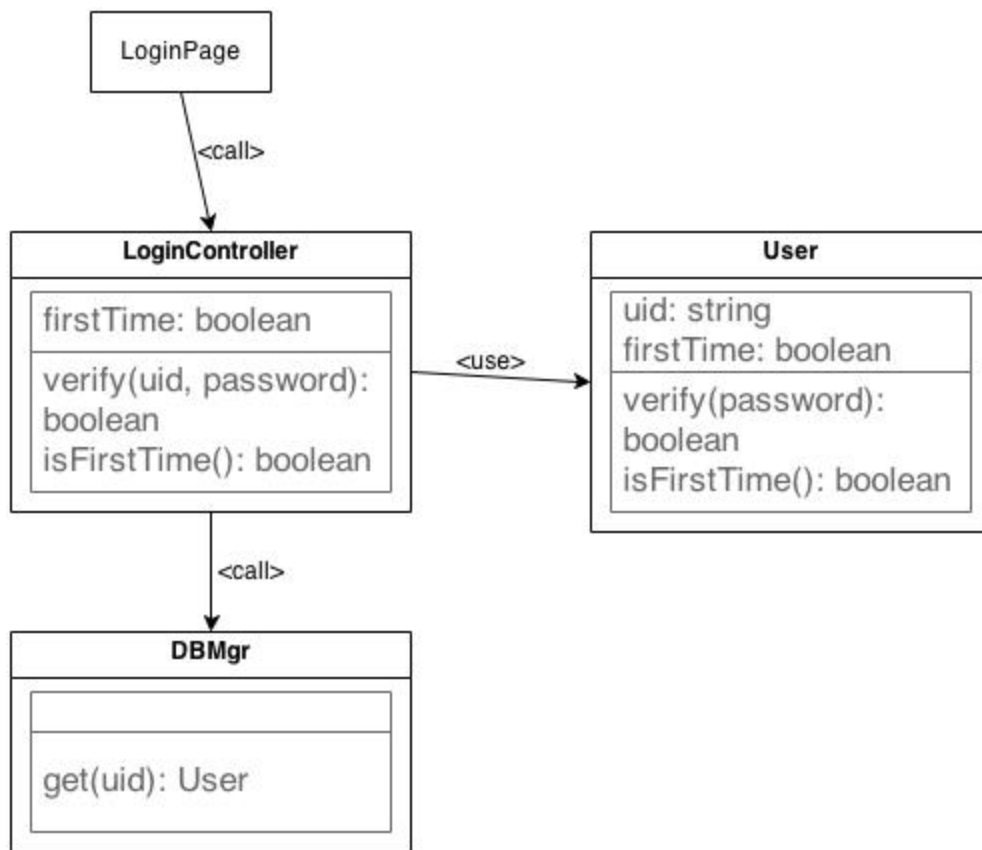
A use case controller is used in a web application like Yelp to control the different use cases for users and businesses as well as control what view to show when the app is being accessed through mobile or desktop.

An expert information pattern is used for a user login system so that the object handling the login request has the required information to carry out the request (password, user)

A creator pattern is used on sites like facebook where people are able to post and make comments. The creator pattern is used to determine who a 'like' on another person's comment on another person's post belongs too and who should be designated as creator.

Problem 11.1

Derive a DCD from the design sequence diagram in Figure 9.21.



Problem 12.1

Produce an expanded use case description, then perform the user interface design steps to produce a user interface design for the Reserve a Car use case for an online car rental application. A description of such an application is presented in Appendix D.1.

