Tutorial 01
Models of software
Architecture: Domain model

SOEN 343

2020 Fall

Exercises taken from the slides of:

Dr. C. Constantinides

Instructions

- Individually answering the following exercises
- The outcome of your work has to be uploaded to Moodle web site to the corresponding slot (tutorial 1 assignment)
- The format is a pdf file
- Note that the slot for submitting the tutorial assignment is open until the end of the week (Sunday 23:59 EST time)

Exercise 1

- Consider an email client software. The client has one ==mailbox== which consists of several ==folders==. Each folder contains a number of ==messages==. A message cannot exist in more than one folders. A user can invoke a ==view== on a ==message== and in fact a user may have various ==views==, each corresponding to one message

(a) Illustrate the appropriate mechanisms in order to identify concepts and their associations.

(b) Create a domain model for the email client.

(c) Define the term *multiplicity expression* and show its applicability on the domain model.

(d) Define the term *aggregation* (and its different types) and clearly illustrate its applicability to the domain model.

Exercise 2.

- Our system is the Registrar office of a University where students go in order to enroll for classes. Consider the following success scenario of use case Process Registration:

- A student arrives at the Registrar's office in order to enroll in one or more classes. The clerk will access the terminal in order to initiate a new enrollment session (you may assume that they already have been authenticated by the system). Each enrollment session captures the date and time it was initiated. The clerk will proceed to enroll the student in each class requested. For each class, the clerk will enter the student name and identification number and the class identification number. In response to each entry, the system will display a description and a confirmation. At the end of the session, the system will display a confirmation of the procedure and the total amount of tuition fees due. The clerk will then initiate a payment of tuition fees and the system will respond with the change due and a receipt

- Create a domain model based on the above scenario.

References

- Applying UML and Patterns: An Introduction to Object Oriented Analysis and Design and Iterative Development by C. Larman. 3rd edition. Prentice Hall/Pearson, 2005

- http://www.cs.sjsu.edu/~pearce/modules/lectures/ooa/analysis/DomainModeling.htm

- http://homepage.divms.uiowa.edu/~tinelli/classes/022/Spring15/Notes/chap11.pdf