

# MN507 – Software Engineering

**Laboratory 7**: Objective Oriented and Service Oriented Design

**Submission Due**: End of laboratory class, submit the file on Moodle at least 10 minutes before the end of laboratory class.

**Total Marks** = 5 marks for 10 weeks

Marks will be given only to students who attend and participate during 2-hour laboratory class. Submission on Moodle is mandatory as an evidence of participation.

**Description of the laboratory exercise**: Form a group to work during the semester:

* 3 members each group: write your names and student ID in every submission.
* Select a leader for the first 3- 4 weeks (leadership will rotate every 3-4 weeks).
* Most of the questions throughout these lab sessions, you can discuss with your group members, but you have to submit your own answer as an individual submission on Moodle. However, some of the lab work/assignments you may need to submit only one submission as a group. Therefore, carefully read the given instruction in every lab sheet/assignment.

# Learning Outcome:

Use appropriate modelling techniques to specify system requirements. Produce and communicate to others the functional specifications of a system.

*MN507 Software Engineering Overview Laboratory 7* Page **2** of **2**

# Knowledge Tasks:

1. Describe attributes and methods of a class in object oriented programming.
2. Describe the differences between object and class in object-oriented programming.
3. Describe encapsulation, inheritance, and polymorphism in object-oriented programming.
4. Discuss the use of throw-away prototype and evolutionary prototype with simple examples.

# Practical Tasks:

**You are to draw sequence diagram for the selected software system.**

1. Develop a sequence diagram showing the interactions involved when a student registers for a course in a university. Courses may have limited enrolment, so the registration process must include checks that places are available. Assume that the student accesses an electronic course catalogue to find out about available courses.
2. You have already drawn the **Use Case Diagram and Use Case Scenarios** for the following.

**Scenario:** Consider the following description of a ATM (automated teller machine) system: An ATM or the automatic banking machine (ABM) is a banking subsystem ([subject](http://www.uml-diagrams.org/use-case-subject.html)) that provides bank customers with access to financial transactions in a public space without the need for a cashier, clerk, or bank teller.

Customer uses bank ATM to Check Balances of his/her bank accounts, Deposit Funds, Withdraw Cash and/or Transfer Funds.

ATM Technician provides Maintenance and Repairs. All these activities also involve with the Bank, whether it is related to custome0r transactions or to the ATM servicing.

* 1. Identify different **classes** involve in during the execution of “**Withdraw Money”**

use case.

**Answer:**

Customer

Bank

Account

Current Account

Savings Account

Atm Transactions

Atm

# Draw the sequence diagram for “Withdraw Money” main use case scenario.

**Answer:**

# Image result for sequence diagram for âWithdraw Moneyâ

* 1. Show all the messages passing between different objects

**Answer:**

Card enter

Request pin

Enter pin

Verify

Valid

Withdrawal amount

Request amount

Check amount

Success/failure

Take cash

Continuation

Receipt

# It is an individual task. Each student is required

1. **Upload your work into your Github**
2. **Submit the Github screenshot in the Moodle**
3. **Submit your work in the Moodle as well.**