# Department of Software Engineering

**SE-210: Software Design and Architecture**

**Class: BESE 9AB**

**Lab 01: Use Case and Class Diagrams**

**CLO2: SELECT APPROPRIATE DESIGN PATTERN AND ARCHITECTURAL PATTERN FOR A GIVEN PROBLEM.**

**Date: Oct 13th, 2020**

**Time: 10:00 am -1:00 pm**

# Instructor: Dr. Hasan Ali Khattak

# 

# Ahmed Hassan Ismail – BESE-9B - 237897

# Lab 1: Use Case and Class Diagrams

**Introduction**

Visual Modeling is a way of thinking about problems using models organized around real-world ideas. Models are useful for understanding problems, communicating with everyone involved with the project (customers, domain experts, analysts, designers, etc.), modeling enterprises, preparing documentation, and designing programs and databases

**Objectives**

Students will get themselves familiarized with the UML designing tools. Students will also learn how to write use cases and how to construct class diagrams. This lab will help students to better understanding of the following:

* Visual modeling.
* Introduction to UML.
* Introduction to visual modeling with UML.
* Use case diagrams: Discovering actors and use cases
* Class diagrams

After the completion of this lab, students will be able to:

* Find the characters from a given scenario and create use case diagrams
* Identify and define entities from a given scenario
* Draw class diagrams

**Tools/Software Requirement**

* Papyrus/Rational Rose

**Description**

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

To model a system, the most important aspect is to capture the dynamic behavior. Dynamic behavior means the behavior of the system when it is running/operating.

**Lab Task**

**Task 1**

Read the following scenario of an ATM system and draw use case diagram by using following specifications:

Actors involved:

* Operator
* Customer
* Bank

Following are the names and brief description of each of the use cases of the ATM system

System Startup Use Case

The system is started up when the operator turns the switch to "on" position. The operator will be asked to enter the amount of money currently in the cash dispenser, and a connection to the bank will be established. Customers can then use the ATM machine.

System Shutdown Use Case

Before shutting down the ATM system, the operator makes sure that no customer is using the machine. He then turns the switch to the "off" position. The connection to the bank will be shut down. The operator can then freely to remove deposited envelopes, replenish cash and paper, etc.

Invalid PIN Use Case

This use case extends the transaction use case.

Session Use Case

A session is started when a customer inserts an ATM card into the card reader slot of the machine. The ATM pulls the card into the machine and reads it. (If the reader cannot read the card due to improper insertion or a damaged stripe, the card is ejected, an error screen is displayed, and the session is aborted.) The customer is asked to enter his/her PIN, and is then allowed to perform one or more transactions, choosing from a menu of possible types of transaction in each case. After each transaction, the customer is asked whether he/she would like to perform another transaction. When the customer is finished with the transactions, the card is ejected from the machine and the session ends. If a transaction is aborted due to too many invalid PIN entries, the session is also aborted, with the card being retained in the machine.

Transaction Use Case

Note: the session use case includes transaction use case.

A transaction use case is started within a session when the customer chooses a transaction type from a menu of options. The customer will be asked to provide appropriate details (e.g. account(s) involved, amount). The transaction will then be sent to the bank, along with information from the customer's card and the PIN the customer entered.

If the bank approves the transaction, any steps needed to complete the transaction (e.g. dispensing cash or accepting an envelope) will be performed, and then a receipt will be printed. Then the customer will be asked whether he/she wishes to do another transaction.

If the bank reports that the customer's PIN is invalid, the Invalid PIN extension will be performed and then an attempt will be made to continue the transaction. If the customer's card is retained due to too many invalid PINs, the transaction will be aborted, and the customer will not be offered the option of doing another.

If a transaction is cancelled by the customer, or fails for any reason other than repeated entries of an invalid PIN, a screen will be displayed informing the customer of the reason for the failure of the transaction, and then the customer will be offered to do another transaction.

The customer may cancel a transaction by pressing the Cancel key as described for each individual type of transaction below, also following use cases use transaction use case.

* Transfer Transaction Use Case
* Deposit Transaction Use Case
* Invalid PIN Extension

**Task 2**

Consider a company which consists of several departments. Departments are located in one or more offices. One office acts as headquarter. Each department has a manager who is recruited from the set of employees.

Draw a class diagram which consists of all the classes in your system along with their attributes and operations, relationships between the classes, and other model elements that you find appropriate.

**Task 3 (optional - bonus task)**

Consider a university having different campuses. Each campus has many departments. A department has a name and it contains many offices and class rooms. A person working at the university has a unique ID and can be a professor or an employee.

A professor can be a full, associate or assistant professor and he/she is enrolled in one department.

Offices and classrooms have a number ID, and a classroom has a number of seats.

Draw a class diagram of the university that clearly demonstrates the basic aggregation and composition aggregation relationships between classes.

**Answer:**

|  |
| --- |
| Solution |
| Task 1    Task 2    Task 3 |

**Deliverables**

Compile a single word document by filling in the solution part and submit this Word file on LMS. This lab grading policy is as follows: The lab is graded between 0 to 10 marks. The submitted solution can get a maximum of 5 marks. At the end of each lab or in the next lab, there will be a viva related to the tasks. The viva has a weightage of 5 marks. Insert the solution/answer in this document. You must show the implementation of the tasks in the designing tool, along with your completed Word document to get your work graded. You must also submit this Word document on the LMS. In case of any problems with submissions on LMS, submit your Lab assignments by emailing it to Ms. Sundas Dawood at sundas.dawood@seecs.edu.pk