Department of
Computer Science and Engineering

Lab No. 07

Title: Develop UML Use Case Diagram for the given project.

Integrated Design Project I

CSE 324

Green University of Bangladesh

1 Objective(s)

- To understand the users view of a project using Use case diagram.

1.1 Sub-Objective(s)

- To represent the goals of system-user interactions.
- To define and organize functional requirements in a system.
- To specify the context and requirements of a system.
- To model the basic flow of events in a use case.

2 Problem analysis

Discovering the functional requirements from the problem statements of an intended system is challenging job. Use case represents the system's functionality, the requirements of the system from the user's perspective. A use case diagram is a dynamic or behavior diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. A Use Case is a set of scenarios describing an interaction between a user and a system which displays (in symbolic form) the relationship among actor and use cases.

Problem statement and motivation for the given project: The project entitled Banking ATM system has a drastic change to that of the older version of banking system, customer feel inconvenient with the transaction method as it was in the hands of the bank employees. In our ATM system, the above problem is overcome here, the transactions are done in person by the customer thus makes the customers feel safe and secure.Thus the application of our system helps the customer in withdrawing money, checking the balance and transaction of the amount with mini-statement and transferring the balance by validating the pin number therefore ATM system is more user friendly.

3 Methodology

To draw an use case diagram of a system we have to use some symbols and notations are shown in Fig.1.

(a) Use cases: Horizontally shaped ovals that represent the different uses that a user might have.

(b) Actors: Stick figures that represent the people actually employing the use cases.

(c) Communication Links: A line between actors and use cases. In complex diagrams, it is important to know which actors are associated with which use cases.

(i) Association: Association link is shown by connecting an actor to a use case by a solid link. (ii) Generalization: Generalization is shown by connecting an actor to a use case by a solid arrow link. A taxonomical relationship between a general use case and a more specific use case. (iii) Include: Include Relationship between a base use case and an inclusion use case, specifying how the behavior of the inclusion use case can be inserted into the behavior defined for the base use case. (iv) Extend: Relationship between the extension use case and the base use case. Specify how the behavior of the extension use case can be inserted into the behavior defined for the base use case.

(d) System boundary boxes: A box that sets a system scope to use cases. All use cases outside the box would be considered outside the scope of that system. For example, Psycho Killer is outside the scope of occupations in the chainsaw example found below.

(e) Packages: A UML shape that allows you to put different elements into groups. Just as with component diagrams, these groupings are represented as file folders.

3.1 Required Software

Anyone of the following tools can be used to draw the UML use case diagram.

1. Visual Paradigm for UML 8.2
2. StartUML
3. Lucidchart and other drawing tools

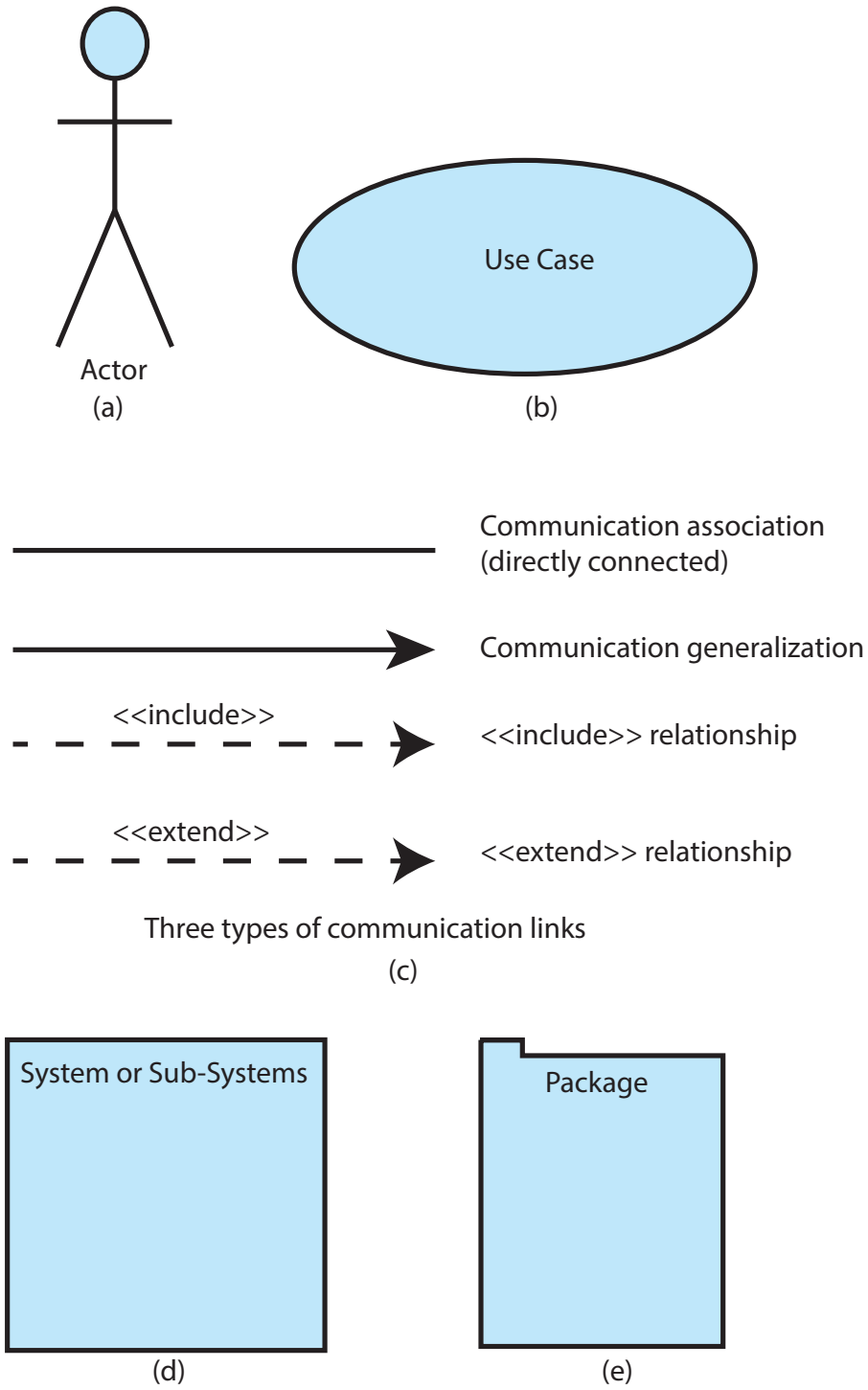

Figure 1: Different symbols and notations are used to draw an use case diagram

3.2 Procedure

You can draw use case diagrams in VP-UML as well as to document the event flows of use cases using the flow-of-events editor of UML 8.2. The steps are as follows.

Step 1: Right click Use Case Diagram on Diagram Navigator and select New Use the pop-up menu (as shown in Fig.2).

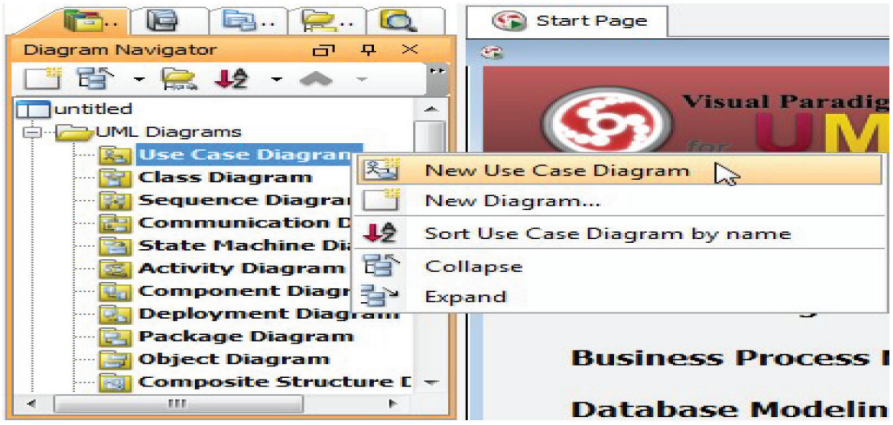

Figure 2: New Use Case Diagram selection from the pop-up menu of the VP-UML

Step 2: Enter name for the newly created use case diagram in the text field of pop-up box on the top left corner (as shown in Fig.3).

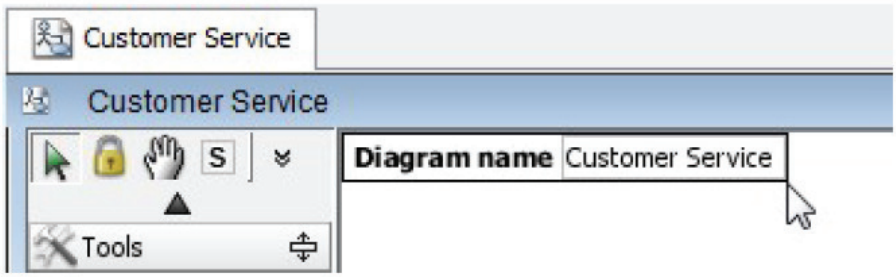

Figure 3: Enter name for the newly created use case diagram

Step 3: To create a system, select System on the diagram toolbar and then click it on the diagram pane. Finally, name the newly created system when it is created (as shown in Fig.4).

Step 4: To Draw an actor, select Actor on the diagram toolbar and then click it on the diagram pane. Finally, name the newly created actor when it is created (as shown in Fig.5).

Step 5: Besides creating an use case through diagram toolbar, you can also create it through resource iMcoonv.e the mouse over a shape and press a resource icon that can create use case. Drag it and then release the mouse button until it reaches to your preferred place. The source shape and the newly created use cases are connected. Finally, name the newly created use cases (as shown in Fig.6).

Step 6: If an use cases is too wide, for a better outlook, you may resize it by dragging the filled selectors. As a result, the name of use case will be line-wrapped automatically (as shown in Fig.7).

Step 7: To create an «extend» relationship follow the Fig.8.

Step 8: To create an «include» relationship, mouse over a use case and press its resource icon Include -> use case according to Fig.9.

Step 9: You can organize use cases with package when there are many of them on the diagram. Select Package on the diagram toolbar (under Common category) according to Fig.10.

Step 10: Drag the mouse to create a package surrounding those use cases (as shown in Fig.11).

Step 11: Finally, name the package (as shown in Fig.12).

Step 12: You may assign IDs to actors/use cases. By default, IDs are assigned with the order of object creation, starting from one onwards.

To define the format of ID, select Tools > Options from the main the Options dialog box. Select Diagramming from the list on the left hand and select the use case diagram tab on the right hand side (as shown in Fig.13).

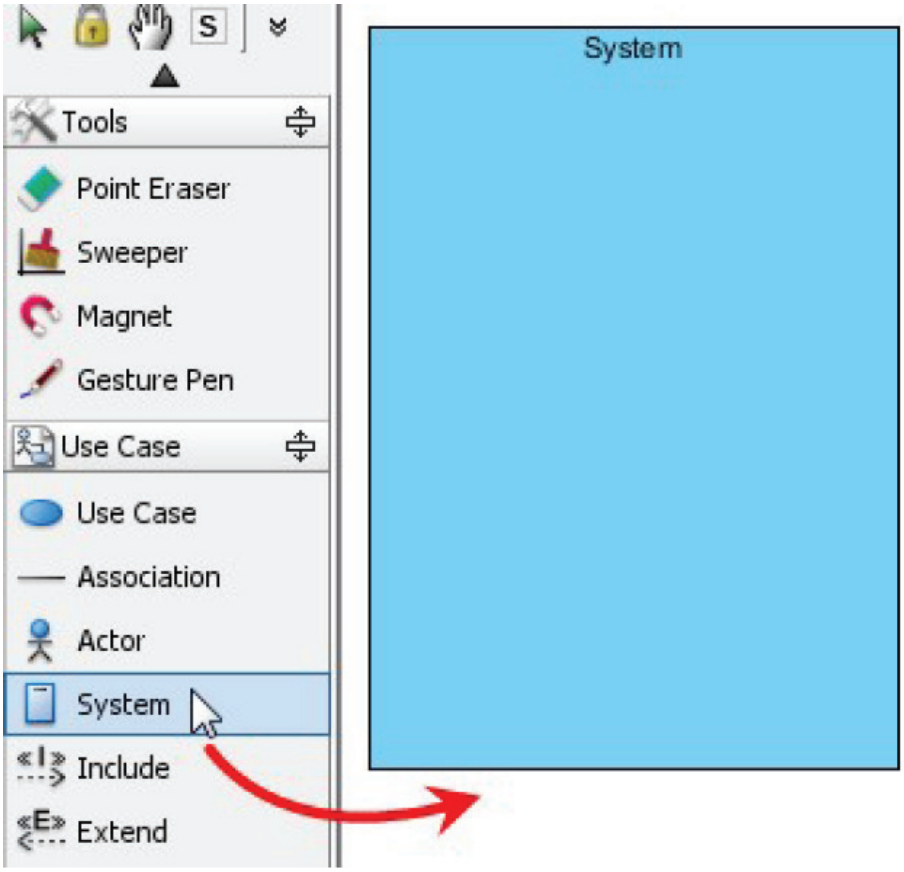

Figure 4: Drawing a system of the use case diagram

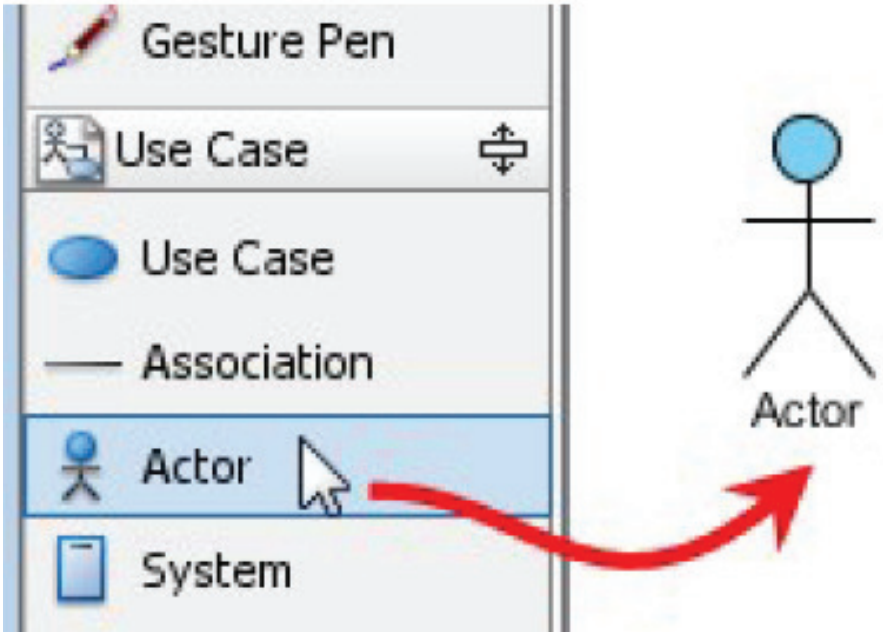

Figure 5: Drawing an actor of the use case diagram

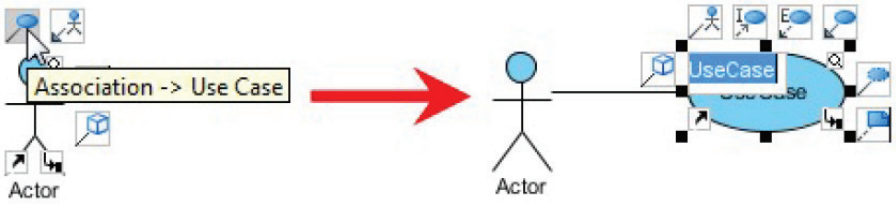

Figure 6: Drawing use cases

Figure 7: Create an use case through resource icon

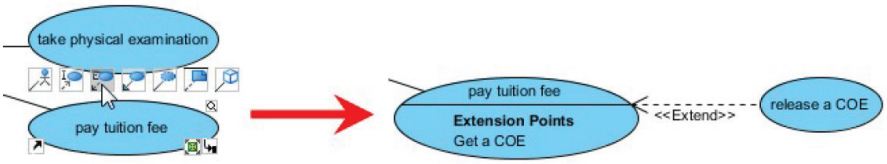

Figure 8: Create an extend relationship

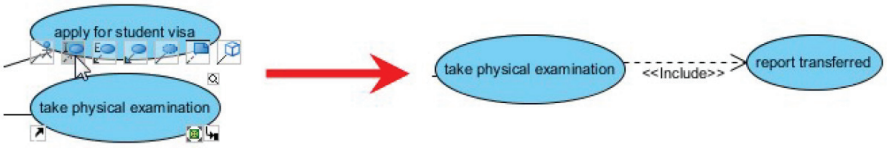

Figure 9: Create an include relationship

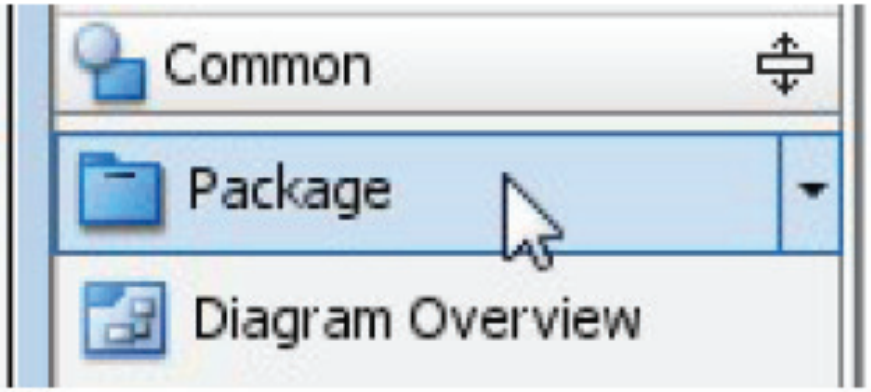

Figure 10: Structuring use case with package

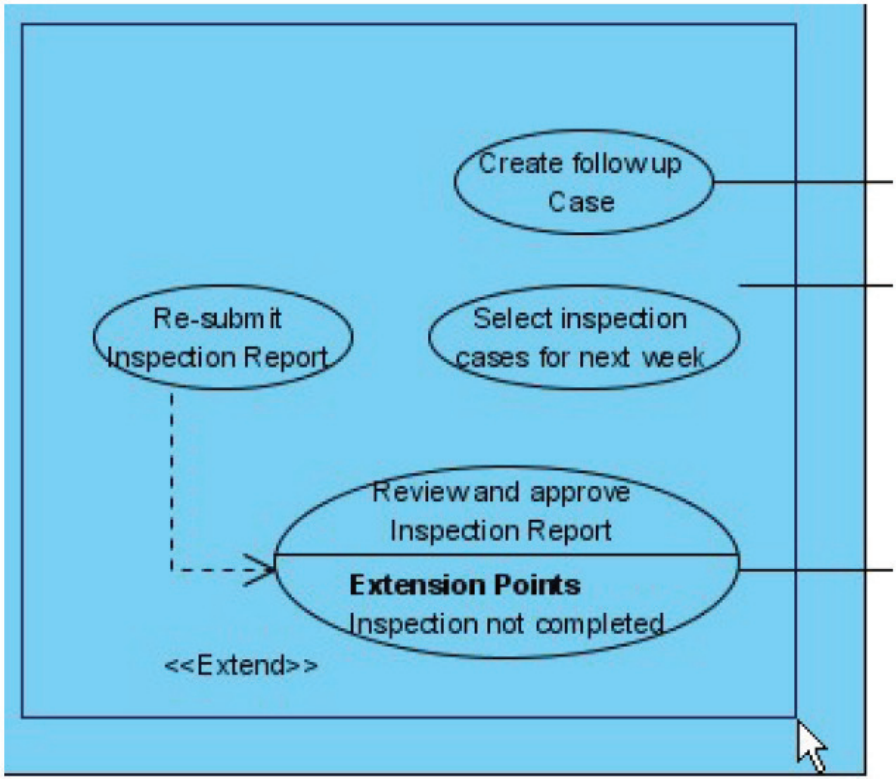

Figure 11: Creating a package surrounding those use cases

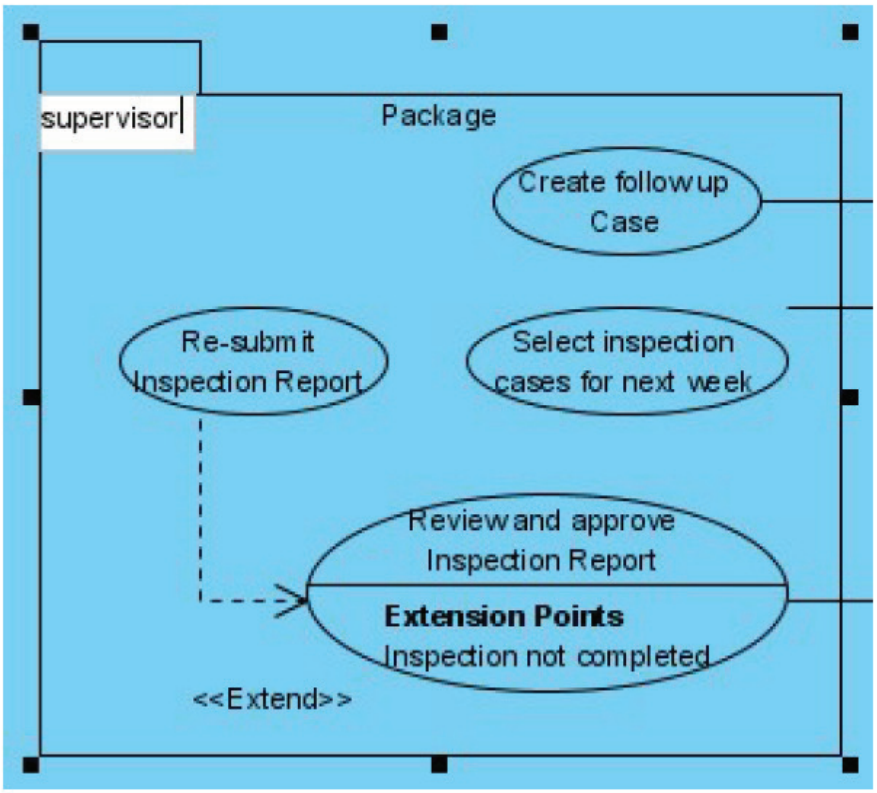

Figure 12: Naming the package

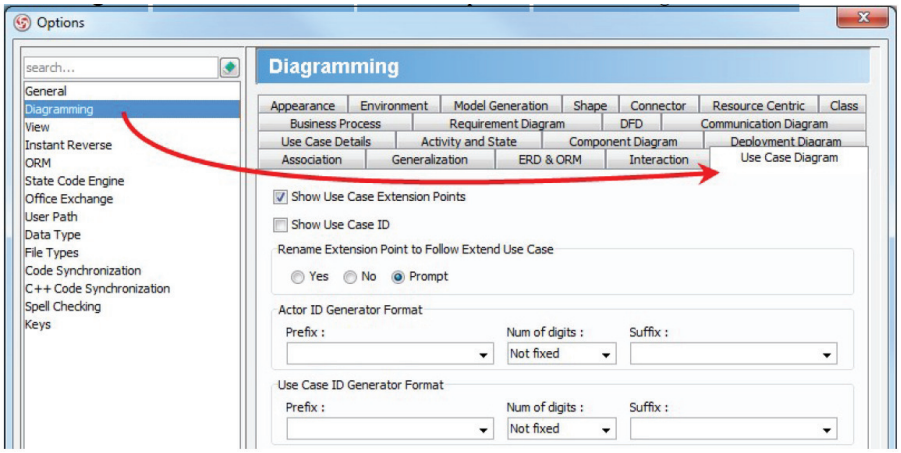

Figure 13: Assigning IDs to actors/Use cases

4 Implementation

The use case diagram of the Banking ATM system is shown in Fig.14.

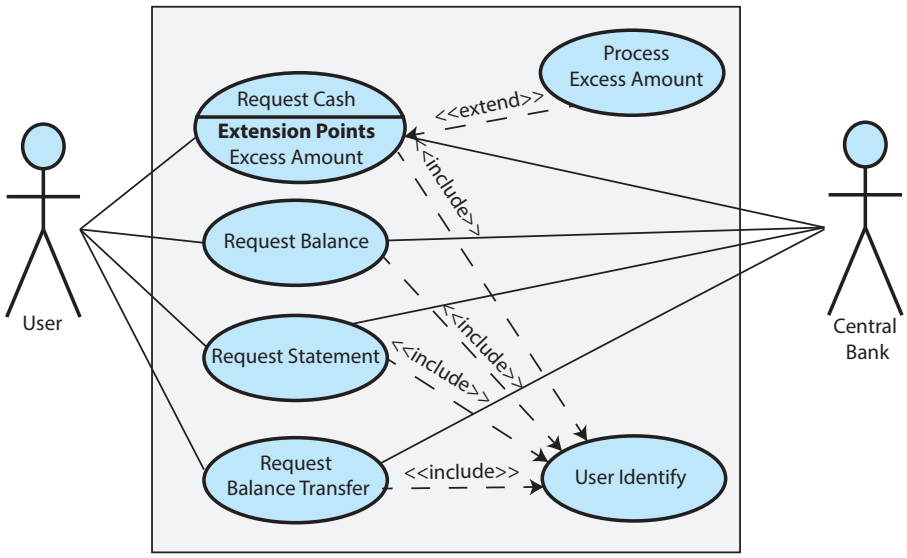

Figure 14: Use case diagram of the Banking ATM system

5 Input/Output

For the use case 'Request Cash' the output scenario documentation is shown in Fig.15 of the program is given below.

6 Discussion & Conclusion

Based on the focused objective(s) to understand about UML use case diagram, the additional lab exercise made me more confident towards the fulfilment of the objectives(s). This use case diagram is a visual representation of the process required to write and publish documentation of a system.

Start of Primary scenario/transaction

1. The user inserts their ID card into the system.
2. The system reads the magnetic strip from the card.
3. If the system cannot read the card then **<<Scenario 1>>**
4. The system contacts the banks central computer to request the PIN number for the card and their account details.
5. If bank central computer cannot access users account then **<<Scenario 2>>**
6. The system prompts the user for their PIN.
7. The user enters their PIN.
8. If PIN cannot be authenticated **<<Scenario 3>>**
9. The user is prompted for the amount of the withdrawal.
10. The user enters the amount of withdrawal.
11. The system checks with the banks central computer
12. If the user has insufficient funds **<<Scenario 4>>**
13. The cash is dispensed and the customer's account at the Bank Central Computer is debited with the withdrawal amount.
14. The card is returned to the user and a receipt issued.

End-Of-Transaction

Scenario 1: The users card is returned. End of Transaction
Scenario 2: The users card is returned. End of Transaction
Scenario 3: The user is given two more attempts to enter a correct PIN. If this fails the card is kept and the transaction ends. Otherwise resume primary scenario.
Scenario 4: The user is given the opportunity to enter a lesser amount or cancel the transaction. If cancel is chosen, the card is returned and the transaction ends. If the lesser amount is acceptable then resume primary scenario.

Figure 15: Output scenario documentation for the use case 'Request Cash' of the Banking ATM system

7 Lab Task (Please implement yourself and show the output to the instructor)

1. Draw an Use Case diagram for the given project 'Library Management System'.

7.1 Problem analysis

A library database system is an infrastructure that allows users to search books and book content, add/remove, and download selected books. The problem faced is that library users require an efficient method to find a specific book or keyword(s) within a book given a continuously expanding library. Some scenarios of the 'Library Management System' are as follows:

1. User who registers himself as a new user initially is regarded as staff or student for the library system.

(i) For the user to get registered as a new user, registration forms are available that is needed to be fulfilled by the user.

(ii) After registration, a library card is issued to the user by the librarian. On the library card, an ID is assigned to cardholder or user.

2. After getting the library card, a new book is requested by the user as per there requirement.

3. After, requesting, the desired book or the requested book is reserved by the user that means no other user can request for that book.

4. Now, the user can renew a book that means the user can get a new due date for the desired book if the user has renewed them.

5. If the user somehow forgets to return the book before the due date, then the user pays fine. Or if the user forgets to renew the book till the due date, then the book will be overdue and the user pays fine.

6. User can fill the feedback form available if they want to.

7. Librarian has a key role in this system. Librarian adds the records in the library database about each student or user every time issuing the book or returning the book, or paying fine.

8. Librarian also deletes the record of a particular student if the student leaves the college or passed out from the college. If the book no longer exists in the library, then the record of the particular book is also deleted.

9. Updating database is the important role of Librarian.

8 Lab Exercise (Submit as a report)

- Draw an Use Case diagram for the given project for your team.

9 Policy

Copying from internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.