Professor: *Dr. Mohammed Ayoub Alaoui Mhamdi*

Bishop's University

CS 410 / CS 560 – Software Engineering

Final exam

Winter 2020

The exam is worth 50 points total and has two problems. Be sure to read the whole exam before attempting any of it. The exam is open notes. Use the provided white space to respond to each question. Please, write legibly.

Full name	Contribution
Yi Ren (002269013)	
Wentao Lu (002276355)	

Problem I

We want to set up an ATM bank system for the withdrawal of money, instead of obligating the customers to come to the counter during the opening hours of the bank. Specifications:

1. Distribution to any debit cardholder via a card reader and a cash dispenser.

2. View account balance for bank customers

3. Transactions are secured

4. The ATM bank system must be reloaded once the amount of stored money reaches a minimum threshold. In fact, once it is reached, an automatic notification is sent to the a

private company for the secured transportation of money to reload ATM bank system with cashes.

The bank can have several ATM systems in the city.

The customer can be also an employee of the bank or an employee for the private service. At the first, an authentication process for the customer is performed. If it is successful the ATM system displays the different options, like withdrawal of money, viewing account balance, and paying bills. If the required amount of money is superior to the available amount in ATM system, or superior to the amount of money that the customer has, or there is a constraint of the customer of the debit card; the transaction is rejected. Otherwise, it is successful. If the authentication process for the customer fails more than three times, her or his debit is blocked and she or he must call the customer service to unblock his debit card.

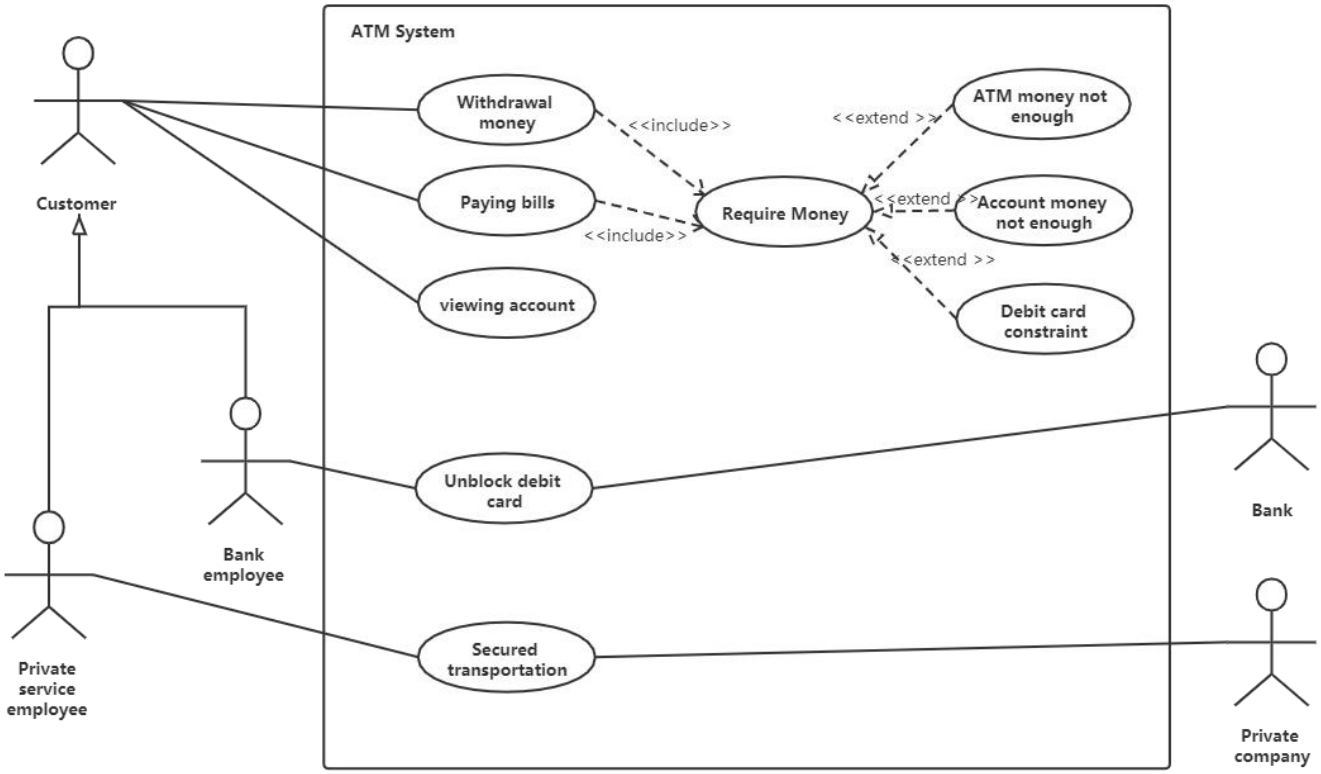
We note that a customer can have up to two type of account in this bank. The first type is called checking account, the second type is called savings account..

1. To which roles would you assign the following tasks? (Stakeholder, programmer, etc.) (10 points)
    a. The tools and libraries to develop different modules of ATM system. (2 points)
    Programmers are most familiar with the tools and libraries to develop the ATM system. In particular, experienced programmers are most likely to know what tools best fit the project, so we should leave this task to developers team.
    b. Negotiate with companies of software development about the cost of the project and its completion on schedule. (2 points)
    Negotiation on project cost and delivery schedule happens between the customer and the software development team, so this role should be assigned to the customer or user.
    c. Give details about progress in the development process of the project. (2 points)
    The project manager is responsible for reporting the development progress to the customer on a regular basis.

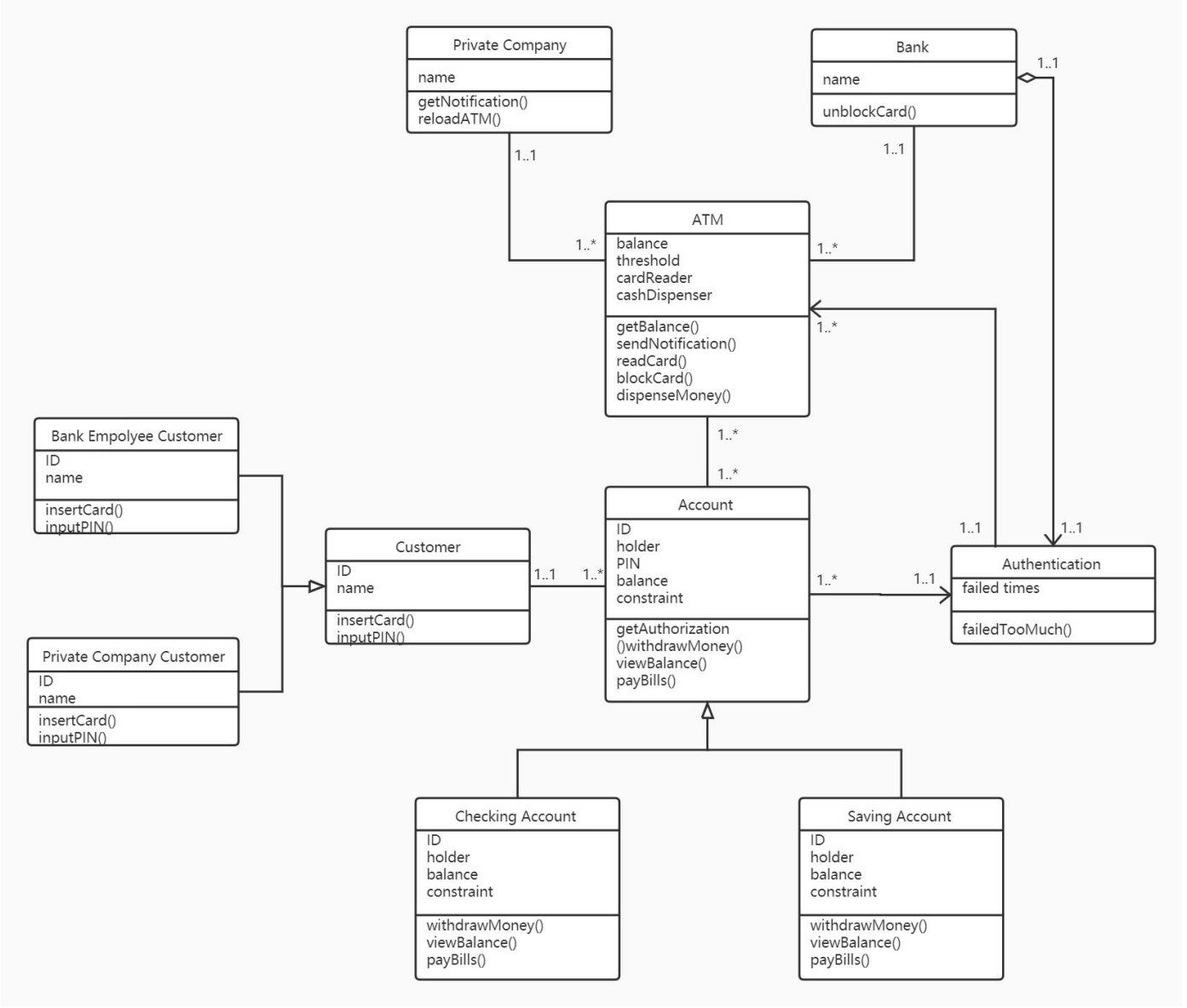
d. Approve or reject the change request for a component in the project. (**2 points**)
This task should be assigned to the project manager or the developer team who is able to precisely evaluate the impact of a change request.

e. Detect and report problems and errors in the delivered project. (**2 points**)
After the project is delivered, customers should report found bugs and errors to the developer team who would then perform corrective maintenance.

2. Write the high-level use case "Withdraw money" initiated by a customer, and refine if it is possible this high-level use case. (**4 points**)



3. Propose a class diagram to model this ATM management system. (**6 points**)



4. Consider that we want to add a card credit support for this ATM system, is it well suited to modify the content of the module that mange the customer's debit card? Justify your response. What kind of solution will you suggest? (**6 points**)

   Yes, a credit feature can be easily incorporated into our design. Although the ATM system now only allows customer to withdraw money, we just to need add a new method saveMoney() to the Account class.

In our case, the OOP design makes it easy to add this feature because only the Account class needs to be changed, then both the checking account and saving account will automatically inherit it, so that all the customer's accounts will have this feature. Other parts of the system are not changed at all, also, the credit money activity can simply reuse the other necessary functionalities we have already built such as customer authentication. This also makes it easier to do testing and maintenance. Therefore, as long as the budget and time allows, the credit feature is well suited to our design.

Problem II: Analysis and design of Minesweeper game

The goal of this case study is to design a minesweeper game like the one that ships with the Microsoft Windows © operating system. The aim of the game is to find all the squares on the board containing mines as quickly as possible without touching them.

The game consists of a rectangular board, a stopwatch and a mine counter. The board is a grid of boxes. At the start of the game, all the squares on the board are covered, the mine counter indicating the number of mines remaining to be located. The stopwatch counts the number of seconds since the start of the game. The game begins when the first box is discovered. When a box is discovered, its content is displayed. The content of a box can

be: nothing, a mine or a number indicating the number of mines present in the neighboring boxes. The following scenarios can occur when a box is discovered, depending on its content:

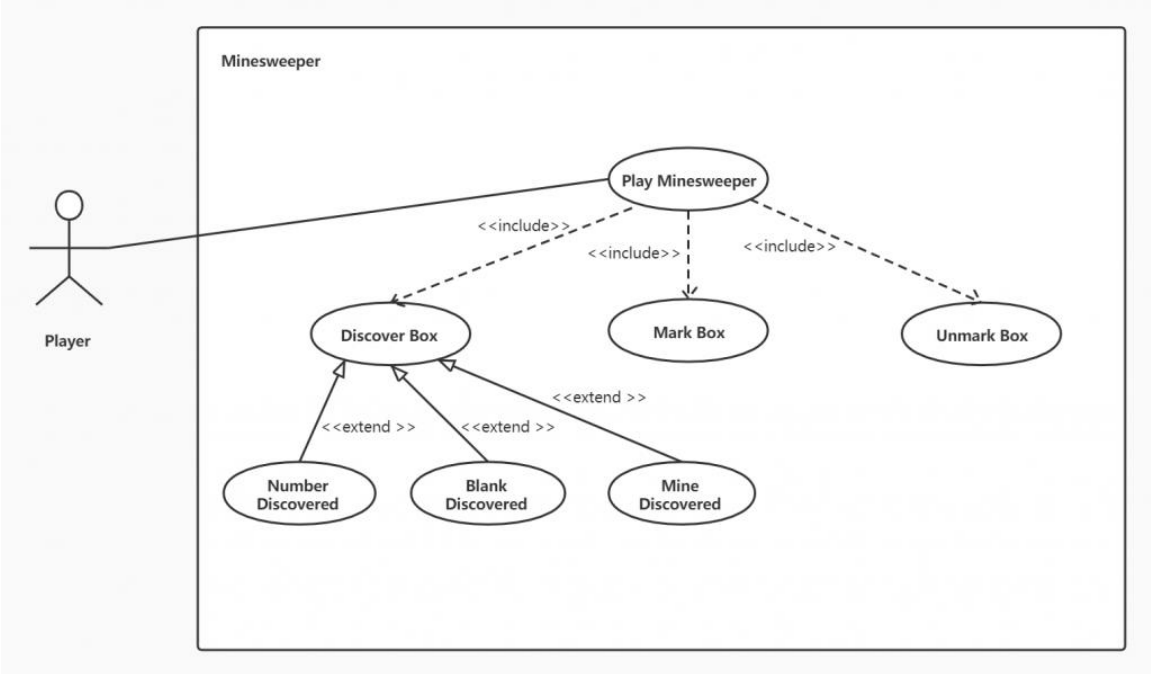
1. A number - Nothing happens.
2. A blank - All neighboring boxes are exposed, in condition that were not reported by a flag. If one of these neighboring boxes contains nothing, the discovery process automatically continues from this box.
3. A mine - The game is over, and the player has lost.

If it is still covered, a box can be marked according to the following rules:

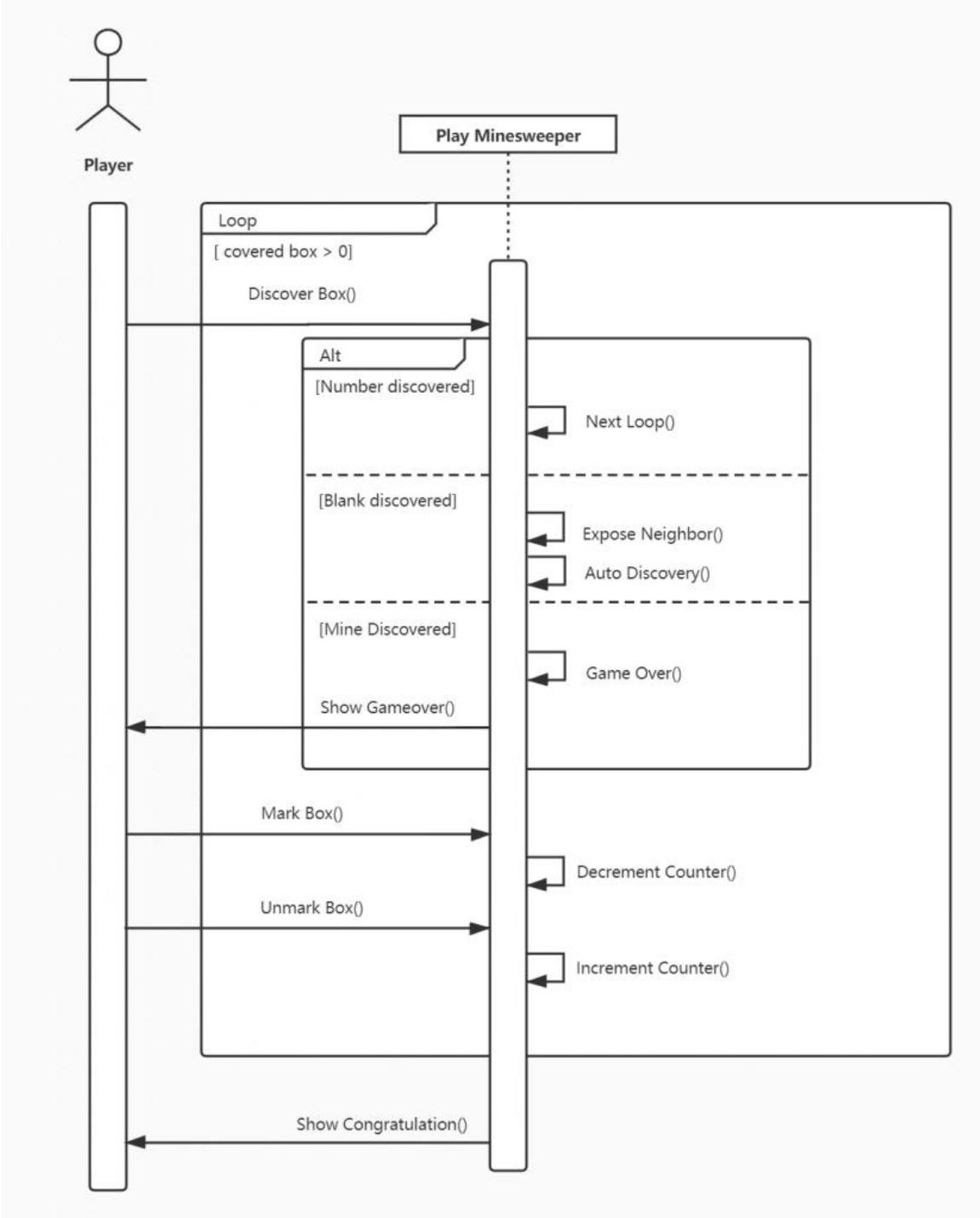
Marking a box that is neither discovered nor marked decrements the remaining mine counter and a flag appears on the box. He indicates that this box potentially contains a mine. A square marked with a flag cannot be discovered.

Marking an already flagged box with a flag will restore it to its original state, i.e. covered and unmarked. The mine counter is then incremented by 1.

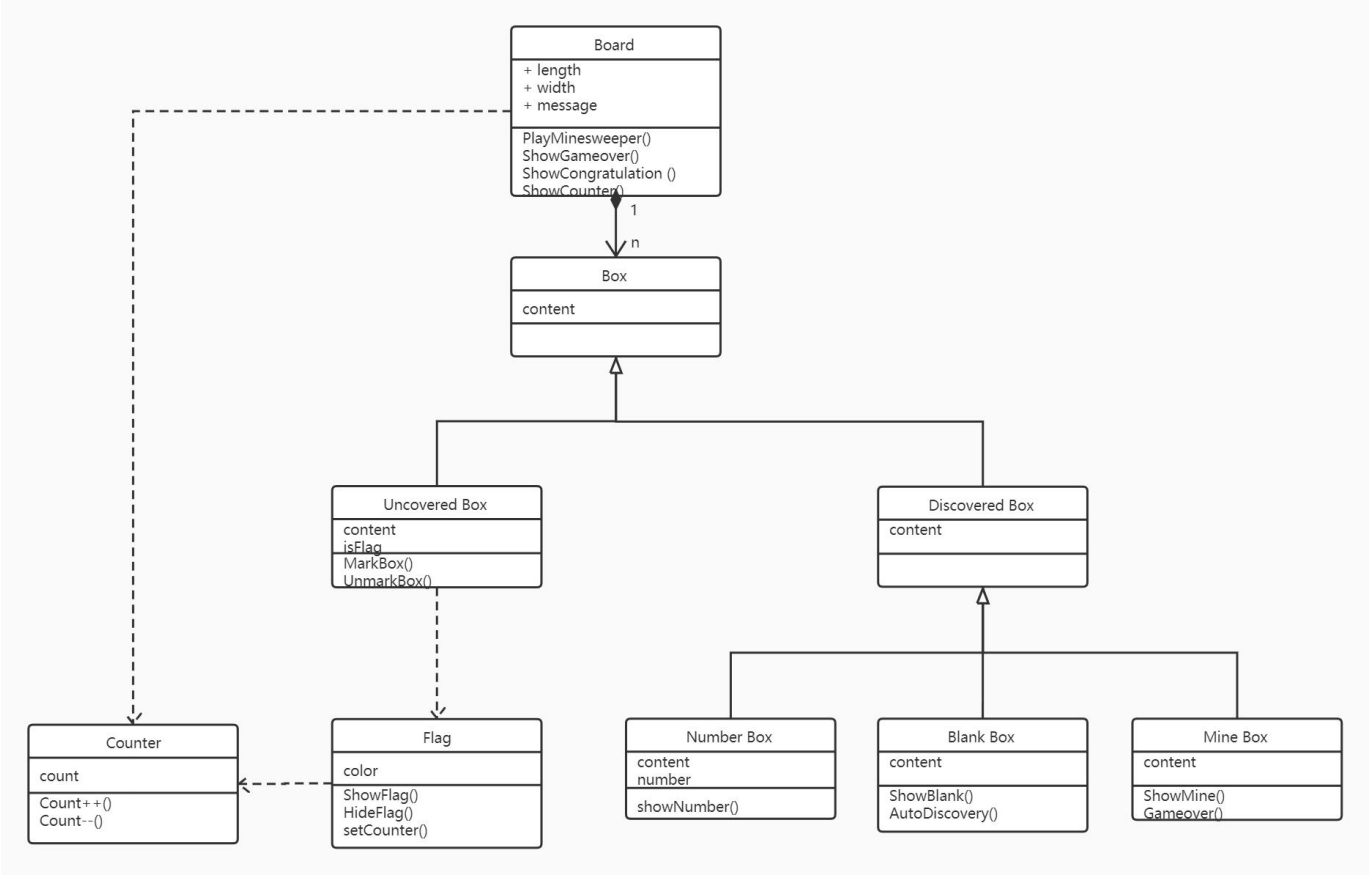
1. Develop a use case diagram for the Minesweeper game. (**6 points**)



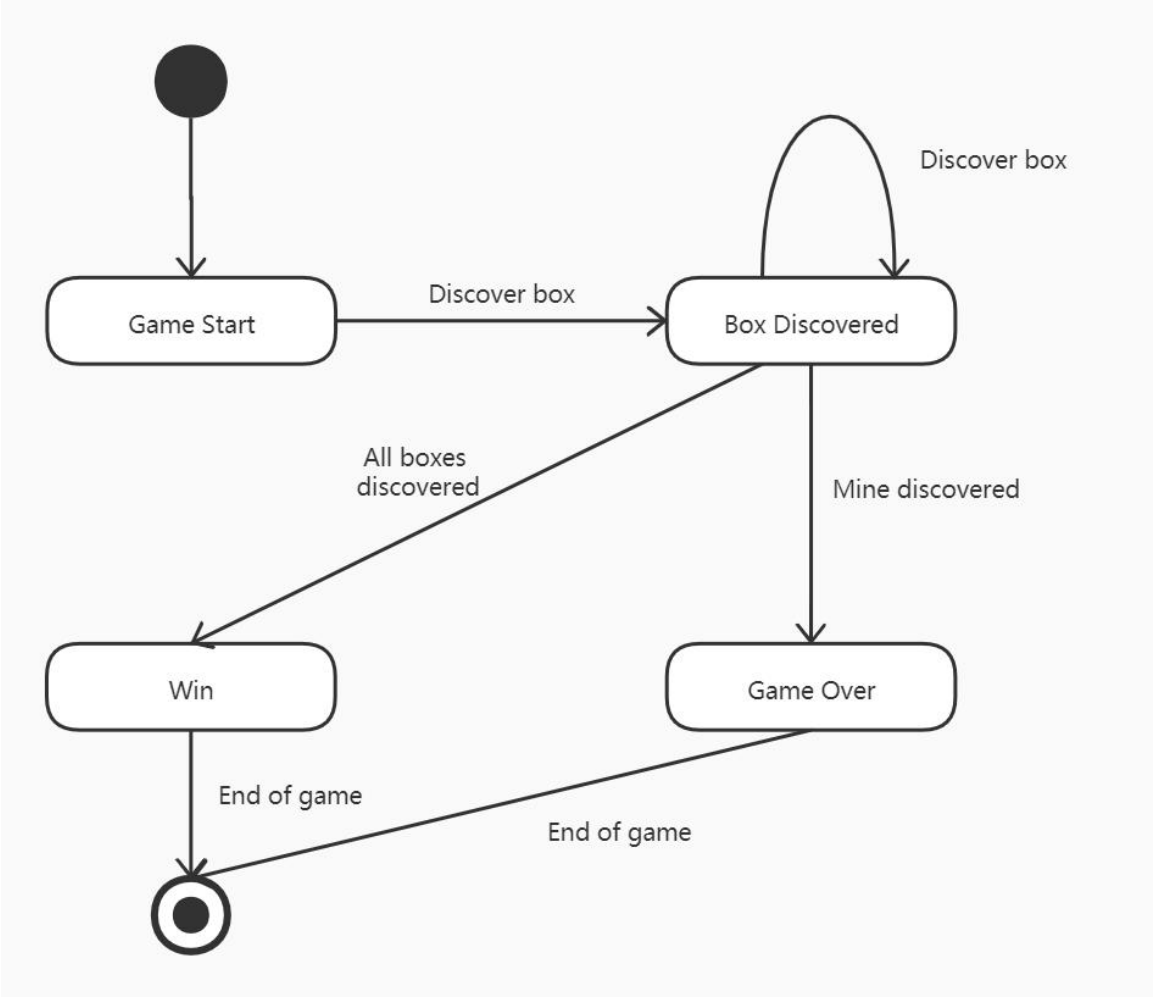
2. Develop a sequence diagram for the "Play Minesweeper" use case. (**6 points**)



3. Propose a class diagram for this system. (**6 points**)



4. Draw a state diagram for this system. (**6 points**)


Discover box
Game Start
Discover box
Box Discovered
All boxes
discovered
Mine discovered
Win
Game Over
End of game
End of game