

SOFTWARE ENGINEERING (503074)

(WEEK 03)

MAI VAN MANH

EXCERCISE 1

1.1 Discover ambiguities or omissions in the following statement of requirements for part of a ticket-issuing system, Ask questions and write them in the [Q&A Template.xls](#) file.

An automated ticket-issuing system sells rail tickets. Users select their destination and input a credit card and a personal identification number. The rail ticket is issued and their credit card account charged. When the user presses the start button, a menu display of potential destinations is activated, along with a message to the user to select a destination. Once a destination has been selected, users are requested to input their credit card. Its validity is checked and the user is then requested to input a personal identifier. When the credit transaction has been validated, the ticket is issued.

1.2 Write a set of functional and non-functional requirements for the ticket-issuing system, setting out its expected reliability and response time. [Answer this exercise on Microsoft Word or Notepad.](#)

EXCERCISE 2

2.1 Do you know that it costs a lot of money to get a “Certified Java Programmer” certificate? It could cost you thousands of euros. Let’s imagine we will develop a browser-based training system to help people prepare for such a certification exam.

A user can request a quiz for the system. The system picks a set of questions from its database, and compose them together to make a quiz. It rates the user’s answers, and gives hints if the user requests it.

In addition to users, we also have tutors who provide questions and hints. And also examiners who must certify questions to make sure they are not too trivial, and that they are essential. [Make a use case diagram to model this system. Work out some of your use cases.](#) Since we don’t have real stake holders here, you are free to fill in details you think is essential for this example.

→ You can use StarUML or any online tools to draw diagrams.

2.2 Suppose we want to develop software for an alarm clock. The clock shows the time of day. Using buttons, the user can set the hours and minutes fields individually, and choose between 12 and 24-hour display. It is possible to set one or two alarms. When an alarm fires, it will sound some noise. The user can turn it off, or choose to ‘snooze’. If the user does not respond at all, the alarm will turn off itself after 2 minutes. ‘Snoozing’ means to turn off the sound, but the alarm will fire again after some minutes of delay. This “snoozing time” is pre-adjustable.

[Identify the top-level functional requirement for the clock, and model it with a use case diagram.](#)

2.3 Using your knowledge of how an ATM is used, develop a set of use cases that could serve as a basis for understanding the requirements for an ATM system.