**Assignment #1**

1)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.

Actor 1

(Mom)

Actor 2

(Me)

Use case: Snooze.

Primary actor: User

Pre-condition: An alarm is firing.

Post-condition: -

Main flow:

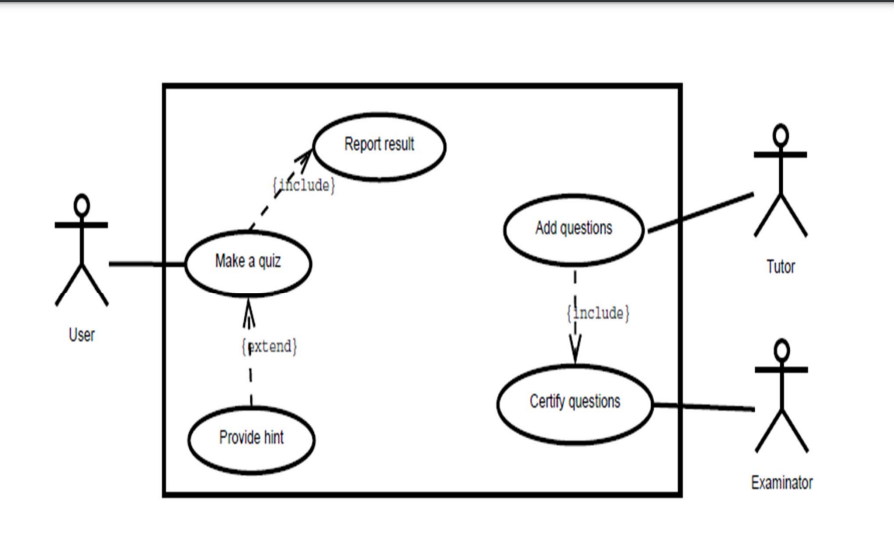
1. The use-case is activated when the user hits the snooze button.

2. The alarm is turned off.

3. Wait for snooze time.

2)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 exam-  
inators who must certify questions to make sure they are not too trivial, and that they are  
sensical.  
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 sensical for this  
example.

****

Use case: Make quiz.

Primary actor: User

Pre-condition: The system has at least 10 questions.

Post-condition: -

Main flow:

1. The use-case is activated when the user requests it.

2. The user specifies the difficulty level.

3. The system selects 10 questions, and offers them as a quiz to the user.

4. The system starts a timer.

5. For every question:

5a. The user selects an answer, or skip. [Extension point]

6. If the user is done with the quiz, or the timer runs out, the quiz is concluded.