Software Engineering with Objects and Components
Practical Work: Tutorial 3

Please read this tutorial sheet before you arrive at the tutorial. You are required to do some preparation for the tutorial. Before the tutorial:

1. **Get together with your team and pull together your work on the *software requirements specification*, *use case diagram* and *class diagram* for your facet of the system. This should be readable by other teams, it needs not to be complete or polished. You have to bring this description to your tutorial. *Hint: use the templates (i.e., Software Requirements Specification template and Use Case description template) tailored for your purposes.***

2. **Prepare a draft of your preliminary deliverable 1. Your draft should include the following: The name of the tutorial group; the team identifier (i.e., O, P or D); The names of each of your team members; A brief update of your use cases: Did you change anything since last week presentation? Did you take into account any feedback/comment? The class diagram(s) you have produced.**

The aims of this tutorial are:

1. To present changes on your use cases
2. To present your preliminary design in terms of class diagrams
3. To provide/receive feedback to/from the other teams
4. To begin your validation by CRC Cards

After the tutorial: You should now be very close to completing deliverable 1. You may want to consolidate and double check your deliverable 1 before submission. Moreover, your submission should be a structured document clearly identifying the different parts of deliverable 1.

Tutorial Instructions

Inspection. Each team will have 10 minutes to review each other work on the **requirements**, **use cases** and **class diagrams**.

Question Time. The tutor will moderate questions after the inspection. You should be prepared to take questions about your preliminary draft deliverable.

Hint: Take notes of any question/discussion during the tutorial. After the tutorial, you should review your work according to the tutorial feedback (e.g., questions, errors, clarifications, changes, etc.).

Tutorial Outcomes

By the end of this tutorial your group should

1. Have some feedback on your class diagrams.
2. Have a reviewed draft on your UML modelling design of deliverable 1.
3. Have started the validation of your design.

Tutorial Activities

Activity 1 - Validation [30 mins]

☞ *Individually* start to write CRC cards for few classes and run a validation scenario, e.g., play a use case description with the CRC cards in order to validate your design. *Note that a validation scenario may involve different use cases too.*

✎ *Describe a validation scenario that may involve one or more use cases.*

Validation Scenario

✎ *Write the CRC cards for the classes involved in the realisation of the validation scenario.* CRC Cards explicitly represent multiple objects simultaneously: The **Name** of the class it refers to; The **Responsibilities** of the class (these should be high level, not at the level of individual methods); The **Collaborators** that help discharge a responsibility.

Class Name	
Responsibilities	Collaborators

Class Name	
Responsibilities	Collaborators

Class Name	
Responsibilities	Collaborators

Class Name	
Responsibilities	Collaborators

Class Name	
Responsibilities	Collaborators

Class Name	
Responsibilities	Collaborators

Class Name	
Responsibilities	Collaborators

Class Name	
Responsibilities	Collaborators

✎ *Write the unfolding of your validation CRC game.*

Class	*Responsibiliy*	*Collaborator*

CRC Card Validation Game

1.			
2.			
3.			
4.			
5.			
6.			
7.			
8.			
9.			
10.			

⋮

✎ *Take notes of any required changes to your classes (e.g., different responsibilities, different collaborators, new classes, etc.).*

♯	*identified issues, changes or comments*

Activity 2 - Inspection [10 mins]

☞ *Collectively* Each team will have 10 minutes to inspect the draft **Class Diagram** of another team.

♯	*identified issues, questions or comments*

Activity 3 - Discussion [10 mins]

☞ *Collectively* Identify / discuss open issues or changes that you need to address into your requirements, use cases and classes. At the end of the tutorial, each group should have a list of issues and changes that need to be addressed.

♯	*identified issues, questions or comments*