

# Software Engineering with Objects and Components

## Practical Work: Tutorial 1

**Note that you are required to do some preparation for the tutorial. Before your tutorial, on your own, read over the main documents, which represent the basis for your system requirements.**

This is your first tutorial. The aims of this tutorial are:

1. To get to know the members of your tutorial group.
2. To form the teams in your tutorial group.
3. To agree on a name for your software company (tutorial group).
4. To gather system requirements.
5. To practice creating use cases.

## Tutorial Instructions

You have 20 minutes to gather some system requirements, to create some use cases for your system and to prepare a short presentation to the rest of the group. Each team will give a presentation of the results. Your presentation should include the following:

- The names of each of your team members.
- The main requirements you have identified.
- The use cases you have produced.
- A proposed name for the software company your tutorial group represents.

## Tutorial Outcomes

By the end of this tutorial your group should

1. have agreed on a name for your software company
2. have a preliminary requirements and use case model for your part of the system.

Moreover, your team should also take this opportunity to organise how and when you are going to coordinate the preparation for subsequent tutorials.

## Tutorial Activities

The activities involve specific tasks described in what follows. Note that these tasks are very tightly timed — your tutor will enforce these so the activity fits in the available time.

## Forming the Teams [10 mins]

☞ Your tutor in collaboration with you decides on the composition of the three teams in the tutorial group. These are: **Orders (O)**, **Plants (P)**, and **Deliveries (D)**.

	Team Orders (O)	Team Plants (P)	Team Deliveries(D)
1			
2			
3			
4			
5			

## Gathering Requirements [15 mins]

☞ *Individually* each member of the team reads the preliminary project documents and identifies the main requirements (e.g., functional or non-functional) as well as open issues (e.g., unknown or unspecified requirements).

### Functional Requirements

Req. #	Requirement

### Non-functional Requirements

Req. #	Requirement

### Open Issues

#	Issue

☞ *Collectively* each team discusses the identified requirements in order to consolidate any understanding about the system.

## Creating Use Cases [10 mins]

☞ *Collectively* each team chooses some of the (functional) requirements for creating a list of use case scenarios as they read the document. Consolidate your lists and generalise the scenarios to create a preliminary collection of use cases for your facet of the system.

### Use Cases

Use Case #	Actor(s)	Use Case Description

*draw some use cases from the list above*

☞ *The whole team* should attempt to merge the list of potential use cases to create a composite list.

## Presenting Preliminary Requirements and Use cases [15 mins]

☞ *A rapporteur* for the team presents the preliminary identified requirements and use cases for their subsystem. **[5 mins each team]**