# Notes for Lab4
# SFWR ENG 3KO4 / 3MO4
# Dr. Kamran Sartipi

## ❖ General:

This short document provides brief guidelines on how to design a Component Diagram and corresponding Statechart for the architectural design of a software system.

### ➢ Component diagram (usage guide):

❑ In a component diagram, the "expressive naming" of the component's interfaces is very important. The name of a service should summarize the core functionality of the service. The generic names such as "information", "data", "service", "operation", "database", "query", "log" that are used without any qualifiers should be avoided.

❑ A dashed arrow means that the component at the source of the arrow requests the service pointed to by the head of the arrow. The direction of the arrow does not necessarily mean that data is being sent to the sink component. However, data can be sent or receive by invoking a service.

❑ The collection of the component interfaces (services) in the Component Diagram should have sufficient functionality to meet the requirements specified by the SRS description. This can be verified by investigating the requirements one by one and see how each can be performed using the available component services.

❑ Every component interface should be labeled.

❑ A component usually provides more than one service, however the kind of services must be related.

### ➢ Statechart diagram (usage guide):

❑ We use Statecharts to specify the behavior of a system based on the states that the system can take, and transitions between the states that can be triggered by events and/or conditions. As a result of a transition one or more actions can be performed as well.

❑ At the architectural level, the interface of the components (component services) and also the user-interaction with the components (through user-interfaces of the system) can be used as the labels of the transitions.

When the system is in a state it can perform some operations in that state. As a result of those internal operations the system may request for the services of other components (through their interfaces). This service request is interpreted as triggering the transitions and the system will change its state and enters into another state.