

Inf2-SEPP 2023-24

Tutorial 2 (Week 3, on SE)

Requirements Engineering

Study this tutorial sheet and make notes of your answers **BEFORE** the tutorial.

1 Introduction

In this tutorial you will look at some example systems and create for them use-case diagrams, use case descriptions and lists of requirements. The tasks are provided in increasing level of difficulty and are intended to be attempted in order.

2 Task 1: Requirements for an electronic health clinic

After the ReformHealthcare party's surprise win in the 2010 UK General Election, a new type of health clinic is to be set up, using treatments whose effectiveness is highly controversial. Because the health outcomes associated with the new clinics will be of wide interest, a new computer system must quickly be developed to track patient treatments and outcomes.

All patient appointments must be entered into the system by the clinic administrators. During the appointment, the doctor or nurse will enter details of the patient's condition and any treatment prescribed. Patients are requested to report on any changes in their health, one week after each appointment. They can do this either by filling in a web form or by telephoning the clinic, in which case an administrator will enter the information.

The system must be able to produce reports of statistics such as the number of patients treated in a given period, their conditions, and their reported state of health a week afterwards.

- (a) What is a *functional requirement* and what is a *non-functional requirement*? Explain, giving at least 3 examples of each for the electronic health clinic. For non-functional requirements, you may want to start from the Wikipedia entry on this topic.

In formulating the requirements, use carefully structured English like shown on slide 19 of Lecture 4.

- (b) What are *stakeholders*? Using the system description and the system's context, suggest different stakeholders for it, with a brief explanation of why each has a stake.
- (c) What *elicitation technique(s)* would you use with your identified stakeholders, to clarify their requirements? Briefly justify your choice for 2-3 groups of stakeholders for whom your answer could be different.
- (d) Explain how use cases can be useful for managing requirements of systems with multiple stakeholders, and give some drawbacks of use cases for managing requirements.
- (e) Draw a use case diagram for the system described in part (b). As you do so, pay attention to the notation for actors, use cases and their associations. Comment briefly on any important ambiguities or problems you find in the system description, and state any assumptions.

3 Task 2: Requirements for a food box system

The Scottish Government has put forth plans for a central, online food box order and delivery system, to help support individuals who are shielding during Covid-19. (...)

Shielding individuals will register and be uniquely identified using their Community Health Index (CHI) number. (...) During this registration process, the system communicates with Public Health Scotland's electronic record system (an external system) to check that the provided CHI number indeed belongs to a shielding individual, and if so retrieves their personal information (full name, address and phone number). Registration is refused if a person is not currently a shielding individual.(...)

Once registered, shielding individuals will have the option to request a food box from (...) a supermarket, once a week only. (...). The system uses information it holds about the individual's post code to suggest supermarkets which can deliver to their location, the normal time it takes for them to deliver, and provides links to the supermarkets' dedicated "food box" web pages. Therefore, the user can order directly from the supermarket and no longer use our system for this purpose. Delivery is also arranged by the supermarket. However, once the order has been placed, the supermarket system will notify our system about it, and later about each time the order changes status so that the delivery of the food box can be tracked.(...)

The system can register supermarkets as available food providers using their business name, phone number, website URL and address.

Considering the critical role of the system in the community and the ever-changing conditions and restrictions, local councils will have access to food box order data so that they can produce reports and statistics over periods of time.

- (a) Draw a use case diagram for the food box system, including associations with both primary and supporting actors (the latter where appropriate). Comment briefly on any important ambiguities or problems you find in the system description, and state any assumptions.

Things to consider:

- What happens on the food box system versus external systems or the environment?
 - What constitutes elements of interaction versus the system's internal processing?
 - How do we choose the scope of the use cases?
- (b) Take two of the most important use cases and describe them using the template provided and explained at the end of this tutorial.
- Keep your descriptions high level. They should be focused on the interaction and not go as far as mentioning design or graphical user interface elements. Also, continue to pay attention to the list of things to consider from part a).
- (c) Give examples of 2 stakeholders of the food box system who are not also use case actors, and of 2 use case actors that are not also stakeholders. Justify your replies by referencing the definitions of these terms.
- (d) Write 2 measurable non-functional requirements for the food box system.

4 Task 3 (Advanced): A lift system

Consider a single lift system where on each floor by the lift doors there is a button for calling the lift and inside the lift there are buttons for selecting the desired destination floor. The lift controller system monitors the buttons and position of the lift, and controls the lift motor for raising and lowering the lift, the doors opening and closing, lights in each of the buttons, and lift position displays both in the lift and above the doors at each floor. The light in each button should normally be lit when the button is pressed and then turned off at some appropriate point later in time. Assume for simplicity there is just one user at a time.

As an extension you could add a 'cancel' button. If a user has waited too long they may instead decide to use the stairs and may cancel their request.

Write a description of the most important use cases that you can identify for this system, using the template from the end of the tutorial.

4.1 Use-Case Template

Use case name: Often a short verb phrase.

Primary actor: The actor with the goal the use case is trying to satisfy. Is usually but not always the initiator of the use case.

Supporting actors: Others the system communicates with while carrying out the use case.

Summary: A one or two sentence description of the use case. Make clear the goal that the primary actor wishes to achieve. Try not to just reiterate the steps in the main success scenario.

Precondition: What the system should ensure is true before the system allows the use case to begin. Useful for telling the programmers what conditions they don't have to check for in their code.

Trigger: The event that gets the use case started.

Guarantee: What the system will ensure at the end of the use case in the event the use case execution is a success. Sometimes it can be useful to distinguish **success guarantees**, **failure guarantees** and **minimal guarantees** (guarantees for what holds at the end of all scenarios).

Main Success Scenario: The steps the primary actor and system would go through to achieve the goal. Focus on the most common and simplest course of events. If there are alternatives, record them in the Extensions field or in a separate use case.

See the lecture slides for an online shopping example. Note that each step is numbered and starts with the name of the agent (the actor or the System) that carries out the step. In describing steps, show the intent of the actor, not the low level details of e.g. user interface interactions.

Sometimes the first step is already described in the trigger. If so, don't bother repeating it.

Extensions: Variations on the Main Success Scenario. Start each with an identifier related to the step at which it starts and a statement of the condition for when the extension applies. Then list the alternate steps and if and when you return to main scenario. Ensure it is clear whether a variation results in the use-case goal being achieved.

Notes: Any comments on the use case. E.g. mention associated non-functional requirements or open issues.