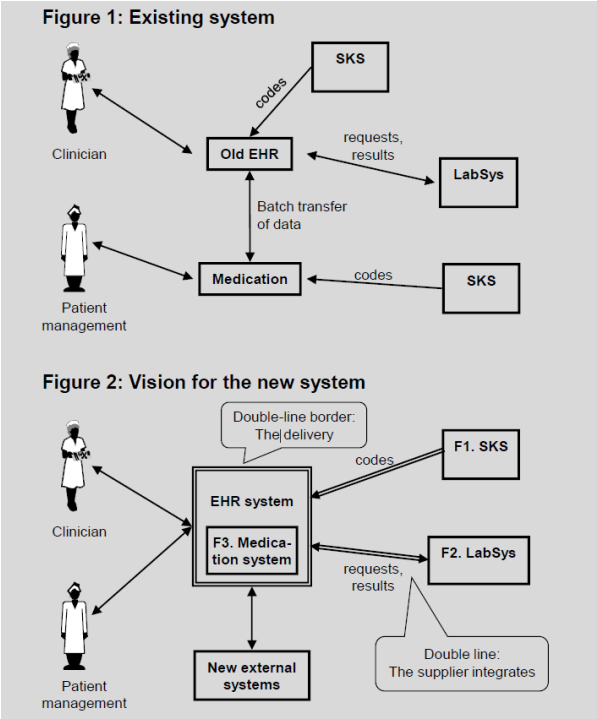
RE Group Project

THE NEWBIE'S GUIDE, MUKE

Project Background

- Describe the proposed project
  - What is the problem
  - Who has the problem
  - Why is it a problem (include how the problem is currently addressed)
  - How do you propose to resolve it
  - Where is this applicable

Diagrammatic view



Context Diagram showing potential users and externa systems to be integrated

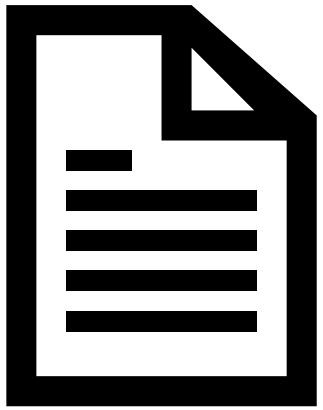
The double arrows show integrations you will specify requirements for.

Soren Lauesen, 2017

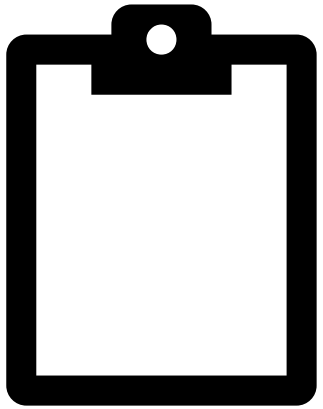
Example Business Goals (for a mobile app for a property disposal company)

ID	Description	Related Requirements
BG1	To reduce property waste	FR1, FR5
BG2	To increase efficiency of sales Staff	FR3, FR9, QR1

Your Task

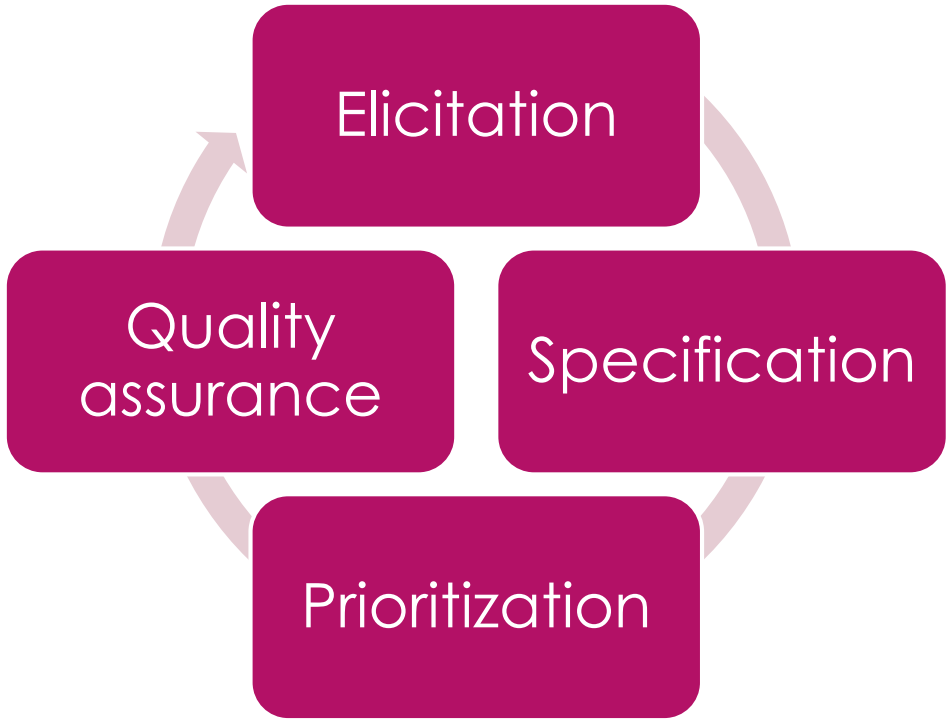


Write Requirements Specification for your project = Specification Document

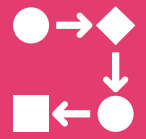


Write about your experience and decisions = Experience Report

Process


Elicitation
Specification
Prioritization
Quality assurance

Elicitation



Pick techniques that fit your project (see Elicitation lecture).

Discuss your decisions in the experience report.

Ensure that you provide more evidence for some of your techniques e.g.,

- if you conducted a survey or interview,
- how many and who were respondents,
- give highlights of questions asked; attach interview or survey form (as appendix)

Requirements Document

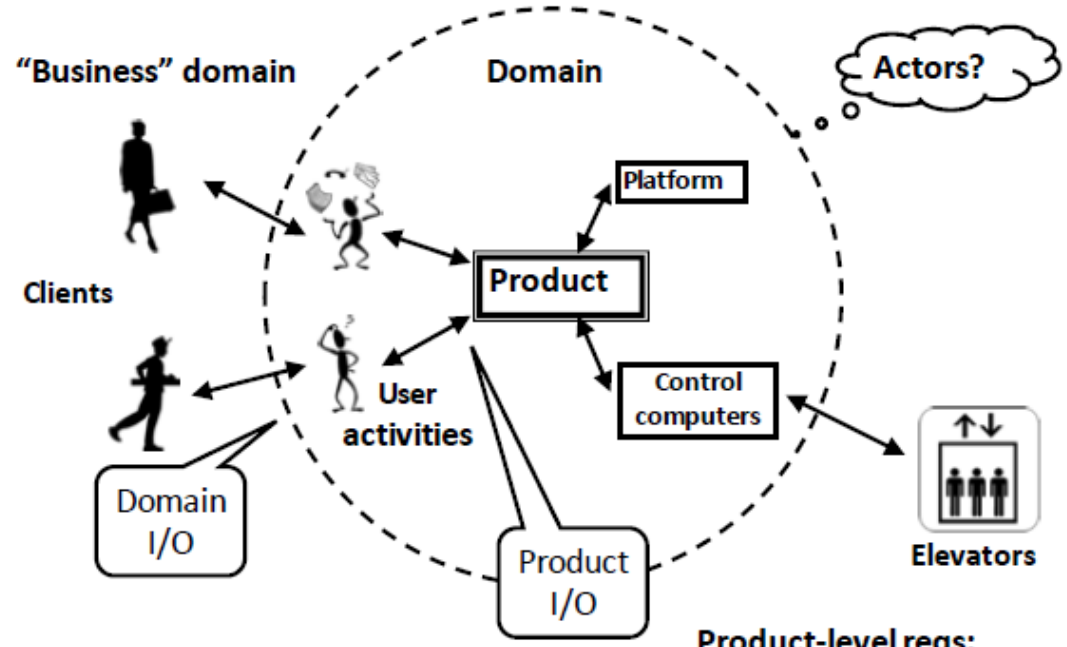
SEE TEMPLATE ON GITLAB

Scopes

What is Requirements Engineering?

- Business case (or: goal-level) requirements
- User (or: domain-level) requirements
- System (or: product-level) requirements
- Design-level requirements

Fig 1.5A Domain and product level



Domain-level req:
The product shall support the following user activities: . . .

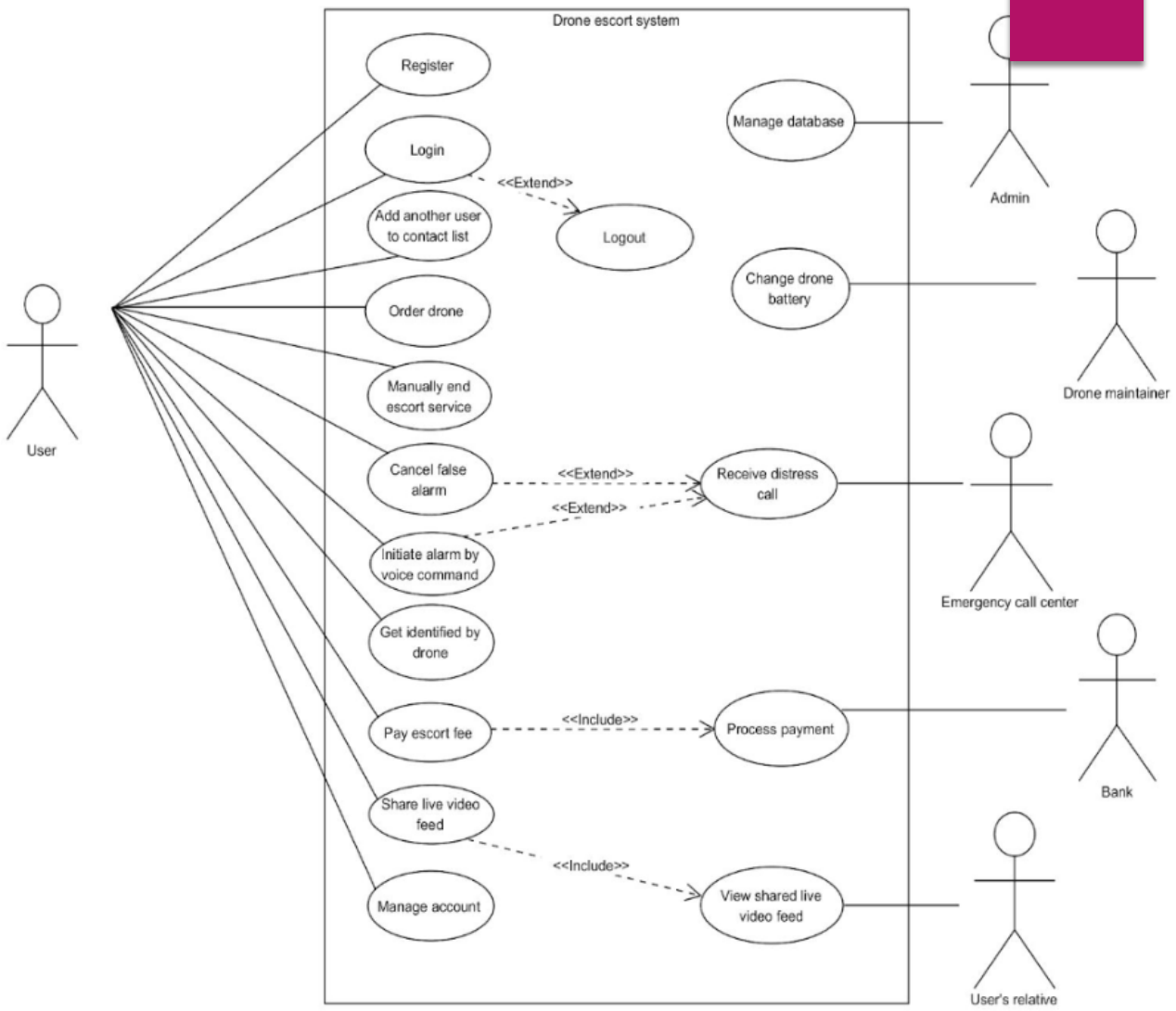
Product-level reqs:
The product shall accept the following input: . . .

From: Soren Lauesen: Software Requirements
© Pearson / Addison-Wesley 2002

[Lauesen, 2002]

High-level view of Core Requirements

- Use case diagram of your core functionalities



Example from Group 6, 2019

Add a stakeholder map

- Stakeholder is anyone who has interest or is affected by the proposed system
- Remember that your context diagram only shows USERS of the system
- Your stakeholder map must show what level of interest or effect each stakeholder has on the system

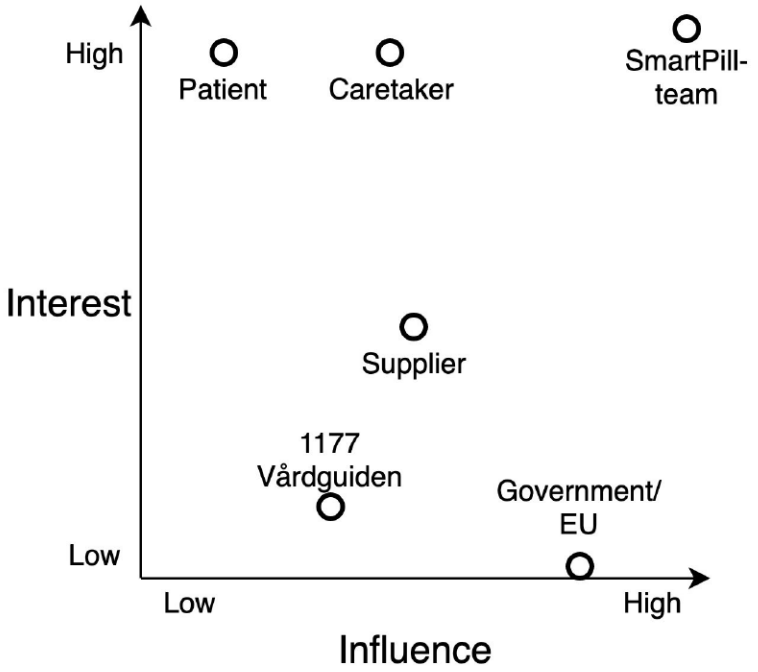


Fig X: Stakeholder map for the SmartPill Dispenser

Example from Group 11, 2019

Critical Quality Requirements

Consider your business case

What qualities are more critical?

Availability (uptime?)
Performance (speed?)
Usability?
Security?

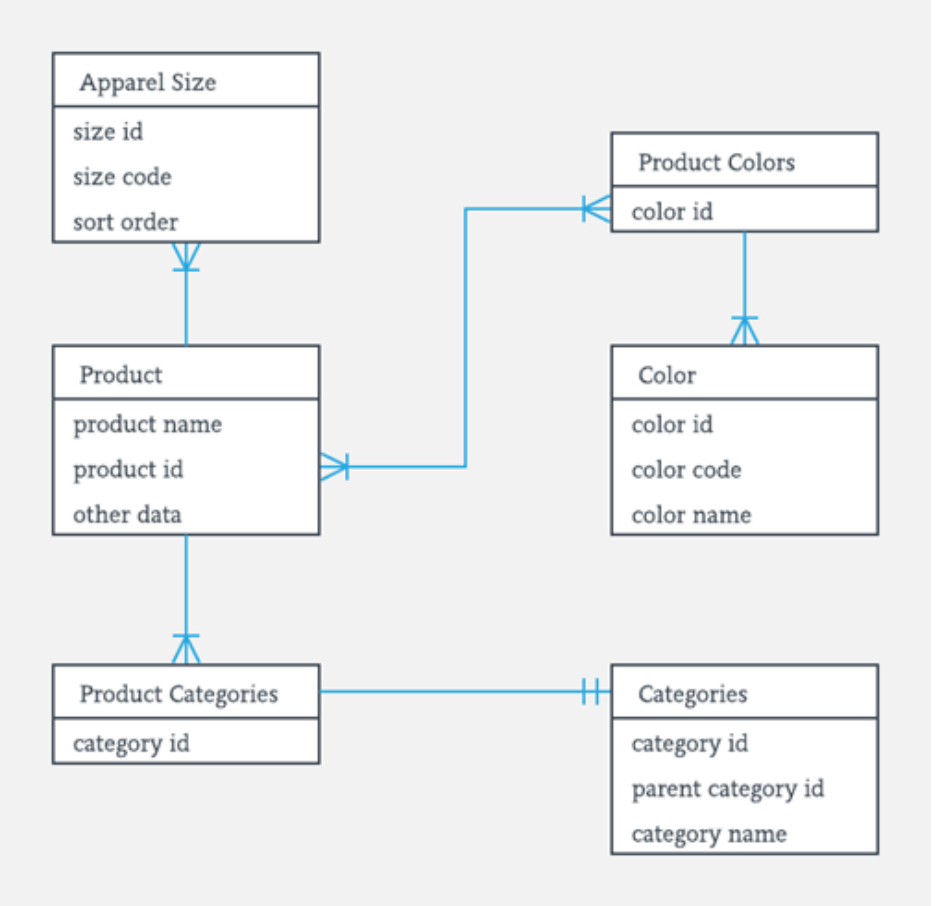
Later these should be described in detail with specifics e.g., planguage

Section 2: User Requirements

Whatever method you choose for specification, discuss them in the experience report; why did you choose them; what worked well and what didn't work well, etc. What could you have done differently

Describe your data

- Use ER Diagram, or class diagram
- plus detailed description of data using a data dictionary



Detailed Functional Requirements

- Use-cases, task descriptions, or user stories
- Abide by the format-rules of each technique you choose
- Provide detail

C. Tasks to support

The system must support all user tasks in this chapter, including all subtasks and variants, ar
the problems. Column 1 of the tables describe what user and system will do together. Who d
depends on the chosen solution.

A task is carried out from start to end without essential interruptions. If necessary, the case must be parked and resumed later. Although subtasks are numbered, they don't have to be carried out in this sequence, and many of them are optional. The user decides what to do and in which sequence. A subtask may also be repeated during the same task.

Some subtasks may be performed in alternative ways. It is shown with a, b, etc. Letters p, q, etc. indicate something that today is a problem with this subtask.

Work area 1: Patient management

This work area comprises calling in patients, monitoring waiting lists ...

User profile: Doctor's secretaries. Most of them are experienced IT users with good domain knowledge. They communicate easily with medical staff.
User profile: Clerical staff ...
Environment: Office ...

C1. Admit patient before arrival

This task creates an admission or continues a parked admission. Most admissions can be handled as one piece of work. The rest have to be parked, e.g. because some information is missing. It is important that the system ensure that parked admissions are not forgotten (see E1-1)

Start: Message from medical practitioner, message from another hospital ... message with missing data, or a reminder about a parked admission.
End: When the patient has been admitted or recorded on the waiting list, or when the admission has been parked while the missing data is on its way.
Frequency: In total: Around 600 admissions per day. Per user: A maximum of 40 per day.
Difficult: (never)
Users: Initially a doctor's secretary, but the case may be transferred to someone else.

Subtasks and variants:	Example solutions:	Code:
1. Record the patient. (See data description D5).		
1a. The patient is in the system. Update data.		
2. Admit also a healthy companion.		
3. Record the admission, including the initial diagnosis. (See data description D1 and D6).		
3a. Transfer data from medical practitioner, etc.	The system uses the MedCom formats.	
3p. Problem: Some electronic messages don't follow the MedCom format.	The system allows manual editing of the transferred message.	
3q. Problem: The patient may have several admissions at the same time at different hospitals and departments. It is hard to see who is responsible for nursing and where the bed is.		
4. Find a meeting time for the patient and send an admission letter or a confidential email.		
4a. Put the patient on the waiting list.		
4b. Essential data is missing. Park the case with time monitoring.		
4c. Transfer the case to someone else, possibly with time monitoring.		
4d. Maybe reject the case.		
5. Request an interpreter for the meeting time.		

Prioritization

- Provide prioritized list of requirements and group them into releases (1,2, and future)
- Discuss your methods for prioritization e.g., $100 method or which ever

Section 3: System Requirements

DISCUSS INTEGRATION REQUIREMENTS

UI PROTOTYPE

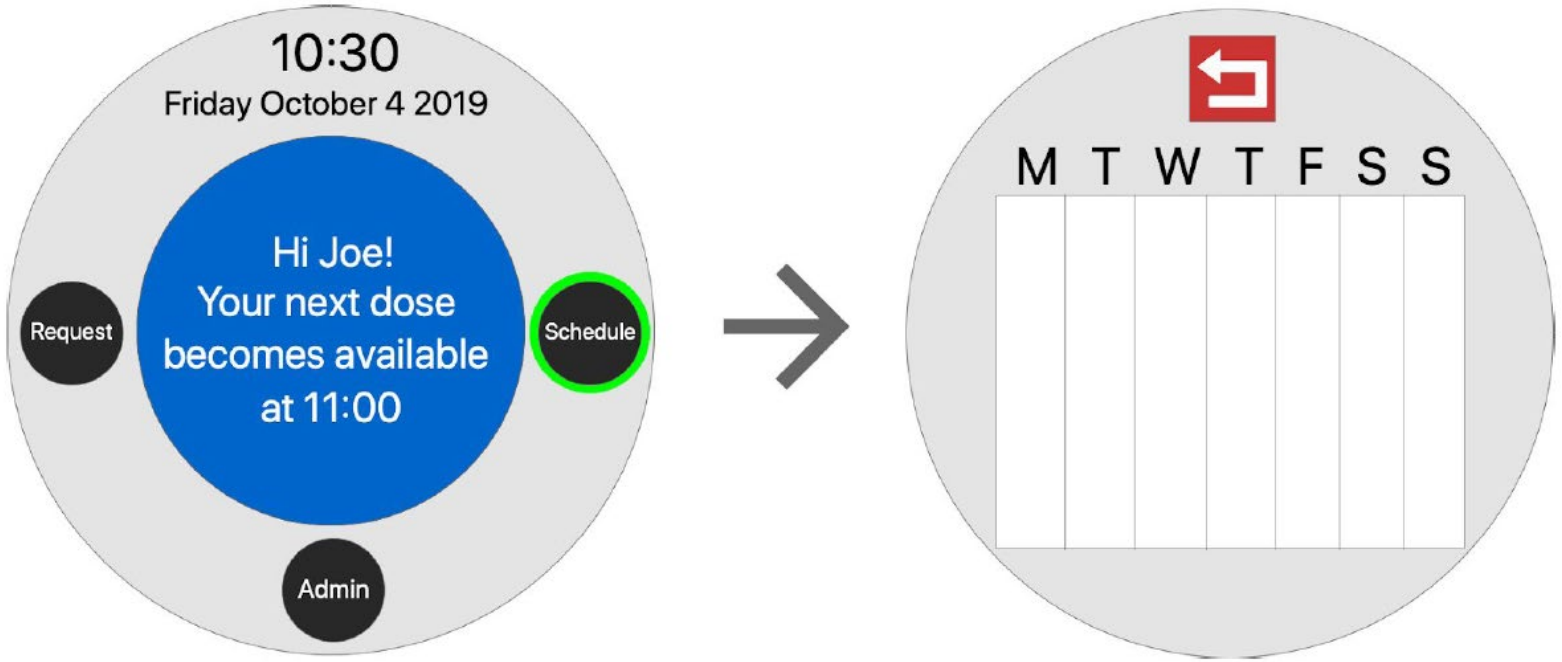
CONSTRAINTS

Example requirement for system generated events in a diabetic patients wearable app (Group 11, 2019)

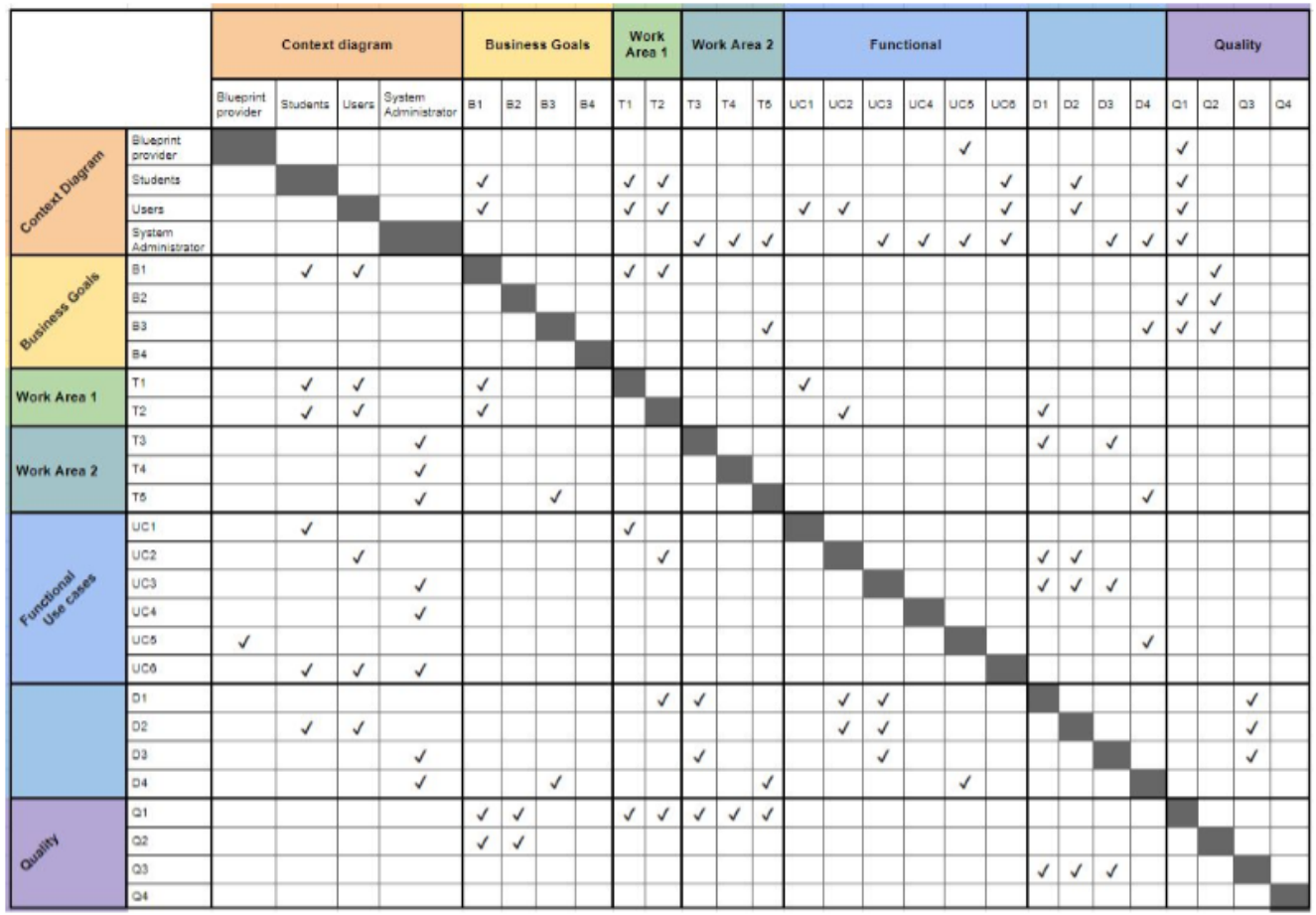
E1. System generated events

The system must generate these reminders:	Example solutions:	Code:
1. The device alerts the patient when it is time to take their medication (in accordance with task C1). If the patient does not interact with the device within a certain time interval, the alert is repeated.	The system makes alert alarm through the embedded loudspeakers or if the patient is away from home, it alerts through mobile phone by sending out recorded phone calls or messages.	

Example UI Prototype



Provide Traceability matrix for your requirements



		Context diagram				Business Goals				Work Area 1		Work Area 2			Functional									Quality			
		Blueprint provider	Students	Users	System Administrator	B1	B2	B3	B4	T1	T2	T3	T4	T5	UC1	UC2	UC3	UC4	UC5	UC6	D1	D2	D3	D4	Q1	Q2	Q3	Q4
Context Diagram	Blueprint provider																		✓						✓			
	Students					✓				✓	✓									✓		✓			✓			
	Users					✓				✓	✓				✓	✓				✓		✓			✓			
	System Administrator											✓	✓	✓			✓	✓	✓	✓			✓	✓	✓			
Business Goals	B1		✓	✓						✓	✓															✓		
	B2																								✓	✓		
	B3													✓										✓	✓	✓		
	B4																											
Work Area 1	T1		✓	✓		✓									✓													
	T2		✓	✓		✓										✓					✓							
Work Area 2	T3				✓																✓		✓					
	T4				✓																							
	T5				✓			✓																✓				
Functional Use cases	UC1		✓							✓																		
	UC2			✓							✓										✓	✓						
	UC3				✓																✓	✓	✓					
	UC4				✓																							
	UC5	✓																						✓				
	UC6		✓	✓	✓																							
	D1										✓	✓				✓	✓										✓	
	D2		✓	✓												✓	✓										✓	
	D3				✓							✓					✓										✓	
	D4				✓			✓						✓					✓									
Quality	Q1					✓	✓			✓	✓	✓	✓	✓														
	Q2					✓	✓																					
	Q3																				✓	✓	✓					
	Q4																											

Example from Group 8, 2019

Requirements specification (R1, R2, R3)

Releases and deliverables

Content	Relevant lectures										Relevant releases		
	L1 Intro	L2 Eli	L3 Doc	L4 Int	L5 Neg	L6 Cre	L7 Tra	L8 VV	L9 Agl	L0	R1	R2	R3
High-level description													
– Goal and scope	●	◐	○	◐	◐	○	○	○	○	○	●	◐	○
– Business case and stakeholder map	○	●	◐	◐	◐	◐	○	○	○	○	●	●	○
– Core functionality	○	◐	◐	●	◐	●	○	◐	○	○	●	●	◐
– Critical attributes	○	○	○	●	◐	●	◐	◐	○	○	●	●	◐
Operational description													
– Functional requirements	○	○	●	●	●	○	◐	○	○	○	◐	●	◐
– Data requirements	○	○	○	●	○	○	◐	◐	○	○	○	●	○
– Proposed priorities (2-3 releases)	○	○	○	○	●	○	○	○	○	○	○	●	◐
System/product requirements													
– UI prototype	○	○	○	◐	○	○	○	○	○	○	○	○	◐
– Technical requirements	○	◐	○	○	○	○	○	○	○	○	○	◐	●
– Detailed data requirements	○	○	◐	○	○	○	○	○	○	○	○	◐	◐
– Acceptance tests	○	○	○	○	○	○	○	◐	○	○	○	◐	◐
Cross-cutting													
– Traceability	○	○	○	○	○	○	●	○	○	○	○	●	●
– Rationale and appropriate detail	○	◐	○	○	○	◐	○	○	◐	○	○	●	◐
– Requirements quality	○	○	○	○	○	○	○	●	○	○	○	◐	◐

Experience Report

General Structure

- IEEE template
- Brief Background of the project
- For each RE activity (elicitation, specification, prioritization, and quality assessment of the RE document), describe
  - What techniques you applied
  - Why did you apply them considering your project context
  - What worked well with them and what didn't
  - What would you do differently
  - Give details on some aspects, e.g., if one elicitation technique yielded more requirements than another, make that evident. E.g., state whether **interviews** with customers added extra requirements to what you may have already **brainstormed** as a group.
  - Do not forget to add section where you discuss overall group dynamics. How you worked as a group; planning, combination, commitment, etc. Think of your team agreement.

Quality Assurance

You will be given another group's specification and experience report

You will be required to read and assess the quality of the requirements presented (see lecture on reviewing guidelines)

E.g., checking ambiguity, missing links between requirements and other aspects

You will write a separate quality assurance report and write about your techniques for quality assurance in the experience report

Always review criteria

Criteria 1/5 (Specification)

Project assessment

Assessment criteria	*Prerequisites for project grade 3*	*Additional prerequisites for project grade 4*	*Additional prerequisites for project grade 5*
(A) **Specification** (content; *checked in Requirements Document*)	3A) Demonstrate acceptable ability to apply several suitable *specification techniques* (e.g., task descriptions, feature requirements, planguage), and several *types of requirements* (in particular: quality, function, data), at several *abstraction levels* (e.g., goal, domain, product, design).	4A) Demonstrate advanced ability to adequately combine different *specification techniques*, *requirements types*, and *abstraction levels* with appropriate degrees of completeness tailored to the context.	5A) Demonstrate excellent ability to combine *specification techniques* in a motivated trade-off among qualities (e.g., completeness, unambiguity, verifiability) in relation to the benefit of the effort.
(B) **Specification** (scope; *checked in Requirements Document*)	3B) Demonstrate acceptable ability to define a *system's boundaries* and its interaction with external entities.	4B) Several requirements have explicit *rationale* that reduces risks of misinterpretation.	5B) Requirements explicitly balance multiple stakeholders' *priorities*.
C) **Specification** (reflection; *checked in Experience Report*)	3C) Demonstrate acceptable ability to *reflect* on specification experiences.	4C) Demonstrate advanced ability to *critically reflect* on specification trade-offs.	
(D) **Specification** (dependencies; *checked in Requirements Document*)		4D) A relevant subset of requirements has explicit *inter-dependency links* that create a consistent and coherent requirements model.	5D) A selected set of important *inter-dependency links* among requirements are managed as requirements evolve, based on a cost-benefit discussion.

Individual Contribution Assessment

Fill in	Fill in the online form at every release
Rate	Rate everyone including yourself
Indicate	Indicate any problems by adding extra comments