



W1 – The Project

DAT232/DIT285 Advanced Requirements Engineering

Eric Knauss

eric.knauss@cse.gu.se



September 9, 2025



① Organization

House Keeping

Learning Objectives

② The project

Tasks and roles

Releases and deliverables

Project assessment

③ Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

④ Wrapping up

Outline



House Keeping

Project status

- Groups have been randomly assigned based on your preferences
- Late registrations have been added to groups
- Make sure you make contact with your supervisor
 - **Group 0-3:** Hina Saeeda
 - **Group 4-7:** Nayat Astaiza Soriano
 - **Group 8-11:** Osama Ateeq
 - **Group 12-15:** Suvrangshu Barua
 - **Group 16-19:** Lilia Beniaminova
 - **Group 19-23:** Jakob Lund Persson
 - **Group 24-27:** Lin Qiqi



Learning Objectives



Knowledge



Skills



Judgement

Identify a common RE challenge in a given software development context.

Choose an appropriate RE practice in a given software development context.

Compare suitability as well as advantages and disadvantages of given RE practices in a given software development context.

Explain the current state of practice and research in requirements engineering.

Plan suitable RE practices in a team with respect to a given software development context.

Effectively apply a suitable RE practice in a team in a given software development context.

Analyze the effect and quality of the outcome of a set of or individual RE practices in a given software development context.

Assess new requirements engineering knowledge (challenge, principle, practice) and relate them to the framework in this course.

Suggest suitable actions to overcome a lack of requirements knowledge in a software development context.

Consider inter-team, program level and social/ethical implications of a set of RE practices in a given software development context.

Critically assess the effectiveness of a set of RE practices from the perspective of the student's master program (e.g. Software Engineering & Technology/Management, Interaction Design, Game Design, Data Science, ...)



① Organization

House Keeping

Learning Objectives

② The project

Tasks and roles

Releases and deliverables

Project assessment

③ Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

④ Wrapping up

Outline



First steps

The project starts now!

- Groups randomly assigned after first lecture
- Meet in your group and align on project mission
- Contact your supervisor in case of any questions



Project objectives

- Connect theory to practice
- Provide concrete experience of practical RE
- Have real stakeholders
- Provide realistic problems

Context and Roles

Three main tasks:

- Write a specification from customer perspective in 3 iterations
- Reflect on experiences during these 3 iterations
- Review specifications from other groups

Project team:

- Randomly assigned

The project

General rules

- 80h per person
- Mainly customer work, but also peer-reviews of other groups' specifications
- Meet your group supervisor: As often as needed, at least in Course Week 2, 4, and 6



Tasks

As a customer:

- Align on project mission
- Do real elicitation, i.e. go out and interview potential users
- Write a specification in 3 iterations

As a reviewer:

- Plan the review of another group's specification within your group
- Review specification(s)
- Integrate findings into a joint review report

As a student:

- Engage in effective group work
- Reflect on experiences
- Document and present experiences

Releases and deliverables

Organization

House Keeping

Learning Objectives

The project

Tasks and roles

Releases and deliverables

Project assessment

Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

Wrapping up



UNIVERSITY OF
GOTHENBURG



CHALMERS

UNIVERSITY OF TECHNOLOGY

- 3 iterative releases
- Each release > previous release
- Each release:
 - Requirements Document
 - Project Experience Report
 - Team agreement
 - Individual Contribution Assessment (per group member)
- For R2, R3:
 - Change history
 - Indication of how grading criteria was addressed



Project deadlines

Releases and deliverables

Phase	Deliverable	Deadline
Definition	Project mission	Week 1: Friday 1pm
Iteration 1	Release R1	Week 4: Monday 8am
	Review R1	Week 4: Friday 4pm
Iteration 2	Release R2	Week 6: Monday 8am
	Review R2	Week 6: Friday 8am
Iteration 3	Conference presentation	Week 8: Monday 4pm
	Release R3	Week 9: Monday 8am

Each deliverable: In PDF via Canvas
(Conference Presentation also as PPT/PPTX)



Project mission

Releases and deliverables

A “Good” Project mission

- You have a deep knowledge of the domain
- You have a genuine interest in the system
- It is feasible to collect requirements from different sources. Examples:
 - Interview potential users at Nordstan,
 - Read app reviews of related apps,
 - Analyze advertisements of potential competitors,
 - Send surveys, ...
→ Elicitation Lecture
- The mission has a business case
- Ideally, the mission has interesting possibilities

Requirements specification (R1, R2, R3)

Releases and deliverables



Content	Relevant lectures										Relevant releases		
	L1 Intro	L2 Eli	L3 Doc	L4 Int	L5 Cre	L6 Neg	L7 Tra	L8 VV	L9 Agl	L0 Eth	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	○	○	○	○	○	○	●	●	○	●	○	●	●



Project experience (R1, R2, R3)

Releases and deliverables

- Experiences and reflections in relation to learning objectives.
- Description of:
 - Chosen methods for elicitation, specification, validation, prioritization (What and How?)
 - Motivation of chosen methods/techniques (Why?)
- Example questions:
 - What have you learned?
 - What would you do differently?
 - What did work well/did not work well?
- No evaluation of the course/lectures.
- Format: IEEE Conference Proceedings, max. 8 pages



Individual Contribution (R1, R2, R3)

Releases and deliverables

- One hand-in per group member
- Assessment of each others' contributions
- Optional explanations
- Other group members cannot see your hand-in
- Format: Microsoft Forms Survey

If needed, we will take action based on the contributions.



Review (Weeks 4 and 6)

Releases and deliverables

Reviewing group:

- Reviews specification (R1: ad hoc, R2 according to guidelines in lectures),
- Gives feedback (with clear criticality),
- Asks for clarification

Customer group:

- Improves specification



Conference Presentation (Week 7)

Releases and deliverables

Prepare and rehearse a short presentation (customer group perspective)

- 2 minutes on project mission
- 3 minutes on project results including used techniques
- 3 minutes on important experiences and learning outcomes



Project assessment

R1, R2, Project Mission, Conference Presentation, Review:

- Pass/Fail

R3 **and** Validation Report from reviewing R2:

- U/3/4/5

Check the grading criteria **regularly**

- General, high-level feedback in the supervision



Writing style

Project assessment

- We will look at the general quality of your reports!
- Captions to all tables and figures.
- All tables and figures referenced in text.
- Table of content where reasonable.
- Paragraphs = *self-contained unit of a discourse*.
- Use of defined format for Experience Report.
- Absolutely **no plagiarism** accepted → Academic integrity module



① Organization

House Keeping

Learning Objectives

② The project

Tasks and roles

Releases and deliverables

Project assessment

③ Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

④ Wrapping up

Outline



Organize your team

Best Practices from your Teachers

- Divide tasks, but collaborate to achieve overall consistency.
- If assigned to a task, take responsibility:
 - Ensure that task is covered in Requirements Document and Experience Report.
 - Keep other team members informed about status and touch points.
 - Ask for input from other team members
 - Ensure that task is covered in cross-cutting activities (e.g. Requirements Elicitation)



Organize your work

Best Practices from your Teachers

- Only R3 will be graded. In R1 and R2, you will receive valuable feedback.
 - As long as you submit something, we will usually set the assignment as passed.
 - No mandatory content for Release 1.
 - Aim for a good draft of Chapter 1 in the suggested template.
 - Develop initial ideas about content in Section 2.
- Do not hesitate to adjust the template to your needs.
- Consider adding a table about how you aim to fulfil grading criteria.

A possible way to assign work for R1

Best Practices from your Teachers



Task area	Requirements Document	Experience Report	Team Coordination
1. Goal/Scope	Ensure that goal and scope are described as well as that a context diagram exists. Ensure that all parts (also from other task areas) fit together and are in scope.	Ensure that this task area is described, motivated, and reflected on.	Keep rest of team aware of scope, trigger discussions about consistency among task areas.
2. Business Case/Stakeholders and Elicitation	Describe Business Case and Stakeholders. Coordinate Elicitation activities.	Ensure that this task area is described, motivated, and reflected on.	Make sure that elicitation needs from other task areas are covered.
3. Core Functionality and Functional Requirements	Ensure that core functionality is described and that a use case diagram exists. Identify appropriate styles to describe core functionality (Section 1.3) as well as functional requirements (Section 2.2).	Ensure that this task area is described, motivated, and reflected on.	Enable team members that work in other task areas to check whether the described functionality is necessary and complete.
4. Critical Attributes	Ensure that relevant Quality Attributes and Constraints as well as a Quality Grid exist and are described (Section 1.4). Identify appropriate styles to describe for non functional requirements (Section 1.4 and 2.3).	Ensure that this task area is described, motivated, and reflected on.	Discuss how qualities and constraints relate to business case and core functionality.
5. Data requirements	Draft a conceptual model of data requirements based on Context Diagram, Use Case Diagram, and Core Functionality. Identify appropriate data requirements styles.	Ensure that this task area is described, motivated, and reflected on.	Explore whether data requirements are necessary and complete. Relate data requirements to quality attributes.

Advice taken from previous project conferences

Best Practices from previous students

- Weekly group meetings
- Weekly supervisor meetings (?!)
- Work meetings with the groups (not only coordination)
- Parallel and pair work, not only individual
- Traceability is useful, try to use early
- Context diagram not useful
- Context diagram very useful
- Personas can help if a stakeholder group is hard to reach
- Communication, alignment, different perspectives are key for this project
- Peer-reviews as source of inspiration
- Sequence-Diagram can be helpful, if you have a lot of technical actors (other systems)



Additional Best Practices

Based on Muke's guide to Newbies

- Here are a few suggestions on how to interpret and approach the assignments
- They complement the information about the project in the previous section

Your Task

Organization

House Keeping

Learning Objectives

The project

Tasks and roles

Releases and
deliverables

Project assessment

Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

Wrapping up



Write Requirements
Specification for your
project = Specification
Document



Write about your
experience and decisions =
Experience Report



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

Organization

House Keeping

Learning Objectives

The project

Tasks and roles

Releases and
deliverables

Project assessment

Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

Wrapping up



UNIVERSITY OF
GOTHENBURG



CHALMERS

UNIVERSITY OF TECHNOLOGY





Diagrammatic View

Figure 1: Existing system

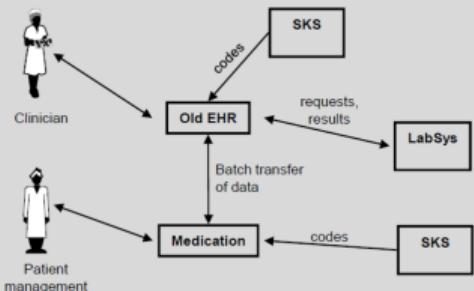
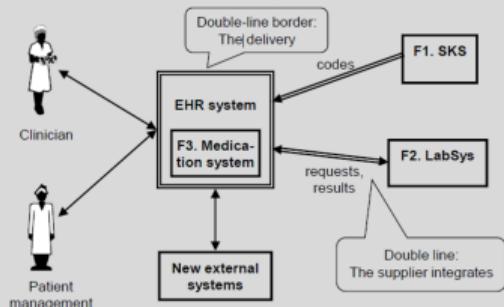


Figure 2: Vision for the new system



Context Diagram showing potential users and external systems to be integrated

The double arrows show integrations you will specify requirements for.

Example taken from [Lauesen, 2017]



Business cases

Example: a mobile app for a property disposal company

ID	Description	Related Requirements
BG1	To reduce property waste	FR1, FR15
BG2	To increase efficiency of sales staff	FR3, FR9, QR1

Good start, but think about how to make them measurable and formulate them as a goal. Why would you want to reduce property waste? Which value is generated?

Organization

House Keeping

Learning Objectives

The project

Tasks and roles

Releases and
deliverables

Project assessment

Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

Wrapping up



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

Elicitation

Organization

House Keeping

Learning Objectives

The project

Tasks and roles

Releases and
deliverables

Project assessment

Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

Wrapping up



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

- 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)

Organization

House Keeping

Learning Objectives

The project

Tasks and roles

Releases and
deliverables

Project assessment

Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

Wrapping up



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

Documentation

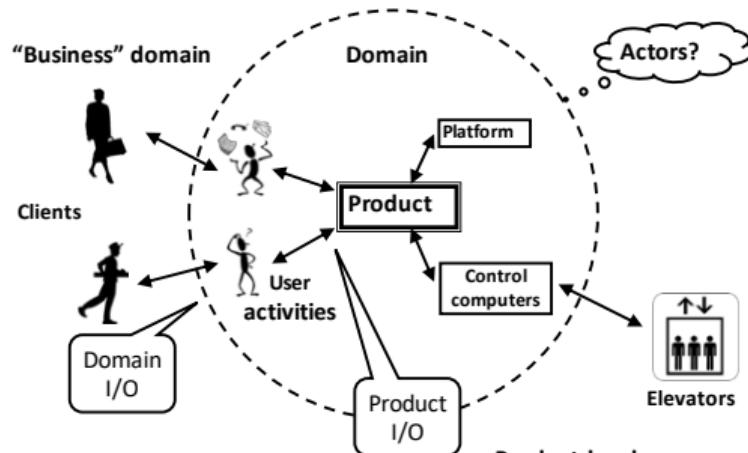


- Business case
(or: goal-level)
requirements
- User
(or: domain-level)
requirements
- System
(or: product-level)
requirements
- Design-level
requirements

Scopes

What is Requirements Engineering?

Fig 1.5A Domain and product level



Domain-level reqs:

The product shall support
the following user
activities: ...

Product-level reqs:

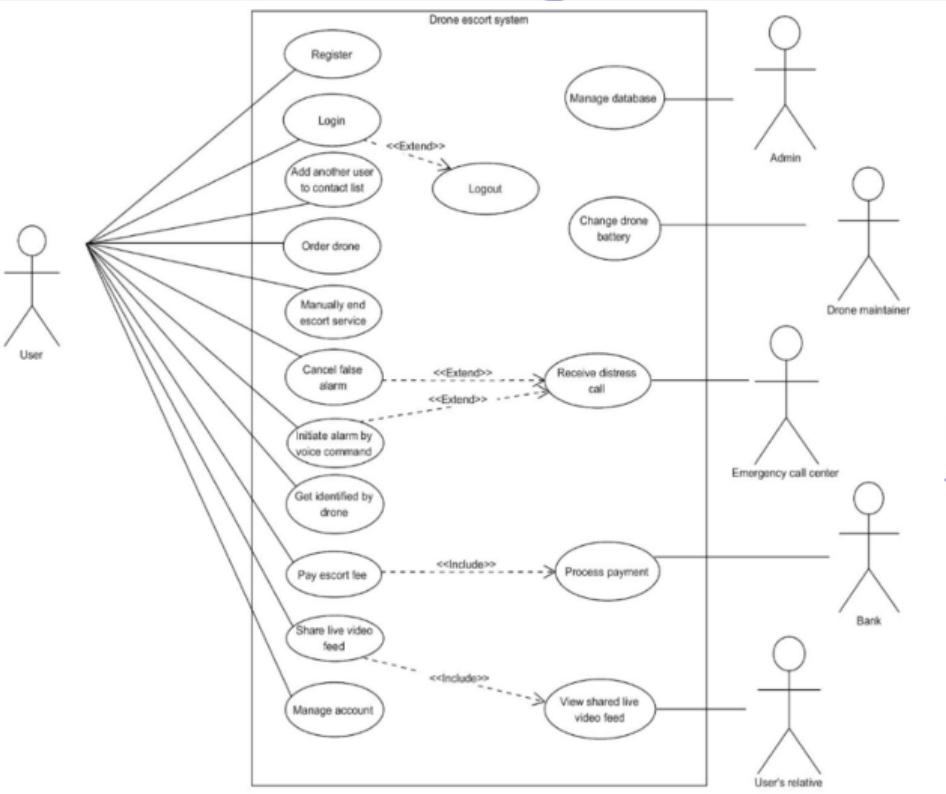
The product shall accept
the following input: ...

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



- 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

Stakeholder Map

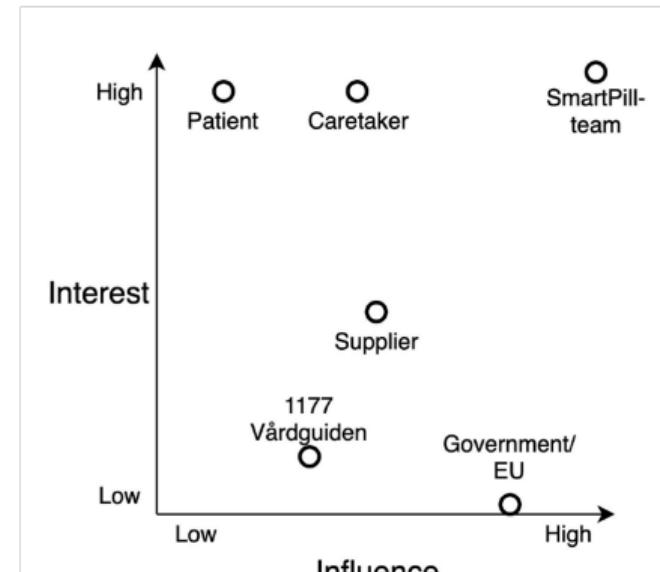


Fig X: Stakeholder map for the SmartPill Dispenser

Example from Group 11, 2019



Quality Requirements

① Consider your business case

② What qualities are more critical?

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

③ Later these should be described in detail with specifics e.g., planguage

Section 2: User Requirements

Organization

House Keeping

Learning Objectives

The project

Tasks and roles

Releases and
deliverables

Project assessment

Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

Wrapping up

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



UNIVERSITY OF
GOTHENBURG

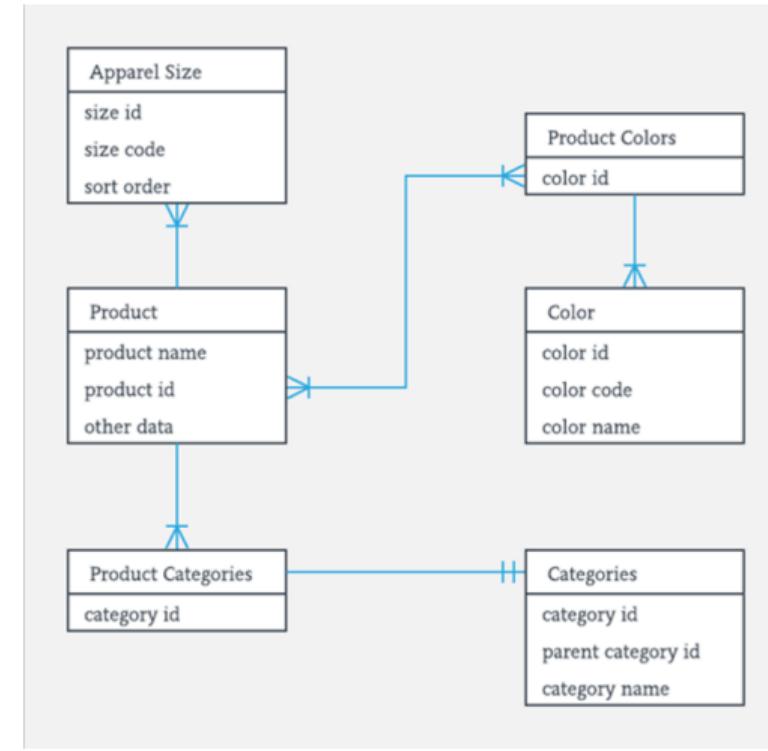


CHALMERS
UNIVERSITY OF TECHNOLOGY



Data Requirements

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





C. Tasks to support

The system must support all user tasks in this chapter, including all subtasks and variants, and mitigate the problems. Column 1 of the tables describe what user and system will do together. Who does what 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.
Difficulty: (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.		

Detailed Functional Requirements

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

Example taken from [Lauesen, 2017]



Prioritization

- Provide prioritized list of requirements and group them into releases (1,2, and future). **(Think about interdependencies and what a minimal viable product could be)**
- Discuss your methods for prioritization e.g., \$100 method or which ever **(Please think your prioritization through! Don't use MOSCOW blindly.)**

Section 3: System Requirements

Organization

House Keeping

Learning Objectives

The project

Tasks and roles

Releases and
deliverables

Project assessment

Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

Wrapping up

- Discuss Integration Requirements
- UI Prototype
- Constraints
- (Consider detailed data requirements here)



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

Example requirement for system generated events

Taken from Group 11, 2019: a diabetic patients wearable app

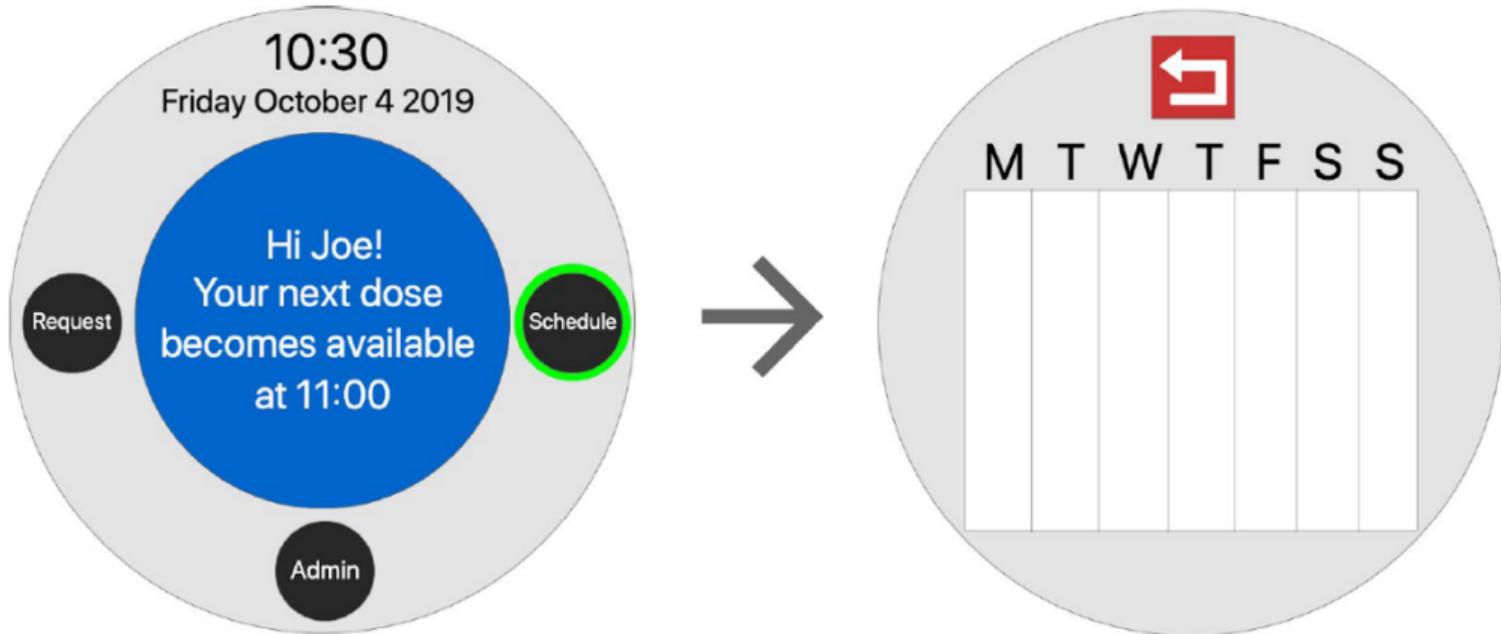
E1. System generated events

The system must generate these reminders:	Example solutions:	Code:
<ol style="list-style-type: none">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

Taken from Group 11, 2019: a diabetic patients wearable app





Example Traceability Matrix

Taken from Group 8, 2019

		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		✓	✓		✓																				✓	
		T3				✓																					✓	
Work Area 2		T4					✓																				✓	
		T5					✓																				✓	
Functional Use cases		UC1			✓																						✓	
		UC2			✓																						✓	
		UC3				✓																					✓	
		UC4					✓																				✓	
		UC5	✓																								✓	
		UC6		✓	✓	✓																					✓	
		D1									✓	✓				✓	✓										✓	
		D2			✓	✓												✓	✓								✓	
		D3					✓						✓					✓									✓	
		D4					✓							✓					✓								✓	
Quality		Q1						✓	✓			✓	✓		✓	✓	✓	✓										
		Q2						✓	✓																			
		Q3																										
		Q4																										



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 requirements document
- 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 (Review of R2 only)



Individual Assessment

Organization

House Keeping

Learning Objectives

The project

Tasks and roles

Releases and
deliverables

Project assessment

Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

Wrapping up



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

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



1 Organization

House Keeping

Learning Objectives

2 The project

Tasks and roles

Releases and deliverables

Project assessment

3 Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

4 Wrapping up

Outline



Wrapping up

Additional notes 1(2)

- How much should you write?
 - For PM, 1 page!
 - For Experience reports, stated in project description as MAX limit
 - All other deliverables, check for page limits (see project description)
- Plagiarism (= unattributed copying of material from other sources) will
 - Yield an immediate FAIL on course
 - Be reported to university disciplinary board
 - Mind *Academic integrity* modules on Canvas, prerequisite for take-home exam, but also helpful for project deliveries.
- Generative AI: Disclose how you used it (e.g. link to chat protocol)



Wrapping up

Additional notes 2(2)

- Student participation

- You are expected to take responsibility
- Lectures focus on the course literature, but will not cover everything. Additional information will also be covered during the lectures
 - Read book and research papers early and beforehand
- Active participation
- Discuss!!!
- Reflect and relate to experiences



Wrapping up

Todo

- Theory: Get familiar with course literature
 - Attend Elicitation lecture tomorrow
 - Attend Documentation lecture and ICC seminar on Tuesday
- Project: Get familiar with your group and the project
 - Set up work environment (ask me for git, if needed)
 - Brainstorm project mission
 - Submit one project mission per group
 - Meet in groups. Make it work. Start elicitation.
 - Book time via email with project supervisor W2, W4, W6
(Each group has a project supervisor)

Organization

House Keeping

Learning Objectives

The project

Tasks and roles

Releases and
deliverables

Project assessment

Best Practices

General advice

Elicitation

Documentation

Prioritization

System Requirements

Experience Report

Quality Assurance

Wrapping up



UNIVERSITY OF
GOTHENBURG



CHALMERS
UNIVERSITY OF TECHNOLOGY

References |



Lauesen, S. (2002).

Software Requirements.
Pearson / Addison-Wesley.
the course book (Lau).



Lauesen, S. (2017).

Guide to Requirements SL-07: Problem-oriented requirements v5.
Complements the course book (Lau) with hands on advice and a template.