Project Description ETSN15 Requirements Engineering

http://cs.lth.se/krav

Revision date: December 19, 2025

1 Objectives

The main goals of the project from a course perspective are to:

1. connect theory to practice,
2. give concrete experiences of requirements engineering challenges,
3. increase your motivation by group activities focused on realistic problems.

2 Assignment of project team and project mission

You should contribute to your project team formation and assignment accordingly:

1. The project members of each team is finalized by Monday of the first course week.
2. You should bring your ideas on a candidate Project Mission for your team to the lecture on Tuesday of the first course week, in a domain that you have knowledge and interest in.
3. You should participate in selecting and developing a draft of Project Mission v1 for your team before the exercise on Wednesday of the first course week.

3 Context and roles

Each project team shall perform requirements engineering work based on a Project Mission. The project team develops system models with requirements of different types at appropriate levels, based on their interpretation of the Project Mission. Your team members act a primary stakeholder in addition to other (potential) external stakeholders. The actual scope of your project is negotiated with a teacher in the course.

The project team consists of at least 4 and not more than 8 members, preferably 5–6 members. You should appoint the roles below among your team members (one person can have several roles, and some roles can be shared by two persons, depending on the number of team members):

- **PM** Project, Process, Prioritization & Release Manager (1 person)
- **SM** Stakeholder Manager (1 person)
- **TM** Tools, Documents, Experiences & Version Manager (1-2 person)
- **EM** Elicitation & Prototyping Manager (1-2 person)
- **QM** Quality Requirements Manager (1 person)
- **DM** Data Requirements Manager (1 person)
- **VM** Validation Manager (1 person)

4 General project rules

1. The project comprises 80 hours per person.
2. The total effort should be evenly distributed among participants.
3. In weeks W2, W4, and W7 a meeting should be scheduled with the project supervisor, where the project team reports on status, challenges and plans. All project members should participate in meetings with the project supervisor.
4. The manager roles imply management, planning, and coordination responsibilities, but managers should not do all the work: *all team members should contribute to all parts of the project work!*.

5 Project deliverables

Phase	*Deliverables*	*Deadline*
Planning	Project Mission v1	Week 1: Wednesday 10:15
	Project Mission v2	Week 2: Tuesday 23:59
Iteration 1	Release R1	Week 3: Sunday 23:59
Iteration 2	Release R2	Week 5: Sunday 23:59
	Validation Checklist	Week 5: Sunday 23:59
	Validation Report	Week 6: Thursday 23:59
Iteration 3	Conference Presentation	Week 7: Monday 08:00
	Discussant Questions	Week 7: Wednesday 15:00
	Release R3	Week 7: Sunday 23:59

All deliverables should have a title, version number, team id, product name, and names of all team members.

5.1 Project Mission v1

You should consolidate your ideas on software-based products and select one of them and create a first sketch of an initial Project Mission v1 including at least:

1. A product name, different from all other product names in the course.
2. A short, high-level product description of its background and purpose.
3. A brief description of some different, important user roles.
4. A brief description of some external system with which your product interacts.

Bring your PM v1 to the first exercise, where you will work on refining your product's scope and context, etc.

5.2 Project Mission v2

Your team should prepare a second version of the Project Mission where the scope of the project is further refined. The purpose of version 2 of the Project Mission is to act as an agreement that specifies what your team intends to develop.

1. The Project Mission v2 is recommended to include the following information:
   (a) Table of contents.
   (b) Product name, background, general purpose and main goals.
   (c) Product context diagram: user roles, external systems, interfaces, etc
   (d) Participants and potential external stakeholders.
   (e) Planned activities with start/stop dates for achieving each deliverable in Section 5. Use a diagram or table to show the planned activities and estimated hours spent per week per participant.
   (f) Responsibilities of project members.
2. Check that the following questions can be answered by PM v2:
   (a) What is the project about?
   (b) Who is participating in the project as members and as input providers?
   (c) What should be done in the project?
   (d) When should the results be delivered?
   (e) Who is responsible for what?
   (f) When shall whom work with what?

5.3 Work approach, releases and deliverables

You should work iteratively and divide your work into 3 main iterations, each ending with a release with all your accumulated work products. (You may have more sub-iterations with additional internal releases.) The releases (delivered for the course) are denoted R1, R2, and R3. You should follow these instructions and guidelines:

1. For each release, the quality of your deliverables should represent a noticeable improvement/refinement.
2. Each release `Rn` of team `X` should be delivered in *one single, self-contained* **zip-file** named `X-Rn.zip` with everything delivered.
3. There should be an **overview description** of each release to make navigation and assessment easy, e.g. in a file called `index.html` or `README.txt` or similar.
4. Each release should be divided into two explicit parts: **System Requirements** and **Project Experiences**, each with its own **table of contents**.
5. Release R2 should include a draft **Release Plan** as part of the System Requirements (see 6e below). The Release Plan should be further refined in R3.
6. Release R3 should include final versions of all deliverables, including at least: System Requirements with Release Plan, Project Experiences, Validation Report, Validation Checklist, and Conference Presentation.

   System Requirements include the following:
   (a) *Different types* of system requirements, e.g. **data**, **function**, **quality** at different levels, e.g. **goal**, **domain**, **product**, **design**.
   (b) *Several* specification techniques, e.g. **context diagrams**, **features**, **virtual windows**, **task descriptions**.
   (c) Each requirement should have a **unique identity** (name or number).
   (d) A **Release Plan** should be part of R2 & R3, in which a subset of the requirements are prioritized and planned to be implemented in three imagined future product releases called R4, R5, R6.
   (e) *Design-level requirements* should be specified for the sub-set of requirements that are planned for a future imagined product release R4 (see 6e). This sub-set of requirements shall be implemented as mock-up designs in R3 using, e.g. **screens and prototypes**, **analog drawings**, **clickable presentations**, **executable GUI mockups** or similar.

   Project Experiences include the following:
   (a) Description of your requirements engineering work, including experiences and reflections in relation to learning objectives.
   (b) Description of the chosen methods/techniques for elicitation, specification, validation, and prioritization.
   (c) Motivation for *why* you chose the used methods/techniques.
   (d) Reflection on the usage of these methods/techniques in terms of what was successful and what was challenging. Example questions for reflection: What have you learned in relation to the learning objectives in this course program? What would you have done differently based on what you know now? What have you learned in relation to the learning objectives?

(e) Reflection on the interaction among internal and (potential) external stakeholders through the different steps of the Requirements Engineering process.
(f) A personal statement by each team member that briefly explains each individual's contributions to the project results (*what*) and the total number of hours of project work per person (*how much*).
(g) The Project Experiences should *not* include course evaluation issues, but focus on your own work and learning outcome.

Validation Report To gain experience and input to your own project, you will validate another project team's release R2 and hand in your validation report together with your team's R3. Your team should produce relevant and useful issues for improvement. Each issue should be ranked for criticality.

Validation Checklist To help another project team to validate your release R2, you will provide them with a requirements validation checklist tailored to the context.

Conference Presentation Prepare and rehearse a short presentation.

(a) The total presentation time and further guidelines are given during the course.
(b) Spend approx. 10% of the presentation time on the project's mission.
(c) Spend approx. 45% of the time on project results and techniques used.
(d) Spend approx. 45% of the time on experiences and learning outcome.
(e) Slides should be in {.ppt|.pptx|.pdf}.

Discussant Questions Prepare questions and act as discussants at the project conference for the project that you previously validated.

(a) The total discussant time and further guideliens are given during the course.
(b) Prepare questions on, for example, choice of RE techniques, experienced RE challenges and solutions during the project.

Course Evaluation (Not part of the assessment, optional.) A separate, free-form Course Evaluation document can be handed in by the team. If team members have different views, it is valuable if these differences are reflected. For each relevant course element (lectures, exercises, labs, project etc) answer questions such as: What worked well? If something needs improvement, *why* and *how* would you like it to be changed? Send the Course Evaluation to `bjorn.regnell@cs.lth.se`

6 Project assessment

1. The deliverables Project Mission and Conference Presentation is pass/fail only.
2. The project grade of fail/3/4/5 is based on Release 3 including Validation Report and Validation Checklist according to the criteria in the table on the **next page**.

Project Grading Grid

Assessment area	*Required for project grade* **3: Accepable** Demonstrate ability to ...	*Also required for project grade* **4: Good** Demonstrate ability to ...	*Also required for project grade* **5: Excellent** Demonstrate ability to ...
Specification	**3A)** apply more than one suitable specification technique (e.g. task descriptions and screen prototypes), and more than two types of requirement (e.g. data, function, quality), and more than three abstraction levels (e.g. goal, domain, product, design). **3B)** define a system's boundaries and its interaction with external entities. **3C)** reflect on specification experiences and reason about choices of specification methods in relation to different contexts.	**4A)** combine different degrees of completeness and different levels of abstraction. **4B)** use at least four different specification techniques adequately tailored to the context. **4C)** provide explicit requirements rationale that reduce risks of misinterpretation. **4D)** use hierarchies and requirements relations to manage evolving requirements structures.	**5A)** combine specification techniques in an explicitly motivated trade-off between qualities and costs, where a high degree of specification completeness is achieved for a carefully selected subset of requirements. **5B)** provide motivated estimations of target quality levels using well-defined scales.
Elicitation	**3D)** apply more than one elicitation technique in a relevant way. **3E)** reflect on elicitation experiences. **3F)** perform stakeholder analysis.	**4E)** reason about the need for further elicitation in relation to specification quality. **4F)** demonstrate good use of prototyping to elicit realistic user requirements. **4G)** interaction with at least one external stakeholder.	**5C)** elicit creative ideas and deep domain knowledge in realistic contexts, while interacting with several types of real-world stakeholders.
Validation	**3G)** assess the quality of requirements and find relevant problems of several different types. **3H)** apply more than one validation technique including prototyping. **3I)** reflect on validation experiences.	**4H)** find, prioritize and discuss requirements quality problems of different types, while reaching beyond form issues. **4I)** adapt the validation to the context and provide rationale for the chosen validation techniques.	**5D)** reason about the relation between requirements quality problems and risks from different stakeholder's viewpoint. **5E)** utilize links among different types of specifications in validation efforts to find and address potentially harmful inconsistencies.
Selection	**3J)** use more than one prioritization technique in a relevant way. **3K)** reflect on prioritization experiences.	**4J)** create a release plan for a subset of prioritized features, while taking into account precedence constraints.	**5F)** combine priorities from several stakeholders and use priorities and scheduling constraints to iteratively create a relevant release plan. **5G)** use prioritization to focus improvements of specification quality and elicitation efforts for a well-motivated subset of requirements.