Project Description ETSN15 Requirements Engineering http://cs.lth.se/krav

Björn Regnell, Elizabeth Bjarnason, Johan Linåker

Study period: 2023-VT1, Revision date: January 17, 2023

1 Objectives

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

1. connect theory to practice,
2. give a concrete experience of practical requirements engineering,
3. promote student motivation through real stakeholders, and
4. provide a group-learning setting that is focused on realistic problems.

2 Assignment of project team and project mission

By Monday of the first course week, you should give input to the team and mission assignment as follows:

1. Read through all Project Missions from the startup companies.
2. Give your preferences according to given instructions.
3. You will be assigned a project team and a project mission based on preferences and availability

3 Context and roles

Each project team will act as an *under-consultant* to a startup company and support them with requirements engineering for their product idea. The startup companies will act as *product owners* and are responsible for conveying their vision of their product idea. They will provide a Project Mission upon which the project team is to develop a system model including requirements of different types at appropriate abstraction levels. The product owners will support the project team on request, but the project team is to act independently with the Requirements Engineering process throughout the project. The product owner is to be seen as a stakeholder among many during the Requirements Engineering process.

The project team consists of 4-7 members and these managers should be appointed among team members:

P3RM Project, Process, Prioritization, & Release Manager (1 pers)
SPOC Stakeholder & Product Owner Communication (1 pers)
TDEVM Tools, Documents, Experiences & Version Manager (1-2 pers)
EPM Elicitation & Prototyping Manager (1-2 pers)
QRM Quality Requirements Manager (1 pers)
DRVM Data Requirements & Validation Manager (1 pers)

The manager roles imply management, planning, and coordination responsibilities, but managers should not do all the work: *all members should contribute in all parts!*

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.

5 Project deliverables

Phase	Deliverables	Deadline
Planning	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: Tuesday 12:00
	Discussant Questions	Week 7: Tuesday 12:00
	Release R3	Week 7: Sunday 23:59
	Course Evaluation	March 31st, Thursday 23:59

All deliverables should have a title, version number, team id (capital letter and number), system name and names of the project members.

5.1 Project Mission v2

Your team should prepare a second version of the Project Mission where the scope of the project is further defined after dialog with your product owner (startup company). The purpose of this version is to act as an agreement that specifies what your team intends to develop, and what the product owner can expect.

1. The Project Mission v2 is recommended to include the following information:
   (a) Table of contents
   (b) Background and other information from Project Mission v1
   (c) Main goals and system context, including a context diagram
   (d) Participants and potential stakeholders
   (e) Description of planned activities and deliverables with deadlines
   (f) Diagram showing, per participant, the planned activities and time spent per week
   (g) Responsibilities of project members

2. With the above content it is useful if the following questions can be answered:
   (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 who work with what?

5.2 Deliverables

1. 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.)
2. The releases (delivered for the course) are denoted R1, R2, and R3.
3. For each release, the quality of your deliverables should represent a noticeable improvement.
4. Each release should be divided into two explicit parts: **System Requirements** and **Project Experiences**, each with its own **table of contents**.
5. 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`.
6. A release `Rn` of team `X` should be delivered in *one single, self-contained* **zip-file** named `X-Rn.zip` including *all* deliverables.

7. Each deliverable may link to further resources such as html pages, pdf documents, screen images, text files, executables, etc., all contained in the delivered zip file. No external links outside the zip are allowed.
8. The second version of the system requirements (R2) should include a first version of the release plan.
9. The last release R3 should include final versions of: System Requirements, Project Experiences, Validation Report & Checklist (final version by R2 and included in R3), and Conference Presentation. Course Evaluation is delivered post course.

System Requirements includes 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 subset of the requirements should be prioritized and release planned into the releases R3 (final course delivery), and (imagined future releases) R4 and R5.
   (e) Design-level requirements are to be specified for the sub-set of requirements that are planned for release in R3 (see previous point). This sub-set of requirements shall be implemented as mock-up designs in the final course delivery (R3) using, e.g. screens and prototypes, analog drawings, clickable presentations, executable GUI mockups.

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 communication and interaction with the product owners and within the project team 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.

(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.) A separate, free-form Course Evaluation document should 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?

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 R3 and your Validation Report & Checklist according to the criteria in the table on the next page.

Assessment area	Requirements for project grade **3** Demonstrate **acceptable** ability to ...	Also required for project grade **4** Demonstrate **advanced** ability to ...	Also required for project grade **5** Demonstrate **excellent** 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.	**4E)** reason about the need for further elicitation in relation to specification quality. **4F)** demonstrate good use of prototyping to elicit realistic user requirements.	**5C)** go beyond initial stakeholders and given frames, while challenging the domain boundaries and eliciting creative ideas and deep domain knowledge in real-world contexts.
Validation	**3F)** to assess the quality of requirements and find relevant problems of several different types. **3G)** apply more than one validation technique including prototyping. **3H)** reflect on validation experiences.	**4G)** to find, prioritize and discuss requirements quality problems of different types, while reaching beyond form issues. **4H)** 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, both from a product owner and developer viewpoint. **5E)** utilize links among different types of specifications in validation efforts to find and address potentially harmful inconsistencies.
Prioritization	**3I)** use more than one prioritization technique in a relevant way. **3J)** reflect on prioritization experiences.	**4I)** 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.