

**Project:** System for Student Activity Management, Group 7

**Date, who:** 09-10-2020, Group 3

## **Introduction**

In this peer review, an inspection method will be applied in order to evaluate the requirements specification and return quality feedback aimed towards improving the report by e.g. pointing out defects and areas that need improvement.

The chosen inspection method is a checklist. This is due to the nature of the project, which has to satisfy certain course criteria in order to be acceptable. Therefore, rather than looking at general guidelines and existing checklist form of what a realistic requirements specification needs to contain, a simplified version based on the course template and criteria can be used.

By using the checklist form provided by Lauesen as a foundation and adding main components required by course directives, a comprehensive checklist was created. After that, a number of content checks that were deemed unnecessary were removed. Some of the content checks, such as stress cases or acceptance tests, are not required by any criteria, but are still included in the checklist as it allows for realization of how the report could excel beyond what is simply “required” by course directives.

When a checklist had been finalized, the group filled out their individual forms before discussing and merging their results.

## **Overview of peer review:**

The high-level description of the report feels on its way to completion and provides detailed information at an appropriate abstraction level. Besides minor improvements such as making the stakeholder power/sentiment ratings into a 2D chart to provide more clarity and provide more detailed problem definition for business goals, not many areas of improvement were found while reading the chapter. The traceability diagram feels a little too early and could be moved down to chapter 3 where additional requirements, besides from functional ones, can also be added to provide a more clear picture at a lower-level description.

In chapter 2, the user requirements specifications, provides detailed data, functional, and non-functional requirements along with some prioritization and description of key attributes. A main improvement area noticed in this chapter was the lack of traceability and proof of interdependencies in certain sections. This was especially noted with the non-functional requirements that could definitely need traceability to e.g. the quality grid concerns or business goals which would provide the report a more sense of completeness. The consistency checks (using CRUD) came positive with only some entities missing Update and Delete tasks/events.

The final chapter, system requirements, provides a nice overview of the planned UI as well as functional requirements and well written and detailed descriptions of their actualization.

In conclusion, the report is very well made and is characterized by detailed descriptions, all group members agreed to having a hard time finding defects and/or areas of improvement throughout the inspection. We believe the report is on a good way to completion and beyond what the project itself requires.

(Content categories with an asterisk (\*) are not necessary for the project but good to check for)

Contents check	Observations - found and missing	Problems or comments?
Goal	Found	Clear goal definition.
Scope	Found	
Context diagram	Found	Well made with clear scope. Double lined arrows needed to show if the supplier is also integrating outlying systems. Later in the document a student committee system is discussed but is not shown in the context diagram.
Business goals	Found	Some business goal problem definitions feel vague and could use a better motivation for its existence and the need for it in the system, while still keeping it high level. For example, BG2 states that student committees will have the option to offer events for students, but does not define why it is needed. Sounds more like a feature rather than a business goal.
Stakeholder map	Found	The power/sentiment levels are presented in a linear fashion rather than a 2D map. For example, student committees might have high interest, as the system is a method of broadcasting events, while holding low power as they are not the main user group.
Core functionality	Found, represented in a use case diagram together with a table.	
Use case diagram or goal model	Found, use case diagram.	Potentially too low level?
System requirements	Found	Nice with detailed description.

Contents check (cont.)	Observations - found and missing	Problem or comments?
Trace goals → requirements	Found, two traceability diagrams used. One mapping Business goals to use cases, and one mapping core functionality to use cases and requirements.	Traceability diagram looks great but is unnecessarily complex. A matrix, for example, would only need one table and be very easy to trace effortlessly from business goals to requirements. BG4 not traced to any requirements. Some use cases are not traced. Feels a little too early? Could be placed in chapter 3 together with all other requirements for better final low level overview.
<b>Data requirements:</b>		
Database contents	ER diagram found	Several lines in the diagram are crossing. ID attribute should probably be left out as it is on a design level.
Data dictionary	Found	Missing data for third party integrations
Input/output formats	Input: found. Output: missing	Output is default?
Initial data and states	Incomplete	Attributes should have initial data/states.
<b>Functional requirements:</b>		
List of features and/or user stories	Found	A neat list with requirements divided into proper levels.
Domain-level requirements	Found	R12005 mentions external student committee systems, this was not visual in the previous context diagram.
Product-level requirements	Found	
Design-level requirements	Found	
Every requirement correlates to a core functionality and/or business goal	Found, in traceability diagram.	
Prioritization	Found, \$100, MoSCoW, and classical ranking.	Additional explanation of the actual results and its implications would be great.
Next release requirement coverage	Found	Detailed explanation with clear tables.

<b>Contents check (cont.)</b>	<b>Observations - found and missing</b>	<b>Problem or comments?</b>
<b>Special cases:</b>		
Stress cases (*)	Found	Covered in non-functional requirements.
Use cases dealing with special events or errors (*)	Missing	
<b>Quality requirements:</b>		
Quality grid	Found	Connecting some of these concerns with goals or requirements to increase traceability and show interdependence would be great.
Non-functional requirements	Found	Neatly divided into levels of goal, domain, product, and design level. Could also need reference to goals or quality grid concerns.
Key attributes, Planguage	Found	Why are these three key attributes key? adding some description and/or connecting with quality grid concerns or requirements is needed.
<b>Other deliverables:</b>		
Support and maintenance (*)	Missing	
<b>Glossary:</b>		
Domain and product terms (*)	Missing	
<b>Structure check</b>		
ID for each requirement	Found	Very nice format.
Verifiable requirements	Found	
Purpose of each requirement	Found, for functional requirements in the traceability diagram.	See comments on non functional requirements.

<b>Contents check (cont.)</b>	<b>Observations - found and missing</b>	<b>Problem or comments?</b>
<b>Structure check (cont.)</b> Examples of ways to meet requirement	Found, in some cases (traceability diagram and system requirements)	
Text explanation of diagrams, etc.	Found	Good explanations outside of figures. Captions could however be more informative.
<b>Structure check (cont.):</b> Importance and stability	Found, by prioritization.	
No duplicate information	Requirements duplicated	Entire Section 3.1.1 contains info from Section 2.2
Index	Found	No List of Figures or Tables
An electronic version	Found	
<b>Consistency checks:</b> CRUD check: Create, read, update, delete all data?	Missing	User: missing UDO Offer: missing UD UserOffer: missing UD
All events handled by task or function?	Found	
Task and/or message data in database?	Found	
UI prototype data available in database and vice versa?	Found	In UI but not in ER: location, activity notes. In ER but not in UI: Majority of user, user offer, offer, user role, role, activity: endtime
UI prototype support all tasks?	Majority found, some were however missing.	R12003: internal email system R13008: student committee R13015:share R13017 view and manage a long term bucket. activitiesRegistration, Push notification, reminderView, all the stuff relate to offer are missing

Contents check (cont.)	Observations - found and missing	Problem or comments?
<b>Reviews:</b>  Goals and critical issues covered?	Found	All business goals are covered when considering the entirety of the report. Increased traceability throughout the sections would however allow for readers to quickly confirm it early in the report.
Requirements justified?	Found	Could however use a traceability matrix in the end to connect the many requirements and attributes to the bigger picture.
<b>Tests:</b>  Acceptance tests (*)	Missing	