Use Cases and Non-functional Requirements Presented in Compact System Description A3s

Anders Fuglesteg Nilsen

Buskerud University College

Frogsvei 41, 3611 Kongsberg

Gerrit Muller

Buskerud University College

Frogsvei 41, 3611 Kongsberg

Abstract. Translating stakeholder requirements into system requirements is important for guiding the detailed design towards a solution that meets the stakeholder needs. The methods typically used today in the company do not sufficiently capture performance requirements.
We combined the functional and non-functional aspects of a system in one diagram. We used compact system description A3s to communicate the diagram and other system information to stakeholders.
We showed that presenting functions through use cases, annotated with non-functional requirements, help capture the performance aspect of the system, and that compact system description A3s communicate the most important aspect of the system well.

Introduction

Company. Kongsberg Devotek AS is an engineering consultancy company developing systems comprised of mechanics, electronics, control, and software for customers in defense, oil and gas, medical, industry, maritime, and automotive. Devotek does not claim ownership of any products developed for customers, the business model is that Devotek shall be a trusted partner that can work as an extension of the customer's organization.

Case. The project in which the case study is applied is competition sensitive. The customer does not allow any information about the project to leave Devotek. The author altered or removed all customer product and project specific contents from this report. We present the research data collected from the customer through use of the method as data from fictitious customer Cooling Systems (CS). They need an actuator that can operate globe valves in an industrial cooling application.

Problem statement. We have experienced that some of Devotek's system architects fail to establish all the relevant performance requirements, which may lead to loopbacks during the design. In addition, our experience is that presenting requirement specification documents to stakeholders does not contribute to stakeholder feedback. If Devotek ignores this then they might lose customers, thus revenue and reputation as a technology development provider.

We want to provide the system architects in Devotek with a method, which sufficiently covers the performance aspects of systems, e.g., speed, force, time, torque, etc., while establishing system requirements. The method has to communicate the requirements effectively.

We expect that a method based on Use case modeling combined with Non-Functional Requirements (NFRs) will help to capture performance aspects**.** The method shall contribute to translating stakeholder needs, design guidelines, and experience into system requirements for a given system concept, by applying use cases annotated with non-functional requirements. The method does not replace requirement specification documents typically used in Devotek, but will serve as a tool for deriving and communicating requirements. Devotek has used compact system descriptions on A3 sized papers for communication purposes in earlier projects. We will develop the method to fit into the Devotek system description A3.

What shall the method improve? We can offer a method that will provide the following improvements:

- Support the system architect with establishing system requirements that cover the performance aspect of the system sufficiently.
- Improve the readability of the requirements compared with requirement specification documents used today.
- Contribute to more valuable requirement specification documents by serving as a source of requirements that the system architects otherwise may forget.

Current way of working

Existing framework for writing requirements. Figure 1 shows the process for establishing system requirements in Devotek. The Devotek Management Norm, which includes all Devotek processes, states that a person with the system architect role, shall perform all steps under "Establish system requirements". The enterprise Wiki describes the activities related to each process step.

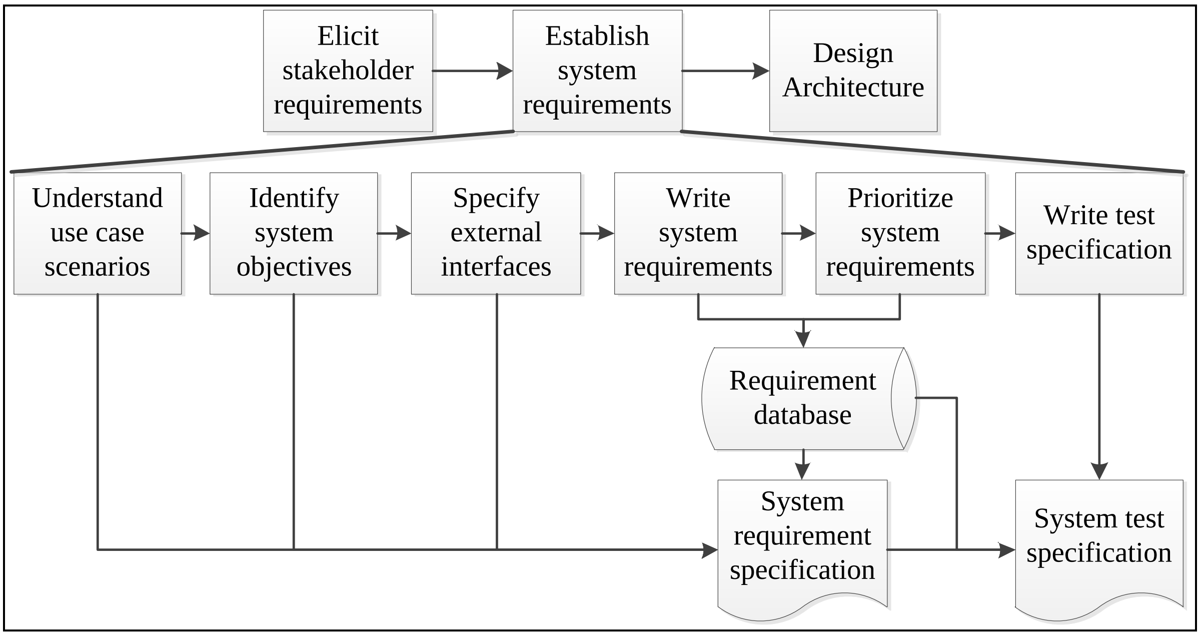


Figure 1 - Devotek process for establishing system requirements and resulting artifacts

Problems developing system and component requirements. The process step "Elicit stakeholder requirements" includes these activities.

- Define the customer need, opportunity, or idea.
- Identify stakeholders, elicit their desired capabilities and characteristics, and define key acceptance criteria.
- Develop system context diagram.
- Generate, evaluate, and select concept.

The result of this process step is descriptions and/or drawings of a selected concept and a stakeholder requirement specification. The system architect has to transform the stakeholder needs, capabilities, and characteristics, documented in the stakeholder requirement specification, into system requirements specific to the selected concept. The system requirements shall fulfill the needs, capabilities, and characteristics of the stakeholders as far as possible within project and technology constraints. As seen in Figure 1, the “Establish system requirements” process step, establishes system requirements in three steps:

- “Understand use case scenarios” establishes the functional requirements (FRs)
- “Identify system objectives” establishes the non-functional requirements
- “Specify external interfaces” establishes external interface requirements

“Write system requirements” is the process of collecting requirements from the three previous steps and organizing them as a set of requirements. We have observed, in previous Devotek projects, by using the process, that the non-functional requirements of the system, do not add relevant information to desired capabilities and characteristics already stated in the stakeholder requirement specification. The stakeholder requirement specification covers the “ilities” well, except for performance aspects, e.g. speed, force, torque, efficiency, etc., of the system. We have experienced in previous Devotek projects that failure to account for performance requirements, may cause loopbacks in design that increase project cost.

Introduction to Linear valve actuator case

The author chose to define an artificial system for actuation of valves in an industrial cooling application. Figure 2 shows the system of interest, the linear valve actuator, in a typical application.

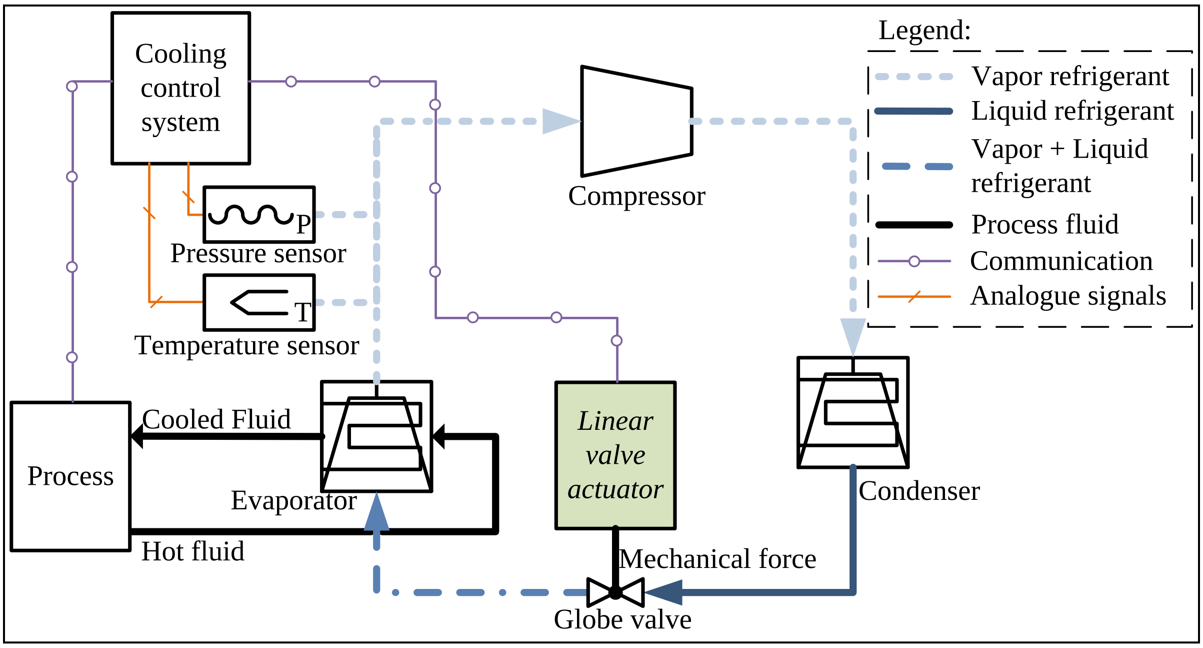


Figure 2 - Linear valve actuator seen in the cooling process

The system shall cover a range of linearly actuated valves controlling the flow through a cooling pipeline. The actuator will serve as a component in the Cooling Systems catalogue. The main objective of the actuator is to provide sufficient force on the valve stem to move the valve piston between open and closed position and vice versa. Figure 3 shows the forces acting on the valve stem during operation. Friction force, seating force, and pressure difference over the piston are the main contributors to the valve force.

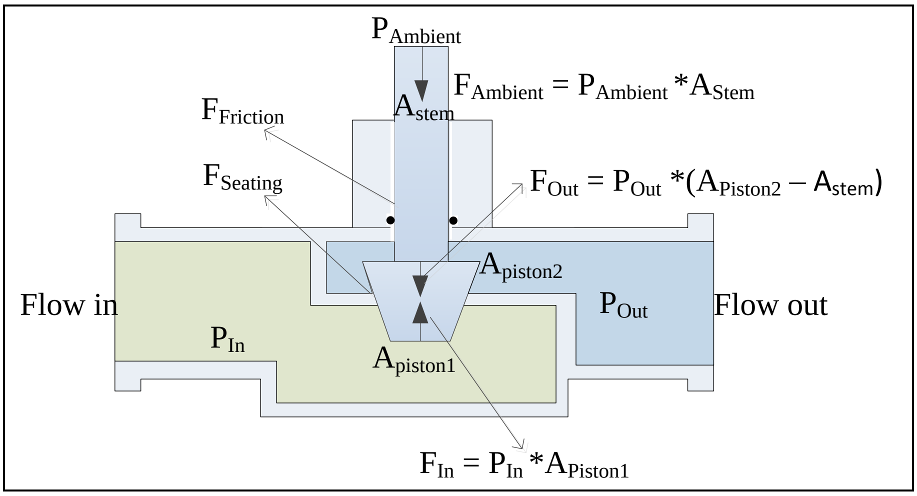


Figure 3 - Valve forces

Research methodology

Method for acquisition of data. The research method applied is action research (Riel, 2010). The compact system description A3s where developed iteratively. The author printed the compact system description on A3 paper, and presented to stakeholders. The stakeholders then had the possibility to provide feedback orally or by sketching directly on their printed version. After the A3 effort, the involved stakeholders participated in a survey. We conducted the survey by handing out a questionnaire with 10 Likert scale (Jamison 2004) questions, and 2 yes/no questions. We gave the same questionnaire to stakeholders that had: actively used and, only seen the compact system description A3.

How to measure improvement. We will validate the method using the following measurements:

- Provide a questionnaire to stakeholders that capture their impression of the improvement between before and after the method was applied.
- Count the number of requirements derived by using the method.

Why approach is appropriate. We have selected action research as the approach for researching this case. Action research is the systematic, reflective study of one's actions, and the effects of these actions, in a workplace context (Riel 2010). As defined by Riel, action research serves as an appropriate approach for doing research on a method applied in an industrial context. Riel lists surveys, interviews, and focus groups as tools that can help the action researcher to understand the impact of the action taken in social contexts within organizations. Among the tools listed by Riel, surveys are most likely to support analysis (Muller 2012). We therefore chose to use a survey with Likert scale to evaluate the stakeholder's subjective impression of the proposed method.

Use case modelling combined with non-functional requirements

Most software people use the use case submethod only for behavioural descriptions. In embedded systems design this submethod is also very useful for quantitative descriptions of the system, for instance for performance (Muller 2004). The method described by Muller focus on describing both the system functionality and non-functional aspects by deploying use cases. Muller separates use cases into typical, worst case, exceptional, or change use cases. He presents the use cases as a list of verb noun pairs together with a list of quantitative requirements related to the use cases. In our proposed method, we associate the NFRs directly with use cases, system boundary or association between external entity and use case.

Suppacul states in his article “Integrating FRs and NFRs: A Use Case and Goal Driven Approach. “(Suppacul 2004): *To provide precise context for NFRs, we propose that NFRs be integrated at certain points in the use case diagram called NFR Association Points*. We will adopt the association points proposed by Seppacul to fit our proposed method, which focus more on performance aspects than the method proposed by Seppacul. Figure 4 shows the points where Seppacul proposed to attach NFRs.

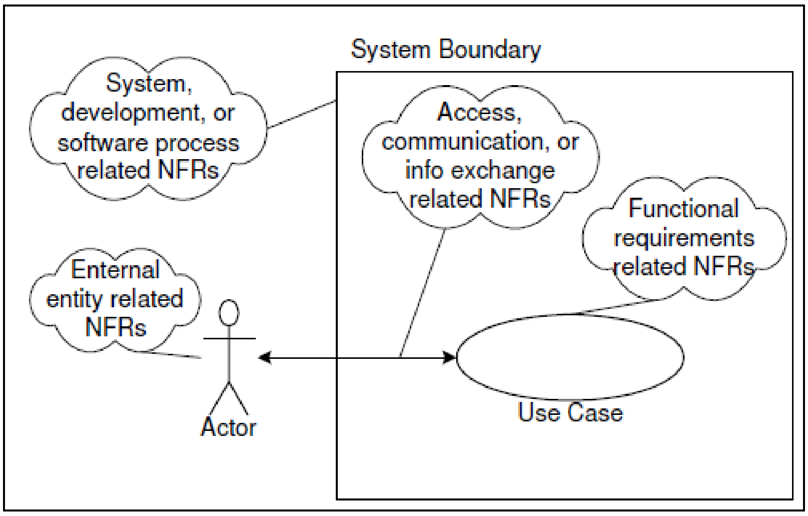


Figure 4 - NFR association points (Suppacul 2004)

Figure 5 shows the proposed method for combining NFRs and use cases, covering functional requirements, in one model. As the purpose of the method is to derive and communicate requirements, especially performance related requirements, focus is not to include all requirements, but the most important ones. The system architect in cooperation with relevant project team members has to decide which requirements are most important. In general, the most important requirements drive the design further. They give the designers goals for which their components much comply with, and failing to accommodate them may lead to loopback during the design.

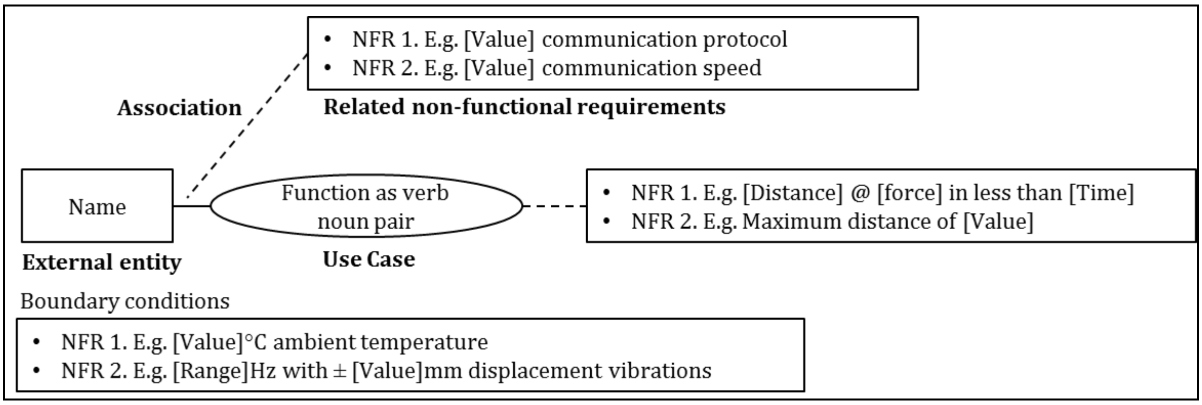


Figure 5 - Proposed method for representing NFRs in Use case diagrams

The model in Figure 5 includes the following elements:

- External entity (Actor or external system interacting with the function);
- Use Case (Function that the system shall execute);
- Related Non-functional requirements (List of non-functional requirements related to the associated element); and
- Association (Link that visualize connection between use case diagram element and non-functional requirement)

The approach aids system architects in eliciting performance requirements as well as functional requirements of the system. We have experienced that system architects tend to leave out the system performance requirements if not specified by the customer. This constitutes a risk to project success. The approach can mitigate part of the risk, by providing a tool that enables the system architect to understand which NFRs will fully define the functions.

Compact system description using A3s

The system architect will develop new requirements when using the proposed method. To verify that these requirements will meet the needs of the stakeholders, the system architects should present their findings through a medium that encourages stakeholder feedback. For this purpose a compact system description, relating different views of the system in one compact A3 sized piece of paper, will help the system architects to get feedback on their work (Borches 2010).

Figure 6 shows the how the architecture description is by definition a flattened and poor representation of an actual architecture, and that the overview of the architecture is again only a fraction of the architecture description (Muller 2013).

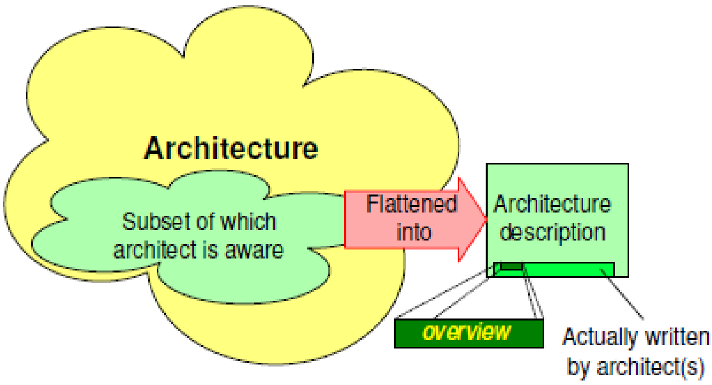


Figure 6 - Architecture Overview as part of the architecture description (Muller 2013)

Muller also formulates the role of system architecture description as:

- Guiding and constraining framework
- Spanning from opportunity exploration via development, manufacturing to support and retirement
- Supporting communication and decision-making
- Providing an audit trail from problem/opportunity to solution

The role of the system architecture and the CAFCR model (Muller 2004), together with examples presented by (Frøvold 2011) has set the foundation for information to include in the A3 system description.

The focus of the system description A3 is to present the most relevant information on the topic of the A3. We used color codes to link associated pieces of information together as proposed in (Borches 2010) where applicable. Figure 7 shows our proposed template for the compact system description A3, hereby referred to only as A3.

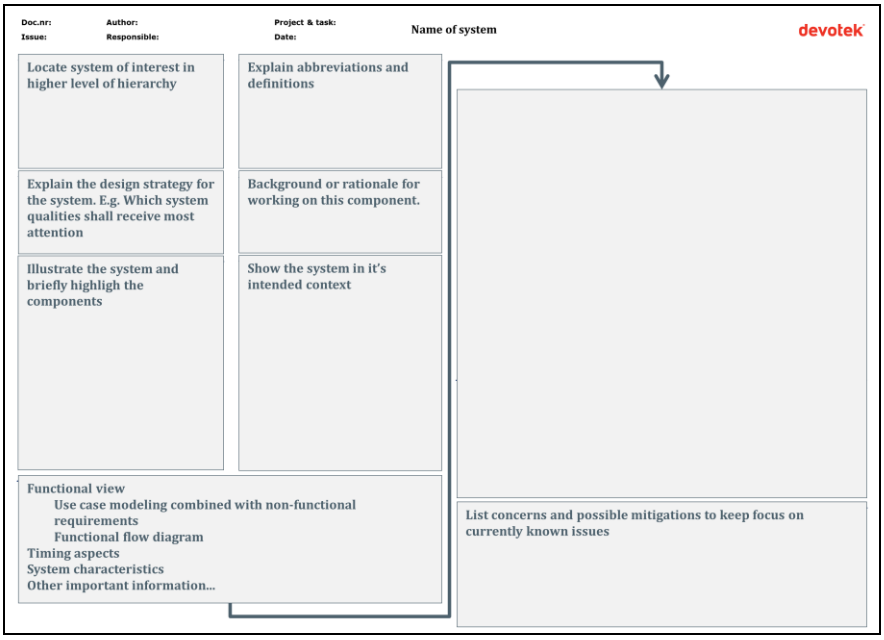


Figure 7 - Template for system description A3

Analysis of how requirements are written today

As described in Figure 1, system architects should develop system requirements in three steps according to the Devotek Management Norm. The foundation for these steps is a stakeholder requirement specification document and descriptions and/or drawings of a selected concept. Experience has shown that the system architects will write functional system requirements based on the stakeholders desired capabilities and characteristics, which according to their perception is a functional requirement. In effect they skip the process "understand use case scenarios" altogether. They do the same for non-functional requirements, thus skipping the process step "identify system objectives".

For the functional requirements, the completeness of the specification is not that crucial, because the functions of the system are often defined also in the non-functional requirements. E.g., the system shall open the valve in less than 5 seconds. Although this is a non-functional requirement, it also contains the function, open the valve. For the non-functional requirements, the shortcut around the process can be of risk for the project success. Even if the process described in the Devotek Management Norm is followed, the system architects tend to underspecify performance requirements.

Representing use case models combined with NFRs in A3

We developed the method of combining use cases and NFRs in an iterative process. We structured the diagram, shown in Figure 5; to fit within half the width of a horizontally oriented A3 sized paper. Limiting the width of the diagram allowed arrangement of the content of the A3 into two main columns.

Previous versions of A3s deployed in Devotek projects used tables to represent NFRs and a list to represent the main functions of the system of interest. In this method we wanted to represent all requirements previously shown in lists and tables in one diagram. We included an association to the link between external entities and use cases to represent external interface requirements. We included the boundary conditions in the diagram to avoid having a separate table with headings and descriptions requiring additional space on the A3.

Method applied on linear valve actuator case

We developed the method while continuously receiving feedback from the stakeholders involved with the confidential project left out of this article. For this reason, we are unable to explain all decisions made in the confidential project through the artificial project. In such situations, we present similar problems applicable in the artificial case.

We started developing the LVA A3 from a template previously used in Devotek projects. Figure 8 shows a snapshot of the final version of the A3 with enlarged section titles as overlay. We will describe the content of the A3 from top left corner to bottom right corner in the following paragraphs.

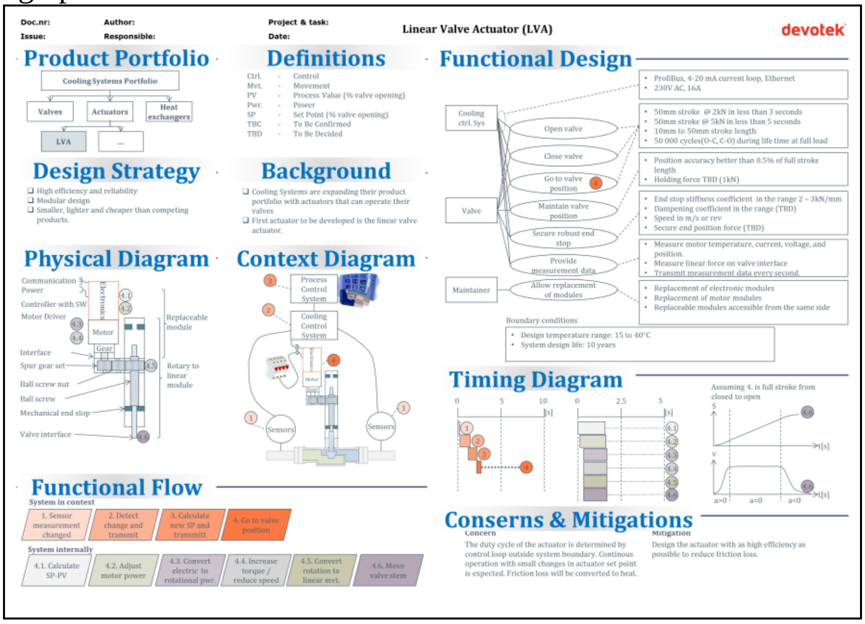


Figure 8 - LVA A3 final version with enlarged section titles as overlay

Product portfolio. We wanted to make the systems position in the customer product range explicit. The product portfolio also serves as a navigational map when representing a set of related A3s together in one file. Figure 9 shows an enlarged version of the product portfolio section.

Definitions. We needed to include a section with definitions to keep the content of model elements in other sections short, explain additional information on a term e.g. SP in Figure 10, and reduce the size of diagrams in other sections.

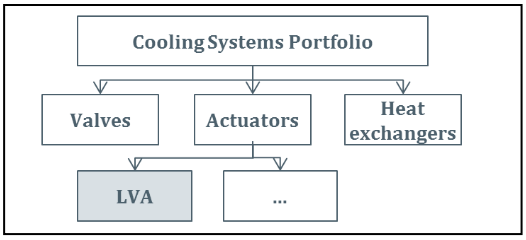


Figure 9 - Product portfolio section

Ctrl.	-	Control
Mvt.	-	Movement
PV	-	Process Value (% valve opening)
Pwr.	-	Power
SP	-	Set Point (% valve opening)
TBC	-	To Be Confirmed
TBD	-	To Be Decided

Figure 10 - Definitions section

Design Strategy. We included the design strategy to make clear which system qualities the project team should give the most weight in concept selection phases. Figure 11 shows the design strategy for the LVA system.

Background. We included a section called background, which serves as a rationale for why we are doing this development effort. We wanted to write the background as short as possible while still conveying the message of why the customer needs the system. Figure 12 shows the contents of the background section.

- ❑ High efficiency and reliability
- ❑ Modular design
- ❑ Smaller, lighter and cheaper than competing products.

Figure 11 - Design strategy section

- ❑ Cooling Systems are expanding their product portfolio with actuators that can operate their valves
- ❑ First actuator to be developed is the linear valve actuator.

Figure 12 - Background section

Physical Diagram. (Engebakken 2010) states that if a model is close to reality, the stakeholders will easily understand the model. We have experienced the same when presenting models to stakeholders. We therefore wanted the physical diagram to represent what we think is most likely the final solution. The LVA physical diagram, shown in Figure 13 is an illustration showing the components included in the actuator. If we have a 3D model available, we would use a snapshot of that instead of the illustration.

Context Diagram. We wanted to show which entities the system interface during the operational lifecycle and we chose to use a context diagram for the purpose. The context diagram, shown in Figure 14, for the LVA illustrates not only the external entities interfacing the LVA directly, but also other entities that influence the LVA indirectly.

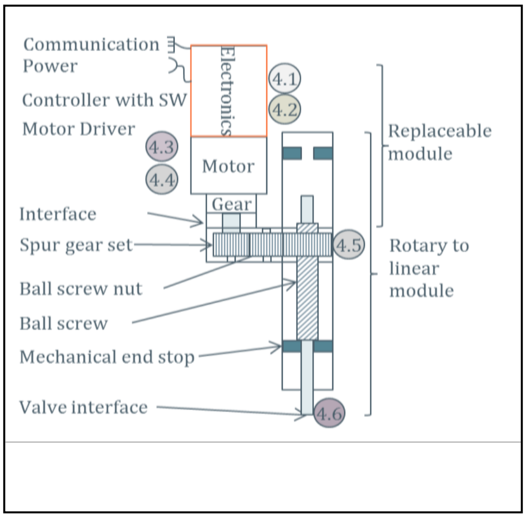


Figure 13 - Physical diagram section

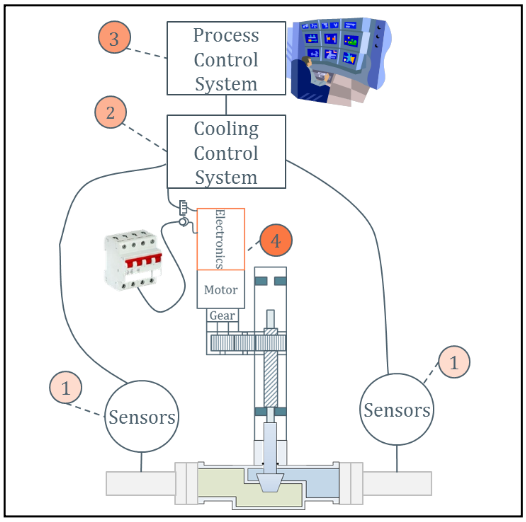


Figure 14 - Context diagram section

Functional flow. We wanted to show how the process, which the LVA is part of, uses the LVA, and what the actuator has to do to fulfill its role. The functional flow, system in context, illustrates how the LVA participates as an actuation mechanism in a closed control loop involving all entities in the context diagram. The functional flow, system internally, shows the functional flow inside the actuator. The flow inside the actuator is the detailed level of function 4. Go to valve position. Figure 15 shows the contents of the functional flow section.

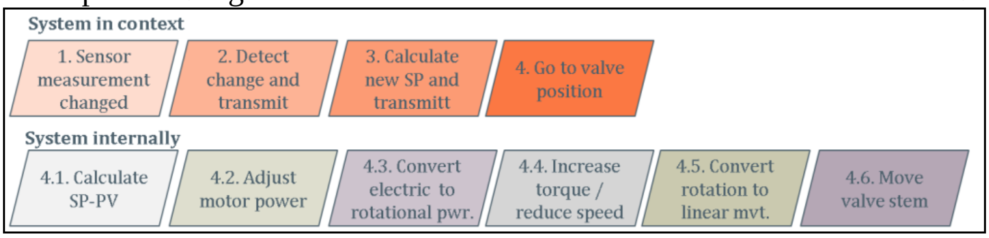


Figure 15 - Functional flow section

Functional Design. We chose to give the method of combining use cases and non-functional requirements the largest part of the A3. We wanted to present all the requirements previously shown in tables and lists in one diagram. To accomplish this, we added the association point between external entity and use case to capture important interface requirements. We also added the unassociated field boundary conditions to capture quality and environmental requirements not related to the functions or interfaces. Figure 16 shows the content of the functional design section.

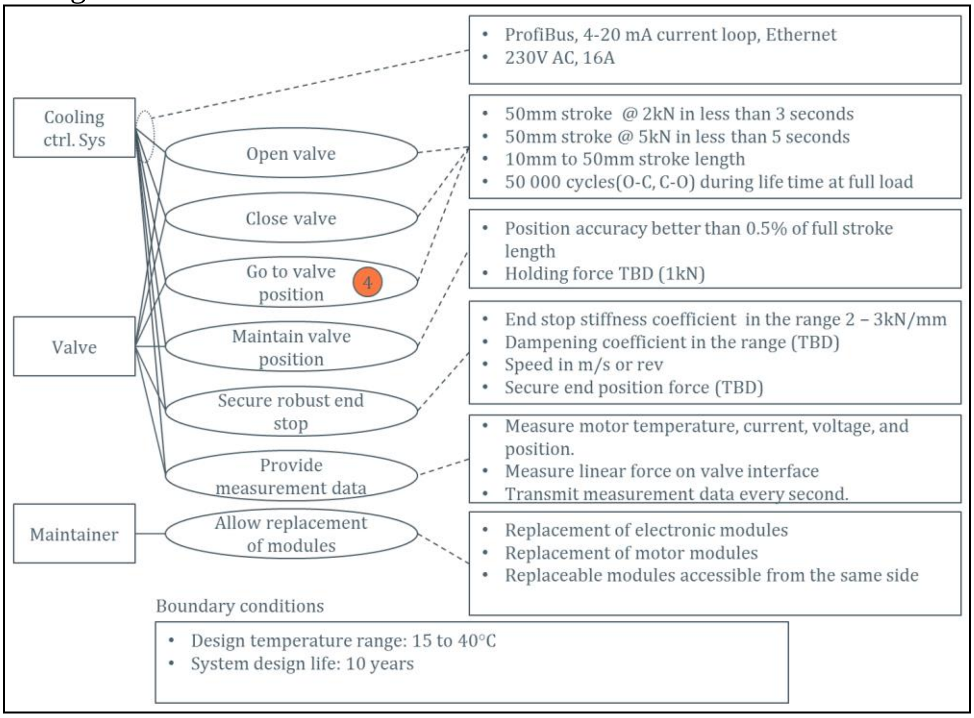


Figure 16 - Functional design section

Timing Diagram. We wanted to represent how much time the LVA spent on actuation in relation to the total elapsed time, from a measurement signal changed to the actuator have corrected the resulting deviation. We chose to use a timing diagram for this purpose. We created the timing diagram with close relation to the functional flow diagram. The leftmost timing diagram shows the time used for each of the activities in the system in context flow. The middle diagram shows the time spent in the system internally. We also presented the function, 4.6 Move valve stem, as a first order model to illustrate the dynamics of the system output. Figure 17 shows the content of the timing diagram section.

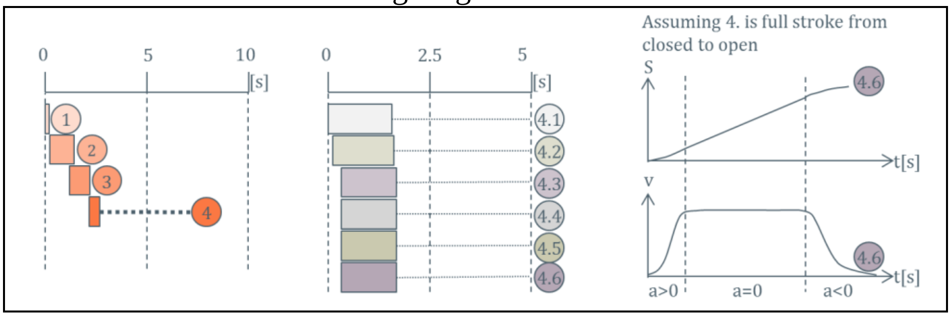
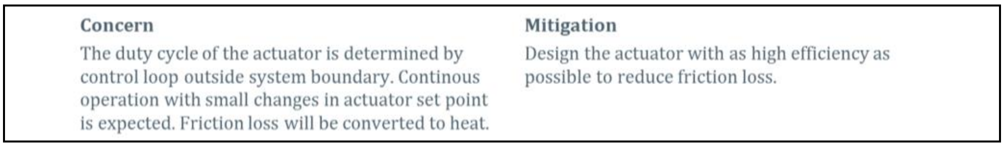


Figure 17 - Timing diagram section

Concerns & Mitigations We wanted to highlight some of the concerns regarding the system together with a mitigation strategy. We chose to represent this in a table with concerns in the first column and mitigation strategy in the second column. Figure 18 shows the contents of the concerns & mitigation strategy section.



Concern	Mitigation
The duty cycle of the actuator is determined by control loop outside system boundary. Continous operation with small changes in actuator set point is expected. Friction loss will be converted to heat.	Design the actuator with as high efficiency as possible to reduce friction loss.

Figure 18 - Concerns & mitigations section

Evaluation

Analysis of acquired data. We collected data by distributing a questionnaire to 10 members of the project team working on the LVA system. We printed the questionnaire on paper and attached the latest version of the LVA A3. The questionnaire contained 5 Likert scale questions and 1 yes/no questions addressing the proposed method, and 5 Likert scale and 1 yes/no question addressing the LVA A3 in general. The survey participants answered the Likert scale questions on a 5-point scale, ranging from strongly agree to strongly disagree. We analyzed the resulting data by defining numbers to the Likert scale, where strongly agree is 5 and strongly disagree is 1. In addition to the structured data, the survey participants were encouraged to write comments on the back of the questionnaire. We analyzed the results of the survey data using Net Promoter Score (NPS) (Reichheld 2003) and visual representation of the score of all respondents. We calculated the Net Promoter Score as #strongly agree - (#neutral + #disagree + #strongly disagree) according to (Muller 2012). We have defined a positive Net Promotor Score as an indication of a validated statement.

Evaluation of use case modeling combined with NFRs. Figure 19 sums up the survey results for assessment of use case modeling combined with NFRs.

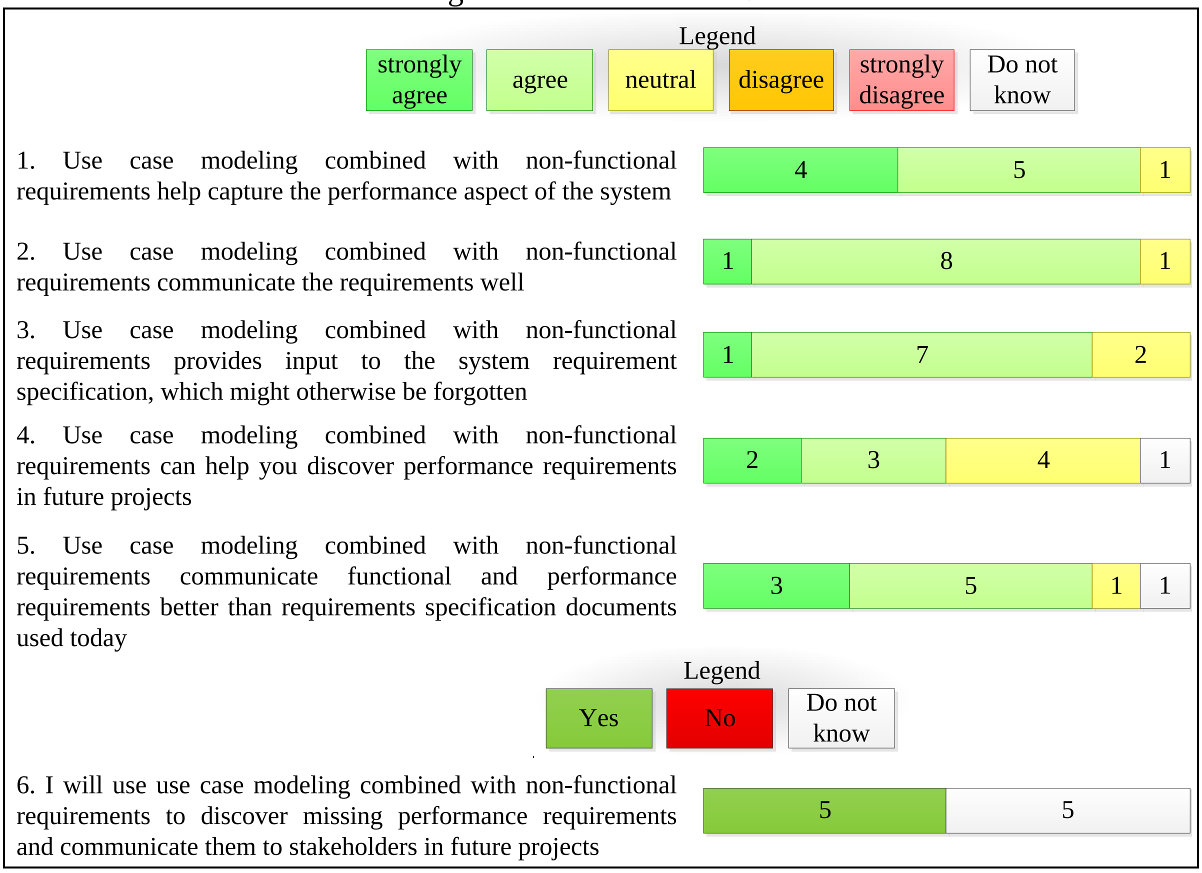


Figure 19 - Survey results for assessment of the proposed method

Question 1 and 5 received a NPS greater than 0. The result shows that the proposed method helps to capture the performance aspect of the system. The method will also communicate the

functional and performance requirements better than requirements specification documents used today.
Question 2,3, and 4 received a NPS equal to or less than 0. The survey population does not think that the proposed method:

- communicate the requirements well;
- provides input to the system requirements specification, which they otherwise might forget; and
- can help them discover performance requirements in future projects.

The 2 survey participants that strongly agree with the statement in question 4 have software background. They work more with use cases, which this method builds upon, than people with mechanical or electronics background.
The survey population agrees that the method helps in capturing the performance requirements in question 1, but does not agree that the method provides input to the system requirement specification, which they otherwise might forget in question 3. We suspect that the contradiction arose because of the difference in words used in question 1 and 3.
Five respondents answered that they will use the method to discover missing performance requirements in the future, and none claimed that they would not use the method.

Evaluation of method applied in A3. Figure 20 sums up the survey results for assessment of the LVA A3.

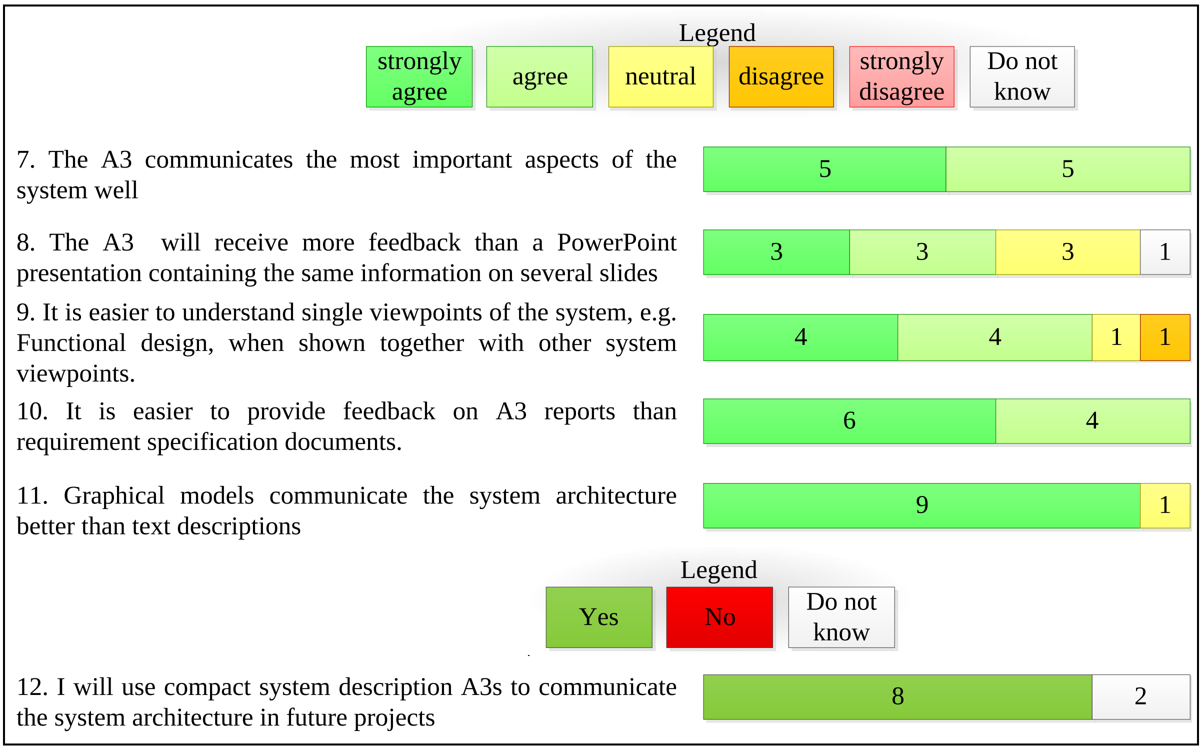


Figure 20 - Survey results for assessment of the A3

Question 7, 9, 10, and 11 receive a NPS greater than 0. The results show that:

- the A3 communicate the most important aspect of the system well;
- it is easier to understand the single viewpoints of the system when shown together with other system viewpoints;
- it is easier to provide feedback on A3 reports than requirement specification documents; and
- graphical models communicate the system architecture better than text descriptions.

Question 8, which states that the A3 will receive more feedback than a PowerPoint presentation containing the same information on several slides, received a NPS of 0. One of the survey

participants commented that he valued the unrestricted space of a PowerPoint presentation According to question 12, 8 of the survey participant's state that they want to use A3s to communicate system architecture in future projects.
Overall, the population of the survey is in favor of the A3 presented to them in the survey and states that they will continue to use A3s in the future.

Analysis of requirements. The requirements presented in Figure 16 originate from several input documents and feedback from both customer and Devotek project organization. We derived 8 performance requirements not specified by the customer at the outset of the project.

- 50mm stroke @ 2kN force in less than 3 seconds
- 50mm stroke @ 5kN force in less than 5 seconds
- 50 000 cycles (O-C, C-O) during life time at full load
- Holding force TBD (1kN)
- End stop stiffness coefficient in the range 2 - 3kN/mm
- Dampening coefficient in the range (TBD)
- Speed in m/s or rev
- Secure end position force (TBD)

The first two requirements where derived from datasheets from 5 applicable valve types, customers desired stroke time, and available electrical power. The third requirement was present in customer documentation, but Devotek had failed to discover the requirement. We presented the A3 to the customer showing only "Operating cycles (TBD)", and they could provide us with the location of the requirement. We derived the fourth requirement by the association to the function Maintain valve position. We found the proposed value, 1kN, by calculating known parameters found in the valve datasheets. We derived the last 4 requirements by analyzing which NFRs define the function "Secure robust end stop", shown in Figure 16.

Credibility of data. The author's relation with the survey participants may affect the credibility of the quantified data gathered from the survey, because participants may have felt that they should not be too negative towards the work of someone they know. This bias is limited by using the NPS, where agree is regarded as neutral. Devotek might have derived some of the performance requirements derived by deploying use case modeling combined NFRs at some point in time, regardless of method.

Validation of data, analysis, and results. We collected the survey data from a population of 10 project members with different backgrounds: three participants with electronics background, two with software background, three with mechanical background and 1 with cybernetics background. 10 participants is a small, but representable population because of the diverse background. The analysis of the results presents the responses of each questionnaire question as a bar showing number of respondents for each answer possibility. As the Likert scale provides ordinal data, we have not performed any statistical analysis of the data, but we used the principles of Net Promoter Score to assess the results.
The analysis of performance requirements derived by deploying use case modeling combined with NFRs showed that we derived requirements using the method. We could validate the data with higher accuracy if we could benchmark the data towards other methods, e.g. current way of working in Devotek. The results are however valid as they show that important performance requirements were captured using the method.

Limitations or constraints of the results. We have collected data mainly inside Devotek, which is a consultancy company working mainly in development of new products unknown to Devotek at the outset of the projects. Although the product is new to Devotek, the customer might know the product well. The gap between Devotek's product knowledge and the customer’s product knowledge will, according to the author’s experience, yield that the customer does not supply Devotek with the information they need. We believe that the results would be different in an organization with ownership of the products under development.

Conclusions

We expected that combining use cases with non-functional requirements would help to capture performance aspects of the system. We found by performing a survey on people working with development of the LVA system that:

- the method can help capture the performance aspect of the system; and
- the method communicates the functional and performance aspect of the system better than requirement specification documents used today.

The survey also showed that the survey participants do not think that:

- the method communicates the requirements well;
- the method provides input to the system requirement specification; and
- the method can help them discover performance requirements in future projects.

Half of the survey participants think that they will use the method in future projects and the other half does not know whether they will use it or not.

We expected that using compact system description A3s with the proposed method will communicate the requirements effectively. The survey results show that compact system description A3s:

- communicate the most important aspect of the system well;
- makes it easier to understand single viewpoints of the system when shown together with other system viewpoints;
- makes it easier to provide feedback than on requirements specification documents; and
- using graphical models communicate the system architecture better than text descriptions.

The results of the survey do not support that compact system description A3s receive more feedback than a PowerPoint presentation, containing the same information on several slides.

Devotek derived 8 performance requirements while using this method in the linear valve actuator project. This supports that combining use cases with non-functional requirements in compact system description A3s, help in deriving performance requirements.

Future research

This research shows that the method helped in deriving performance requirements for the system, but the survey showed that the project team is not convinced that the method communicates the requirements effectively. This shows that the author of the diagram was able to make use of the method, but the method does not communicate the requirements sufficiently to stakeholders. We suggest to do further research on the representation of the information in the diagram to understand how to represent the data more effectively. Measurements such as number of stakeholder meetings with requirements as topic, and number of changes to stakeholder requirements due to increased knowledge gained from combining use cases and non-functional requirements, will measure the intended improvement better. Evaluation of the method communicated by other mediums, such as PowerPoint presentation or through a SysML (Friedenthal 2009) architecting tool, could assess together with this research the value of the method more unbiased by the communication medium.

References

Borches, D. 2010. *A3 Architecture Overviews, A tool for effective communication in product evolution.* PhD Thesis, Enschede: Wohrmann Print Service, 2010.

Engebakken, E. 2010. *SUPPORTING THE SYSTEM ARCHITECT: MODEL-ASSISTED COMMUNICATION*. System Research Forum, Vol. 4, No. 2 (2010) 173-188,doi:10.1142/S1793966610000211

Frøvold, K. 2011. *Early Validation through the A3 method*. Accessed online on [13.02.2013] from http://ksee.no/wp-content/uploads/2011/09/KSEE-2011-Kristian-Frovold-v3.pdf

Jamieson, Susan. (2004). *Likert scales: how to (ab)use them*. Medical Education. Accessed online on [03.04.2013] from http://xa.yimg.com/kq/groups/18751725/128169439/name/1LikertScales.pdf

Muller, G. 2004. *CAFCR: A Multi-view Method for Embedded Systems Architecting; Balancing Genericity and Specificity.* Ph.D. thesis. Delft University of Technology.

Muller, G. 2012. *Systems Engineering Research Methods*. Accessed online on [03.04.2013] from www.gaudisite.nl/SEresearchMethodsSlides.pdf

Muller, G. 2013. *How to Create an Architecture Overview*. Accessed online on [14.03.2013] from http://www.gaudisite.nl/OverviewHowToPaper.pdf

Reichheld, Frederich. (2003). *The One Number You Need to Grow*, Harvard Business Review. Accessed online on [16.04.2013] from http://hbr.org/2003/12/the-one-number-you-need-to-grow/ar/1

Riel, M. 2010. Understanding Action Research, Center For Collaborative Action Research. Pepperdine University. Accessed online on [15.03.2013] from *http://cadres.pepperdine.edu/ccar/define.html*

Sanford Friedenthal, Alan Moore and Rick Steiner. 2009. *Book, A Practical Guide to SysML.* Elsevier Science.

Suppacul, S, L. Chung. 2004. *Integrating FRs and NFRs: A Use Case and Goal Driven Approach*. Accessed online on [15.03.2013] from http://rfidlab.iecs.fcu.edu.tw/~cyt/cyt/data/paper/IntegratingFRsAndNFRs.pdf

Biography

Anders Nilsen.

Anders Nilsen received his certificate of apprenticeship as electrician in 2006. He studied automation and information technology at Telemark University College, where he graduated in 2010. In spring 2013, he received his Master's in Systems Engineering from Buskerud University College. He has worked with systems engineering for Kongsberg Devotek from 2010 to present date.

Gerrit Muller.

Gerrit Muller studied physics in Amsterdam. He worked from 1980 until 1999 in industry. Since 1999, he has worked as a senior research fellow at Philips Research and the Embedded Systems Institute in Eindhoven, focusing on developing system architecture methods and the education of new system architects, receiving his doctorate in 2004. In January 2008, he became a full professor of systems engineering at Buskerud University College in Kongsberg, Norway. He continues to work at the Embedded Systems Institute in Eindhoven in a part-time position.