

Seminars

Nikhil Yarlagadda

9405251993

niya15@student.bth.se

Seminar 0:

Reflections:

Article 1: “The art and science of release planning”

This paper mainly deals with the importance of the release planning, difficulties while dealing with it. It also states the art and science of release planning which involves the human intuition and linear problem. By combining these two, a hybrid model for dealing with the release planning is proposed which helps to identify what features to be implemented first. This mainly depends upon the information of no of sprints, experts. This paper provides me a basic idea regarding the release planning. The proposed method is also easy to implement and I also agree with the proposed method, when there is a chaos it would be helpful for selecting the requirements to be implemented in which release in the context of release planning.

Article 2: “A case study evaluation of the guideline-supported QUPER model for elicitation of quality requirements”

The authors of this paper mainly concentrated on the evaluation of QUPER model and the guidelines for implementing the method. This model is for dealing with non-functional requirements. This paper clearly provides the steps for implementing the QUPER model with a small example and also an extra step included. This model is easy to understand and I have implemented this method for my assignment and faced some difficulty during the identifying step cost dependencies as the clear explanation is not provided. There is a need to implement this model in other areas and also the transferability of guidelines should be researched further in order to adopt this model for other domains.

Article 3: “Introducing support for release planning of quality requirements –an industrial evaluation of the QUPER model”

This paper mainly explains about the QUPER model. As the methods are mainly dealing with the functional requirements, there are very few methods for dealing with the quality requirements. Among them QUPER model is one. This paper is also the base for the article “A case study evaluation of the guideline-supported QUPER model for elicitation of quality requirements”.

The research is very little and the guidelines is also not provided. This paper fails to consider the interdependencies between the requirements. There should be a need to evaluate this model in many industries as this is the base.

Article 4: “A market-driven requirements engineering process: results from an industrial process improvement programme”

This paper proposes a method REPEAT for handling requirements in the context of release planning. This method clearly explains the processing of requirements from initial stage i.e requirement creation to the final stage delivery. This method is very helpful for industries when dealing with the continuous requirements during the release planning. It is very easy to understand and steps provided in this paper to implement this method are also clear.

The articles mainly concentrated on the methods for handling requirements in the case of release planning. Some data from the articles is helpful for answering the seminar discussion questions. Some of the articles[1]–[3] read for answering the seminar questions from web also helped for answering the seminar discussion questions related to the requirements.

References:

- [1] S. Konrad and M. Gall, “Requirements Engineering in the Development of Large-Scale Systems.”
- [2] G. Sabaliauskaite, A. Loconsole, and E. Engström, “Challenges in Aligning Requirements Engineering and Verification in a Large-Scale Industrial Context.”
- [3] K. Wnuk, “Understanding and Supporting Large-scale Requirements Management,” 2010.

Seminar 1:

Seminar Question:

GAP Analysis:

It is a simple tool helps in identifying the gap between the current situation and the future desired state, along with the tasks/factors needed to complete in order to close this gap. It contains a list of characteristic factors (attributes, competencies, performance levels) of the present situation and a list of factors need to achieve the future objectives which help in identifying the gap that need to be fulfilled. Every gap analysis template contains few essential components they are:

- State Descriptions: This is the first step in the gap analysis, helps in identifying the current and future desired state.
 - Current State: every gap analysis starts with introspection and in this state all the factors to be improved.
 - Future State: It represents the ideal condition that the organization to be in.
- Bridging the gap: helps in identifying and describing the gap before finding the ways to fix it.
 - Gap Identification: helps in identifying whether a gap exists between the current and future state.

- Gap Description: It records all the elements that make up the gap between the current and future state. The description should be consistent with the current and future states.
- Factors and Remedies: This helps in identifying the factors responsible for the difference between the current and future performance. This data helps in finding the solutions and action plans to handle the performance gap.
 - Factors responsible for gap: the identified list of factors should be specific, objective and relevant.
 - Remedies, Actions and Proposals: This is the last step in the gap analysis, identifying all the possible solutions for bridging the gap between the current and ideal state. The remedies should be action oriented and specific.

This is an effective tool for analyzing and understanding the organizations. It is useful at the beginning of the project and when identifying the tasks needed to be completed. Some of the benefits of gap analysis are

- Effective gap analysis increases the performance and production of an organization.
- It also measures the amount of time, money and resources needed to fulfil an organizations potential and reach the desired state and helps in cost reduction.

CVA Analysis:

Customer value has recently become a primary focus among many strategy researchers and practitioners as an essential element of a firm's competitive strategy. Many firms are engaged in some form of customer value analysis (CVA). Customer value analysis is an innovative research technique that assesses how your organization is viewed relative to others in the marketplace. It involves a structural analysis of the antecedent factors of perceived value (i.e. perceived quality and perceived price) to assess their relative importance in the perceptions of their buyers[4]. It includes opinions of both customers and competition's customers to assess the product's relative performance and provide more accurate and useable information for planning and strategic positioning purposes. It concentrates on the following steps for conducting CVA Analysis:

- What your customers want
- Which suppliers are performing well or poorly against these wants
- Whether or not you offer customers good value relative to your competitors
- How much your products are really worth
- What improvements to your product would be worth the most customers
- How to set prices to be competitive and to capture the full worth of your product
- The role of marketing communications in shifting how customers view your brand.

IVA Analysis:

It is defined as a technique in which the ongoing product status is compared with the predefined product strategies and checks whether the product is in line or not by considering the limited resources such as cost, time, risk and knowledge[5].

References:

- [4] W. S. DeSarbo, P. Ebbes, D. K. H. Fong, and C. C. Snow, "Revisiting customer value analysis in a heterogeneous market," *J. Model. Manag.*, vol. 5, no. 1, pp. 8–24, 2010.
- [5] T. Gorschek and A. M. Davis, "Requirements engineering: In search of the dependent variables," *Inf. Softw. Technol.*, vol. 50, no. 1–2, pp. 67–75, 2008.

Articles:

Quality Requirements in Industrial Practice - An Extended Interview Study at Eleven Companies

The role of the software increases and becomes a substantial part of industrial and consumer products. In order to create a successful software product and assure its quality, it is not enough to fulfill the functional requirements but also to find the right balance among competing quality requirements. Quality requirements play an important role in the software product development and the ability to develop a software product that meets customer's requirements and offer high value for both the organization and the customer increases the likelihood of market success and also acts as a competitive advantage.

Despite their importance, they are poorly understood and generally stated informally in a non-quantifiable manner and are difficult to validate. Along with these, the improper handling of the quality requirements leads to the expensive software products and increased time-to-market which in turn leads to the dissatisfaction of the customers.

In the case of market-driven requirements engineering, as the flow of requirements is continuous from multiple stakeholders and the problem of handling Quality requirements is even more complex. It is very important to deal with these quality requirements for the success of the software product and also to gain competitive advantage. The authors of this paper mainly focused on elicitation, analysis and negotiation, management and handling of quality requirements in the industry. Along with this, it also concentrates on the comparison between the two company categories Business-to-business and Business-to-consumer. The main purpose of this study is to gain an in-depth understanding of Quality requirements in the market-driven context.

In order to achieve this, an interview study is performed to identify specific challenges associated with their selection, trade off and management of Quality requirements in industrial practice. The data is collected from 11 product managers and 11 project leaders from 11 software companies. The results obtained from these interviews are as follows:

A total of 23 quality requirements are identified and asked to choose the top five most important quality requirements. The top most important requirements are as follows:

- In general usability, performance, reliability and stability are considered as top four
- For B2B companies' safety, performance and reliability are considered as the top three.
- For B2C companies' usability, performance and stability are considered as the top three.

A set of six different interdependency types are identified in the industry based on the interviews conducted at two different companies. They are AND, OR, REQUIRES, TEMPORAL, CVALUE and ICOST. Along with this one extra interdependency type was identified TXOR. The findings are based on the interdependency types i.e. QR to QR and QR to FR and type of company B2C and B2B.

- The common interdependency types identified are as follows:

- For QR to QR in general, most common were REQUIRES, CVALUE and ICOST and the least was TEMPORAL. For B2B the most common is REQUIRES. For B2C the most commonly used is CVALUE.
- For QR to FR in general, TEMPORAL was considered as the most common and OR was considered as least common. For B2B TEMPORAL as most common and OR and AND as least common. For B2C REQUIRES as most common and OR as the least common.
- The most important interdependency types are identified as follows:
 - For QR to QR in general, REQUIRES. For B2B REQUIRES and ICOST as most important. For B2C REQUIRES and CVALUE as most important
 - For QR to FR in general, REQUIRES. For B2B TEMPORAL as most important. For B2C REQUIRES as most important.
- Interdependencies can have a critical impact on product development in terms of planning, design, and quality. But the identification of dependencies is a complex task and the potential number of dependencies may be very large. Therefore, understanding of which interdependency types should be considered is the most important task.

The findings for the cost estimations of quality requirements are as follows:

- During the cost estimation, there is no distinction between functional and quality requirements.
- Expert opinion is the predominant method for the cost estimation.
- B2B will have more inaccurate cost estimates compared to B2C in the worst case scenario.

From the findings the main reasons for the dismissal of the quality requirements are as follows:

- Close to 1 out of 5 all the Quality requirements are dismissed from the projects with little or no consequence analysis is performed.
- For B2C, performance requirements are more often dismissed because of the improper estimates. For B2B, quality requirements with least importance are dismissed i.e. Usability.
- Poor cost estimations and quality requirements have lower priority than functional requirements.
- Lack of resources.

Dismissal of quality requirements leads to the decrease in the value of the product in the competitive market. The dismissal rate should be lowered by improving the specification and the quantification of the most challenging ones. It is easier to go on instead of dismissing them. In general, the results obtained from the interviews indicate there might be different in the relation to the type of the company. In order to make a successful software product these quality requirements should be selected, managed, and handled carefully by the members of the industry.

Reflections:

“Requirements Abstraction Model”

This paper proposes a model called requirements abstraction model. This model helps in breaking down the requirements from higher level to low level. As the requirements obtained from the market will be of different types. This model is very helpful for understanding the requirement clearly before implementing by breaking into low level. I have implemented this model and it is easy to implement and understand. This model is very useful in the industries and evaluated

practically and the results obtained are positive. It would be difficult for handling large number of requirements manually as it involves four levels for each requirement. It is also helpful if tool to implement this model.

“A method for early requirements triage and selection utilizing product strategies”

This paper presents a method called MERTS and it provides a clear view for achieving consensus and for understanding the product strategy. It is helpful for the organizations as the strategic and technical product managers can easily discuss the requirements and they can decide the requirements selection. This method makes the selection process easy and further research to be done.

“Requirements Engineering. In search of dependent variables”

This paper presents the importance of requirements engineering and their impact on several quality levels. The levels explained in this paper is very clear explaining the effect of requirements on the project, product, company, and society. The requirements should be selected keeping these effects in mind and along with the changes from present to future.

“Quality Requirements in Industrial Practice – an extended interview study at eleven companies”

This paper mainly concentrates on the quality requirements and addresses the important quality attributes. It also provides the reasons and problems during the handling and selection of quality requirements along with the interdependencies among the quality requirements. The results presented in this paper are acceptable and because of the delivery time many organizations are not considering these quality requirements regardless of their importance in gaining the competitive advantages.

“A cost-value approach for prioritizing requirements”

This paper mainly provides information about the cost – value approach for prioritizing the requirements. The method proposed in this paper is clear and easily understandable. I have implemented this method for my assignment. This method is very flexible when there is need to identify the requirements which are needed to be implemented first. But the process involved with this method is very complex when compared with other methods. This method fails to consider the interdependencies as there is a need for the consideration of interdependencies in the context of release planning.

For answer the research discussion questions mainly about the release planning the article “The art and science of release planning” from seminar 0 helped a lot. It provides a clear idea about the release planning as it mainly deals with the release planning. For answering the discussion questions related to the large scale requirements are done with the help of articles[1]–[3] read for seminar questions in seminar 0. As the articles mainly deals with the large scale requirement engineering challenges and the context.

Seminar 2:

Seminar Questions:

How do you connect your requirements to your architecture?

Software Architecture is mainly concerned with the components of the system and their interactions. The software architecture becomes an essential step in the software development. According to the author[6] the good architecture is described as when the system is implemented according to the architecture it meets all the requirements and the resource budgets. The requirements are taken as an input for designing the software architecture and these requirements contain both non-functional and functional requirements. The organizations should look into software architecture which helps in achieving some of the benefits like higher productivity, higher adaptability and better code maintainability. This architecture enables the stakeholders to understand the system and communicate with each other.

Some methods available for designing architecture from the requirements are as follows

- Pattern-based architecture design: provides significant semantic context about concerns, relationship of the system with similar other systems, expected path evolution and overall computational paradigm. Based on their scales they are divided into two types design patterns and architecture styles.
- Multi-view model: address the different aspects of the users with different views.
- Evaluation and Transformation based design: addresses the functional requirements first and later followed by non-functional requirements by iterative evaluations and transformations.

Along with these models, there are some other models available based on the requirements type for connecting with the software architecture. Unified Modelling Language diagrams are also used.

Can you connect all requirements directly? What do you do if you cannot?

According to the author[6] the software architecture is concerned with the components of the system and their interactions. Along with this, it is also concerned with the high level structure, in contrast with the detailed software design. According to the author[7] some of the requirements are directly connected to the architecture while some others are not. The requirements which are clearly specified and at architecture level are connected directly with the architecture. The requirements which are complex, not detailed enough and not at architecture level are not possible to connect with the architecture and they are not considered. In order to connect these requirements with the architecture they should be broken down to the architecture level i.e. by providing detailed information for the requirements.

References:

- [6] L. Liao, "From requirements to architecture: The state of the art in software architecture design," *Citeseer*, pp. 1–13, 2002.
- [7] R. C. de Boer and H. van Vliet, "On the similarity between requirements and architecture," *J. Syst. Softw.*, vol. 82, no. 3, pp. 544–550, 2009.

Articles:

Scaled Agile Framework V4.0

SAFe is an online, freely revealed knowledge base of proven success patterns for implementing lean-agile software and systems at enterprise scale. Many large and small scale enterprises reported, after applying SAFe they have obtained many business benefits such as:

- Increase in productivity
- Increase in quality
- Faster time to market
- Increase in the employment engagement and job satisfaction.

The Scaled agile framework by Dean Leffingwell, SAFe 4.0 supports both the software and system development. It contains four levels and one layer they are Portfolio level, Value stream level, Program level, Team level and Foundation layer.

Foundation Layer:

In order to get the desired results, the foundation principles and values must be understood, exhibited and upheld. SAFe stand on the foundations of its core values, Lean-agile mindset, lean-agile principles, implementation patterns and lean-agile leadership principles.

Portfolio Level:

It provides the basic constructs for organizing the lean-agile enterprise around the flow of one or more value streams. Each value stream develops the systems and solutions necessary to meet the strategic intent. This portfolio also provides the basic budgeting and other governance mechanisms and other governance mechanisms. It also ensures the investment in these value streams are returned. It includes the following

- strategic themes that guide the portfolio to the larger, and changing business objectives.
- Portfolio management represents the stakeholders who are accountable for delivering the business results.
- Enterprise provides guidance on how to support multiple portfolios in the larger enterprise.
- One portfolio Kanban system for both business and enabler epics.

Value stream Level:

It is intended for builders of large and complex solutions, solutions that require multiple ART's as well as the contribution of suppliers. This level for those building the world's largest software and system. It includes

- Economic Framework -helps in building the governance with value streams.
- Managing fixed and variable solution intent with MBSE, set based and agile architecture.
- Spanning palette for roles and artifacts.
- Kanban backlog and epics.
- Value stream cadence and synchronization across trains.
- solution context which provides the customer focus(value).

Program Level:

It describes the teams, roles and activities are organized around agile release train metaphor, a team of agile teams that deliver a continuous flow of incremental releases. Each agile

release train is along lived self-organizing team of agile teams, a virtual organization (5-12 team members) that plans and commits and executes together. The ART aligns teams to a common mission via single vision, roadmap and product backlog. Each train has dedicated people and resources necessary for every iteration. The roles involved are RTE, system engineer and project management. Trains provide architectural, engineering and user experience guidance to help teams build systems that support current and upcoming user and business needs.

Team level:

SAFe teams are self-organizing, self-managing and cross functional teams. The SAFe agile team is a group of individuals who have the ability and authority to build and test solution in a short iteration time box. In order to manage their work these teams can apply scrum or Kanban methods. Most SAFe teams use scrum as their primary team based project management framework. Scrum includes the following roles namely

- Product owner: responsible for defining user stories and prioritizing the product backlog. The product owner also plays a key role in quality and is the only team member to accept user stories.
- Scrum master: responsible in helping the members to achieve the goals.
- Development team

In the SAFe, scrum architecture contains product backlog and short iterations(releases) for every two weeks. The user stories in the backlog are prioritized by the product owner and moved to iteration for the development. At the end of each iteration a release is delivered. The products are delivered with high quality by using practices such as continuous integration, test-first and refactoring.

Reflections:

“Towards a reference framework for software product management”

This article provides a clear information and structure of the software product management. The internal processes that are carried out during the software product development and the tasks that are involved in each process (Road mapping, release planning, requirements management, portfolio management) are clearly understood. This provides me a clear overview of the management process.

“Market-Driven Requirements Engineering for Software Products”

This paper mainly focuses on the market drive requirements engineering process. It clearly provides the information about the release planning, road mapping and the process involved in the requirements, Challenges faced in the MDRE process. The provided information gives a clear view MDRE for software requirements.

“Are You Biting Off More Than You Can Chew? A Case Study on Causes and Effects of Over scoping in Large-Scale Software Engineering”

The authors of this paper find out the causes and effects of over scoping in large scale industries, the over scoping cannot be eradicated even by following the agile practices so there needs to be some management for scope. The presented study helps the organizations, but these findings are related to one area further research should be done in order to generalize the results.

“Factors Affecting Decision Outcome and Lead-time in Large-Scale Requirements Engineering”

The paper mainly concentrates on the factors effecting the decision outcome and lead time. The results obtained in this paper are mixed results. The results are presented based on the survey and case study both of them varies in their findings related to the factors. The research should be done further for better understanding of the factors effecting.

“An industrial survey of requirements interdependencies in software product release planning”

This paper concentrates on the requirements interdependencies in the software release planning. The provided information will be very helpful for the industries in handling the requirements selection for the releases. The interdependency types and the coupling between the requirements will help the organizations to select or not. As mentioned in the paper, there should be further research on this in order to use effectively.

For answering the seminar discussion questions, related to road mapping the article “Market-Driven Requirements Engineering for Software Products” helped a lot. It provides a clear idea of release planning and roadmap process. For the question portfolio management, the blog “scale agile framework” and the article “Towards a reference framework for software product management” helped a lot in getting an idea about it. Further searched from web in order to gain a clear idea.

Seminar 3:

Article:

What Happened to Our Features? Visualization and Understanding of Scope Change Dynamics in a Large-Scale Industrial Setting

When developing a software platform for the product lines, deciding on which features to include into the scope of the upcoming project is not an easy task. It is not possible to implement all the features in the same release, some features are to be postponed to upcoming projects or next release. The process of selecting features which are to be implemented in the current release, which are to be postponed for the next release is called scoping. This scoping plays a key role in achieving the economic benefits in the product line development. The features that are to be implemented depend upon the factors such as changing markets and evolving technologies. Effective scoping thus requires continuous assessment of how changes in the domain impact scoping decisions. Decisions may have to be changed as circumstances change, resulting in a dynamic evolution of the scope of software asset investments. As many changes in the scoping process for product line projects are identified as one of the main problem of product line scoping. To solve this, issue the authors of this paper proposed a technique called Feature survival charts. This paper also contributes the implementation and evaluation of the proposed technique, a set of scoping measurements proposed, evaluated and analyzed.

The proposed method is evaluated by implemented in the three projects. All the selected projects are not at the same level and selected based on the following factors such as length, number of features considered in the scope, and the possibility to visualize and analyzing scope changes. The participants responsible for the evaluation is process manager, requirements manager, and key performance indicator. The company uses several milestones and toll gates to control the project progress. The projects are carried out with the help of the requirements team and design teams. The scope changes are clearly presented in this paper with different colors. The results also show as the time varies there is no increase in the scope in these three projects instead there is a decrease in the scope by reducing the features. The time related and feature related measurements are used to track the scope progress. The time-related scope measurements contains one measurement

- Number of positive and negative scope changes per time stamp/baseline

The feature -related scope measurements includes the following measurements. They are

- Time to feature removal
- Number of state changes per feature
- Time to birth
- Reason for scoping decision

These time and feature related measurements are used to track the scope changes. These measurements are evaluated empirically with the data obtained from the cases and also analyzed theoretically. The proposed measurements are ranked by the three practitioners working with the scoping. The most useful measurement ranked by these practitioners is Time to remove a feature and reason for scoping decisions and the least useful measurement is number of states changed per feature. Along with this, the reasons for the de-scoping are also analyzed. The some of the reasons that are identified are stakeholders' business decision, lack of resources and portfolio changes are main. The projects are compared using the charts. The proposed technique and measurements are very helpful for the identifying the scoping changes. The graphs also helped in identifying the features and the time frames responsible for the scoping issues related to the uncertainties in the estimations and charts are very helpful for finding the instabilities in the scoping process. From the evaluation, it is also known that charts can effectively focus investigations of reasons behind scoping decisions, valuable for future process improvements. The results obtained are positive and the technique is improved further in future by concentrating on optimizing the usability.

Reflections:

“What Happened to Our Features? Visualization and Understanding of Scope Change Dynamics in a Large-Scale Industrial Setting”

This paper mainly focuses on the scope changes in a large scale. Feature survival charts are proposed in this paper to handle and identify the scope changes in the project along with this a set of measurements are also proposed and evaluated. This method helps in understanding the scope changes clearly and it would be very useful in tracking the scope changes. This technique should be improved further for better usability.

“Obsolete Software Requirements”

This paper mainly concentrates on the obsolete requirements. The requirements which are remained in the databases because of the continuous changes in requirements. The need for identifying these requirements and reasons for obsolete requirements are mentioned. The findings

from this paper are acceptable and helps in understanding the reasons and ways for handling the obsolete requirements. There is a need to identify these requirements because absence of identifying these requirements lead to ineffective project management and increase in the time and cost of the project. The results obtained are mixed results the research should be done to one area for better results and better understanding.

For answering the research questions related to requirements and architecture decisions and into the projects the articles read from the seminar 2 in order to complete the seminar questions related to the architecture connection helped a lot for getting idea about these questions. The article “On the Similarity between Requirements and Architecture” and “From Requirements to Architecture: The State of the Art in Software Architecture Design” and the “Scaled agile framework” further from web.

Seminar 4

Reflections:

“Assessing challenges of continuous integration in the context of software requirements breakdown: a case study”

This paper mainly focuses on the importance of continuous integration and the challenges faced during the requirements breakdown, continuous integration and in the context of the challenges faced for requirements breakdown at the time of continuous integration. As continuous integration includes the several commits which helps in improving the quality and the process of the development. The addressed challenges will be helpful for the organizations dealing with CI. These challenges are addressed from one industry and based on their structure need for further investigations to get generalized results.

For answering the discussion questions related to horizontal scaling the article “What Happened to Our Features? Visualization and Understanding of Scope Change Dynamics in a Large-Scale Industrial Setting” provided a clear idea, along with this some other articles from previous seminars related to scope provided a basic idea about scoping and for remaining questions from web. For the discussion question quality requirements several papers from previous seminars provided an idea about the quality requirements and their urge “Quality Requirements in Industrial Practice – an extended interview study at eleven companies”.