SEMINAR NOTES

Pakki Neetika
931110-0524

Section 1: *Summarising the articles and rectifying the mistakes that were identified:*

"Requirement Abstract model"

The authors of this paper discuss about the problems faced while handling of the requirements and comparing them with each other in market driven requirements engineering. In order to solve this problem, the authors came up with a model (RAM) which involves four abstractions level that allows placement of requirements on different levels and supports abstraction or break down of requirements to make them comparable to each other. Requirement abstract model (RAM) mainly focuses on the product perspective, supporting a continuous requirement engineering effort, aimed at taking requirements of multiple types (abstraction level) as input, and offers a structure for the work-up of these requirements, i.e., breaking down abstract requirements into detailed ones and vice versa.
This model involves three steps.
Specify:
This involves specifying the initial requirements and eliciting information to an extent of it being understood by the project manager performing continues requirements engineering. This step includes four attributes;
a. Description: this attribute describes the centre essence of the requirements.
b. Reason/ benefits/ rationale: this attributes consist of 2 parts. Why the requirements are specified and benefits of the requirements specified.
c. Risk: this attributes describes the risks with the requirements that are not specified.
d. Title: this attribute should reflect the content of the requirement.
Place:
This step analyse the requirements and place them in the level they belong. There are four levels: product level, feature level, functional level and the component level. In product level, the main focus will be on the goals, product strategies and indirectly on the organization strategies. Requirements which are features of the product are placed at the feature level. Features are characteristics of the system; requirements at this level provide abstract descriptions of these characteristics. At the functional level each requirement is described in such a way that it clearly shows what a user or system can do. In the component level the Requirements will have detailed information. Component Level can also act as clarification to one or more Function Level requirements.
Abstraction:
This step involves abstracting and/or breakdown of a requirement, depending on the initial placement of the original requirement. New requirements are created on adjacent abstraction levels or linking to already existing one, depending on the situation. Every requirement has to be abstracted to the product level. For this 2 rules are set up.
R1. No requirement may exist without having a connection to the Product Level.
R2. All requirements have to be broken down to Function Level.
The authors of this paper also discuss about the evolutionary development of RAM as it was validated in industry against an organization faced with the improvement issues. There are two types of validation.
Static validation: The static validation consists of reviews and walkthroughs of RAM
Dynamic validation: the dynamic validation consists of a live industry requirements engineering effort where the model was put through its phases.
The author's claims that further work is done on the validation and evolution of the RAM method which includes gathering of data (qualitative and quantitative) from two organizations that supports the decision materials for further model improvements and also acts as input to the organizations when considering the full scale adoption of a tailored version of RAM in their product planning and development process. Furthermore RAM v.2.0. is implemented which is an usable and flexible light

weight model that covers requirement engineering activities performed not only during product planning and management activities, but also within and post projects.

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

The authors of this paper mainly focus on the importance of handling the quality requirements along with the functional requirements and also the challenges that are faced by the organizations while managing the quality requirements. During software product development, the non functional requirements (quality requirements) play a critical role and improper handling of quality requirements may lead to several challenges such as dissatisfaction of customers, more expensive software products and increased time to market. But handling quality requirement is a very challenging task and so these requirements are poor understood. Mainly in market driven development, where the situation of handling the quality requirements is even more complex due to the large number of requirements stemming from multiple internal and external sources and the continuous flow into the development organization.
The authors mainly focus on the importance and analysis of QR in different companies. For this an interview study is performed to identify specific challenges associated with their selection, trade off and management of QR in industrial practice. The data is collected by carrying out a semi structured interview which is conducted to the one project manager and one product leader each from 11 different industries that are selected by the gate keeper of corresponding industries that examines QR in practice. The collected data also includes the perspective of comparing B2B and B2C analysis which presents new data from all 11 cases. 4 RQ's where formed and analysis of QR in related to each of the RQ is done by interviewing the 22 participators of 11 companies from various perspectives. The main objectives of this study are as follow
To find out what QR's are considered most important and are there any distinguishably characteristics in relation to costumer type (B2B or B2C)?
To find out what interdependencies between QR are present in the companies?
To investigate how estimations of QR cost is performed, and what is the accuracy of these cost estimates?
To find out to what extent are QR dismissed from the project after project initiation?
The results from the interview study done on the objectives are as follow.
Not all QR's are given the same importance in all the companies. Each requirement is given different priorities based on the type of company and its specifications. For B2B usability is considered most important and for B2C safety is considered most important.
A set of six predefined interdependency types were used by the interviewees to characterize perceived interdependencies. REQUIRES and CVALUE are considered as the most common and important interdependency types to identify. B2B viewed REQUIRES as the most common and important one, B2C considered CAVLUE. The order of implementation is affected by interdependencies and the delays or exclusions of requirements may be the result of QR interdependencies, which also affects what is implemented.
There is no distinction between functional requirements and quality requirements during cost estimation.
Expert opinion is the predominant method for estimation.
In the worst case, B2B has much more inaccurate estimates than B2C.
All QR are dismissed from the projects at some stage during development, with little or no consequence analysis performed
For B2C, performance requirements are more often dismissed due to the difficulties in proper estimation and for B2B, quality requirements are not considered important
Poor cost estimation and the fact that QR has lower priority than FR are the main reason for dismissal.
The authors of the paper conclude by stating that it is important for them to understand and accept the fact that the dismissal of the QR may solve a short term problem but on a long term basis it will reduce the competitive advantages and the value of the system. However the main problem is that QR are not taken into consideration during product planning and thus not included as hard requirements in the projects, making the realization of QR a reactive rather than a proactive effort. Product management may thus not be able to plan and rely on QR to achieve competitive advantages.

Conclusion of "A method for early requirements triage and selection utilizing product strategies":

MERTS is intended to aid product managers in performing requirements triage effectively and efficiently in a repeatable manner providing traceable decisions. The only drawback is that MERTS seems to be more resource intensive to use, although per correct answer we think that MERTS is at least as efficient as the NL option. The benefit of MERTS is the ability to document the strategies in a way that offers explicit decision support for all decision makers when performing requirements triage.

"Towards a Reference Framework for Software Product Management"

The authors of this paper have developed a framework that provides the structure to a body of knowledge for software product management. Taking this into consideration the authors have developed a reference framework based on overview of state-of-the-art literature and industrial case studies. This framework explains the process carried out by the product manager during the software product management and also helps in identifying the stakeholders for the product and their relation during the software product management.
This framework divides the software product management into four parts:
Portfolio management: It deals with the products in the product portfolio.
Product release planning: It is used to identify the different releases of a product
Product road mapping: It deals with the set of requirements that are to be released.
Requirements management: It deals with the content of each individual requirement A clear explanation is not provided on how a software product management process is carried out entirely by an organization and the structure of the software product management..
These four processes are carried out as follow:
Portfolio management process involves four tasks
• Partnering and contracting
• Market trend identification
• Product life cycle management
• Product line identification
The input for this process helps in making the decision about the development strategy and lifecycle. A software product line is identified in this process which serves as an input for product road mapping.
Product road mapping process involves these tasks
• Theme identification
• Core asset identification
• Road map construction
This process receives input as product lines from portfolio management. This helps in identifying the themes and core assets. This roadmap also acts as an input to the requirements organization.
Requirements management process involves these three tasks
• Requirements gathering
• Requirements identification
• Requirements organizing.
In this process the inputs are gathered from different stakeholders and were indentified. The product requirements are grouped according to the product and core asset.
Product release planning mainly involves these five tasks.
• Requirements prioritization
• Requirements selection
• Release definition
• Release verification
• Launch application
Takes the input from the requirements management process where some of the prioritized requirements are selected. For these requirements, a release is prepared and different internal stakeholder validates it and sent it to the board for approval. Once the approval is received, the implementation process starts and the launch preparation package is delivered to the stakeholders. As

this framework also helps in identifying the stakeholders, some of the stakeholders that are identified and mentioned in this framework are

• External Stakeholders
o Market
o Partner companies
o Customers
• Internal stakeholders
o Development
o Support
o Service
o Research and innovation

This paper gives an idea and a clear understanding about what a software product management is by explaining each and every step involved in the software product management clearly.

"Scaled Agile Framework (SAF)"

SAF version 4.0 is a major milestone for the framework. This version incorporates learning from all prior releases into a single, scalable and more modular framework. This framework supports both software and system development, from the modest scale of under 100 practitioners to Solutions that require thousands of people to create and maintain. This framework consists of 1 layer and 4 levels.
Foundation layer: this layers acts as guidance for Lean –Agile Leaders, Communities of practice, core values and implementation. And the four levels are;

- Portfolio level: Guidance for strategy formulation and portfolio communication, organizing and funding value streams and cross value stream coordination. this is the highest level of concern in SAFe. This level provides the basic constructs for organizing the Lean –Agile Enterprise around the value via one or more Value stream, each of develops the systems and solutions necessary to meet the strategic intent.
  This layer contains the following:

1. Enterprise- Enterprise needs to fulfil some element of the business strategy.
2. Strategic themes- These themes are specific, itemized business objectives that connect a SAFe portfolio to the evolving enterprise business strategy.
3. Epic owners- They have the responsibilities begin early in the life cycle of the epic.
4. Enterprise Architect- They work with the business stakeholders and solution and system architects to drive holistic technology implementation across value streams.
5. Value streams - It is a long lived series of steps used to deliver value, from concept or customer order to delivery of a tangible result for the customer.

- Value stream level: Guidance for those building the world's largest software and system. This level is organized around program increments, which are synchronized across all the agile release trains in the value stream. It provides for cadence and synchronization of multiple ARTs and suppliers and the Solution Demo. There are three key roles to coordinate and advance the value stream;

1. Execution and improvement
2. Content management
3. Technical excellence

- Program level: this is the level where development teams and other resources are applied to some important, ongoing development mission. It consists of description of agile release trains, teams of agile teams that build solution Capabilities and subsystem. SAFe program level teams, roles and activities are organised around the Agile Release Trains (ART) metaphor, a team of Agile Teams that delivers a continuous flow of incremental releases of value. There are three functions that helps in ensuring successful execution of the version and roadmap initiatives at the program level:

1. Program Execution
2. Content Management
3. Technology

- Team level: The team levels describe how Agile Teams power the ART. They apply SAFe Scrum XP or Team Kanban, along with the built in quality practice that helps ensure a quality end product. SAFe teams are self organizing, self managing and cross functional teams. These teams deliver valuable, tested, and working software and systems every two weeks. This level provides an organization, artefact, role, and process model for the activities of Agile Teams.

"Market-Driven Requirements Engineering for Software Products"

This paper mainly discusses the overview of the special characteristics of market-driven requirements engineering and describes the most important challenges of the area. The main objective of market-driven development is to deliver the right product at the right time. MDRE also covers the specific activities needed in a market-driven context, such as release management and market analysis. In this paper the specific challenges of requirements engineering in a market driven software development context is discussed. The MDRE case is more focused on prioritization, cost estimation and release planning, and these activities are all conducted by the developing organization. Some of the important characteristics are listed below.
a. The developing organization makes all decisions but also takes all risks.
b. There is a continuous flow of requirements throughout the product lifetime.
c. The requirements volume is potentially very large and continuously growing.
d. A majority of the requirements are informally described.
e. The product is evolving continuously and delivered in multiple releases.
f. Release planning focuses on time-to-market and return-on-investment.
There are some RE challenges in market-driven software development

- Balancing market pull and technology push.
- Chasm between marketing and development.
- Organizational instability and market turbulence.
- Simple tools for basic needs.

The key factor for a market driven company is to continuously improve in managing these challenges in such a way that it stays ahead of competitors. When designing an MDRE process for a specific company, it is important to realize that there are many factors that determine what the best concrete process implementation is.
Process quality issue in MDRE is the quality of decisions that are made about produced artefacts. The selection is carried out by the Alfa/beta model of MDRE selection quality. An Alfa requirement is defined as a requirement with high quality that it is ideally selected. Beta requirements are defined as the requirements that should be rejected because of its low quality. The main challenge in MDRE process is to find and select the alfa requirements and reject beta requirements. In MDRE data management, mainly contains two ingredients
1. Requirements state model: this model used for progress tracking of requirements refinement. Requirements are received at any time, but the development of a product is made in releases that are produced at discrete points in time. Two modes are identified.
a) Continuous mode: in this mode, requirements are received and registered by the product manager.
b) Release mode: The development of product releases is initiated at designated times according to the roadmap planning and the requirements management activities are in release mode.
There are statuses such as candidate, approved, specified, discarded, planned, developed, verified and released in order to monitor the progress on the requirements.
2. Requirements repository: requirements repository where relevant attributes of candidate requirements are stored.
Roadmap and release planning are one of the important factors to be considered in the MDRE context. A roadmap is a document that provides a layout of the product releases to come over a time frame of three to five years. Taxonomy is established that classifies roadmaps according to their location in an applications-objectives space. This taxonomy scheme classifies the roadmaps broadly into the following four categories:

a. Science and Technology Roadmaps
b. Industry Technology Roadmaps
c. Corporate or Product-Technology Roadmaps
d. Product or Portfolio Management Roadmaps
The three major uses of road mapping are:
a. Development of a consensus about a set of needs and the technologies required to satisfy the needs.
b. Provision of a mechanism to help experts forecast technology developments in target areas.
c. A framework to plan and coordinate developments either within an organization or in an entire industry.
The structure of the road mapping process consists of four phases to stratify from strategy making to the development of the software product.
Phase 1: Initiation Phase:
Phase 2: Preparation Phase
Phase 3: Finalization Phase
Phase 4: Follow-up Phase
Later in this paper an example Road mapping at Baan is considered. In the conclusion the author as stated that in MDRE context, there are so many challenges observed in the areas like prioritization, management of dependencies and tool support for handling large scale requirements.

"What Happened to Our Features? Visualization and Understanding of Scope Change Dynamics in a Large-Scale Industrial Setting"
Results: The authors of this paper has conducted an industrial case study in a large-scale setting where a technique called Feature Survival Charts for visualization of scoping change dynamics has been implemented and evaluated in three projects. These three different projects have started the work at different point of time. From the data collected from the projects, the results shows that the authors have experienced almost one scope per feature for each project which reflects the poor understanding of the scoping process. And also the result of a qualitative analysis from the graph shows that from all analysed projects, negative scoping is observed though out the analysed period.

Conclusion: After the results are used by the case company to adjust the process towards more flexibility in scope setting decisions, and a clearer scope responsibility, the authors have stated that solution of the technique has confirmed to outperform the previously used table-based textual method to track the scope changes in the case company which gives a better overview of the scoping process of the whole project on a single page size graph. The studies shows that the visualisation technique can be applied to large scale projects, which demonstrates the scalability of the method and it can be implemented as a standard practice and is currently in widespread usage at the case company and it is believed that the manner in which the graphs together with measurements are used to increase the understanding of the performance of the scoping process is generally applicable.

Section 2: *Homework questions*

Seminar 0 (q)
Question 2
There are many challenges that are identified in LSRE. Few of the challenges are listed below:
1. Complex dependencies between requirements: With the increase in the number of requirements, their interdependencies will also increase. Handling and managing a large number of requirements along with their dependencies is a complex task.
2. Cost estimation: as the inflow of the requirements in the case of large scale is continuous, it is difficult to estimate the accurate cost.
3. Requirements prioritization: with the inflow of the requirements, there will be changes in the priorities that is given to the requirements.
4. Finding right balance between selecting commercial requirements and internal quality requirements.

5. Uncertain environments: this includes both internal and external elements. External uncertainty includes changes in the market, the operating environments, business processes, and threats. Internal uncertainties include program/project execution as well as design, implementation, and performance challenges.
6. Resource allocation: It is very challenging to allocate resources for such a large pool of requirements. Inappropriate allocation of resources to any requirement leads to many problems.
7. Handling of multiple customer or the stockholders: it is difficult to satisfy the needs of all the stakeholders and establish proper communication must be established among them since the field of large scale requirement engineering covers a vast area with huge number of stockholders involvement.

Question 3
The order of magnitude in RE is based on the size of the sets of requirement that are handled by the organization in developing a software intensive system [1]. The order of magnitude also relieves the effort that is required in managing the complete sets of interdependencies of the requirements.
The order to magnitude in RE are categorised in 4 levels.
Level 1 - Small scale requirement engineering (SSRE)
The order of magnitude (or) the number of requirements the organisation can handle is: 10
Effort required: less effort
Level 2 – Medium scale requirement engineering (MSRE)
The order of magnitude (or) the number of requirements the organisation can handle is: 100
Effort required: managing the set of requirements at this level is feasible but requires large effort.
Level 3 – Large scale requirement engineering (LSRE)
The order of magnitude (or) the number of requirements the organisation can handle is: 1000
Effort required: practically unfeasible but feasible among small bundles of requirements.
Level 4 – Very large scale requirement engineering (MSRE)
The order of magnitude (or) the number of requirements the organisation can handle is: 10000
Effort required: Managing a complete set of interdependencies among small bundles of requirements is unfeasible in practice.

Levels of requirement engineering	No. of requirements
Small scale requirement engineering	~10
Medium scale requirement engineering	~100
Large scale requirement engineering	~1000
Very large scale requirement engineering	~10000

Seminar 1 (q)
Question 1: GAP/CVA/IVA
Gap analysis:
This analysis mainly focuses on what the business is doing currently and where it wants to reach. This analysis is carried out by identifying the gaps by the process of comparing the actual performance with the desire performance. In order to perform the gap analysis, it is important to understand the expectation level of the performance in a company so that comparison of the current performance level with the expected performance level is done.
This analysis can be performed at the strategic level and the operational level.
It provides the company/ organisation with a foundation for measuring investment of time/ money/ human resources required to reach the desire goal.
This analysis can be used as a ranking of good, average and poor.

Planning gap is the process of comparing the forecast profits to the desire profits of the company. Planning gap is divided into two main elements:
Usage gap: This is the gap between the total potential for the market and the current usage by the consumers in the market. Usage gap= market potential – existing usage
Product gap: A particular organisation is removed from the part of the market because of product or service characteristics.
Gap process includes the following steps:

- Identifying the existing process: it is important to know what process/ method the company/ organisation is using.
- Identifying the existing outcome: to identify the current status of the outcome that the company is producing.
- Identifying the desire outcome: to identify what the company/ organisation wants.
- Identifying the process to achieve the desire outcome: to identify the procedure or the process that the company/ organisation uses to achieve the desire outcome.
- Identify and document the gap: to find the difference between the current outcome and the desire outcome.
- Develop the means to fill the gap: develop and using the correct tools or techniques required to fill the gap and reach the desire goal.
- Develop and prioritise the requirements to bridge the gap.

Customer value analysis (CVA):
Customer value analysis is an innovative research technique that assesses how an organization is viewed relative to others in the marketplace. This analysis includes opinions of both customers and competitor's customers so that the product's relative performance can be assessed which helps to provide more accurate and usable information for planning and strategic positioning purposes. This analysis includes few activities that are performed by the companies. The activities are identify the attributes that matters to its customers and to the competitors' customers, show exactly how these attributes are defined by the customers, quantify the company's performance and the competitors' performance, show corresponding prepositions of different customers and what can be done according to this, provide a fact based, data driven system for making decisions, provide ways to overcome the competition and tracking ongoing progress.

Internal value analysis (IVA):
It is a technique to measure whether a product is following the proposed strategy by meeting the product line at every stage by considering limited resource such as cost, time, risk and knowledge. In other words it is a method which is used to measure whether a product in a company is in line with its strategies or not by taking limited resources and other products into account.
References [1] 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.
Technical product management: it is the process of collecting and using data on the products that a business or organization sells handles or makes. The term technical project management describes a person, not a role. It implies that it does not describe a product manager who needs to actually perform technical tasks (software architecting and coding), but they need to perform product management role in close coordination with the technical team. The activities to be performed by a technical manager include: 1. conducting technology assessments. 2. Analysing the competitive overlook. 3. Maintaining the product portfolio landscape. 4. Monitoring and organising industry innovations. 5. Writing product requirements and use case scenarios.

Question 4: Road mapping:
Road mapping: A roadmap is a strategic plan that provides the steps an organisation needs to take to achieve stated outcomes and goals. It provides vision to the links among tasks and priorities for action in the near, medium and long term. Metrics and milestone provided by the roadmap allow the organisation having a regular tracking of progress towards the goals. There are specific key elements of a successful roadmap:

A roadmap will contain a clear statement of desire outcome that follow a specific pathway for reaching the goal. The pathway contains the following:
Goals: A clear and concise set of targets that, if achieved, will result in the desired outcome; quantified goals.
Milestones: the interim performance targets for achieving the goals, pegged to specific dates.
Gaps and barriers: a list of any potential gaps in knowledge, technology limitations, market structural barriers, regulatory limitations, public acceptance or other barriers to achieving the goals and milestones.
Action items: actions that can be taken to overcome any gaps or barriers that stand in the way of achieving the goals; typical solution actions include technology development and deployment, development of regulations and standards, policy formulation, creation of financing mechanisms, and public engagement.
Priorities and timelines: a list of the most important actions that need to be taken in order to achieve the goals and the time frames, taking into account interconnections among those actions and stakeholder roles and relationships.
There are four phases involved in road mapping:
Phase 1: Planning and preparation: a complete understanding about the present situation of the company is done after which a plan is prepared based on the current situation.
Phase 2- Visioning Setting: in this phase setting a desire pathway for technology development is done.
Phase 3-Roadmap development: once the vision is set, certain activities will be performed in this phase. The activities are holding road map workshops, preparing the draft road map document, conducting a roadmap review.
Phase 4- Road map implementation and adjustment: in this phase launching the road map and putting in place tracking systems is done. Adjusting the road map according to the situations and monitoring the progress is also done.

Seminar 2 (q)
Question 1: BOSTON MATRIX:
Boston matrix that is developed by Boston Consulting Group (BCG) is a marketing tool that analysis and manages the product portfolio which helps the investor to decide on which product he/she has to invest.
How BCG is used to manage product portfolio?
A portfolio of products can be analysed using the Boston Group Consulting Matrix. This will classifies the products in four categories based in the market share and the market growth where market share answers the question "does the product being sold have a low or high market share?" and market value answers the question "are the numbers of potential customers in the market growing or not?" below are the four categories;
Dog: The products that are in dog category have low market share and low market growth. This means that these products do not consume nor will they generate much cost.
Cash Cow: the assets of the products are much higher than the market growth in this category. So they generate most income then that is consumed.
Question mark: these are the product that consumes large amount of cash by providing a high growth in the market and as they have low market share they don't generate much income. These products can be places in the star category if the market share rises and also can be placed in cash cow if the market growth is slow.
Star: these products produce and consume large amount of cash because their market growth is high and at the same time its market share is equally high. When the market growth falls for these products, they will become cash cow.
Construction of BCG is done on few assumptions:
Market share can be gained by investment in marketing
Market share gains will always generate cash surpluses
Cash surpluses will be generated when the product is in the maturity stage of the life cycle
The best opportunity to build a dominant market position is during the growth phase

Seminar 2 (q)

Question 2:

The good architecture means when the system is implemented according to the architecture, it meets the requirements and resource budgets. The requirements are the key input to the software architecture design. There are different model that to used to design software architecture by considering the requirements as their input. Some of the models are listed below.

1. Pattern-Based Architecture Design: Patterns are often categorized into two levels based their scales: architecture styles and design patterns. The main difference of design pattern from architectural styles is that design patterns are often only applied to one part of architecture. There are three parts in this design; context, structure and consequences. The idea of pattern-based architecture design is to use such a knowledge base to guide the design of software architecture. Because the system is much larger and more complex, it is not easy to apply these patterns to practical contexts.
2. Multiple-View Model: Multiple-view model that addresses the different aspects of the system with different views. 4+1 view model presented by Rational Software Corporation is the most well-known model. This model may be the most mature and widely used one. Each view has some specific notations or tools to support. This method provides a framework and it is up to the architect to decide which views are useful for a project. This model divides a complex architecture into some loose-coupling views from different perspective. This makes it much easier for various users to understand the architecture. This allows the architect to apply the aforementioned design methods to each single view, such as different architectural styles can be applied to different views without disturbance. The drawback of such a model is that there is not a clear boundary of architectural design and detail design.
3. Evaluation and Transformation Based Design: This design advocates for meeting the FRs first and then meeting the NFRs by iterative evaluations and transformations. The process starts with the functionality based architectural design. Then this design is evaluated using qualitative or quantitative assessment techniques. The estimated NFR values are compared to the requirements. If the estimations are as good as or even better than the requirements, the architecture design is finished. Otherwise, some transformation has to be done to optimize the original design in order to meet the NFR. There are two problems with this design; problem 1 is how to estimate the NFR values for an architecture design and problem 2 is how to transform an existing design and get better NFR values.
4. Architecture-Based Product Lines Design: Architectural based product lines tries to reuse the architecture design in a family of software systems and are proving to be a significant success for many organizations. The architecture design in product lines is more formal and time-consuming than the other methods.

Reference: On the Similarity between Requirements and Architecture Remco C. de Boer and Hans van Vliet *The Journal of Systems and Software*, Vol. 82, Issue 3, March 2009, pp. 544–550

Section 3: Reflections for all the articles.

"The art and science of release planning"

From this paper I have understood a release planning is and the problems that are faced during release planning. This paper use two types of approaches; art and science approaches. The art approach mainly involve the human activity for release planning where as in science approach, formulated algorithms were used for a successful release planning. But both the approaches have their own drawbacks, and for this reason a hybrid approach is used which is a combination of both arts and science. This approach is more suitable and useful and gives a positive result. By going through this article I was able to answer questions related to release planning in the seminar discussion questions.

"A case study evaluation of the guideline-supported QUPER model for elicitation of quality requirements" and "Introducing Support for Release Planning of Quality Requirements – An Industrial Evaluation of the QUPER Model"

The articles of both the papers discuss the importance of quality requirements and the problems that are faced if these requirements are not included in the release planning along with a model named QUPER that was developed with the aim to support high-level decision-making in release planning of

quality requirements. The authors of the first paper give us detail knowledge on the application of QUPER model for decision making on the quality requirements with a practical example. I felt that this model is very easy to understand and implement and the results show that this model is most commonly used and gives the right result since it is a step by step process while making a decision on selecting the right quality requirement. This model is the only method which addresses quality and cost constraints of QR. In the second paper it presents a quality performance (QUPER) model which supports release planning and road mapping of quality requirements and it also presents one case of QUPER tailoring, implementation, and most important evaluation, conducted at Sony Ericsson in 3 steps. The main purpose of this is to investigate the implementation of QUPER in industry. From the results of this investigation, it was spotted that the main identified challenge was difficulties to identify and specify the values for the differentiation and saturation breakpoints. And also different understanding of the breakpoints value among the staff was raised as a challenge.
While comparing both the articles i.e. "Introducing Support for Release Planning of Quality Requirements – An Industrial Evaluation of the QUPER Model" and "A case study evaluation of the guideline-supported QUPER model for elicitation of quality requirements", it was found out that the authors of both the papers has considered the quality requirements and have show practical results of the implementation of the QUPER model but in one paper the author has also included the cost dependencies among the quality requirements.

"A market-driven requirements engineering process: results from an industrial process improvement programme"
This paper discuss about the REPEAT process that manages the requirements throughout the release cycle by including activities such as elicitation, selection, management, documentation and validation and release planning of the requirements. I have learned that this process allows the experts to carefully consider which requirements to implement in which release, so that the requirements that are believed to give the highest value to the lowest cost are implemented first. In this way I believe that the end result of this process will be more accurate since the requirements go through and handled by many references (actors) involved in this process and also cross verification are also done by some of the actors. Even after all the benefits gained from this process, the authors of this paper have notices many challenges that occur while working on this process. This article helps me to answer questions related to MDRE and large scale requirement engineering.

"Requirements Abstraction Model"
I personally felt that Requirements Abstraction Model (RAM) is very easy to understand and also easy to implement it on the requirements. If all the three steps are followed in the right way and in the second step, if the right dependencies of the main requirement are placed in each level which builds up the connection from level 1 to level 4 and vice versa, that feature is all set to go. Since this model also includes both static and dynamic validation, chances of getting positives results are more.

"A method for early requirements triage and selection utilizing product strategies (MERTS)"
MERTS which is built based on the needs identified in literature and industry helps the product managers to deal with the challenges that occur in the continuous effort of early requirements triage. From the results of the controlled experiments done in a lab environment with 50 software engineering graduate students as subjects on MERTS I have learned that the product strategies though MERTS are highly effective and efficient and these strategies are ready for industrial trials.

"Requirements Engineering. In search of dependent variables"
According to my point of view, this paper provides the contributions that serve the companies to take care of the dependent variable at the time of requirement engineering process changes are made and serves as a reference as it presents a clear frame work of all the variables in different views. The results of the framework help the requirements engineering to assess the quality. I feel the structure of the framework is well structured because as changes in requirements process occur we can observe their impact on dependent variables till five distinct levels (Requirements phase, Project, Product, Company, Society).

"Quality Requirements in Industrial Practice – an extended interview study at eleven companies"
This paper focuses on the importance of considering the quality requirements in the product plan. The author of this paper has conducted interviews at eleven companies which are divided into 2 categories (B2B and B2C) in order to find out the objectives that are related to the priorities given to the quality requirements. I have learned that the different priorities are set to different quality requirements and this will be changing from company to company. This is proved from the results that the companies that were interviewed given different priorities to the quality requirements. Seminar questions that are related to quality requirements are answered with the help of this article.

"A cost-value approach for prioritizing requirements"
From this article I have learned that the requirement prioritisation is done based on their cost and the value. This kind of approach is called cost value approach which helps to make prioritization simple and straightforward giving out accurate and convincing results. In my opinion, this approach is suitable in prioritizing requirements for the projects which have less number of requirements. This approach does not consider interdependencies so it may be a disadvantage to apply it in large scale requirements engineering. It also has few advantages and can be implemented in large scale requirements engineering if it consider interdependencies.

"Towards a reference framework for software product management"
My opinion on this framework is that it gives a clear explanation on how a software product management process is carried out entirely by an organization and the structure of the software product management and also helps in identifying the stakeholders for the product and their relation during the software product management. I also got an idea and a clear understanding about what a software product management is as the paper explains each and every step involved in the software product management clearly. This article answers the questions related to portfolio in the seminar discussion questions.

"Market-Driven Requirements Engineering for Software Products"
From this paper I have learnt about MDRE and the activities included in it that helps in delivering the right product at the right time. I have also learnt about the two modes that help in managing the MDRE process when the requirements flow is continuous and also I got a clear idea about the road mapping and its importance from this paper. All the questions related to road mapping were answered with the information given in this article.

"Are You Biting Off More Than You Can Chew? A Case Study on Causes and Effects of Over scoping in Large-Scale Software Engineering"
I find this paper very interesting because it discuss about the effects caused by the over scoping which is very important for any organizations in the competitive society. Scoping is the process of selecting which requirements to include into the next release of a software product. From the results of the interview and case study that is conducted, i have learnt that there are many negative effects caused because of over scoping.

"An industrial survey of requirements interdependencies in software product release planning"
The main aim of this paper is to find out strategies for reducing the effort needed for identifying and managing interdependencies. The authors of this paper mainly concentrated on identifying the nature and frequency of requirements interdependencies and their importance in software release planning. From the results I have observed that it is important to find the methods or techniques that identify the singular requirements, highly dependent requirements or free clusters. This will make it easy to categorising the requirements along with their interdependencies and will also reduce the complexity in software product release planning.

"What Happened to Our Features? Visualization and Understanding of Scope Change Dynamics in a Large-Scale Industrial Setting"
From this article I have learnt about the scoping process and the cause and effects of negative scoping. I have also got a clear idea about the visualization technique. This article helped me to answer questions related to scoping and what scope changes are likely to happen in requirement lifecycles.

"Obsolete Software Requirements"
From this article, I have learnt what obsolete requirements are and what causes the obsolete requirements. I have also learnt that it is not an easy task to handle, identify and remove the obsolete requirements. And if the management of these requirements are poor, then there will be an immense effect on the organization or company.

Section 4: *Reflections upon the contribution of the papers with respect to the seminar discussion questions*

All the seminar discussion questions in each seminar are related to the articles given in the seminars. After going through the articles in each seminar, it was helpful to answer the discussion questions that are asked in the next seminar. In the first seminar discussion questions, there are all basic questions about requirements and in the second seminar discussion questions; we have questions that were related to the articles covered in the first seminar. For example we have articles about release planning and MDRE. These articles helped me to answer the questions like what is market-driven requirements engineering, how do you deal with MDRE and etc. And also these seminar questions can be answered by the articles given in the same seminar. In this way with the help of all the articles given in the seminars, I could answer all the questions. Few questions like requirements and architecture design, into the project, tools support for the requirements and horizontal scaling in the seminar discussion are answered by the articles searched in web.

Can you connect all requirements directly? What do you do if you cannot?
It is not possible to connect the requirements directly to the architecture. But there are many techniques to connect requirements to architecture. The techniques can be 1. Use of goal-oriented requirements models to formally generate feature, state chart, and component connector models of a system's architecture. 2. A method to produce multi-agent architectures from a specification language that is intended to be used for expressing agent requirements. 3. Architecture-aware requirements models by applying taxonomy of architectural dimensions to requirements.
Reference: Summary of article (Wnuk et al. "Factors Affecting Decision Outcome and Lead-time in Large-Scale Requirements Engineering")