JOURNAL OF SOFTWARE ENGINEERING & INTELLIGENT SYSTEMS
ISSN 2518-8739
30th April 2017, Volume 2, Issue 1, JSEIS, CAOMEI Copyright © 2016-2017
www.jseis.org

A survey of decision support and cognitive load in requirements engineering

[1] Falak Sher, [2] Shahbaz A.K. Ghayyur
[1]Faculty of Computing, Universiti Teknologi Malaysia, Malaysia
[2]International Islamic University, Islamabad, Pakistan
Email: [1]falakawan@gmail.com, [2]shahbaz.ahmed@iiu.edu.pk

ABSTRACT

Decision-making is one of the most complex activities in requirements engineering (RE). There are certain factors which affect directly or indirectly to the quality of decision-making. Certain characteristics and guiding principles also exist and theoretically grounded which can play a very important role in improving the quality of decision-making. But, there is a need to improve and evaluate these characteristics and their guiding principles for their effectiveness and prioritization. One of these characteristics is cognitive load that is a major cause for effectiveness of quality of decision-making in RE. It can be reduced by following the guiding principles giving detailed and overview information about the system during decision-making in different phases of RE. Similarly, different techniques also exist in literature which can help in reducing Cognitive Load during decision-making process in RE. However, it is required to evaluate the usefulness of these characteristics in terms of their effectiveness, valuable in a certain context, and their priority in order to increase efficiency. Hence, a survey is conducted in this research in order to evaluate the characteristic of cognitive load and the use of decision support tools, techniques and their effectiveness in making decisions for RE and to reduce cognitive load.

Keywords: decision support; requirements engineering; cognitive load; decision-making;

1. INTRODUCTION

Requirements Engineering (RE) is a very critical activity for those who are in the field of research and development in software industry for the last many decades. It has been recognized that due to complex nature of the area of Software Engineering (SE) it is very difficult to make decisions during its different phases. Our ultimate perspective throughout this research is to focus on validation aspect of characteristics required for decision-making in RE to improve the quality of decision-making in RE decision support systems (REDSS).

Empirical validation for such characteristics which influence REDSS should focus on to enhance the decision-making capabilities to perform decision-making activities in the RE decision process. Presently, there is a need to go for justification in terms of practical implementation of existing guiding principles and characteristics to facilitate decision-making requirements of requirements engineer in the field of RE for decision support. Moreover, there is a need to work on existing RE tools to validate their existing capabilities in terms of support for RE to augment their decision capabilities in decision support systems (DSS). It is also required to consider different levels where there is a need for support to help decision makers in their decision-making process to improve the quality of their decisions in the field of RE and research. Therefore, we have decided to target the practical aspects of the industry to explore current situation of existing tools and introduction of new directions to overcome the problems of the industry as well as researchers for decision-making in RE. Hence, in order to achieve this goal a survey based approach is adopted. Initially empirical evidences are gathered from the literature in order to strengthen our research claim.

Zave (1997) provides one of the clearest definitions of RE: “Requirements engineering is the branch of software engineering concerned with the real-world goals for, functions of, and constraints on software systems. It is also concerned with the relationship of these factors to precise specifications of software behaviour, and to their evolution over time and across software families” [1]. RE is the “branch of systems engineering concerned with the desired properties and constraints of software-intensive systems, the goals to be achieved in the software’s environment, and assumptions about the environment” [2]. Nuseibeh and Easterbrooks (2000) define RE as a series of decisions that lead from recognition of a customer problem to be solved (or a need to be satisfied) to a detailed specification of that problem [3]. The decision-making complexity stems from the limited capacity to understand everything, the limited time in which to make decisions, and the limitations of our schemas (cf. bounded rationality in decision-making [4] and multitude of attributes to compare the choice strategies [5]. Cognitive load is a “construct representing the load that performing a particular task imposes on the cognitive system” [6]. To put it in other words, cognitive load means a mental exertion. Its purpose is to interpret and process information in order to decide an action within a given space of time [7]. There are two common ways to categorize requirements that are:

- Functional requirements
- Non-functional requirements

Non-functional requirements can be sorted into three groups [8]:

- Organizational requirements
- Product requirements
- External requirements

Another way of categorizing requirements is [9]:

- Goal level requirements
- Domain level requirements
- Product level requirements
- Design level requirements

Requirements specification is a term given two different meanings in the literature. It is used as either describing the RE activity of specifying requirements which can be understood by their stakeholders or as the name of the document covering the requirements that is a complete description of what the system should do [10]. Someone has to carry out the activity in order to receive a collection of specified requirements. To divide the term into two different ones will not add any value. Thus, we include both meanings in our research, and let the textual context show the current meaning.

It is important, not only to specify requirements of different types, but to write clear, understandable, and unambiguous requirements. Otherwise, the users of the requirements may interpret them in an incorrect way and thereby use them inappropriately. Requirements can be written in natural or formal language [11]. Natt och Dag and Gervasi (2005) advocate natural language when requirements are specified, which they do for communicative reasons [12]. Natural language is the primary communication language of people, which makes it more understandable for more readers than formal language. Natural language is useful for validation of requirements, and is not domain-specific or specific for a certain level of abstraction. Therefore, it is more flexible than formal language [12]. On the other hand, natural language can be, and often is, ambiguous and difficult to understand. This can lead to misinterpretations [11, 13], and as a result problems occur during the system development process causing quality problems in the system.

It is necessary to specify requirements in order to develop successful systems, but low quality requirements reduce the chances of reaching that goal. Difficulty in understanding the requirements is one such quality problem. However, this is not the only threat to quality. Other common requirements problems, listed by Kotonya and Sommerville (1998), are inconsistent and incomplete requirements as well as wrong requirements which do not fulfil the needs of the users [13]. Other essential problems are that it is expensive to make changes among agreed requirements, and that there are confusions and mix-ups between different stakeholders [13].

A reflection of specifying requirements at different levels is the importance of traceability and dependencies between the requirements at different levels and within each level [14]. To make decisions concerning, for instance, a requirement change proposal at an organizational level will most likely have an effect on other requirements. There are also cascade effects concerning the traces between the levels. A requirement at an organizational level is most likely concretised in several requirements at the project level. This can also be the reverse case, since a requirement at a project level can be a concretisation of several requirements at a higher level. The decision-maker needs to have information about these relationships in order to make a well-informed decision, where at least some consequences are known and can be taken into account.

1.2 Activities in RE

1.2.1 The RE process

RE process has several activities through which different kinds of information flow and knowledge increases; not least concerning requirements as shown in Figure 1. The RE process consists of the following activities: elicitation, analysis, negotiation, validation, documentation, and management. In requirements elicitation, the requirements are discovered. These "raw" requirements need to be refined, which is conducted during requirements analysis. There are multiple stakeholders involved in the process who have different views and needs. Thus, requirements negotiations are necessary in order to agree on a set of requirements. The requirements should also be validated to ensure the quality of the future system. The requirements are documented, and requirements management is also necessary.

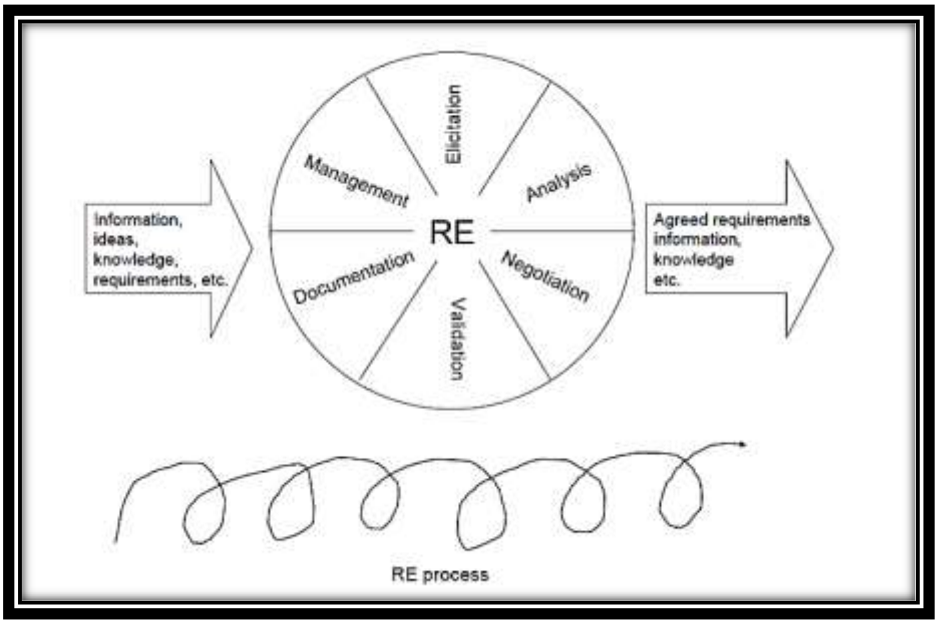


Figure. 1 RE process

1.2.2 Requirements elicitation

A central activity in RE is to generate the requirements of the system. In requirements elicitation, the needs of the users and other stakeholders should be learned and understood in order to communicate this information to developers. The sources of requirements are of different types, such as stakeholders' opinions, documentation, and existing system [13, 15]. Requirements elicitation can be compared to data collection, i.e. data concerning relevant aspects are brought together. "Data collection" has to be conducted several times within the RE process, since it is not possible to "collect" a complete set of relevant "data" at one time. The comprehension of what "data" is needed during RE process also emerge while working with the process. To put it in the words of Zowghi and Coulin (2005) requirements elicitation "must allow for communication, prioritization, negotiation, and collaboration with all the relevant stakeholders. It must also provide strong foundations for the emergence, discovery, and invention of requirements as part of a highly interactive elicitation process" [15]. There are five fundamental types of activities in the process of requirements elicitation [15]:

- Understanding the application domain
- Identifying sources of requirements
- Analyzing the stakeholders
- Selecting the techniques, approaches, and tools to use
- Eliciting the requirements from stakeholders and other sources

1.2.3 Requirements analysis

While, requirements analysis and requirements elicitation are interdependent and iterative, none of these activities have high value on their own. In the previous section, we compared requirements elicitation with data collection. The collected data has to be analysed in order to be understand and interpret it. Accordingly, requirements analysis can be compared to data analysis. We have to organize, scrutinize, and add meaning to the "data", in order to make it useful. During "data analysis" we become aware of missing information and insufficient knowledge. Hence, the "data analysis" drives further "data collection" efforts. Requirements elicitation and requirements analysis can be conducted almost synchronously or at separate times.

According to Kotonya and Sommerville, the purpose of requirements analysis is to establish a complete and consistent set of requirements [13]. The need for a requirement has to be established and it must be ensured feasible within the budget and schedule. The requirements document should be scrutinized in order to find missing requirements, requirements conflicts, ambiguous requirements, as well as overlapping requirements [13]. However, not only requirements should be the result of requirements analysis. Sutcliffe claims we need information in order to understand the system context and to be able to model and write scenarios [11]. There are different types of information that should be gained during this activity:

a) dynamic information,
b) static information,
c) contextual information, and
d) intentions

While dynamic information describes events, actions, procedures, and tasks, static information is, for example, entities, agents, attributes, relationships, properties and states. Contextual information concerns the setting of the system, while information about intentions includes goals, arguments and justifications. This information can be used for refinement of requirements, interpretation, modelling, and design [11].

1.2.4 Requirements negotiation

The stakeholders of a system and a system engineering process have different perspectives, goals, and needs. This inevitably has an effect on the requirements, and trade-offs between these conflicts are necessary [11, 13, 16]. This is handled in the activity called requirements negotiation, which is a process where requirements conflicts are discussed and solved through compromises [13]. Requirements negotiation can iteratively occur several times in the RE process. It does not take place in a specific order in relation to other RE activities. Negotiations can also be made during other activities, for example, if a goal analysis is conducted cooperatively between stakeholders, then negotiations concerning the relative importance of goals are certainly needed.

An important purpose of requirements negotiation is to agree on a set of requirements. More specifically, as described by Sutcliffe, there should be agreement with regard to the most appropriate design options and trade-offs between conflicting requirements [11]. There should also be a selection of requirements for prioritisation. Grünbacher and Seyff (2005) mention that another important result of requirements negotiation is an understanding of why there is a disagreement among the stakeholders [16]. Such disagreements are threats and they need attention and should be dealt with in the project management.

1.2.5 Requirements validation

The term validation has slightly different meanings for different authors in the literature. Sutcliffe describes that validation “involves getting users to understand the implications of a requirements specification and then agree that it accurately reflects their wishes” [11]. This statement can be contrasted to the view of Kotonya and Sommerville who define requirements validation as being “concerned with checking the requirements document for consistency, completeness, and accuracy” [13]. Kotonya and Sommerville further write that “the aim of requirements validation is to ‘validate’ the requirements, i.e. check the requirements to certify that they represent an acceptable description of the system which is to be implemented” [13]. Sutcliffe is more oriented towards the users and how to get the users to comprehend the consequences of the requirements. We consider the user perspective important, although not enough. There are other stakeholders important in requirements validation, for example, a customer or another project sponsor. An obstacle of user requirements validation is that there are not always real users available. The availability depends on what type of application that is developed. The availability of users is most likely higher in the development of in-house bespoke systems, than market-driven development.

1.2.6 Requirements documentation

A large amount of resources is invested in the RE process, which results in a lot of information. The information needs to be documented so that it can be used by different stakeholders. Similar to specification, the term ‘documentation’ has two meanings; an activity or an artefact. Both denotations are used and the textual context shows the current meaning. Requirements documentation should be performed continuously in the RE process so that no important information is lost. A requirements document can be stored and accessed using different media; either a computer or paper. The media can also be combined, e.g., requirements can be stored in a database and from that a paper-based requirements document can be generated. Paper-based documents are still important. To put it in the words of Hoffmann et al. (2004) “the days of paperless development are still far away, especially in fields where interaction with suppliers is important" [17]. An important document is the requirements specification. Eriksson (2007) lists other requirements documents, for example [18]:

a) Preliminary study documents – The result of investigations preceding the requirements engineering process
b) Vision documents – Describe the vision of the system
c) Use cases – Describe the interaction between the actors of the system
d) Supplementary specifications – Additional requirements
e) Change requests – Describe requests of requirements changes
f) Sequence diagrams – Describe communication flows between the system actors
g) Function specifications – Requirements that are broken down from the requirement specification
h) Screen layout – Describe the screen layout of the system
i) Design specifications – Requirements that are broken down from the function specification
j) Graphical user interface standards – Company or project-specific guidelines concerning the user interface
k) Component specifications – Describe in detail the components of the system

1.2.7 Requirements management

Other important parts of RE is requirements management and requirements change management. Since the requirements are continuously changing, they must be effectively dealt with. There are other significant areas, such as quality assurance in RE, requirements prioritization, requirements traceability and dependencies, impact analysis, and requirements management tools. We have summarized important areas related to requirements management in Figure 2.

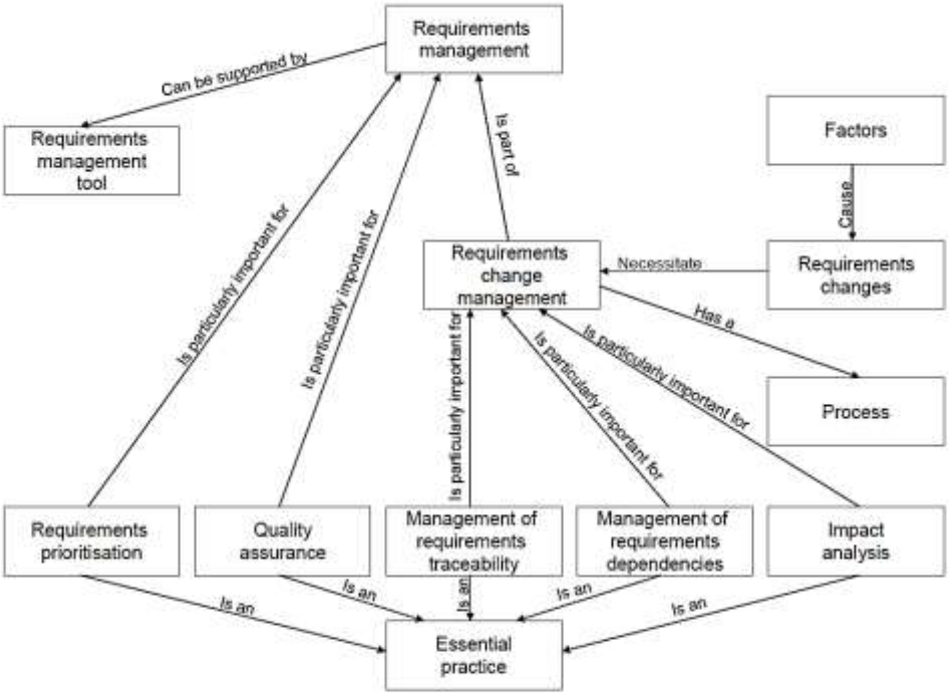


Figure. 2 Areas of requirements management

1.3 Requirement engineering tools

RE involves many complex and important activities. The skills and knowledge of the requirements engineers are vital for successful RE work. However, as the size of projects or systems increase, then skill and knowledge is not always enough. The requirement engineers possibly need assistance in order to effectively carry out the RE tasks. Such assistance can be partly provided by RE tools. RE tools are defined by Matulevicius (2005) as "software tools that provide automated assistance during the RE process and support the RE activities" [19]. Often, the RE tools are termed as the requirements management tools [17, 20-22].

The other RE activities and tasks also need support, such as generation of ideas of innovative solutions for the next generation of the system, problem solving with regard to requirements error reports, and dissemination of requirements information. Consequently, we use the term RE tools, instead of requirements management tools, in this research paper. According to INCOSE, there are different types of RE tools which are depicted in Figure 3 [22].

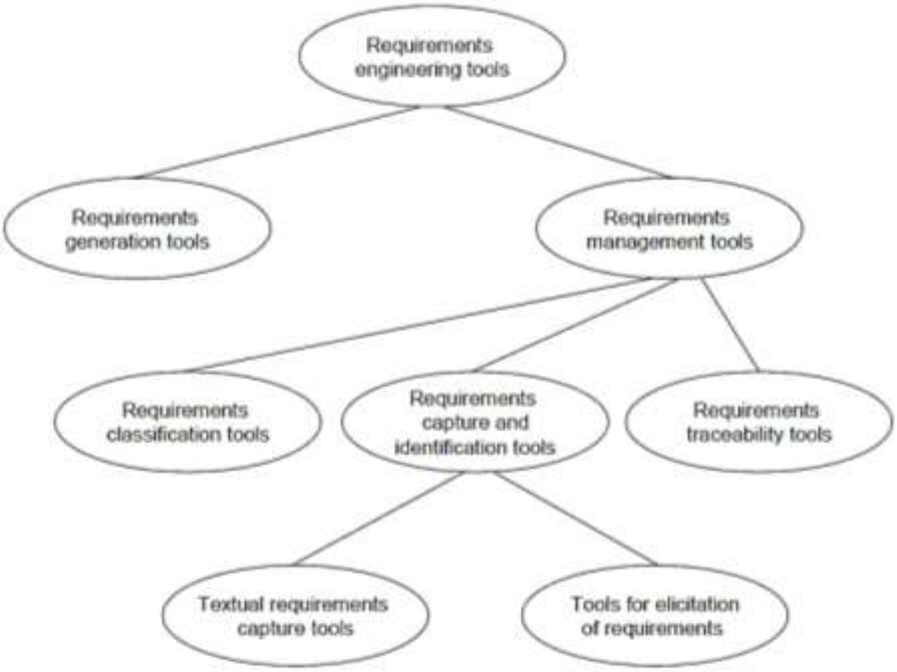


Figure. 3 Categories of RE tools

A requirements generation tool produces lower level requirements in a systematic way. The generation is based on, for instance, design constraints and results from system simulations [22]. It is not obvious from INCOSE's description, if the generation is automatic or supported and manually performed. However, we guess that both are possible to some extent, although we argue that human intervention is important. There are important aspects that are implicit and knowledge-based, which calls for a person. INCOSE describes a requirements classification tool as enabling classifications of requirements with the purpose of facilitating scheduling and tracking of requirements analysis activities [22]. Requirements capture tools compile information from several text sources. They aid in finding relationships in the documents. A requirements identification tool separates requirements in a set of information from superfluous information. A requirements traceability tool makes it possible to manage links between requirements and other artefacts, such as models, requirements changes, and information sources [22]. This categorization shows that not all aspects of RE are supported by tools. For example, there is no requirements negotiation tool or requirements validation tool. Perhaps such tools exist, though not explicitly part of the categorisation of RE tools, which indicates that there is room for improvement in this area.

A problem with the RE tool categorization of INCOSE is that the functionality of a certain RE tool can be broad and the tool can be classified in several RE tool categories [19]. This makes the classification more difficult to use, which decreases its value. Another problem with the categorization is that many RE tools on the market are available as COTS components [21, 23]. Thus, depending on what components the purchaser chooses, the RE tool provides different functionality. Hence, it can be categorised in different ways.

RE tools are evolving towards integration with other tools used in the development process, and they also progress towards product life-cycle management [2]. However, most RE tools are based on a requirements database, which store requirements and related documents. This facilitates gaining an overview, organising, and finding requirements [18]. Eriksson states that several RE tools can manage requirements, error reports, as well as test cases. They can be linked to each other and their relationships can be displayed in traceability metrics. Some tools support prioritisation and requirements checking [18].

There are many lists of characteristics, requirements, and improvement suggestions for RE tools in the literature [2, 10, 17, 18, 20, 22]. Examples of the content of the lists are that the RE tool must support base-lining and configuration management, be user-friendly, support standard systems modeling techniques and notations, allow the user to freely define a requirements management model, improve facilities for the geographically distributed collaborative work, and inter-tool communications. Our impression is that there is a large number of relevant and important characteristics, requirement, and improvement suggestions. Interestingly enough, none of these lists include characteristics or requirements for RE decision support.

RE tools have advantages compared to office and modelling tools (e.g. Word, Excel, and Visio). A case study demonstrated that the quality of requirements documents was higher, when an RE tool had been used compared to when standard office and modelling tools were used [10]. More specifically, as described by Eriksson, RE tools have dedicated functionality for efficient management of requirements. It is possible to depict an overview of relationships between requirements [18]. RE tools enable checking and approval of requirements. Neither of this can be done efficiently and smoothly with standard office and modelling tools [18]. With the help of RE tools, the goals of systematic requirements management can be achieved [17]. Other advantages are that RE tools force requirements engineers to write and structure the requirements in a uniform way [18]. RE tools help overcoming the complexity of activities embraced by the RE discipline [21]. Project managers, requirements engineers, as well as other project participants can benefit from proper management of this complexity with an RE tool [23]. A survey, conducted by Matulevičius (2005), demonstrates that mainstream RE practice relies on standard office and modelling tools instead of specialised RE tools [19].

There can be many potential reasons why RE tools are not used in every RE process. One reason can be that not all companies and all projects can benefit from RE tools. According to Eriksson (2007) and Hoffmann et al. (2004), a company needs to have a stable and mature work process to utilise RE tools [17, 18]. The requirements have to be written in a standardised way, which calls for an agreed way of working with requirements. The need for an RE tool increases along with several factors: a) when the number of requirements exceeds about 100-150, b) when there are multiple persons involved in the RE work, c) when the project, budget, risks, and participants increase, d) when requirements are frequently reused, as well as e) when the project is conducted in cooperation between several organisations [17, 18]. Hence, immature organisations, small projects, projects with few requirements, and/or single organisations can be overwhelmed by the administrative burden caused by an RE tool. It can be more beneficial in such cases to manage the requirements in a more simplistic manner.

Some other reasons for organisations not using RE tools can be the drawbacks and remaining challenges of current RE tools. Many tools lack a user-friendly interface and they have, therefore, received negative feedback from engineers [17]. Hoffmann et al. (2004), also mention that the effort sometimes exceeds the benefit, for example, when maintaining traceability [17]. This means that the cost of maintaining traceability links goes beyond the gains of always having them up to date. Another problem is the difficulty of making tools from different suppliers work together [18]. Since, RE is an intrinsic part of systems engineering, integration with other tools is important in order to achieve a smooth work process. Lang and Duggan (2001) claim that the social process of software development has not been taken into account in RE tools [20]. The collaborative needs of multidisciplinary and distributed teams are not support enough [10]. However, as [19] puts it, the long-term value of an RE tool survey is limited, since their features and qualities continually evolve.

We have argued that the RE process consists of decisions and decision-making activities which are critical for both the system to be as well as the systems engineering process. The RE decision-maker’s abilities and capabilities can be enhanced if appropriate RE decision support is provided. Thus, an important RE tool category should be RE decision support systems (REDSS). In this research, we use the term REDSS to denote a visionary, non-existing tool and the term RE tool represents existing tools. The term REDSS implies that the fundamental concept of such a tool would be decision support. This does not exclude the possibility of implementing decision-supporting functionality and qualities into existing RE tools. Including decision-supporting features in RE tools is desirable.

There is also a lack of descriptions of the types of characteristics an REDSS should have and what requirements we should have of such a system. RE decision support characteristics have the potential to suggest and direct research and development efforts concerning decision support for RE decision-makers.

1.4 Requirements engineering as decision-making

Requirements serve as verbalisation of decisions concerning the functionality and qualities of a system. Thus, the RE process can be viewed as a decision process and requirements can be viewed as decisions [24, 25]. Requirements engineers can then be regarded as decision-makers. RE decision-making is complex, has several difficulties, and is of vital importance for both the development process and the system. Therefore, RE decision support can increase the effectiveness and efficiency of RE decision-making.

1.4.1 RE comprises decisions

RE is largely a decision-making process in the words of Evans et al. “For the engineering of computer-based systems, the term [and the associated process of ‘requirements’ might well be replaced with the term ‘decisions’ and a decision process” [24]. Stakeholders’ decisions about the quality and functionality of a system are expressed in requirements. Other important issues in the RE process, such as organisation, staffing, and planning, are also decided upon. The poor decision-making results in RE failure [26]. Thus, by addressing decision-making improvement in RE, the probability of successful systems engineering increases [24]. Ruhe (2005) stress that the most successful companies in the future will be those that have an integrated approach for strategic decision-making, requirements management, and road mapping processes [27]. The successful companies will be those who effectively utilise “their intellectual capital generated by the decision-making process and would link this process to the essential supporting information” [27].

Decisions are made throughout the whole RE process [24]. Aurum and Wohlin compare an RE process model [28] with a decision process model [29], and they claim that the two have much in common in that their activities have similarities. Using the model of Mintzberg, et al. (1976), micro decisions can be identified [29]. Micro decisions are concerned with the decision-maker level, i.e. how they actually carry out decision-making. Macro decisions, on the other hand, focus on management activities of an organisational level. Micro and macro decisions are mutually dependent and intertwined. Both types are present in RE [24]. Macro decisions can be categorized as belonging to three levels in an organisation: strategic planning, management control, and operational control [30]. In the RE process the decision matter at the strategic level mainly concerns organisational considerations, such as the consistency of requirements with the product strategy or business goals. Tactical decisions, i.e. management control, focuses on the project level, for example, human resource planning. The lowest level, operational control, involves making decisions on realisation issues and decisions on quality, classification, and properties of requirements [24, 26]. Examples of decisions that need to be made in RE are [24, 27]:

a) Which functional and non-functional requirements should be selected in relation to given time and budget constraints?
b) How should the requirements be organized?
c) How should the requirements be classified?
d) What is the importance of the requirement?
e) Who are the requirements’ stakeholders?
f) What is the priority of the requirements?
g) How does the requirement depend on other requirements?

Such decision-making is not always straightforward and there are several challenges that decision-makers face in the RE process.

1.4.2 Difficulties of decision-making in RE

There are difficulties in RE decision-making, for instance, that it is knowledge intensive activity, and that human decision-makers in general have cognitive limitations [24]. However, there are several other obstacles that the RE decision-makers have to deal with. We can use Orasanu’s and Connolly’s list of eight factors that characterize decision-making in natural settings to structure difficulties of RE decision-making [31]. The factors are stated below.

a) Ill-structured problems,
b) Uncertain, dynamic environments,
c) Shifting, ill-defined, or competing goals or values,
d) Action and feedback loops,
e) Time stress,
f) High stakes,
g) Multiple players, and

h) Organizational goals and norms.

1.5 Decision-making and Decision Support System

1.5.1 Definition of decision

Decision is a commonly used word. Large amounts of literature have been written on how people make decisions, how we should make decisions and consequences of decisions. However, the definition of the concept, decision, is still ambiguous. The reason for this could be that the term is taken for granted; everybody "knows" what a decision is. If so, there can be a risk that we interpret the term, decision, differently. Two definitions of the term are: a decision is a "specific commitment to action" [29] decision is a "reasoned choice among alternatives" [32]. However, these definitions focus on different aspects. Mallach (1994) focuses on the choice, which in this case must be preceded by an evaluation of alternatives [32]. This means that decisions are regarded as two steps within a decision process. The decision process is often considered to include more steps, which makes defining decisions as the last part of a decision process somewhat awkward. In the definition of Mintzberg et al. (1976), decisions are regarded as a consequence, and it also implies that some type of action is always the consequence [29]. While it is reasonable to consider a decision the result of a decision process, it does not necessarily cause action. The result can be the decision not to act. In this research the central concepts are defined as follows:

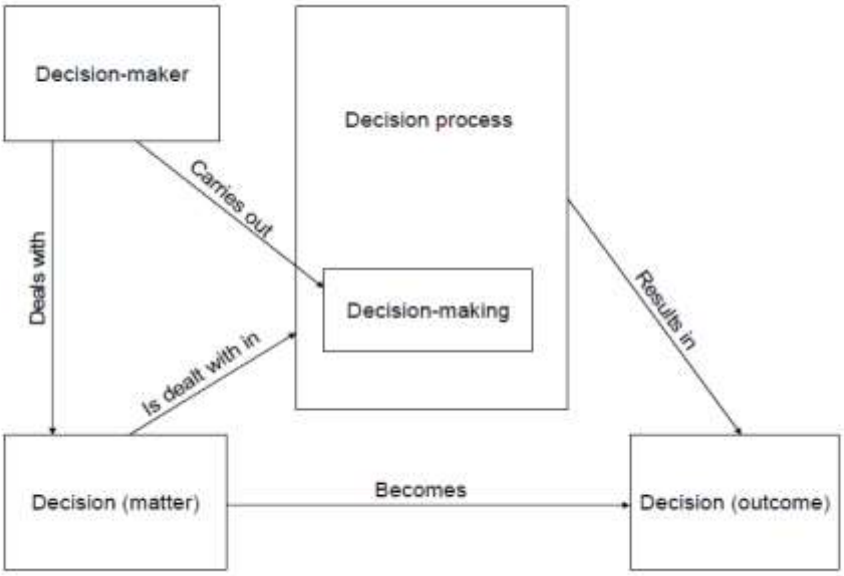


Figure. 4 Result of a decision process

A **decision** has two meanings; decision matter and decision outcome. A decision matter is the issue that is dealt with in the decision process. The decision matter becomes a decision outcome when the choice is made. The decision outcome is the chosen alternative that is to be acted upon.

Decision-making is considered to be the mental or physical activities done by a decision-maker when dealing with decisions.

A **decision-maker** is a person who carries out decision-making activities, alone or together with others, but is not necessarily the one who authorizes the decision.

A **decision process** is viewed as a number of phases or steps related to each other that consist of decision-making activities.

1.5.2 Types, characteristics of decision support system

According to Turban (1990): "A DSS is an interactive, flexible, and adaptable CBIS [Computer-Based Information System] that utilizes decision rules, models, and model bases coupled with a comprehensive database and the decision maker's own insights, leading to specific, implementable decisions in solving problems that would not be amenable to management science optimisation models per se" [31]. Thus, a DSS supports complex decision making and increases its effectiveness [12].

There are many ways to define the concept of DSS. In literature, DSS is defined as a computer-based information system that supports organization, individuals as well as group of decision-makers in performing more effective decisions while dealing with structured and unstructured problems. The DSS supports series of decision activities performed during decision making process. Similarly, application of DSS ranges many domains of life. DSSs serve the management, operations, and planning levels of an organization and help to make decisions, which may be rapidly changing and not easily specified in advance. Power (2002) categorizes DSS into following categories [33]:

- Data-driven DSS
- Model-driven DSS
- Knowledge-driven DSS
- Document-driven DSS
- Communication-driven and group DSS

1.5.3 Benefits and limitation of DSS

The benefits of DSS are [33, 34]:

- Improve individual productivity
- Improve decision quality and problem solving
- Facilitate interpersonal communication
- Improve decision-making skills
- Increase organizational control

Power (2002) and Marakas (2003) list some limitations [33, 35]:

- A DSS cannot have human decision-making abilities, such as creativity, imagination, or intuition.
- A DSS is limited by its stored knowledge, data and models as well as by the operating computer system.
- The user interfaces are not sophisticated enough for full interaction between the user and the system in natural language.
- It is difficult to design a general DSS that is applicable in multiple contexts, but instead they often have a narrow scope of application.
- Often, a DSS needs to be integrated into decision processes.
- A DSS can only be supportive if a decision-maker chooses to use the system and integrates the analyses into ‘off line’ thinking and analysis.
- DSS is a type of behavioral engineering, and many managers refuse to accept such intrusions.

1.6 Defining cognitive load

Sweller (1998) describes cognitive load as the level of “mental energy” required to process a given amount of information [6]. It refers to the total amount of mental activity imposed on the working memory at an instance in time. Working memory is the stage of memory where information is stored for a short period prior to either being forgotten or transferred to long term memory. Long term memory refers to the relatively permanent memory. We experience cognitive load because of the limitations of the working memory. In this section the types of cognitive load, and their influence on memory influenced by cognitive load, described in detail.

Cognitive load is a “construct representing the load that performing a particular task imposes on the cognitive system” [6]. To put it in other words, cognitive load means a mental exertion. Its purpose is to interpret and process information in order to decide an action within a given space of time [7]. Our mental capacity to consciously process information, which is called controlled processing, is limited. Thus, it is important that the individual uses this limited capacity for the most important and mentally demanding tasks [36].

1.6.1 Types of cognitive load

Following are the different types of cognitive load.

- **Intrinsic Cognitive Load** - refers to the inherent difficulty of the content.
- **Extraneous Cognitive Load** - refers to the load imposed by the instruction design.
- **Germane Cognitive Load** refers to the degree of effort involved in processing.
- **Working Memory** - is used to process information and create schemas in the long terms.

- **Schemas** - learning will only occur if a connection is made to a schematic structure in the long-term memory.

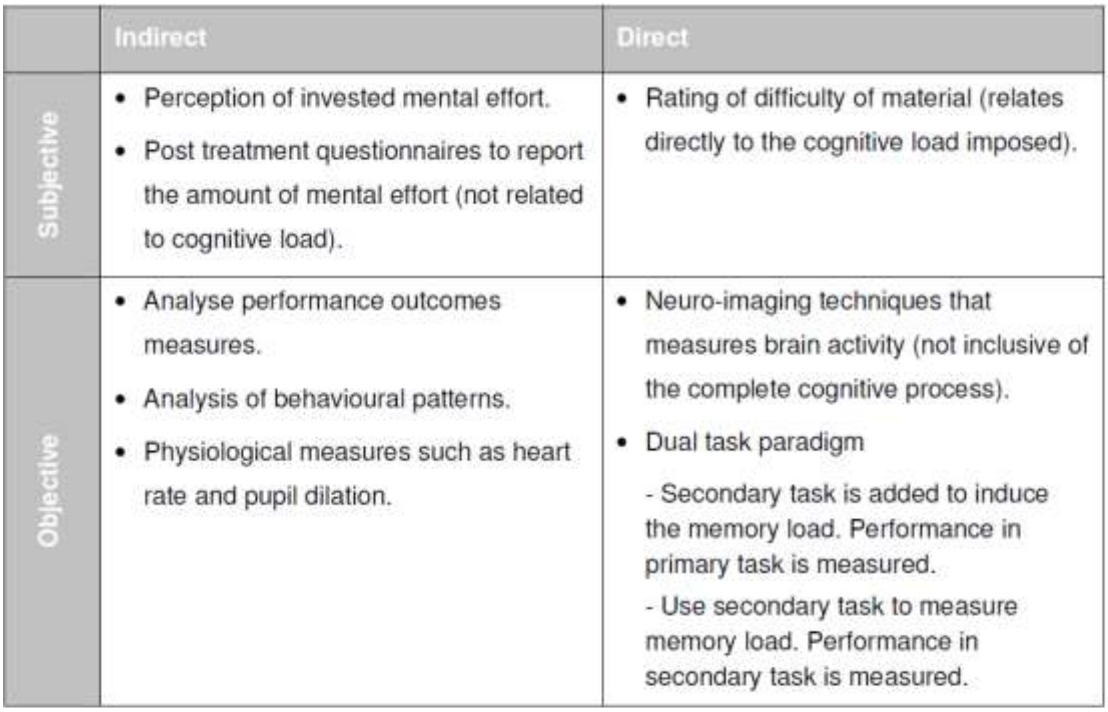
1.6.2 Cognitive load and decision support system

The cognitive load, in the case study, is expressed in three problems:

- Lack of general overview
- Lack of understanding
- High memory load

1.6.3 Measuring cognitive load

The following matrix is used to measure the cognitive load.



	Indirect	Direct
Subjective	• Perception of invested mental effort. • Post treatment questionnaires to report the amount of mental effort (not related to cognitive load).	• Rating of difficulty of material (relates directly to the cognitive load imposed).
Objective	• Analyse performance outcomes measures. • Analysis of behavioural patterns. • Physiological measures such as heart rate and pupil dilation.	• Neuro-imaging techniques that measures brain activity (not inclusive of the complete cognitive process). • Dual task paradigm - Secondary task is added to induce the memory load. Performance in primary task is measured. - Use secondary task to measure memory load. Performance in secondary task is measured.

2. RESEARCH IN RE DECISION-MAKING AND RE DECISION SUPPORT

Research in the field of RE decision-making and RE decision support is in its infancy [27], although there has been and is some research conducted in the field. Ruhe (2005) discusses that decision problems in RE can be seen from two perspectives; a requirement-centric perspective and an activity-centric perspective. The first one, the requirement-centric perspective, is often the viewpoint of software engineering researchers, whose main concerns are the contexts directly related to requirements. Their decision problems begin with the requirements. The second one, the activity centric perspective, is often the position of decision theory researchers who apply their theories in the RE domain. Their decision problems are identified in the RE process and the software engineering process. They focus on a broader context and include other aspects, such as maturity of organisations and availability of information. The two perspectives are not mutually exclusive, but the focus and order of importance are different [27]. Our perspective is an activity-centric perspective, since we primarily focus on decision-making activities of requirements engineers and the problems and difficulties they experience. Based on the nature of the activities we suggest characteristics of a visionary RE decision support system.

A major challenge of this field is to describe and comprehend RE decision-making. When we have substantial knowledge about this, then it is possible to effectively improve and support RE decision-making. Thus, more theoretical and empirical research is needed [27]. To gain insights and guide such research, decision-making theories and models of decision processes can be used. This way we can understand the nature of RE decision-making activities [37]. A research agenda for the field of RE decision-making can be derived from [27], as well as [26]. We need to:

- Perform empirical studies of RE decision-making in a comprehensive and focused way.
- Identify and study the decision problems in the RE process.
- Identify the decision types involved in each RE phase, as well as the meaningful actions or options that each decision-maker carries out for each decision type.
- Develop approaches that properly address requirements decision problems.
- Identify the information type (or knowledge) needed at each phase.
- Examine how non-technical issues, e.g., political, social, organizational, and cultural issues, have an effect on RE decision-making.
- Identify which type of stakeholders participates in each RE activity and accordingly consider specific decision aids for each type of stakeholder.
- Understand the group problem solving processes of RE.
- Validate the impact of improved RE decision-making on the system as well as the systems engineering process.

Based on knowledge about RE decision-making, decision support and decision making improvements should be suggested. Developing support for RE decision making is a major issue for RE research [26]. “The tremendous impact of software on products and services makes software engineering decision support (SEDS) critical activity” [38]. Ruhe (2005) adds that decision support is needed throughout the whole life cycle.

However, the research in [27] argues that RE decision support should not strive for optimality. Many decision situations in RE are not simple enough to enable an absolute optimal solution. The decision problems are “wicked” and trade-offs, uncertainty, and judgments are necessary. Instead, the provided support should augment the decision-making capacity of the human decision-maker. The strengths of humans and the computational power should be combined. Humans, for example, have a good capability in handling soft and implicit constraints and objectives, while computational models have a high capacity, such as memory space and computational complexity, where the human cognitive abilities are limited, [27]. A research agenda and suggestions for what needs to be supported are provided by [26, 27, 37]. We should:

- Develop empirically based guidelines to support decision-making.
- Focus on both improved decision quality and improved cost-benefit decision making, i.e. both effectiveness and efficiency.
- Keep track of RE decisions and their effect on the software product, i.e. record decisions, their rationale, and facilitate traceability.
- Emphasize decision support for RE decisions during uncertainty. Approaches from other disciplines can be used, e.g., probability theory.
- Provide decision support tools for both development teams and project managers.
- Support decision-making and problem solving of groups of stakeholders.
- Support requirements negotiations so that all stakeholders’ interests are taken into account.
- Make generation and evaluation of possible alternative solutions.
- Make better reactions for possible changes.
- Facilitate more transparent and robust decisions that can be understood by the stakeholders.

Thus, the field of RE decision-making and RE decision support is still immature and a coherent body of knowledge does not yet exist. The field has a lot of potential and can hopefully contribute significantly to RE practice in the future, thereby having a positive impact on the quality of systems and their development processes.

2.1 Decision situation of RE decision-makers

2.1.1 The Establishment of Requirements

The first step towards decision process is the establishment of requirements which is mainly focusing the following areas [39, 40].

a) Get requirements, structure requirements, write down requirements using automated tools.
b) Receive requirements, document requirements, analyse and discuss requirements (negotiate).

2.1.2 The identification phase

Begins with decision recognition routine and RE decision-maker conducts several decision-making activities [39, 40]. The categories of decision-making activities are:

a) System-related activities

The system-related activities are to:

- find out what the customer requirements and system requirements mean.
- investigate ambiguities in system requirement.
- initiate themselves into interfaces provided by the customer.
- analyse what matters for the subsystem.
- perform a basic analysis of the desired functionality.

b) Process-related activities

The process-related activities are:

- RE decision-maker creates a general view of the needs and problems in the development process.

c) Decision communication activities

The decision communication activities are:

- The RE decision-makers conduct investigations in which they obtain an understanding of the problem by searching documents and talking to relevant stakeholders.
- Notify those who are responsible for the entire system when there are problems in the customer requirements.

2.1.3 The development phase

Search routine is closely related to screen routine where decision maker seeks for ready-made solution [41].

- RE decision-makers compare the new requirements with existing components and find out if something can be reused.
- In the design routine, custom-made solutions are developed and ready-made solutions are modified [29].
- RE decision-makers create use cases and write requirements such as internal requirements and requirements that specify the interface between the subsystems. Dependencies between use cases are also drawn in this routine.

a) Three decision communication activities

The decision communication activities are stated as follows:

- The RE decision-makers discuss ideas and solutions with those who are responsible for the entire system
- Discuss with other people who are responsible for subsystem requirements
- Each person documents the result of these discussions in their “own” use cases. They also have to stay alert to the customer requirements and system requirements, so that these are covered in the subsystem requirements.

2.1.4 The selection phase (evaluation-choice routine)

Three modes of this routine are [39, 40].

- Analysis (the alternatives are evaluated)
- Bargaining (several decision-makers with different goals that make the choice),
- Judgement (an individual makes the choice)
  - Three system-related decision matters are handled by the RE decision-makers
  - Can the requirements become baseline?
  - How is the subsystem going to behave and what is it going to look like?
  - Which use cases are needed?
  - Work-related decisions that the RE decision-maker makes
  - Which level of detail should the requirements have?
  - What type of information should the requirements contain?
  - In which order, shall the requirements be implemented, i.e., what is the priority of the requirement?

- Which level of effort should an investigation have, e.g., of functions in order to write requirements?
- Authorization routine in the decision process model of Mintzberg.
- In which the decision is approved by someone in order to commit the organization to a certain course of action.

2.1.5 The implementation phase

Several decision activities are performed in this phase for example:

- Communicating decisions, plan actions, and track performance [33].
- RE decision-makers set up the requirements documents to be used.
- They document trade-offs, decisions and rationale for decisions together with the functionality.
- They check design specifications.
- Support the persons, who verify, design, and implement, to interpret the requirements.
- The RE decision-makers are a service function for all requirements stakeholders.
- The decision communication carried out is dissemination.

2.1.6 The follow-up and assessment phase

The consequences of decisions are checked in this phase.

- Identification of new problems [33].
- RE decision-makers check the verification and test specifications
- User group meetings in order to validate the outcome with the users

2.2 Management of requirement changes

Management of requirements changes is highly iterative and each instance of the decision process can take any path through the routines.

a) The identification phase

There are three different ways RE decision-makers recognise problems that initiate the decision process called management of requirements changes [41].

- Error reports from verification or construction.
- Direct requirement change proposals that start the process.
- Requirements errors can also be discovered by the RE decision-maker and in such cases, he or she carries out dissemination activities by writing an error report

In the diagnosis routine, the RE decision-makers carry out investigations.

- Check change proposals, investigate error reports.
- Initiate themselves into input from the customer, depending on what initiated the decision process.

b) The development phase

In the design routine, the RE decision-makers solve error reports, as well as change and add requirements.

c) The selection phase

- In the analysis mode in the evaluation-choice routine, the RE decision-makers check so that a change proposal is not going to become a problem for other subsystems.
- In the bargaining mode, there is one system-related decision matter. This decision matter is negotiated when the requirement in question is shared with other projects:
- Is a requirement change proposal going to be approved or not?
- There are also two work-related decision matters that are managed in this mode:
- When is the requirement change going to be activated?
- When is the requirement change going to be implemented?
- In the judgment mode, there is one system-related decision matter, which is dealt with by the individual RE decision-maker when the requirement in question is project unique.
- Is a requirement change proposal going to be approved or not?
- There is also one work-related decision matter to handle:
- How should the particular requirements change be managed?

- In the authorization routine, we have identified activities. When there are particular important decisions, authorization is needed concerning requirements changes.

d) The implementation phase

- In this phase, the RE decision-makers generate requirements documents
- Documents aimed for the verifiers and implementers that show the differences between former and current requirements documents

e) The follow-up and assessment phase (no specific activity)

2.2.1 Information sources used by RE decision-makers

There are different types of information sources that RE decision-makers use in their decision-making activities [40].

- Requirements and requirement-related information (customer requirements, system requirements, and subsystem requirements)
- Customer (technical data of their existing systems, which are going to interact with the system to be)
- Points of view (from internal stakeholders, such as system manager, resource personnel, e.g., cognitive scientists, project manager, software engineers, as well as other RE decision-makers)
- Records and reports (consist of error reports, reports from investigations concerning functions, design reports, records from requirements check)
- Meetings
- Theory
- The Internet

2.3 Factors that affect the RE decision-makers

The factors that affect the RE decision-makers are listed below [40, 41]:

a) Attitudes towards requirements work

- Low status of requirements work
- Prestige between subsystems
- Departmentalization of work

b) Communication and coordination

- Lack of coordination of way of working
- Little involvement in discussions
- Time-consuming coordination (with respect to calendar time)
- Little communication of decisions

c) Resource

- Usability problems in requirements management tools
- Lack of external expertise
- Lack of introduction to and education in RE

d) Pressure

- Two problems in the factor pressure are identified:
- Time pressure
- Several actors with different needs

e) Cognitive load

- Lack of general overview:
- Lack of understanding
- Lack of memory overview

f) Knowledge

- The domain
- The product
- Requirements engineering

3. INDUSTRY SURVEY

In this section, survey is conducted and we have presented the demographic analysis of software development companies of Pakistan who involved in this research. An overview of respondent designations and experience is also given. We sent our survey questionnaire to 25 teams of 25 different companies and we got response from respondents. The average number of years of experience of a software development company in software development is 8 and the average number of years of experience of the software development company in Requirement Engineering (RE) software development is 5. Among the respondents, almost all the respondents were experienced and working on senior positions. Among the respondents, there were seven software engineer, five Software developers, six project managers and seven system analysts. The average number of years of experience of a respondent is 8 and the average number of years of experience of a respondent in requirement engineering is 5. Moreover, the range of total experience of the software industry is ranging between 5 and 14 years. Similarly, the range of experience of requirement is lying between 2 and 10 years. Table 1 and Figure 1 shows the demographic analysis of the selected companies of Pakistan for this survey.

Table. 1 Demographic analysis

Respondent	Frequency
Software Engineer	7
Software Developer	5
Project Manager	6
System Analyst	7
Total participants	25

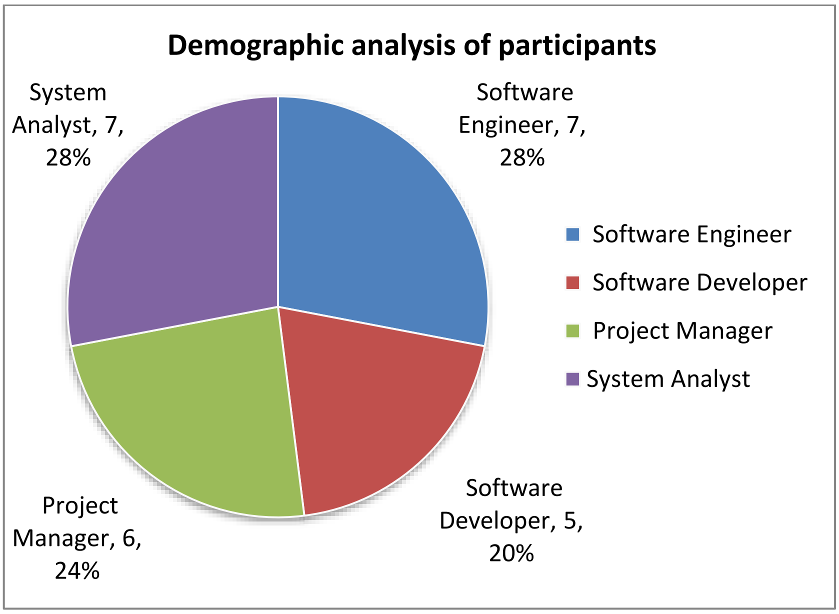


Figure. 5 Demographic analysis of participants

4.1 Comparison of factors that affect Cognitive Load and RE Decision making process

In this section, a comparative analysis is given about the factors that are directly affecting the decision-making process and the quality of decision. The table shows the frequencies of sub-factors used for decision making activities. The following table shows the most commonly used and least commonly used factors for decision making process.

Table. 2 RE factors and questions

Factors	Related questions from Questionnaire								Count
RE Tools	16	17	18	19	21				5
Memory Aid	26	27	30	33	36	37			6
Manpower	13	34	38	42	48	13			6
Information Visualization	20	22	23	24	25	31	40	41	8
Standardization	49	50							2
Time Factor	14	15	32	35	44	45	46	47	8

4.2 Comparison of factors that are directly affecting cognitive load during decision making

In this section an analysis is given about that factor which in particular is directly affecting the process of decision-making during REDSS. These factors also depend on each other. These factors include use of RE tools, the usefulness of these tools, ease of use of these tools, effect of memory load on decision making process, total stress during decision-making process, support in term of information visualization given by RE tools, understanding of cognitive load, visual separation and provision of memory load. For detail analysis T test is performed for which detail is as follows.

This section explains how to conduct a hypothesis test for the difference between paired means. The test procedure, called the **matched-pairs t-test**, is appropriate to apply in this research.

4.2.1 Success and failure based on use of RE Tools and Impact of Change in Industry

Figure 6 shows the success and failure based on use of RE tools and the impact of change in the industry and the results are interpreted below:

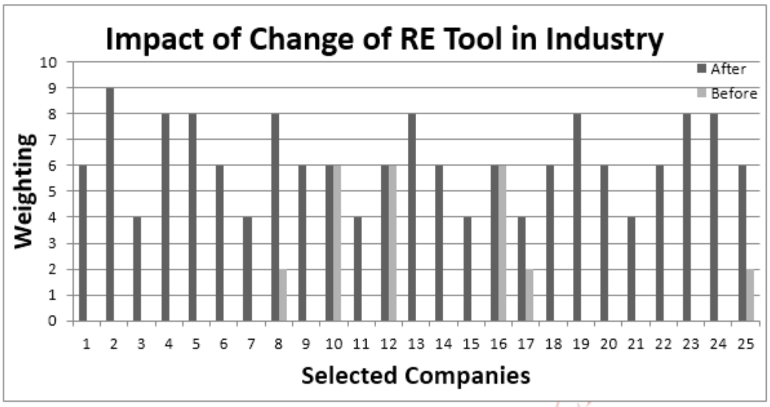


Figure. 6 Impact of change of RE tool

Null Hypothesis H_0: use of RE tools during requirement engineering decision making process has no impact on reduction of cognitive load
Alternative Hypothesis H_1: use of RE tools during requirement engineering decision making process either increase or decrease cognitive load
Analysis of Data Using data, we compute the standard deviation of the differences (s), the standard error (SE) of the mean difference, the degrees of freedom (DF), and the t-score test statistic (t).

s = sqrt [$(\Sigma(di - \bar{d})^2 / (n - 1)$] = 2.634
SE = s / sqrt(n)
= 3.586 / [sqrt(25)] = 0.527
DF = n - 1 = 25 -1 = 24
t = [(x1 - x2) - D] / SE = (d - D)/ SE = 9.945

Degrees of freedom	24
t score	9.945
Cumulative probability: P(T ≤ 9.945)	1.0000

Interpret results. The result given above proves that use of tools is having great impact on quality of decision making.

4.2.2 Success and failure with respect to use of RE Tools in reducing cognitive load

Figure 7 shows the success and failure based on use of RE tools in reducing the cognitive load and the results are interpreted below:

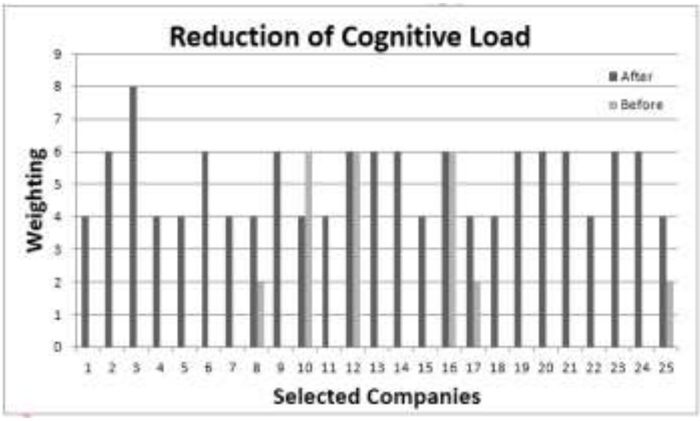


Figure. 7 Reduction of cognitive load

Null Hypothesis H_0: use of RE tools during requirement engineering decision making process is not helpful in reduction of cognitive load
Alternative Hypothesis H_1: use of RE tools during requirement engineering decision making process is helpful either increasing/decreasing of cognitive load
Analysis of Data Using data, we compute the standard deviation of the differences (s), the standard error (SE) of the mean difference, the degrees of freedom (DF), and the t-score test statistic (t).

$s = \text{sqrt} [(\Sigma(di - \bar{d})^2 / (n - 1)] = 2.375$
$SE = s / \text{sqrt}(n) = 3.586 / [\text{sqrt}(22)] = 0.475$
$DF = n - 1 = 25 - 1 = 24$
$t = [(x1 - x2) - D] / SE = (d - D)/ SE = 8.758$

Degrees of freedom	24
t score	8.758
Cumulative probability: $P(T \leq 8.758)$	1.0000

Interpret results. The result given above proves that use of tools are helpful and having great impact of quality of decision making

4.2.3 Success and failure with respect to ease of use of RE tools during experiments

Figure 8 shows the success and failure based on ease of use of RE tools and the results are interpreted below:

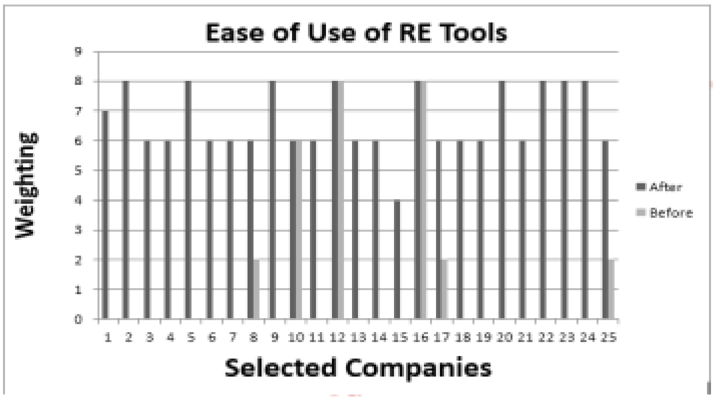


Figure. 8 Ease of use of RE tools

Null Hypothesis H_0: use of RE tools is not easy during requirement engineering decision making process in reduction of cognitive load
Alternative Hypothesis H_1: use of RE tools leads to ease or difficulty during requirement engineering decision making process in managing cognitive load
Analysis of Data Using data, we compute the standard deviation of the differences (s), the standard error (SE) of the mean difference, the degrees of freedom (DF), and the t-score test statistic (t).

s = sqrt [(Σ(di - $\bar{d}$)^2 / (n - 1)] = 2.485
SE = s / sqrt(n) = 3.586 / [sqrt(22)] = 0.497
DF = n - 1 = 25 -1 = 24
t = [(x1 - x2) - D] / SE = (d - D)/ SE = 11.189

Degrees of freedom	24
t score	11.189
Cumulative probability: P(T ≤ 11.189)	1.0000

Interpret results. The result given above proves that use of tools are helpful and having great impact of quality of decision making

4.2.4 Success and failure with respect to memory load during experiments

Figure 9 shows the success and failure based on memory load during experiments and the results are interpreted below:

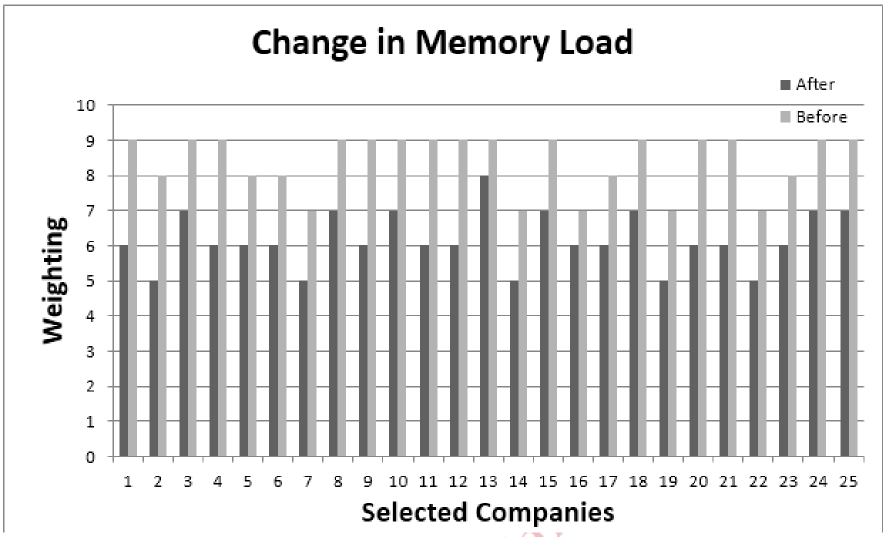


Figure. 9 Change in memory load

Null Hypothesis H_0: memory load during requirement engineering decision making process has no impact on reduction of cognitive load
Alternative Hypothesis H_1: memory load during requirement engineering decision making process either increase or decrease cognitive load
Analysis of Data Using data, we compute the standard deviation of the differences (s), the standard error (SE) of the mean difference, the degrees of freedom (DF), and the t-score test statistic (t).

s = sqrt [(Σ(di - $\bar{d}$)^2 / (n - 1)] = 0.597
SE = s / sqrt(n) = 3.586 / [sqrt(22)] = 0.119
DF = n - 1 = 25 -1 = 24
t = [(x1 - x2) - D] / SE = (d - D)/ SE = -18.754

Degrees of freedom	24
t score	-18.754
Cumulative probability: P(T ≤ -15.087)	0.0000

Interpret results. The result given above proves that use of tools are helpful and having great impact of quality of decision making in term of memory load

4.2.5 Success and failure with respect to stress Level during experiments

Figure 10 shows the success and failure based on stress level during experiments and the results are interpreted below:

Figure. 10 Stress level during experiments

Null Hypothesis H_0: Stress during requirement engineering decision making process has no impact on reduction of cognitive load
Alternative Hypothesis H_1: Stress during requirement engineering decision making process either increase or decrease cognitive load.
Analysis of Data Using data, we compute the standard deviation of the differences (s), the standard error (SE) of the mean difference, the degrees of freedom (DF), and the t-score test statistic (t).

$s = \text{sqrt} [(\Sigma(di - \bar{d})^2 / (n - 1)] = 0.654$
$SE = s / \text{sqrt}(n) = 3.586 / [\text{sqrt}(22)] = 0.123$
$DF = n - 1 = 25 -1 = 24$
$t = [(x1 - x2) - D] / SE = (d - D)/ SE = -18.575$

Degrees of freedom	24
t score	-18.575
Cumulative probability: $P(T < -18.575)$	0.0000

Interpret results. The result given above proves that use of tools are helpful and having great impact of quality of decision making in term of memory load.

4.2.6 Success and failure with respect to Information visualization during experiments

Figure 11 shows the success and failure based on information visualization during experiments and the results are interpreted below:

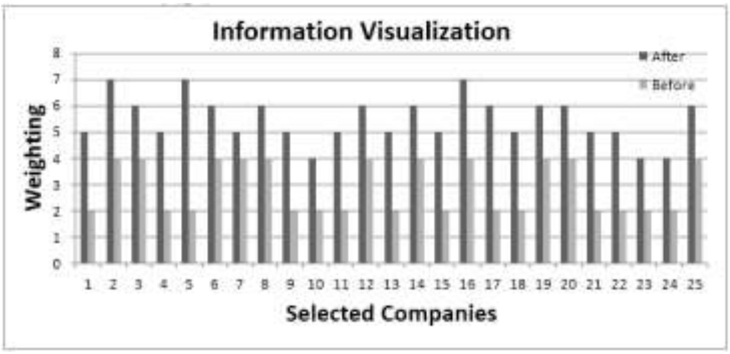


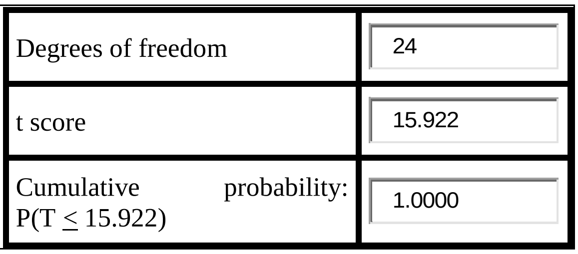
Figure. 11 Information visualization during experiments

Null Hypothesis Ho: Information visualization during requirement engineering decision making process has no impact on reduction of cognitive load

Alternative Hypothesis H1: Information visualization during requirement engineering decision making process either increase or decrease cognitive load

Analysis of Data Using data, we compute the standard deviation of the differences (s), the standard error (SE) of the mean difference, the degrees of freedom (DF), and the t-score test statistic (t).

$s = sqrt [(\Sigma(di - \bar{d})^2 / (n - 1)] = 0.816$
$SE = s / sqrt(n) = 3.586 / [sqrt(22)] = 0.163$
$DF = n - 1 = 25 - 1 = 24$
$t = [(x1 - x2) - D] / SE = (d - D)/ SE = 15.922$



Degrees of freedom	24
t score	15.922
Cumulative probability: P(T ≤ 15.922)	1.0000

Interpret results. The result given above proves that use of tools are helpful and having great impact of quality of decision making in term of memory load.

4.2.7 Success and failure with respect to understanding of cognitive load during experiments

Figure 12 shows the success and failure based on understanding of cognitive load during experiments and the results are interpreted below:

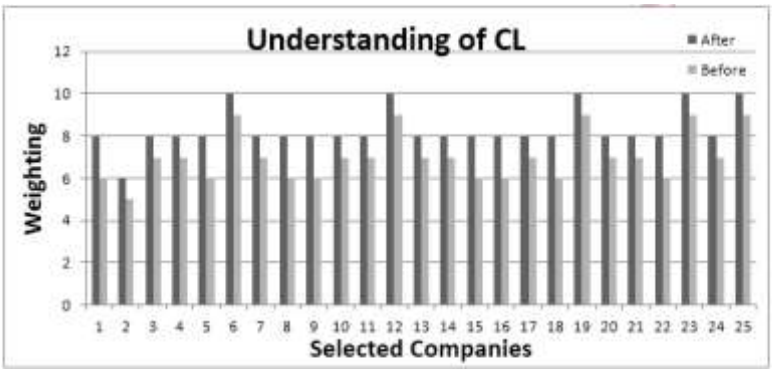


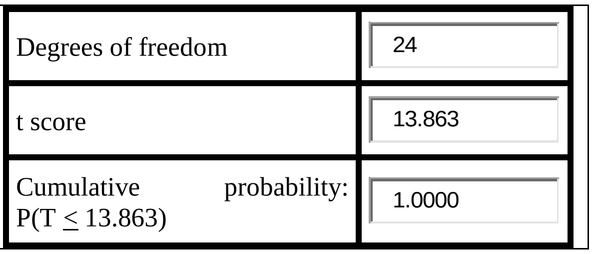
Figure. 12 Understanding of cognitive load

Null Hypothesis Ho: Understanding of cognitive load during requirement engineering decision making process has no impact on reduction of cognitive load

Alternative Hypothesis H1: Understanding of cognitive load during requirement engineering decision making process either increase or decrease cognitive load

Analysis of Data Using data, we compute the standard deviation of the differences (s), the standard error (SE) of the mean difference, the degrees of freedom (DF), and the t-score test statistic (t).

$s = sqrt [(\Sigma(di - \bar{d})^2 / (n - 1)] = 0.476$
$SE = s / sqrt(n) = 3.586 / [sqrt(22)] = 0.095$
$DF = n - 1 = 25 - 1 = 24$
$t = [(x1 - x2) - D] / SE = (d - D)/ SE = 13.863$



Degrees of freedom	24
t score	13.863
Cumulative probability: P(T ≤ 13.863)	1.0000

Interpret results. The result given above proves that use of tools are helpful and having great impact of quality of decision making in term of memory load.

4.2.8 Success and failure with respect to Visual Separation on Cognitive load during experiments

Figure 13 shows the success and failure based on visual separation cognitive load during experiments and the results are interpreted below:

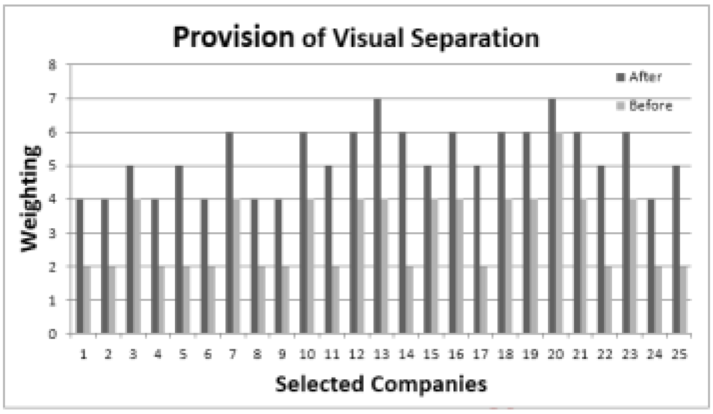


Figure. 13 Visual separation on cognitive load

Null Hypothesis Ho: visual separation support during requirement engineering decision making process has no impact on reduction of cognitive load
Alternative Hypothesis H1: visual separation support during requirement engineering decision making process either increase or decrease cognitive load
Analysis of Data Using data, we compute the standard deviation of the differences (s), the standard error (SE) of the mean difference, the degrees of freedom (DF), and the t-score test statistic (t).

s = sqrt [(Σ(di - $\bar{d}$)^2 / (n - 1)] = 0.707
SE = s / sqrt(n) = 3.586 / [sqrt(22)] = 0.141
DF = n - 1 = 25 -1 = 24
t = [(x1 - x2) - D] / SE = (d - D)/ SE = 15.556

Degrees of freedom	24
t score	15.556
Cumulative probability: P(T ≤ 15.556)	1.0000

Interpret results. The result given above proves that use of tools are helpful and having great impact of quality of decision making in term of memory load.

4.2.9 Success and failure with provision of memory load on cognitive load during experiments

Figure 14 shows the success and failure with provision of memory load on cognitive load during experiments and the results are interpreted below:

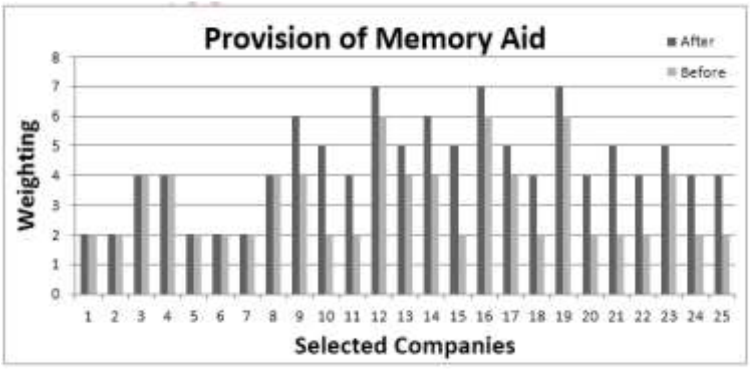


Figure. 14 Provision of memory aid on cognitive load

Null Hypothesis Ho: visual separation support during requirement engineering decision making process has no impact on reduction of cognitive load
Alternative Hypothesis H1: visual separation support during requirement engineering decision making process either increase or decrease cognitive load
Analysis of Data Using data, we compute the standard deviation of the differences (s), the standard error (SE) of the mean difference, the degrees of freedom (DF), and the t-score test statistic (t).

s = sqrt [(Σ(di - $\bar{d}$)^2 / (n - 1)] = 1.052
SE = s / sqrt(n) = 3.586 / [sqrt(22)] = 0.210
DF = n - 1 = 25 -1 = 24
t = [(x1 - x2) - D] / SE = (d - D)/ SE = 5.894

Degrees of freedom	24
t score	5.894
Cumulative probability: P(T $\leq$ 5.894)	1.0000

Interpret results. The result given above proves that use of tools are helpful and having great impact of quality of decision making in term of memory load.

4.2.10 Success and failure with respect to colour text for abstraction help to reduce Cognitive load during experiments

Figure 15 shows the success and failure based on colour text for abstraction help to reduce cognitive load during experiments and the results are interpreted below:

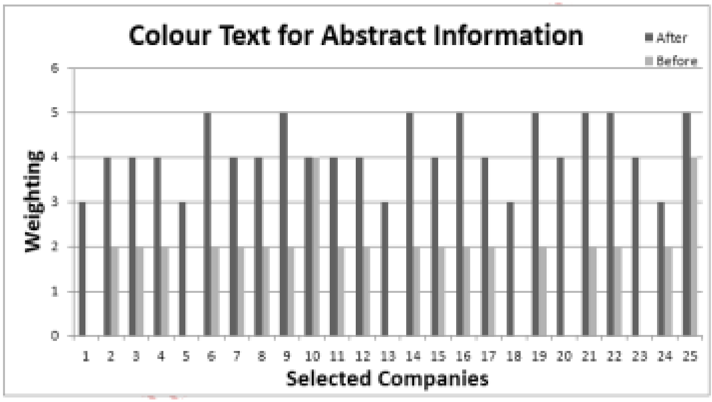


Figure. 15 Colour text for abstract information

Null Hypothesis Ho: colour text for abstraction during requirement engineering decision making process has no impact on reduction of cognitive load
Alternative Hypothesis H1: colour text for abstraction during requirement engineering decision making process either increase or decrease cognitive load
Analysis of Data Using data, we compute the standard deviation of the differences (s), the standard error (SE) of the mean difference, the degrees of freedom (DF), and the t-score test statistic (t).

s = sqrt [(Σ(di - $\bar{d}$)^2 / (n - 1)] = 0.917
SE = s / sqrt(n) = 3.586 / [sqrt(22)] = 0.183
DF = n - 1 = 25 -1 = 24
t = [(x1 - x2) - D] / SE = (d - D)/ SE = 13.311

Degrees of freedom	24
t score	13.311
Cumulative probability: P(T $\leq$ 13.311)	1.0000

Interpret results. The result given above proves that use of tools are helpful and having great impact of quality of decision making in term of memory load.

The remaining figures explain the different factors that have analysed during experimentation of this research.

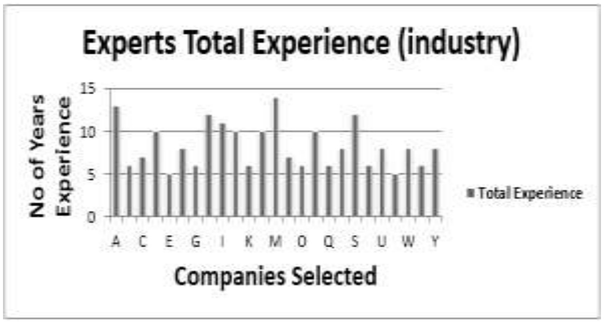


Figure. 16 Experts total Experience for Requirement Engineering in Industry

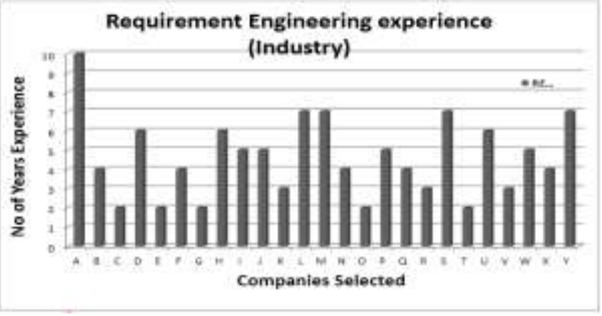


Figure. 17 Experts total Experience and no of projects for Requirement Engineering in Industry

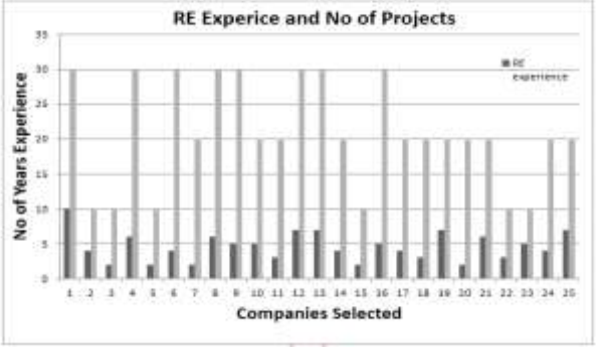


Figure. 18 Experts total experience

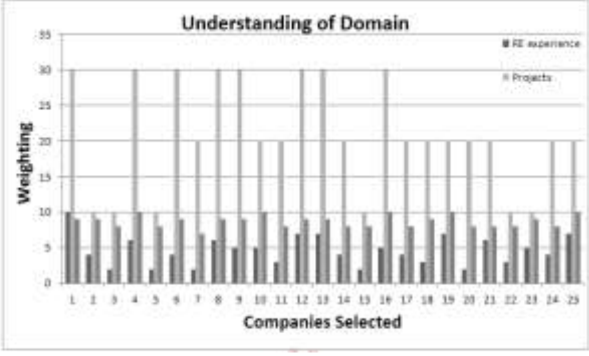


Figure. 19 Understanding of domain and number of projects for RE in Industry

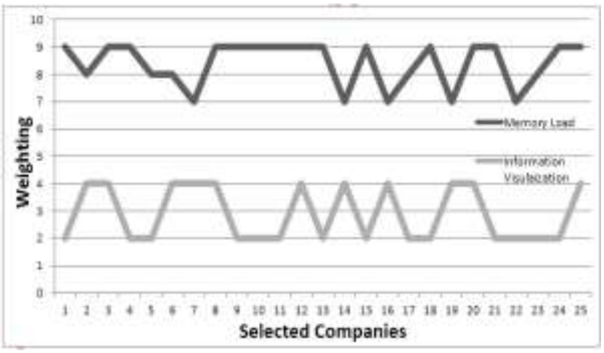


Figure. 20 Comparison of Memory load and information visualization

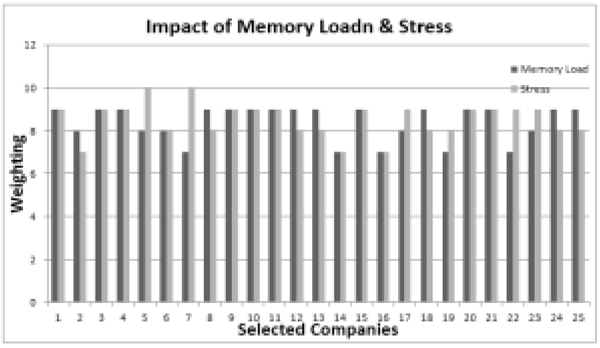


Figure. 21 Memory load and comparison of certification/standard compliances

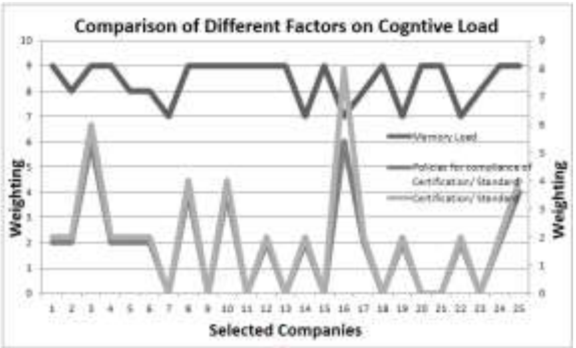


Figure. 22 Comparison of support of standard/ certificates and their compliances

Figure. 23 Organization and compliance of policies

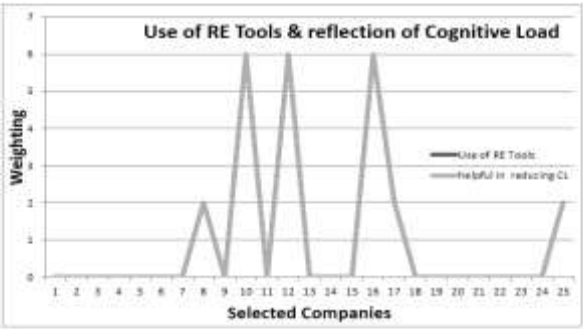


Figure. 24 Use of Re tools and their help in reducing cognitive load

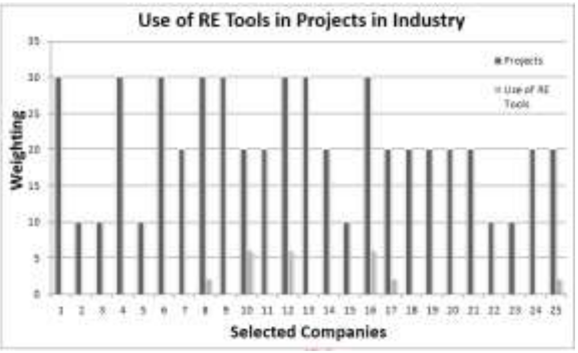


Figure. 25 Comparison or RE tools use different projects

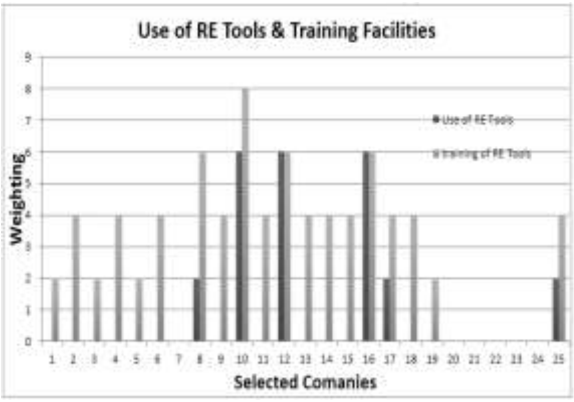


Figure. 26 Comparison of use of RE tools and training support

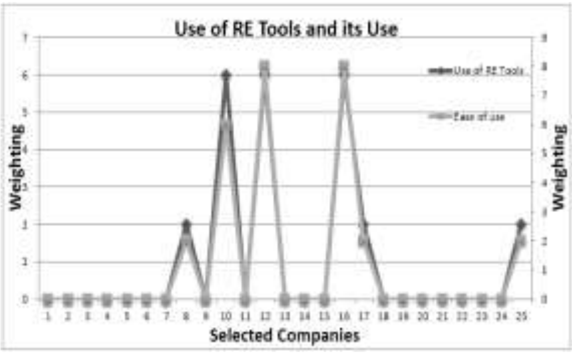


Figure. 27 Comparison of use of RE tools and ease of use

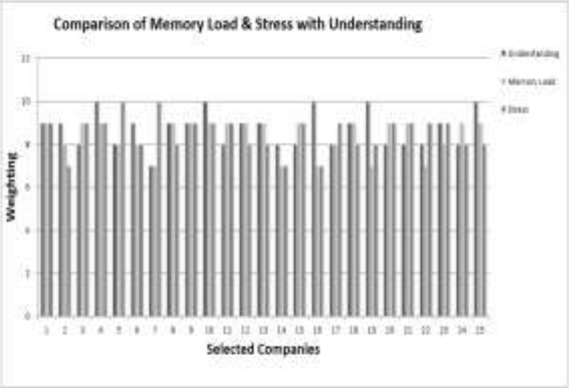


Figure. 28 Comparison of understanding, memory load and stress on project

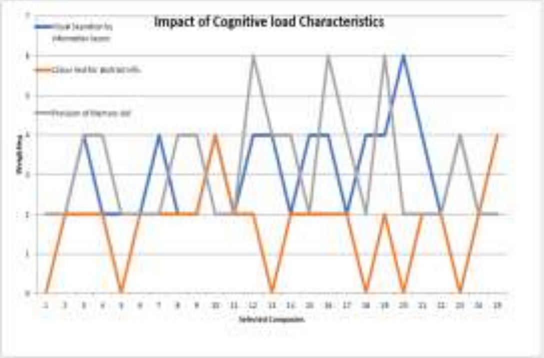


Figure. 29 Comparison of Visual support for reduction of cognitive load

CONCLUSION

The aim of this research work was to explore the impact of cognitive load on decision-making in the domain of RE. For completion of this research work, a survey is performed in 25 software organizations and responses are collected from experts. Special care was taken in designing questions for survey and considered factors that affect RE decision-making. Factors list comprises of use of RE tools, ease of use of RE tools, change in memory load, impact of stress level, information visualisation support, understanding of cognitive load, visual separation, provision of memory aid and colour text abstraction. Experts response was noted in two different phases, once without provision of support for handling cognitive load. Later, the response collection was done after providing necessary support to supplement cognitive load in RE decision-making. For evaluation of responses, T-test statistical analysis was performed to measure the change impact. A significant improvement was observed in the results after facilitating experts with different provisions of handling cognitive load. Moreover, empirical evidences conclude that the reduction of cognition load improves quality of decision-making in RE. Factors required for reduction of cognitive load are very subjective in nature and their representation in RE tools needs further investigation in the social sciences domain. Based on empirical results it is suggested that more knowledge of cognitive load management and training in this area is necessary. Most of the industry experts are still inefficient in term of use of RE tools and following practices in the field of requirements engineering.

ACKNOWLEDGEMENT

Special thanks to International Islamic University, Islamabad, Pakistan for providing support in order to complete this research. Moreover, we are also thankful to all those who have supported us in completion of this research.

REFERENCES

1. Zave, P., *Classification of research efforts in requirements engineering.* ACM Computing Surveys (CSUR), 1997. 29(4): p. 315-321.
2. Ebert, C. and R.J. Wieringa, *Requirements engineering: Solutions and trends*, in *Engineering and managing software requirements*. 2005, Springer. p. 453-476.
3. Nuseibeh, B. and S. Easterbrook. *Requirements engineering: a roadmap*. in *Proceedings of the Conference on the Future of Software Engineering*. 2000: ACM.
4. Newell, A. and H.A. Simon, *Human problem solving*. Vol. 104. 1972: Prentice-Hall Englewood Cliffs, NJ.
5. Hogarth, R., *Judgement and Choice*. 2nd Edition ed. 1987, New York: Wiley.
6. Sweller, J., J.J. Van Merrienboer, and F.G. Paas, *Cognitive architecture and instructional design.* Educational psychology review, 1998. 10(3): p. 251-296.
7. Gulliksen, J. and B. Göransson, *Användarcentrerad systemdesign [User-centred system design]*. 2002: Lund, Sweden: Studentlitteratur (in Swedish).
8. Sommerville, I., *Software Engineering*. 6th ed. 2002, Wokingham, England: Addison-Wesley.
9. Aurum, A. and C. Wohlin, *Requirements engineering: setting the context*, in *Engineering and managing software requirements*. 2005, Springer. p. 1-15.
10. Matulevičius, R. *How requirements specification quality depends on tools: A case study*. in *International Conference on Advanced Information Systems Engineering*. 2004: Springer.
11. Sutcliffe, A., *User-centred requirements engineering*. 2012: Springer Science & Business Media.
12. och Dag, J.N. and V. Gervasi, *Managing Large Repositories of Natural Language Requirements*, in *Engineering and Managing Software Requirements*. 2005, Springer. p. 219-244.
13. Sommerville and G. Kotonya, *Requirements engineering: processes and techniques*. 1998: John Wiley & Sons, Inc.
14. Dahlstedt, Å.G. and A. Persson, *Requirements interdependencies: state of the art and future challenges*, in *Engineering and managing software requirements*. 2005, Springer. p. 95-116.
15. Zowghi, D. and C. Coulin, *Requirements elicitation: A survey of techniques, approaches, and tools*, in *Engineering and managing software requirements*. 2005, Springer. p. 19-46.
16. Grünbacher, P. and N. Seyff, *Requirements negotiation*, in *Engineering and managing software requirements*. 2005, Springer. p. 143-162.
17. Hoffmann, M., et al. *Requirements for requirements management tools*. in *Requirements Engineering Conference, 2004. Proceedings. 12th IEEE International*. 2004: IEEE.
18. Eriksson, U., *Kravhantering för IT-system [Requirements engineering for IT systems]*. 2007: Lund, Sweden: Studentlitteratur.
19. Matulevičius, R., *Process support for requirements engineering: A requirements engineering tool evaluation approach.* 2005.
20. Lang, M. and J. Duggan, *A tool to support collaborative software requirements management.* Requirements engineering, 2001. 6(3): p. 161-172.
21. Carvallo, J.P., X. Franch, and C. Quer, *A Quality Model for Requirements Management Tools.* Requirements Engineering for Sociotechnical Systems, Idea Group Publishing, 2004.
22. *INCOSE: Requirements management tools survey*. . 12/10/2016]; Available from: http://www.paper-review.com/tools/rms/read.php.
23. Heindl, M., et al. *Value-based selection of requirements engineering tool support*. 2006: IEEE.
24. Aurum, A. and C. Wohlin, *The fundamental nature of requirements engineering activities as a decision-making process.* Information and Software Technology, 2003. 45(14): p. 945-954.

25. Evans, R., S. Park, and H. Alberts. *Decisions not requirements: decision-centered engineering of computer-based systems*. in *Engineering of Computer-Based Systems, 1997. Proceedings., International Conference and Workshop on*. 1997: IEEE.
26. Regnell, B., et al. *Requirements Mean Decisions!–Research issues for understanding and supporting decision-making in Requirements Engineering*. in *First Swedish Conference on Software Engineering Research and Practise: Proceedings*. 2001: Citeseer.
27. Ruhe, G., *Decision support in requirements engineering*, in *Engineering and Managing Software Requirements*. 2005, Springer. p. 267-286.
28. Macaulay, L.A., *Requirements engineering*. 1996: Springer-Verlag.
29. Mintzberg, H., D. Raisinghani, and A. Theoret, *The structure of" unstructured" decision processes.* Administrative science quarterly, 1976: p. 246-275.
30. Anthony, R.N., *Planning and Control Systems: A Framework for Analysis [by]*. 1965: Division of Research, Graduate School of Business Administration, Harvard University.
31. Turban, E., *Decision support and expert systems: management support systems*. 1990: Prentice Hall PTR.
32. Mallach, E., *Understanding Decision Support and Expert Systems. Burr Ridge, IL: Richard D. Irwin*. 1994, Inc.
33. Power, D., *Decision Support Systems: Concepts and Resources for Managers.* Studies in Informatics and Control, 2002. 11(4): p. 349-350.
34. Alter, S., *Decision support systems: current practice and continuing challenges*. 1980.
35. Marakas, G.M., *Decision support systems in the 21st century*. Vol. 134. 2003: Prentice Hall Upper Saddle River, NJ.
36. Schneider, W., *Att köra över människors inneboende autopilot [To run over the inherent autopilot of humans]*. Människor, datateknik, arbetsliv, 1993: p. 99-114.
37. Aybuke, A. and W. Claes, *Engineering and Managing Software Requirements*. 2005, Springer-Verlag Berlin.
38. Ruhe, G., *Software engineering decision support–a new paradigm for learning software organizations*, in *Advances in Learning Software Organizations*. 2003, Springer. p. 104-113.
39. Orasanu, J. and T. Connolly, *The Reinvention of Decision Making. In (Eds.) G. Klein, J. Orasanu, J. Calderwood & CE Zsambok.* Decision Making in Action, 1993.
40. Alenljung, B. and A. Persson. *Factors that affect requirements engineers in their decision situations: A case study*. in *11th International Workshop on Requirements Engineering: Foundation for Software Quality; in conjunction with CAiSE'05, 13-14 June 2005, Porto, Portugal*. 2005: Universität Duisburg-Essen.
41. Alenljung, B. and A. Persson, *Supporting requirement-based decision-making in the software engineering process: A position paper.* 2004.

AUTHORS PROFILE

Falak Sher received his MCS from Al-Khair University AJK Pakistan. He received his MS degree in Software Engineering from International Islamic University Islamabad Pakistan. Currently, he is pursuing his Ph.D. in Computer Science from Universiti Teknologi Malaysia. His research interests are in software engineering, decision support systems, artificial intelligence, software requirements engineering, and software quality assurance. He is an experienced teacher of computing at Beaconhouse School System Islamabad, Pakistan and has a teaching experience of more than 15 years.

Shahbaz Ahmed Khan Ghayyur received his BS degree in Computer Science and MS degree in Software Engineering from International Islamic University Islamabad Pakistan. Currently, he is pursuing his Ph.D. in Computer Science from Preston University, Islamabad, Pakistan. His research interests are in software engineering, decision support systems, artificial intelligence, software requirements engineering, and software quality assurance. He is serving as an Assistant Professor in Department of Computer Science and Software Engineering in International Islamic University Islamabad Pakistan. He is also co-editor-in-chief of Journal of Theoretical and Applied Information Technology.