



AENSI Journals

Australian Journal of Basic and Applied Sciences

ISSN: 1991-8178

Journal home page: [www.ajbasweb.com](http://www.ajbasweb.com)



## Exploring the Relationships Between Requirements Relationships Knowledge and Requirements Prioritisation Towards Software Development Project Success

A.A. Ruhaya and W. Bernard

University of Technology, Sydney; Enterprise Strategy Consulting, Sydney, Australia

### ARTICLE INFO

#### Article history:

Received 25 January 2014

Received in revised form 12

March 2014

Accepted 14 April 2014

Available online 25 April 2014

#### Keywords:

Requirements Relationships

Knowledge, Requirements

Prioritisation, Success

### ABSTRACT

In software development project, requirements are very important and they need to be managed effectively for the project to be successful. One of the important parts in managing requirements is prioritising the requirements. However, prioritising requirements is difficult as requirements do not stand alone and they are typically related to one another in several ways. These relationships may affect individual requirement as well as the whole process of software development project. Hence, requirements relationships need to be captured, analysed, and managed carefully to avoid any costly mistake. This research aims to investigate how these types of requirements relationships impact requirements prioritisation as well as the success of software development project. We examined the impacts from the perspective of business analyst. Partial least square (PLS) and structural equation modelling (SEM) were used to analyse the data. The findings from this research can be used as a guide on working with requirements relationships knowledge useful for business analysts and research community.

© 2014 AENSI Publisher All rights reserved.

**To Cite This Article:** A.A. Ruhaya and W. Bernard., Exploring the Relationships Between Requirements Relationships Knowledge and Requirements Prioritisation Towards Software Development Project Success. *Aust. J. Basic & Appl. Sci.*, 8(5): 440-449, 2014

### INTRODUCTION

The success of a software development project is a concern for any related stakeholders. There are various factors of success discussed in literature. One of the many factors is requirements prioritisation. Indeed, previous researchers indicate the importance of requirements prioritisation in ensuring software development project success (Berander and Andrew 2005). The success of a software development project is often determined by the ability to satisfy the user needs (Agarwal and Rathod 2006, Nasir and Shahibudin 2011, Standish group 1995, Wohlin *et al* 2000). Capturing and specifying the correct requirements and planning suitable releases with the right functionality is an essential step towards the success of a software development project. Wrong requirements selection and planning during a particular software development project may create chaos when users resist using the end product. Thus, the proper selection and prioritization of requirements are very important. Berander and Andrew (2005) indicate that, requirements prioritisation is a crucial step towards making good decisions regarding product planning for single and multiple releases. Requirements selections and release planning are also asserted to be influenced by and also may impact the estimation of effort and cost for a software development project (Haerman and Daneva 2008; Karlson *et al* 2007). However, to plan, select and prioritise requirements for a software development project is not a simple task. Most software projects are complex where they have more candidate requirements than can be realised within the time and cost constraints (Berander and Andrew 2005). Accordingly, to properly select and prioritise requirements, the knowledge of the relationships between requirements is needed.

In this paper, the information of the relationships between requirements is addressed as requirements relationships knowledge (RRK). RRK is concerns on how requirements are related to one another and other artefacts during the software development project. Although RRK is not problematic, the knowledge would affect other aspects of software development project and the project as a whole. Requirements relationships knowledge is critically needed in prioritisation of requirements (Carlshamre *et al* 2001, Davis 2005). When performing prioritisation, if we remove or add a requirement in a specific release, we need to know what that does to other requirements in term of necessity, cost, functionality and etc. For instance, if a highly prioritise

**Corresponding Author:** Ruhaya Ab. Aziz, Department of Software, Faculty of Engineering and IT University of Technology, Sydney.  
E-mail: 10603948@uts.edu.au / ruhaya@uthm.edu.my

requirement A depends on requirement B which is very costly and has the lower prioritisation rate, then A will not be released, unless there are sufficient resources to release B too. A product with value added characteristics may be released over several versions rather than putting all the best features in one release to maintain a continuous interest from the customers. However, to allow this decision to be made the combination and the interrelationships among the set of requirements of the products need to be managed and selected carefully to ensure the best features is maintained. Both examples illustrate that how the knowledge of requirements relationships is essential to make decision for requirements prioritisation. Several researchers have addressed the needs of requirements relationships knowledge within their study on requirements prioritisation. For example in release planning, Carlshamre *et al* (2001) indicate that priority is a key determinant in increment planning but it is difficult and possibly impossible to schedule requirements based on priority only without knowing how the requirements are related to one another. Similarly, Davis (2005) refers to prioritisation as 'requirements triage' where he claims that successful triage requires knowledge of six elements including the relationships among requirements. Moreover, Giesen and Volker (2002) have combined the information of requirements interdependencies and conjoint analysis approach to determine stakeholder preferences.

However, is this knowledge really significant in software development project? If yes, how can this knowledge impact requirements prioritisation as well as the success of software development project? There are studies that focus on examining how requirements prioritisation impacts success (Berander and Andrew 2005, Haerman and Daneva 2008; Karlson *et al* 2007). There are also some studies that investigating how RRK impacts requirements prioritisation (Carlshamre *et al* 2001, Davis 2005). But there is limited studies focusing on how RRK impacts requirements prioritisation as well as success of SDP. Thus, this paper aims to discuss these issues further and extend the literature on the interrelationships between RRK, requirements prioritisation and the related issues that have impacts on project success.

The rest of this paper is organised into 4 sections. Firstly, the research model will be discussed. Secondly, research method, which mainly concerns on the development and the validation of the requirements relationships instrumentation design, will be discussed in the next section. Thirdly, this paper will continue to discuss the result of the research and finally, this paper will present a discussion and conclusion including the implications of the work in both research and practice.

#### **Research Model:**

A particular requirements specification is considered in good quality when a set of requirements or software requirements specification (SRS) having all the good characteristics that listed as proposed by IEEE-830-1998 recommended practices for SRS (IEEE-830-1998). According to the practice; requirements should be correct, unambiguous, complete, consistent, ranked for importance and/ or stability, verifiable, modifiable and traceable. Thus, an SRS developed for a particular software project should fulfil all the characteristics listed to ensure the quality of the requirements. One of the characteristics listed is ranked for importance which means to ensure the quality of requirements, requirements should be prioritized. IEEE software engineering terminology defines priority as the level of importance assigned to an item. Consequently, to prioritise requirements correctly the comprehensive understanding of requirements is needed.

However, to fully understand requirements, it is important to know how requirements are related to one another. Knowledge on how each requirement is related to one another may assist stakeholders to make informed decision in accomplishing many things that involve in managing requirements (Dahlstedt & Persson 2005, Ozkaya 2006). The knowledge will also help stakeholders to structure and organise the requirements (Diev, 2007). In a particular software development project, each requirement elicited needs to be analysed and documented in requirements specification document for future references and also in the planning for implementation. In the process, prioritisation and selection of requirements is essential. Without careful selection and prioritisation may doom the whole implementation of a particular software development project. Moreover, as requirements specification document will acts as a contract between the software developers and their customer (IEEE 830-1998), constructing the structured and organised requirements with correct prioritisation is critically needed especially when it involves large amount and complex set of requirements. The decision on the prioritisation and selection will need stakeholders to consider many factors including the importance of the requirements, cost and also the conflict between quality and functionality (Berander and Andrew 2005, Herrman and Daneva 2008). Moreover, to manage conflict and to avoid costly mistake, the knowledge of the interrelationships between requirements (RRK) is critically needed. For instance, an importance requirement might not be selected for implementation if by selecting the requirement, we need to also select some other related requirements which will incur higher cost and may affect the quality of the software system. Indeed, the importance of considering RRK in requirements prioritisation has been asserted by many (e.g. Carlshamre *et al* 2001, Davis 2005, Berander and Andrew 2005, Herrmann & Daneva 2008). Thus, the above discussion leads the following hypothesis:

H1: Requirements Relationships Knowledge (RRK) has significance positive impacts on requirements prioritisation

The importance of requirements prioritisation in a particular software development project has been asserted by many. Previous researchers indicate that there are at least two approaches can be used to resolve conflict between requirements which are requirements negotiation and requirements prioritisation (Barney, Aurum and Wohlin 2008). Both of the approaches are important to make the right decision on the selection of requirements for the next release and also for the implementation. Requirements selections and release planning are asserted to be influenced by and also may impact the estimation of effort and cost for a software development project (Haerman and Daneva 2008; Karlson *et al* 2007). Requirements prioritisation is also importance in software quality (Babar, Ramzan, Ghayyur 2011) and system maintenance (Asghar, Marchetto and Susi 2013). As an important factor for the success of software development project, there are many studies to further identify the best technique in requirements prioritisation (e.g. Kukreja *et al* 2012, Marjaie & Kulkarni 2010, Kukreja 2013). Thus, it is arguably that requirements prioritisation and selection are importance and consequently impact the success of a software development project. Thus, lead to another hypothesis for this research:

H2: Requirements prioritisation has significance impact on success

Accordingly, as RRK has significant impact on requirements prioritisation and requirements prioritisation has significance impacts on the success of software development project, this paper intends to investigate further and hypothesize that RRK will indirectly (dotted line in the research model) has significance impacts on the success of software development project (H3). Furthermore, this paper also hypothesizes that requirements prioritisation mediate the relationships between RRK and the success of software development project.

H3: RRK has significance impacts on requirements prioritisation and success of software development project

H4: Requirements Prioritisation will mediate the relationships between RRK and the success of software development project

The overview of the research model and the related hypotheses are illustrated in figure 1. The constructs and the items represent the construct in the research model are further illustrated in figure 2. This paper will continue with the discussion on how the research model is validated and tested in the next section.

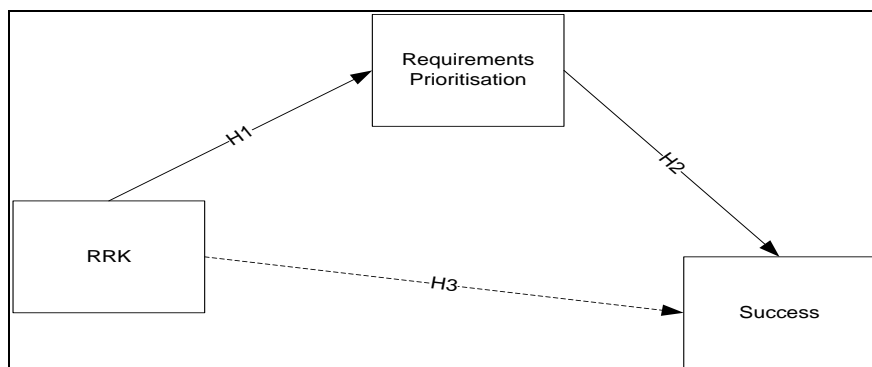


Fig. 1: The research model and related hypotheses.

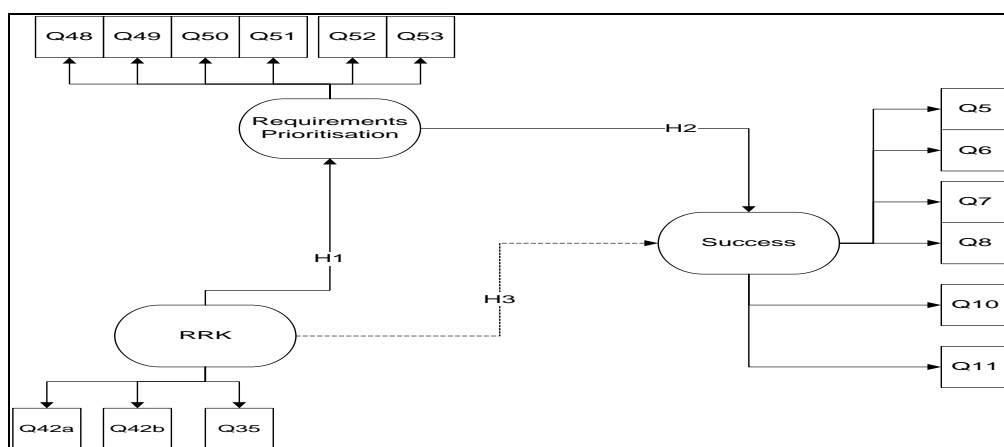


Fig. 2: The initial constructs in the model.

**Research Method:****A. Participants:**

In this study, non-probability sampling of purposive sampling was used. Business analysts and any stakeholders involved in requirements in their software development project were chosen and those not involved were excluded from the sample. 173 business analysts and related stakeholders (to the software development projects) were recruited. Inclusion criteria were as follows: the participants were stakeholders involved in requirements management in their software development project. 60% of the participants were business analysts and system analysts (refer table 3). About 10-20% of the respondents are from Australia and most of the participants are practitioners in Malaysia industry. The survey data shows that, most of the respondents are from the medium and large organisation (refer to table 1 and Australian Bureau of statistics classification of business framework (office of small business, 1999). Accordingly, in table 2, the survey data shows that the industry domain of most of the organization is from Information Technology and Telecommunication (41%) and Infrastructure and government (30%). In addition, the respondents mostly have about 2-5 years (37%) and about 6-9 years (25%) experience in writing requirements which represent about 62% of the whole respondents (table 4).

**Table 1:** Employment Information (source: Survey data, 2011-2012).

Items	Frequency	Percentage
Less than 10	15	9%
11-100	84	49%
101-500	32	18%
More than 500	41	24%

**Table 2:** Industry domain (source: Survey data, 2011-2012).

Items	Frequency	Percentage
Factory automation	3	1.7=2
Financial	10	5.7=6
Infrastructure & Government	52	30
Medical	4	2.3=2
IT & Telecommunication	72	41.1=41
Transportation	2	1
Others	32	18.3=18

**Table 3:** Designation of the respondents(source: Survey data, 2011-2012).

Items	Frequency	Percentage
Business Analyst	18	10.5
Business Analyst Manager	6	3.5
System Analyst	80	47
System Designer	20	11.6
Tester	4	2.3
Others	43	25.1

**Table 4:** Experience in Writing Requirements (source: Survey data, 2011-2012).

Items	Frequency	Percentage
Within one year	32	19
2-5 years	63	37
6-9 years	43	25
10-15 years	23	13
More than 15 years	8	5

**B. Data Collections:**

There were about 380 self-administered questionnaires used for gathering the data from the respondents. Several methods of questionnaire distribution were employed: 1) a number of questionnaires were mailed to the respondents; 2) a number of questionnaires were emailed (on-line survey); 3) a number of questionnaires were completed using drop-off survey method. A total of 210 questionnaires were received but only about 173 questionnaires were usable for analysis. This translates to about 55.3% response rate and only 45.5% were considered effective response rate.

**C. Structural Equation Modeling:**

This study is a part of a research that examined the impacts of requirements relationships on the other elements and activities in a software development project that may impact success. The activities and elements may have direct and indirect relationships; they will impact each other and thus the success of a particular software development project as a whole. Therefore, in order to validate and investigate the relationships further and the impacts that they give to one another, structural equation modelling (SEM) was used. SEM is a statistical technique for testing and estimating causal relationships using a combination of statistical data and

qualitative causal assumption. This approach is more confirmatory rather than exploratory, thus is more suitable for theory testing than theory development. SEM is a very general, powerful multivariate analysis technique that includes specialised versions of a number of other analysis methods as special cases. SEM is not designed for a single statistical technique but it is also a family of related procedures (Kline 1995). Other related terms used are covariance structure analysis, covariance structure modelling, and analysis of covariance structures.

Furthermore, SEM can be categorised into two approaches, which are: 1) covariance-based approach, which is related to tools such as EQS and AMOS; and 2) variance-based approach, which is related to PLS. Thus, in this research, PLS was chosen. This is discussed further in the next section.

#### D. Partial Least Square (PLS):

In this research, partial least square (PLS) was used because of the following reasons: 1) research on requirements relationships is relatively new; and 2) there is no measurement model that is already available. PLS could be a suitable technique to be used when the phenomenon to be investigated is relatively new (Chin 1998). In addition, this research involved complex relationships between latent variables and many indicator variables. In this research, there were about 7 constructs and more than 53 indicators used. Hair *et al* (2012) indicated that if there are more than 7 constructs with more than 50 indicators, thus PLS is a more suitable than other methods. The next section presents the assessment of the goodness of measure of these constructs in terms of their validity and reliability within the research framework.

#### E. Measures and Goodness of Measures:

A questionnaire using five-point Likert scale was used to gather data for each construct of the research model. Some of the instruments were newly developed where most of the questions were created based on the theory from literature and other empirical studies. There were also some instruments that were adapted from previous literature. The final constructs of the model are illustrated in Table 5:

Table 5: Constructs in the model.

Construct	Item	Description
Success	Q5 (SC1)	The outcome of the project meets the business goal.
	Q6 (SC2)	The outcome of the project meets all the specified requirements.
	Q7 (SC3)	The overall quality of the developed application / product is high.
	Q10 (SC4)	The project is completed within scope.
	Q11 (SC5)	The requirements-related tasks (e.g. requirements specification, requirements management) have been completed successfully in the project.
Requirements Relationships Knowledge	Q42a (RRK1)	The relationships between requirements that exist between the components are considered when deciding to implement the solution.
	Q42b (RRK2)	The relationships between requirements that exist between the components are considered when planning the schedule for the design/development team to complete the task.
	Q35 (RRK3)	Before implementing a change to a particular requirement, any possible impact it will cause to other requirements will be considered.
Requirements Prioritisation	Q50 (RP2)	In the requirements prioritisation, the importance of the requirements according to the user needs is checked.
	Q49 (RP1)	In the requirements prioritisation, the time estimated for implementing the set of requirements is checked.
	Q48 (RP3)	In the requirements prioritisation, the cost and effort for implementing the set of requirements are checked.

#### F. Goodness of Measure:

There are two main criteria used for evaluating goodness of measures, which are validity and reliability. The combination of both is essential to ensure the quality of a research (Trochim and Donnelly 2008). Trochim and Donnelly (2008) also indicated that reliability refers to repeatability or consistency. A measure is considered reliable if it gives the same result over and over again. On the other hand, validity is about how well a developed instrument measures the particular concept that is intended to be measured (Sekaran and Bougie 2010). The validity and reliability measures of this research model are discussed in the next section.

#### G. Construct Validity:

Construct validity is concerned about the degree to which inferences can legitimately be made from the operational constructs in a study to the theoretical constructs on which those operational constructs are based on (Trochim and Donnelly 2008). Sekaran and Bougie (2010) indicated that construct validity is used to testify how well the results obtained from the use of the measure fit the theories around which the test is designed. Thus, to assess how the instrument fits the concept as theorised, convergent and discriminant validity can be used. Firstly, the respective value of loadings and cross loadings in Table 6 were examined to assess whether there were any problems with any particular items. A cut-off value for loadings at 0.5 was considered as significant (Hair *et al* 2010). If there were any items with a loading of higher than 0.5 on two or more factors, then they were deemed to be having significant cross loadings. From Table 6, we can see that all the items that measured a

particular construct would load highly on the construct and would have lower loadings values on other constructs hence confirming construct validity.

**Table 6:** Loading and Cross Loading.

	RRK	RP	SC
RR1	0.926	0.410	0.246
RR2	0.930	0.449	0.247
RP1	0.329	0.802	0.201
RP2	0.451	0.910	0.347
SC1	0.249	0.212	0.716
SC2	0.181	0.204	0.802
SC3	0.279	0.279	0.782
SC4	0.159	0.277	0.690
SC5	0.116	0.262	0.733

#### **H. Convergent Validity:**

Secondly, the validity test was continued with the convergent validity. It is the degree to which multiple items are in agreement to measure the same concept. Factor loadings, composite reliability, and average variance extracted (AVE) were used to assess the convergent validity. This practice was proposed by Hair *et al.* (2010). In this research, the test showed that the factor loadings for all items exceeded the recommended value of 0.5 (Hair *et al.* 2010). Next, composite reliability values (refer to Table 7), which illustrate the degree to which the construct indicators indicated the latent, ranged from 0.847 to 0.925. This exceeded the recommended value of 0.7 (Hair *et al.* 2010). Finally, the average variance extracted measured the variance captured by the indicators relative to measurement error. It should be greater than 0.5 to justify the use of the construct (Barclay *et al.* 1995). As shown in Table 7, the AVE was in the range of 0.556 to 0.861.

**Table 7:** Results of measurement model.

Measurement item	Loading	CR <sup>a</sup>	AVE <sup>b</sup>
RRK1	0.926	0.925	0.861
RRK2	0.930		
RP1	0.802	0.847	0.736
RP2	0.910		
SC1	0.716	0.862	0.556
SC2	0.802		
SC3	0.782		
SC4	0.690		
SC5	0.733		

**Table 8:** Summary Results of the Model Construct.

Model Construct	Measurement Item	Standardised Estimate	T-Value
Requirements Relationships Knowledge	RRK1	0.926	38.286
	RRK2	0.930	42.992
Requirements Prioritisation	RP1	0.802	13.517
	RP2	0.910	39.156
Success	SC1	0.716	13.616
	SC2	0.802	22.313
	SC3	0.782	19.591
	SC4	0.690	11.708
	SC5	0.733	14.106

Moreover, summarisation of the results for the measurement model is illustrated in Table 8. The results show that all of the 3 constructs, namely Requirements Relationships Knowledge, Requirements Prioritisation and Success were all valid measures of their respective constructs based on their parameter estimates and statistical significance.

#### **I. Discriminant Validity:**

Thirdly, the test was continued to validate the discriminant validity. Discriminant validity is the degree to which items differentiate among constructs where they show that the measures that should not be related are in reality not related. This validity test was assessed by examining the correlations between measures of potentially overlapping constructs. The items should have the highest loading value on their own constructs in the model, and the average variance shared between each construct and its measures should be greater than the variance shared between the construct and other constructs (Compeau *et al.* 1999). Table 9 shows that the squared correlation for each construct is less than the average variance extracted by the indicators measuring the construct indicating adequate discriminant validity. As a result, the measurement model demonstrated adequate convergent validity and discriminant validity.

**Table 9:** Discriminant Validity of Constructs.

4. Req. Relationships	0.861		
5. Req. Prioritisation	0.214	0.736	
7. Success	0.070	0.110	0.556

**J. Reliability Analysis:**

Reliability is about the quality of measurement. Reliability in a research is the extent to which a measurement procedure yields the same answer however and whenever it is carried out (Kirk and Miller 1986). One of the general classes of reliability is the internal consistency reliability that is used to assess the consistency of result across items within a test (Trochim and Donnelly 2008). In this research, Cronbach's alpha coefficient was used to assess the reliability of the inter item consistency of the measurement items. Table 6 shows the summarisation of loadings and alpha values. All the alpha values listed in Table 6 are above 0.6, which are conforming to what have been suggested by Nunnally and Berstein (1994). The composite reliability values also ranged from 0.847-0.925. Composite reliability values are another approach similar to Cronbach's alpha for internal consistency reliability estimate where a composite reliability of 0.7 or more is considered acceptable (Fornell and Lackler 1981). Thus, it can be concluded that the measurements were reliable.

**Table 6:** Results of Reliability Test.

Constructs	Measurement items	Cronbach's alpha	Loading range	Num. of items
Req. Relationships	RRK1,RRK2	0.839	0.926-0.930	2(3)
Req. Prioritisation	RP1,RP2	0.651	0.816-0.836	2(3)
Success	SC1, SC2,SC3, SC4, SC5	0.799	0.690-0.802	5(7)

The measurement model used in this research has been assessed on the reliability and validity in this section. Next section will continue to discuss the hypothesis testing and the result of the analysis.

**Result:**

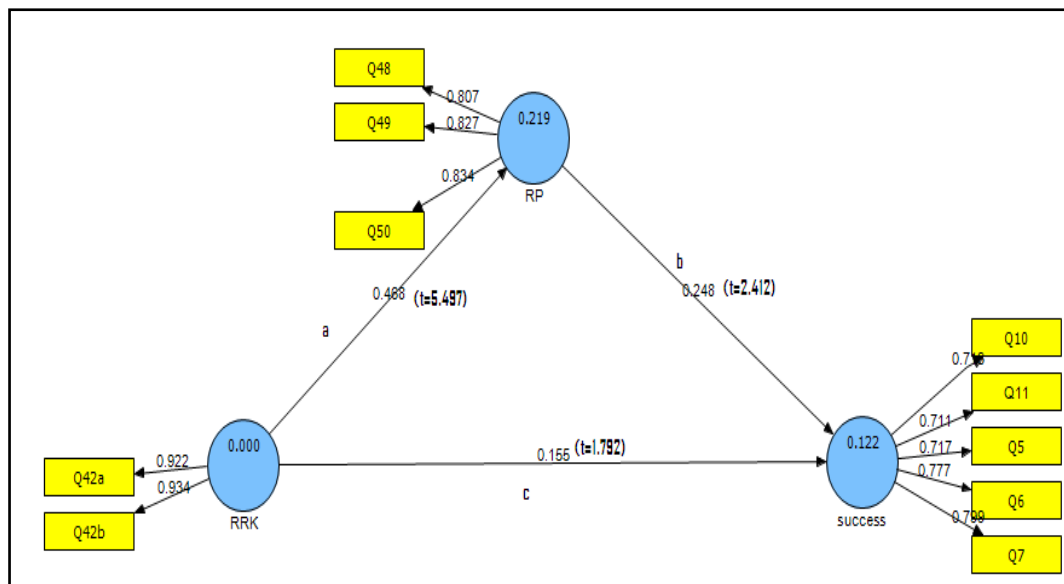
The result from the analysis shows that there are significant relationships between requirements relationships knowledge, requirements prioritisation (RP) and success of software development project. The result of the analysis shows that the path coefficient value for RRK->RP is 0.468 and the path coefficient for RP-> success is 0.248. Both values are more than the range of (0.20-0.30) in which has been indicated as acceptable (Chin, 1998b). The findings indicate that hypothesis H1 and hypothesis H2 were supported in this research. Thus, it can be concluded that there are significant relationships exist between the three constructs.

In addition, mediator effect analysis has also been conducted. The analysis reports that, there is exists mediator relationships between the three constructs. Figure 3 illustrates the analysis which shows that the initial coefficient for the three constructs. In order to allow for mediator analysis, there are certain criteria that need to be fulfilled. Firstly, the predictor (RRK) has significant impact on the mediator requirements prioritisation (RP); secondly, the mediator (RP) has significant impact on the criterion variable success; third, the predictor (RRK) has significant impact on the criterion variable in the absence of the mediators' impact. Thus, to establish the mediating effect, the indirect effect of a x b (see figure 3) has to be significant. In this regard, the z statistic is applied (Sobel, 1982), in which the value is significant at  $p < 0.05$ . If the z value exceeds 1.96 ( $p < 0.05$ ), then the hypothesis 3 can be accepted where there is an indirect impact of RRK through requirements prioritisation on success of software development project. The z value is defined as:

$$z = \frac{a \times b}{\sqrt{b^2 \times s_a^2 + a^2 \times s_b^2 + s_a^2 \times s_b^2}}$$

As shown in figure 3, there is a significant impact of RRK on requirements prioritisation (0.468,  $p < 0.05$ ) as well as requirements prioritisation on success (0.248,  $p < 0.05$ ). However, because there is significant direct impact of RRK on success of software development project (0.284,  $P < 0.05$ ), requirements prioritisation is established as a partial mediator. This mediating effect of requirements prioritisation is confirmed by z statistic (Sobel, 1982):

$$z = \frac{0.468 \times 0.248}{\sqrt{0.248^2 \times 0.085^2 + 0.468^2 \times 0.103^2 + 0.085^2 \times 0.103^2}} = 2.20$$



**Fig. 3:** Requirements Prioritisation as one of the important construct.

The result shows that, RP does have partial mediating effects in which it implies that there is an indirect impact of RRK on success. Accordingly, variance accounted (VAF) value is used to represents the ratio of the indirect effect to the total effect. The VAF value indicates that only 42.8% of the total effect of RRK on success of software development project is explained by indirect effect (Requirements Prioritisation).

$$VAF = \frac{a \times b}{a \times b + c} = \frac{0.468 \times 0.248}{0.468 \times 0.248 + 0.155} = 0.428$$

Thus, it can be concluded that the relationships between the three constructs is significant and the three hypotheses are confirmed by the mediation effects that exist among them in which consequently also support the fourth hypothesis.

#### **Discussion and Conclusion:**

This study emphasizes on the impacts of the independent variables RRK on requirements prioritisation and the success of software development project using the PLS technique. It is a part of a research in which also examines the impacts of RRK on other related factors of success including requirements change, complexity and estimation of budget and schedule. In this paper, the goodness of measure is also assessed by looking at the validity and reliability of the measures using the PLS approach. The results showed that the measures used demonstrate both convergent validity and discriminant validity. Moreover, the reliability of the measures was also assessed by looking at the Cronbach Alpha values and Composite Reliability values. As a result, both the cronbach alpha values and composite reliability values have fulfilled the criteria set up by other established researchers. The results have shown that the measures in the model were reliable.

The findings from the analysis indicated that a software development project is considered successful when the project has achieved several criteria, which are: 1) the outcome of the project meets all the specified requirements; 2) the overall quality of the product is high; 3) The requirements-related tasks (e.g. requirements specification, requirements management) are successfully completed in the project; 4) The outcome of the project meets the business goal; and 5) The project is completed within scope. All the criteria are in fact quite similar to the criteria suggested by previous researchers (e.g. Wohlin *et al* 2000, Agarwal and Rathod 2006). The finding indicated that as long as the outcome meets all the specified requirements and business goal, has good quality, completed within scope, and all the requirements-related activities are completed successfully, the project will be considered successful although the project is not completed within time and budget. Thus, the findings are apparently shown how the success of requirements activities in which including requirements prioritisation will impact success.

Moreover, the findings of the paper confirmed views that RRK has significant impacts on requirements prioritisation. The results of the analysis also confirmed the hypothesis that highlight the important of RRK as one of the significance predictors for success of software development project. Results of the analysis confirmed the direct significant impact of RRK has on requirements prioritisation. Firstly, RRK has significant impact on

requirements prioritisation in which inline with what has been indicated in the literature (Carlshamre *et al* 2001, Davis 2005, Herrman and Daneva 2008). The knowledge of how requirements related to one another provide guide on how a set of requirements can be prioritised and organised in a software development project. According to the analysis of the result, the main characteristics of requirements prioritisation that related to RRK are: 1) In requirements prioritisation, the cost and effort of implementing the set of requirements are checked; 2) In requirements prioritisation, the time estimated of implementing the set of requirements is checked; and 3) In requirements prioritisation, the importance of the requirements according to the user needs is checked. The characteristic of the three items confirms the suggestion of considering the importance of requirements, cost and time constraint in requirements prioritisation during a particular software development project (Berander and Andrew 2005, Herrman and Daneva 2008). Thus, it is important for stakeholders in a software development project to consider the three aspects in their technique or process of requirements prioritisation.

In conclusion, the findings confirmed the phenomena of the interrelationships between RRK->Requirements prioritisation ->Success. Requirements prioritisation has mediated the interrelationships between RRK and Success. Thus, the findings confirmed the four hypotheses listed in this study i.e. requirements relationships knowledge has indirect significant impact on the success of a software development project and requirements prioritisation has partially mediate the relationship. As requirements relationships knowledge has significant impact on requirements prioritisation, and requirements prioritisation has direct significant impacts on success, it can be concluded that requirements relationships knowledge is another significant predictor that will impact requirements prioritisation as well as project success. In future, this study continues to examine further this quantitatively finding with a qualitative study in investigating further how RRK impacts requirements prioritisation and other related factors to achieve success from the perspectives of business analyst.

#### ACKNOWLEDGEMENT

The author is grateful to the Department of Software Engineering, Faculty of Computer Science and Information Technology, University of Tun Hussein Onn, Malaysia for providing the facilities to carry out the research.

#### REFERENCES

- Agarwal, N., U. Rathod, 2006. 'Defining success for software projects: An exploratory revelation', *International Journal of project management* 24: 358-370.
- Asghar, W.M., A. Marchetto, *et al.*, 2013. Maintainability-based Requirements Prioritisation by using Artefacts Traceability and Code Metrics. 17th European Conference on Software Maintenance and Reengineering, IEEE Computer Society.
- Babar, M.I., M. Ramzan, *et al.*, 2011. Challenge and Future trends in software requirement prioritisation, IEEE.
- Barclay, D.W., R. Thompson, Higgins, 1995. The Partial Least Squares (PLS) Approach to Causal Modeling: Personal Computer Adoption and Use an Illustration. *Tech Studies*, 2(2): 285-309.
- Barney, S., A. Aurum, C. Wohlin, 2008. 'A product management challenge: Creating software product value through requirements selection', *Journal of Systems Architecture - Embedded Systems Design*, 54(6): 576-593.
- Berander, P. and A. Andrews, 2005. Requirements Prioritisation. Engineering and Managing Software Requirements. A. Aurum and C. Wohlin. Germany, Springer Verlag.
- Carlshamre, P., K. Sandahl, M. Lindvall, B. Regnell, Nattoch J. Dag, 2001. 'An industrial survey of requirements interdependencies in software product release planning', *Proceedings of Fifth IEEE international Symposium on Requirement engineering*, 27-31 August 2001, IEEE, pp: 84-91.
- Compeau, D.R., C.A. Higgins, S. Huff, 1999. Social Cognitive Theory and individual Reactions to Computing Technology - A Longitudinal-Study. *MIS Q* 23(2): 145-158.
- Dahlstedt, A.G., A. Persson, 2005. 'Requirements Interdependencies: State of The Art and Future Challenges', in A. Aurum & C. Wohlin (eds), *Engineering and Managing Software Requirements*, Springer-Verlag Berlin Heidelberg, Germany, pp: 95-116.
- Davis, A.M., 2005. *Just Enough Requirements Management: Where Software Development Meets Marketing*, Dorset House Publishing, New York.
- Diev, S., 2007. 'Structuring complex requirements', *ACM SIGSOFT Software Engineering Notes Pages 1*, 32(2): 1-5.
- Fornell, C., D.F. Larcker, 1981. Evaluating structural equation models with unobservable variables and measurement error. *J Mark Res.*, 18(1): 39-50.
- Hair, J.F., W.C. Black, B.J. Babin, R.E. Anderson, 2010. *Multivariate Data Analysis*. Upper Saddle River, NJ: Prentice-Hall.

Herrman, A., M. Daneva, 2008. 'Requirements Prioritization Based on Benefit and Cost Prediction: An Agenda for Future Research', 16th IEEE International Requirements Engineering Conference, RE 2008, 8-12 September, Barcelona, Catalunya, Spain. IEEE Computer Society, pp: 125-134.

Karlsson, L., A. Dahlstedt, B. Regnell, J.N.O. Dag, A. Perrson, 2007. 'Requirements Engineering challenges in market driven software development-An interview study with practitioners', *Elsevier Science Direct - Information and Software Technology*, 49: 588-604.

Kirk, J., M.L. Miller, 1986. *Reliability and Validity in qualitative research*, vol. 1, Sage Publications, USA.

Kukreja, N., 2013. Decision Theoretic Requirements Prioritisation: A Two-Step Approach for Sliding towards Value Realisation. International Conference on Software Engineering 2013, San Francisco, IEEE.

Kukreja, N., B. Boehm, *et al.*, 2012. Selecting an appropriate Framework for Value-based requirements prioritisation. Requirements Engineering 2012, Chicago, Illinois, USA, IEEE.

Marjaei, S.A. and V. Kulkarni, 2010. Recognition of Hidden Factors in requirements Prioritisation using Factor Analysis, IEEE.

Nasir, M.H.D., S. Sahibuddin, 2011. 'Critical success factors for software project: A comparative study', *Scientific Research and Essays*, 6(10): 2174-2186.

Nunnally, J., I. Berstein, 1994. *Psychometric Theory*. New-York, NY: McGraw-Hill.

Ozkaya, I., 2006. 'Representing Requirements Relationship', *First International Workshop on Requirements Engineering Visualisation*, IEEE Computer Society, pp: 3-3.

Sekaran, U., R. Bougie, 2010. *Research Methods for Business: A Skill Building Approach*. UK: John Wiley and Sons.

The Standish Group, 1995. *Chaos Report*, T23-T10E Standish Group Report, <http://www.scs.carleton.ca/~beau/PM/Standish-Report.html>.

Trochim, W.M.K., J.P. Donnelly, 2008. *The Research Methods knowledge Base*, Third edn, Atomic Dog a part of CENGAGE Learning, USA.

Wohlin, C., A.V. Mayrhauser, M. Höst, B. Regnell, 2000. 'Subjective evaluation as a tool for learning from software project success', *Information & Software Technology*, 42(14): 983-992.