The Role of Interdependencies in Software Requirements Analysis Estimation

A.Ramachandran

Dept of CSE

UCEP

D.Juliet Thessalonica

Dept of CSE

IRT

S.Ayshwarya Lakshmi

Dept of CSE

UCEP

ABSTRACT

Requirement analysis is sophisticated process to understand the exact requirement in which some are singular and some may depend on other requirements. This Paper presents identification of requirements interdependencies in the software development process. Requirements elicitation is the gathering of relevant knowledge to find out the solution for a problem domain. There are various requirement elicitation techniques available such as: Interviewing and questionnaires, requirements workshops, brainstorming and idea reduction, storyboards, use cases, role playing, prototyping. This paper provides a significant work done on selecting an appropriate technique among them to address the number of unresolved issues concerning the identification and mapping of interdependencies.

Keywords

Elicitation, Interdependencies, Requirements, Role playing.

1. INTRODUCTION

Project Estimation is a vital part of software industries. The accuracy and the reliability of estimation depend upon the time, cost and effort estimates. More than 50-60% 0f new software product development project fail due to improper estimation. Many aspects need to be considered during project selection keeping in mind the goals and requirements of the organization.

The first stage in doing software project development is the requirement specification. The customer who needs the software/system in order to work with is in the process of writing down exactly what the system is to do, and this will be effectively communicated to the developer. Hence this is a well-defined science.

Requirements are the basis for the contract, so it must specified vigorously. It should not be overstepped in to the design, but it should be in certain detail and no misinterpretation is allowed. Requirements must not specify how the system is to do but should specify what the system is to do itself. There are two types of requirements: user requirements and system requirements. User requirements are expressed in natural language, written for customers. System requirements are expressed in structured document, which contains descriptions of the system's functions, services and operational constraints. Functional requirements are expected in terms of systems' functionality/behaviour i.e. how the system should behave in particular situation. Non functional requirements may be more critical than functional requirements. If these are not met, then the system is useless.

Requirements Engineering (RE) is the process in itself, which is a fundamental aspect of software development. RE can be described in five steps: Requirements elicitation, Requirements Analysis, Requirements Specification, Requirements Validation, and Requirements Management. This Paper mainly deals with the Requirements Elicitation.

A higher level of interdependence and lower level of independence leads to stronger estimation bias [1]. A new technique called paper prototyping is implemented [2], which is used to determine interdependencies among requirements. Carlshamre et al. [3] criticizes that 20% of the requirements are responsible for 75% of the interdependencies and only few are singular requirements. It is aimed to learn about the nature of interdependencies among requirements, and classify them to support the estimation efficiency.

The main contribution of this paper is the software requirements analysis. The elicitation techniques allow gathering the requirements from stakeholders and analyzing the interdependency among requirements. Based on the priorities of the requirements an appropriate dependency can be implemented to ensure proper estimation. The results show that the approach could effectively optimize the time, cost and effort estimation.

2. LITERATURE REVIEW

One of the difficult phases in project management is gathering requirements from stakeholders. When there is a good relation between business customers and the project team, a value product can be built to the organisation. The Requirement gatherers should spend more time with the customers to achieve this goal. The requirements elicitation process should rely on assumptions and estimations to form an application.

Deciding [4] dependent requirement for the specific project is nontrivial, yet of major consequence. Some see changing requirements during a project as a problem and expressed it as *requirement creep*. The real world requirements are actually changing, hence the changes need to be ignored to build the wrong product or the requirements need to be changed to build a good product. Requirements come from several sources both internal and external. Requirements traceability [5] has been acknowledged as a basis for understanding requirements Interdependencies. Traceability [14] is concerned with the relationship between requirements. The types of activities affected by requirements interdependencies are: Requirements Management, Change Management, Release Planning, Reuse of components, Reuse of requirements, Implementation, Testing, Maintenance.

After the initial screening, requirements have to be prioritized and interdependencies need to be defined. Prioritization can be performed by customers or by internal experts. Prioritization of requirements can be used as an input to the requirements selection. The different types of interdependencies among requirements are:

- R1 AND R2
- R1 REQUIRES R2
- R1 TEMPORAL R2
- R1 CVALUE R2
- R1 ICOST R2
- R1 OR R2

The different types of interdependencies among requirements are discussed by Carlshamre et al.

Table 1: Requirements grid

| ID	Name
REQ001	cancel order before shipping
REQ002	Login
REQ003	view current order status
REQ004	manage sales to customer
REQ005	visit the site
REQ006	add/remove/update items
REQ007	add/remove items from shopping cart
REQ008	manage payment to employees
REQ009	give feedback
REQ010	manage accounts with bank
REQ011	provide solution to customer complaints
REQ012	browse catalogue
REQ013	discontinue shop
REQ014	Report to the administrator
REQ015	take feedback from customer
REQ016	approve/reject shop creation request
REQ017	manages transactions with customer
REQ018	add/remove/update categories or items
REQ019	Advertise product
REQ020	add/view guest book entry

Table1.represents [3] pair wise assessment of 20 requirements. Using AHP (Analytic Hierarchy Process) [6] it searches for the dependency with the consideration of requirement 1 and 2, 1 and 3, 1 and 4 and so forth. If there are n requirements then n.(n-1)/2 comparisons are required. For example if there are 20 requirements then 190 assessments are required. If singular requirements are identified at the earlier stage the number of assessments can be reduced substantially to (n-s).(n-1-s) / 2. Identifying s singular requirement is less time consuming than n-1 pair wise assessment. There is no need to consider singular requirements for pair wise assessment, rather consider only dependent requirements. The suggestion is that by identifying the 20% of "most dependent" requirements 67% to 79% of all dependencies can be covered.

2.1. Elicitation Techniques

Elicitation is a technique used to discreetly gather information. It is a conversation with a specific purpose: collect information that is not readily available and do so without raising suspicion that specific facts are being sought. It is usually non-threatening, easy to disguise, deniable, and effective. The conversation can be in person, over the phone, or in writing. Requirements elicitation practices include [7] interviews, questionnaires, user observation, workshops, brainstorming, use cases, role playing and prototyping. The elicitation phase is a human activity

system where the analyst can identify three types of [16] requirements: Normal requirements, Expected requirements, Exciting requirements.

Prototyping is beneficial in systems which have more Interdependency. A prototype is a working physical model of a system or a subsystem. Generally, the analyst's objective is to gather information about the user's requirements from the bottom up by allowing the user to interact with the prototype. In effect, the prototype serves as a preliminary version of the system or component from which requirements are extracted and on which subsequent versions are based.
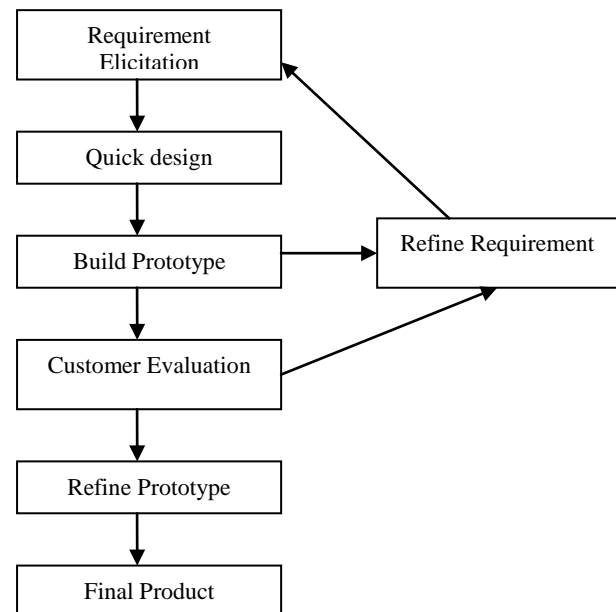
Fig 1: Prototyping Process

Prototyping is iterative. The process starts with a set of partial requirements, and new or expanded requirements are continuously incorporated into the system based on user feedback. This is shown in figure 2. Consequently, the requirements can be viewed as floating, or dynamic. The purpose of the prototype [11] is to clarify the system's requirements. The tasks and queries performed by the prototype demonstrate what the system must do and translate into processes. To many people, manipulating a working model seems more natural than answering questions in an interview or trying to link an abstract model to reality.

Brain Storming is a technique where two or three persons sit together to storm their views regarding system's requirements and then shortlist the requirement using calm method. Role playing is a technique, used when the key stakeholders are inaccessible and one of the analysts recommends the non-stakeholders to play the role of stake holder .A flexible role playing is very useful early in the design process for gathering requirements and user's opinion about a particular idea .If it is intended to have a flexible role playing, however, it can be found useful to have a rough plan, particularly if the person is inexperienced at role playing. Audio and video recordings of role playing can be transcribed so that it can be examined in detail. The choice of specific techniques varies depending upon the specific application.

3. PROPOSED SYSTEM

The proposed Role Playing approach captures the requirements and utilizes the concepts of interdependencies and their relationship to manipulate the effective system.

3.1 Instructions and Background Information

The instructions about the completion of the experiment included the following:

1. The general purpose of the study was to acquire information that would enable the identification of interdependency.
2. The participants [12] should play any one of the role (Analyst/Stakeholder).
3. Requirement analyst is responsible for collecting requirements from stakeholders.
4. Stakeholder is to provide requirements to the requirement analyst.
5. They should use questionnaire-based and prototyping (throwaway) technique.
6. They should provide answers by identifying singular and dependent requirements.

The works of the students were monitored from different locations. The students completed their work that provided the input to the discussion.

3.2 Use Case Description

The task given for the students is to elicit requirement [9] to implement Software Metering System. The Approach will allow the students to gather business requirements such as:

- Determine how many copies of a specified software application are in use on the company network.
- Determine whether there are any unused copies of the specified software application on the network.
- Determine which users regularly use the specified software application.

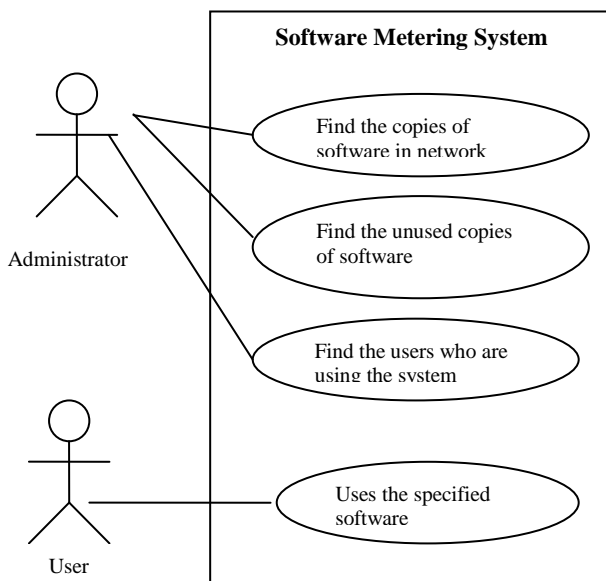Figure 2 represents the use case diagram [15] for software metering system.

Fig 2: Use Case diagram of software metering system

3.3 Class Diagram Description

The approach has made the students to detect the following requirements:

- Requires Configuration Manager 2007.
- To use software metering, the software metering client agent must be enabled.
- Enabling software metering client agent, usage data can be collected.
- Disabling software metering client agent, prevents from collecting data.
- **Name:** The name of the software metering rule should be unique and descriptive.
- **File Name:** The name of the *executable file* to meter should be browsed with the open dialog box. Wildcard characters are not permitted in the *file name*.
- **Original File Name:** If the executable file has been renamed, but needs to be metered by the original name. This field is optional if a value for *File Name* is specified. Wildcard characters are not permitted in the *Original file name*.
- **Version:** The version of the executable file to meter can use wildcard characters.

The relationship between requirements [13] can be shown using class diagram, and is depicted in Figure 3. In many of the prior works on gathering information prototyping, interviewing and questionnaire was broadly used. The goal is to have a different approach using students to perform role playing. The knowledge thus gained through role playing can be organised to develop all UML diagrams, from which the desired code can be developed. Still changing requirements remain a challenging process.
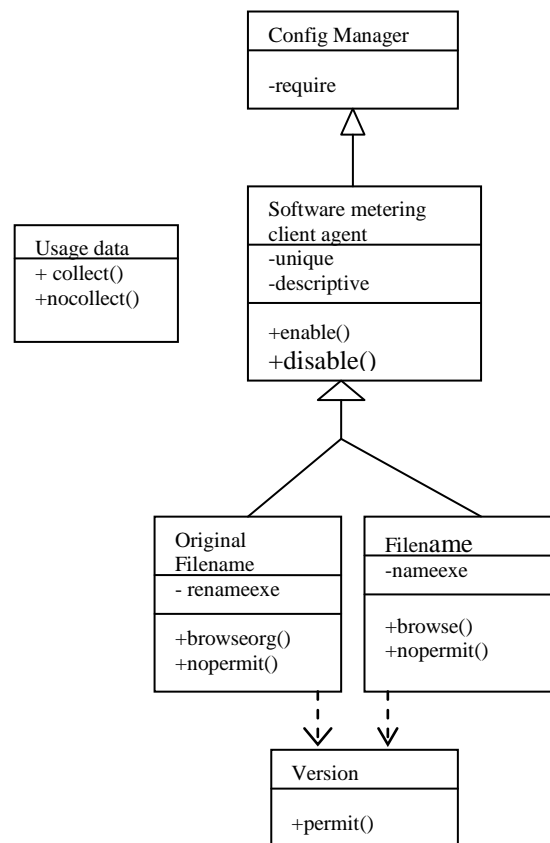

Fig 3: Class diagram of software metering system

This section presented the Use case diagram and class diagram which provides the basis for the design to identify the business requirements.

4. DISCUSSION

The elicitation of requirements is a crucial stage in project development. A more emergent and deliberate view of requirements elicitation requires good communication between stakeholders and the analyst. The degree of understanding however depends on the communication of the user requirements, which involve a diverse range of people who differ on the levels of background, skill, knowledge and status.

As shown in figure 4 Communication activities is a vital part in Role playing. Requirements in fact emerge and are negotiated and indeed at the heart of effective communication lay a shared understanding. Much of the critical information that requirement engineers need is embedded in social words of users and manager as part of the interactive and collaborative activities. The difficulties faced with Communication activities are: Poor Communication between people, Lack of appropriate knowledge and shared understanding, Inappropriate, Incomplete or inaccurate documentation, Lack of a systematic process, Poor management of people or resource.

User and Analyst relation has been portrayed in the literature as a highly problematic in nature and characterized by a culture gap. Roles are classified as either being a user or an analyst held and represents facts of the system.
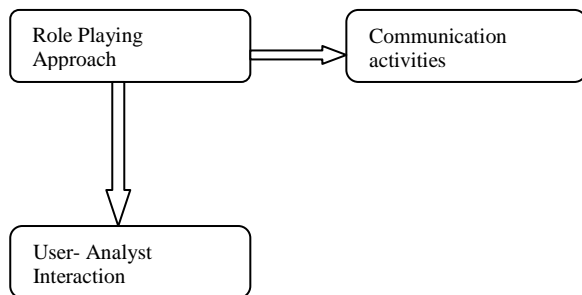

Fig 4: Role Playing System View

When the analyst lacks domain or business knowledge, the consequences tend to misunderstand or ignore requirements and their social context. Role playing evaluates the user to an equal footing with the analyser so that joint decision making is possible as well as satisfying of both parties.

The proposed system can be compared with the brain Storming technique. This technique enables to record ideas during a meeting through a note taking feature called "Brainstorm" It uses requirements diagram to build up the requirements dependency matrix Table 2 represents matrix diagram, which shows interdependency among requirements. The graph shown in figure 6 compares the two techniques in identifying the requirements dependencies with the actual dependencies

Table 2: Matrix Diagram

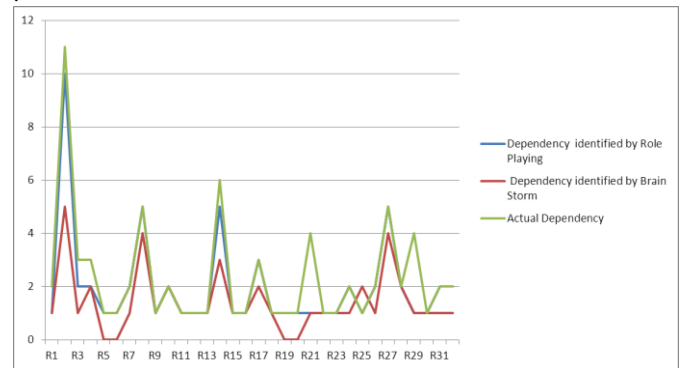|	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
R1							→			←
R2				→						
R3		→				←				
R4									←	
R5	→									
R6			→							
R7					→			←		
R8										→
R9		→								
R10				→						

Fig 6: Comparison Chart

5. CONCLUSIONS AND FUTURE WORK.

Many problems arrive while eliciting [10] and analysing requirements such as incomplete requirements, lack of requirements management, communication problems, problem due to changing economic and business environments. If these problems were studied well the requirements gathering process can be done effectively. This will drastically reduce the bottlenecks in the designing process. The result shown in figure 6 portrays clearly that compared to the Brainstorming; Role playing identified the dependency of requirements to be closer to the actual dependency. In today's world many Organizations struggle [10] in estimating projects. The future work is how to estimate cost, time and effort in requirements analysis phase, so that rework may eventually be reduced.

6. REFERENCES

[1] Magne Jorgensen and stein Grimstad, "Software Development Estimation Biases: The Role of Interdependence," IEEE Transaction on Software Engineering, vol 38, No.3, May/June 2012.

[2] Jaya Vijayan, G.Raju,"A New approach to Requirements Elicitation Using Paper Prototype," International Journal of Advanced Science and Technology, vol 28, March 2011

[3] Carlshamre, P., Sandahl, K., Lindvall, M., Regnell, B. and Natt och Dag, J. (2001) An Industrial Survey of Requirements Interdependencies in Software Product Release Planning, Fifth International Symposium on Requirements Engineering, 27-31 August, Toronto, Canada.

[4] T. Gorschek and A. Davis, "Requirements Engineering: In Search of the Dependent Variables," Information and Software Technology, vol. 50, nos. 1/2, pp. 67-75, Jan. 2008.

[5] Asa G.Dahlstedt and Anne Person, "Requirements Interdependencies – Moulding the state of Research into a Research Agenda".

[6] Joachim Karlsson and Kevin Ryan , "A cost-value approach or prioritizing requirements", IEEE Software.

[7] Ann M.Hickey and Alan M.Davis, "Elicitataion Techniques Selection : How Do Experts Do It?", Proceedings of the 11th IEEE International Requirements Engineering Conference, 2003.

[8] Oscar Dieste, Member, Ieee, And Natalia Juristo, Senior Member, Ieee, "Systematic Review And Aggregation Of Empirical Studies On Elicitation Techniques" Ieee Transactions On Software Engineering, Vol. 37, No. 2, March/April 2011

[9] Bee Bee Chua1, Danilo Valeros Bernardo1, June Verner2, " Understanding the use of Elicitation Approaches for Effective Requirements Gathering" 2010 Fifth International Conference on Software Engineering Advances.

[10] Richard Berntsson Svensson, Tony Gorschek, Member, Ieee, Bjo Rn Regnell, Richard Torkar, Member, Ieee, Ali Shahrokni, Student Member, Ieee, And Robert Feldt, Member, Ieee," Quality Requirements In Industrial Practice—An Extended Interview Study At Eleven Companies", Ieee Transactions On Software Engineering, Vol. 38, No. 4, July/August 2012

[11] Stytz, M.R.; Adams, T.; Garcia, B.; Sheasby, S.M.; Zurita, B.," Rapid Prototyping for Distributed Virtual Environments", Software, IEEE Volume: 14 , Issue: 5

[12] M. Jørgensen, B. Faugli, and T. Gruschke, "Characteristics of Software Engineers with Optimistic Predictions," J. Systems and Software, vol. 80, no. 9, pp. 1472-1482, 2007.

[13] T. Gorschek, C. Wohlin, Requirements abstraction model, Requirements Engineering 11 (1) (2006) 79–101, March.

[14] Ramesh, B. (1993) A Model of Requirements Traceability for Systems Development, Technical report, Naval Postgraduate School, Monterey, CA, USA, September

[15] Maciaszek, L.A. (2001) Requirements Analysis and System Design – Developing Information Systems with UML, Addison Wesley.

[16] Marcel J. Simonette, Fabio L. L. Sanches, Edison Spina," Soft Systems Engineering Tools In Requirements Elicitation", International Journal Of Systems Applications, Engineering & Development Issue 3, Volume 4, 2010.