Web Based Requirement Elicitation Tool

Mr. Shinde Santaji Krishna
Research Scholar,
Department of Computer Engineering,
Shri Jagdishprasad Jhabarmal Tibrewala University,
Vidyanagari, Jhunjhunu, Rajasthan, India

Dr. D. K. Kaushik
Research Supervisor
Department of Computer Engineering,
Shri Jagdishprasad Jhabarmal Tibrewala University,
Vidyanagari, Jhunjhunu, Rajasthan, India

Dr. Shashank Dattatraya Joshi
Research Supervisor
Shri J.J.T.University Rajasthan &
Professor, Department of Computer Engineering,
Bharati Vidyapeeth Deemed University College of Engineering,
Pune, Maharashtra, India

Abstract—Requirement elicitation is complex and critical phase in software development. The process involves various activities like acquiring domain knowledge, identifying sources of requirements, identifying stakeholders, selection of technique, approach and tool to collect requirements. Traditional requirement elicitation system may contain biasing. Requirement elicitation using fuzzy collaborative filtering is a novel method that uses the collaborative approach to prioritizing requirements from different stakeholders. It asks stakeholders to rate requirements, recommends other pertinent requirements to them and prioritizes requirements based on their ratings from various stakeholders. The binary search tree approach is used for requirement prioritization.

Keywords— Biasing, Binary Search Tree, Collaborative filtering, Requirement Elicitation, Requirement prioritization, Stakeholder.

I. INTRODUCTION

Requirement elicitation is the most important phase of the software development life cycle. Today, projects to build large software systems involve huge numbers of stakeholders, the individuals or groups that can influence or be influenced by the achievement or failure of a software project [1]. These stakeholders involve users those pay for the system, Clients who communicate with a system to get their work done [1, 2]. It's objectives to detect the purpose for which the software system is planned [3]. The software project succeeds only if requirements from different stakeholders are identified and prioritized properly. The traditional methods fail to scale to big projects with hundreds, thousands, or even hundreds of thousands of stakeholders [4].Understanding of the customer needs is the process of Requirement elicitation. Users who own business processes are experts in their domain, but they lag to mention their requirements to solve particular business problem in technical word that can be understood by the software developer. Users become irritated when the software fails to meet their needs. Customers who pay for the project pay for the mistakes[5]. Requirement engineer plays the crucial role to gather requirement from stakeholders of business process and put that requirement in technical(Software Requirement Specification)SRS document that can be understood by software developers. Requirement elicitation using fuzzy collaborative filtering is a collaborative approach to identifying and prioritize requirements of different stakeholders.

Various techniques are used by requirement engineer to collect requirements from stakeholders' viz. Face to face meeting, Focus Groups, Interviewing and Questionnaires, Requirements workshop, Storyboards, Joint Application Development (JAD), Prototyping, Use cases, Role playing and Brainstorming[1] .The problems associated with these techniques are many of these processes are time consuming, not effective if requirements to be gathered from large number of users, also these methods are susceptible to observer, interviewer or facilitator bias. If the project size is large and there is large number of requirements then collecting requirements from a large number of stakeholders is a cumbersome task. The Sometimes it may happen that stakeholders are absent during requirement gathering process, and their requirements are not considered which leads to disappointment of that user. Many times biasing occurs while assigning priorities to requirement by different stakeholders.

To overcome these problems, this work proposes an approach that uses collaborative filtering for requirement elicitation as it scales to a huge number of user requirements. Collaboration supports team and individual goals; also, people innovate faster in the collaborative workspace.

II. LITURATURE SURVEY

Neglecting stakeholders is one of the most common errors in software engineering [6]. Existing stakeholder analysis approaches are likely to overlook stakeholders [7]. In addition, stakeholders are regularly sampled during requirements elicitation [8]. Lehtola et al. [1] mentions that requirements prioritization techniques are casual. Individual practitioners prioritize requirements on the basis of their experience. Bhushan et al. [8] discuss collaborative filtering as a recommender system that makes a prediction about most likely item to be rated by user by comparing users' preference with like-minded users. Soo et al [7] proposed a novel method StakeRare, which identifies stakeholders and requests them to recommend other stakeholders and stakeholder parts, builds a social network with stakeholders as nodes and their recommendations as links, and priorities stakeholders using a

ISSN: 2278 – 1323

range of social network measures to decide their project influence. It then requests the stakeholders to rate an initial list of requirements, recommends other related requirements to them using collaborative filtering, and priorities their requirements using their ratings weighted by their project influence. [9] mentions that projects often have more requirements than time, resource, and budget allow. As such, requirements should be prioritized and managed so that those that are serious and most likely to reach customer fulfillment can be selected for implementation.

III. MOTIVATION

A. Requirements prioritization is an ambiguous concept:concept

Though it is important that people have a common kind of the terms they use and activities they do in product development, the terms "requirements prioritization" and "priority" have various meaning in practice.

B. Prioritization practices are informal and dependent on individuals:

There are different methods to present requirements prioritization in the industries. No explicit requirements prioritization methods were in use in the industries. The development persons attempted to build a irregular guess which requirements were the most important ones to customers and users, how gainful requirements were to their own industry, and how all this join with the strategy of the industry, but there were no efficient ways to these study.

C. Requirements are prioritized in many stages:

Assessment about which requirements can be included in the next version of the product and which can be delayed are needed in many stages of product improvement. Requirements meaning is a process during which priority assessment have to be made repetitively. Requirement priorities are required, not only for making assessments as to which requirements to leave out, but also for enquiry purposes after the release and in order to support the communication within the association and with the users.

D. Developers do not know enough about customer predilections:

The product development employees would like to know why a requirement is significant to users or customers. Usually, they have no indication because people are working distinctly in the product development; product development employees do not have straight contacts with users and customers. In addition to this, there are no common ways to communicate customer and user information through the product development procedure.

E. The priority of a requirement is based on many issues:

The requirement's significance to a customer is vital, but usually not the only, factor that has an effect on a requirement's priority. There are many difficulties in defining which issues should be taken into account when setting the priorities. Receiving the true data for to use as the basis for prioritization choices is not always easy[9,10].

Developing scalable, automated recommender system that will help to work all stakeholders in a collaborative fashion. There is a need of the proposed system that is based on fuzzy approach for collaborative filtering as recommender for requirements suggested by different stakeholders. It is assigning priorities to requirements so that system will be helpful for decision makers to decide which requirements would be beneficial for business with minimum risk and maximum profit with customer's satisfaction.

IV. PROBLEM DEFINITION

To develop a requirement elicitation automated recommender system that will help to work with all stakeholders in the collaborative fashion and provide a simpler user interface that will give information about priorities assigned and representation of prioritization requirement is done using binary search tree technique.

Objectives

1. To develop an automated recommender system that will help to work with all stakeholders in the collaborative fashion.
2. To provide a simpler user interface that will give information about priorities assigned to various requirements and then represent the prioritization requirement using binary search tree technique.

Layout of proposed work:

1. Developing the online tool to collect requirements from all stakeholders in the project and classify requirement into different categories.
2. Making prediction of non-rated requirements that will be rated high by the user, using User-Item rating matrix generated using step 1 and Singular Value Decomposition algorithm.
3. Requirement Prioritization using Binary search tree (BST) approach.

Collection of requirements from different stakeholders of the project is done and given as input to the proposed system.

Let Ms denotes the number of stakeholders. P denotes the set of requirements. $X1...XNs$ indicates the user session (requirements recommended by the user). Let XA represents an active user session. Let $MP \subset P$ be all the requirements not yet recommended for the active user, for which system is going to provide predictions. A collaborative filter is a function f that takes as input all past user sessions and produces its result recommendation values for requirements not yet recommended for the active user.

$$XAj = f(X1, X2....XNs), \forall j \in MP$$

ISSN: 2278 – 1323

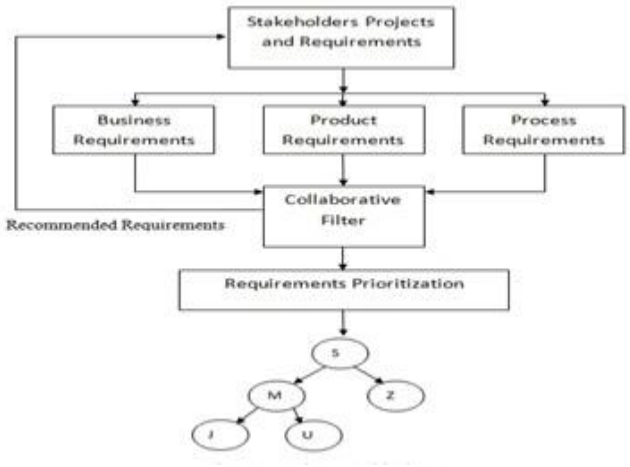

Fig. 1. Requirement Elicitation

Use collaborative filtering as recommender engine for recommendation of requirements

1. Collect requirement from the user using what, when, which and how, etc. questions.
2. Using singular value decomposition make predictions of requirements that will be rated by user most probably.
3. Classify requirements into functional, nonfunctional and business requirement.

Prioritizing requirements in simple, unambiguous manner which will help to make the decision regarding implementation of a requirement.

Algorithm: Prioritization of requirements using of Binary Search Tree.

1. Get all requirements in one stack.
2. Select one requirement as a root node.
3. Select another requirement and compare it to the root node and place in the tree using the binary search tree technique.
4. Perform step three until all requirements have been compared and inserted into the binary search tree.
5. For arrangement purposes, traverse through the entire BST in order and put most necessary requirements at the start of the list. Finally, put the least important requirements, at the end of the list.

Snapshots

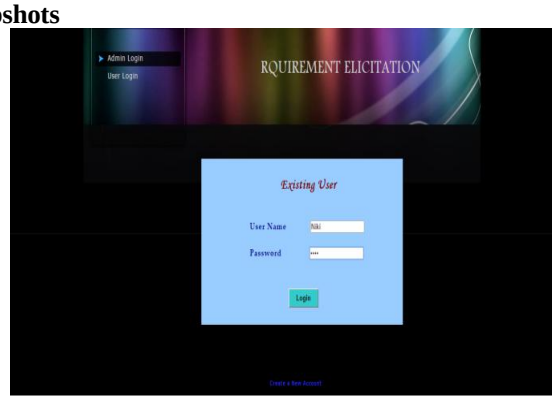

Fig. 2. Existing user

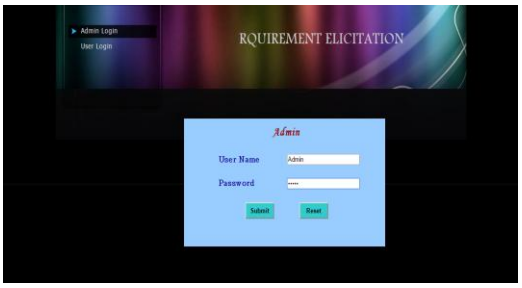

Fig. 3. Admin

Fig. 4. Admin Login

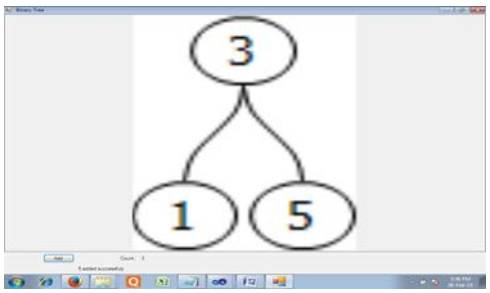

Fig. 5. Bug Detection Binary Search Tree

V. RESULT DISCUSSION:

The proposed system is tested on a some project that assigned to different groups having same technical skills. Some groups have asked to complete the project using the current methodologies of SDLC & others are have asked to complete the same project using proposed system. The group who has used current methods in SDLC has gone through the traditional approaches to requirement elicitation.We found that, the group which has used the proposed system for the collection & prioritization of the requirements required in less time as compared to others groups those who have used the traditional approaches for the collection & prioritization of the requirements.

From the result discussion, it is clear that project completion success rate of proposed system approach is more than that of the traditional system. Also, the accuracy obtained the proposed system is more than that of the traditional system.

ISSN: 2278 – 1323

TABLE I. SUCCES RATE OF PROPOSED SYSTEM AND TRADITIONAL SYSTEM

Project TestedName	%Project Success Rate Ratio(proposed)	%Project Success Rate Ratio(Traditional)
Project name 1	90	75
Project name 2	95	80
Project name 3	92	50
Project name 4	95	0
Project name 5	90	35

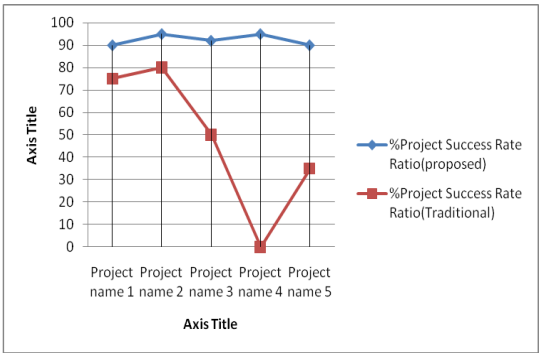

Fig. 6. Succes rate of proposed system and traditional system

VI. CONCLUSION

By automation of requirement gathering process, it is possible to collect the requirement from all stakeholders 24 x 7. By implementing collaborative filtering approach, it is possible to work with a group. System generates recommendations for requirements that are most likely for the user. It is also possible to maintain secrecy of the user that helps to remove biasing happens many times in requirement gathering process. A simple rating mechanism help to assign priorities to requirements that make requirement engineers life comfortable. Binary search tree implementation for priority assignment has helped to take decisions regarding which requirement should be considered first for implementation. Hence, the proposed system has designed as the online tool so that stakeholders can directly participate during requirement elicitation phase, and the system has worked as recommender for participating users.

Proposed system helps to improve the quality of software product development by reducing product development time and with optimum resources.

REFERENCE

[1] G. Eason, B. Noble, and I.N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955.

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[3] I.S. Jacobs and C.P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.

[4] K. Elissa, “Title of paper if known,” unpublished.

[5] R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740-741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[8] B. Nuseibeh, And S. Easterbrook, "Requirements Engineering: A Roadmap," Proceedings Of The Conference On The Future Of Software Engineering. Pp. 35 - 46, 2000.

[9] H. Sharp, G. H. Galal, And A. Finkelstein, "Stakeholder Identification In The Requirements Engineering Process," Proceedings Of The Database & Expert System Applications Workshop. Pp. 387–391, 1999.

[10] C. Duan, P. Laurent, J. Cleland-Huang, and C. Kwiatkowski, “Towards automated requirements prioritization and triage,” *Requirements Engineering*, vol. 14, no. 2, pp. 73-89, 2009.

[11] A. Davis, O. Dieste, A. Hickey, N. Juristo, and A. M. Moreno, "Effectiveness of requirements elicitation techniques: Empiricalresults derived from a systematic review," Proceedings of the 14th IEEE International Conference on Requirements Engineering. pp. 179-188, 2006.

[12] S. Lauesen, *Software Requirements: Styles and Techniques*: Addison-Wesley Professional, 2002. [64] K. Wiegers, “First things first: prioritizing requirements,” *Software Development*, vol. 7, no. 9, pp. 48–53, 1999.

[13] A. Herrmann, and M. Daneva, "Requirements prioritization based on benefit and cost prediction: an agenda for future research," Proceedings of the 16th IEEE International Conference on Requirements Engineering. pp. 125-134, 2008.

[14] R. M. Bell, and Y. Koren, "Scalable collaborative filtering with jointly derived neighborhood interpolation weights," Proceedings of the 7th IEEE International Conference on Data Mining. pp. 43-52, 2007.

ISSN: 2278 – 1323