# Implementing Item-Based Collaborative Filtering System in Requirement Elicitation

**T.S Shiny Angel, Keertana Madan, Utkarsh Pandey**

***Abstract:*** ***In Requirement Engineering, Gathering Requirements plays a vital role in the Software Development Process. There are lot of processes available to gather requirements i.e. Brainstorming, Interview, Observation. This process takes lot of time and effort for the developer to gather and continue the development, and if the requirements which are analyzed are not up to the satisfaction of the user, it will cause issues in end product resulting in loss of human effort, time and cost. To overcome this issue faced by the developers we have developed a tool using item-based collaborative algorithm for users which will recommend users the required set of functional and non-functional requirements based upon the questionnaire given to the user and produce a software requirement specification (SRS).***

***Keywords: Item-Based Collaborative Filtering, Requirement Engineering, SDLC, Software Development Life Cycle, Requirement Elicitation***

## I. INTRODUCTION

During Software Development Life Cycle (SDLC), Requirement Elicitation plays a major role in the process. It is important for every organization to develop quality software products. The most difficult part of the requirement's elicitation is obtaining a comprehensive and trustworthy set of requirements. In our approach we will develop a prototype-Implementation of Item-Based Collaborative Filtering in Requirement Elicitation for gathering requirements from the customers in an intelligent way. Requirements elicitation is the first step in Project Management. It is the process of extracting all the requirements from a customer/user by various means like brainstorming, interviews, brain-writing, etc. This approach towards automating the process will be useful for the software engineers and domain experts, help them gather requirements in a faster way and generate Software Require- ment Specification (SRS). The domain expert can log into the system and prepare a questionnaire asking close ended questions to the users to extract requirements. For the process of eliciting requirements, the user shall answer a questionnaire prepared by the domain expert in the system. The answers to this questionnaire shall be saved and used to ultimately prepare an SRS for the user in the required format (IEEE format used mostly). Our tool, implementing the algorithm, will increase the possibility that the customers get real time situations and nearly complete requirements.

**Revised Manuscript Received on June 22, 2019.**

**T.S Shiny Angel,** Department of Software Engineering, SRM Institute of Science and Technology, Chennai, India - 603203

**Keertana Madan,** Department of Software Engineering, SRM Institute of Science and Technology, Chennai, India - 603203

**Utkarsh Pandey,** Department of Software Engineering, SRM Institute of science and Technology, Chennai, India – 603203

## II. RELATED WORK

There are lot of researches been done implementing automation in requirement gathering process. The concept was studied by Neil W. Kassel and Brian A. Malloy in their survey titled "An Approach to Automate Requirements Elicitation and Specification". In this survey, they studied the requirement elicitation, a relational database was created for gathering user's information and requirements that was developed based on the repository, this database can be used for data collection, organization and knowledge testing, they initialized the process of requirement gathering implementing automation.

In specific systems, another tool GMARC was developed by Computer System Architects using an approach that automatically capture user's requirements based upon the existing information present in the domain. It is difficult how the tool GMARC requires an expert requirement engineer to enter precise domain information, which could be challenging for the naive users. Our tool aims to reduce the amount of time and cost by automating the process.

Another approach to automate the requirement gathering process was by Nor Aiza Moketar, Massila Kamalrudin, Safiah Sidek titled "An Automated Collaborative Requirements Engineering" (ASE '16, Singapore). TestMEReq tool was developed that could assist the engineers to communicate with the customers to validate the requirements, as this approach was much more useful just for the validation which would not benefit the user as the user has to create the set of requirements by themselves and buy the software to validate the requirements gathered, which would increase the time and cost required for development, In comparison with our tool, is user-friendly for professional as well as naive users to generate the requirements required.

## III. PROPOSED SYSTEM

We understand human interaction is a key element of software requirements elicitation process and eliminating all cost that may not be possible entirely. Our tool has been implemented by machine learning algorithm. The Algorithm used is Item-Based Collaborative Filtering. Every Item creates an item-item similarity matrix of every 'requirement' and lists the similarity value of each 'requirement' with respect to the other. Then, for every user, it calculates a weighted sum of the user's preferences based on their needs and using that it calculates the top similar 'requirements' and recommends it for the user. Finally, after assessing the requirements and effective validation resulting into a Software Requirement Specification (SRS).

## IV. SYSTEM ARCHITECTURE

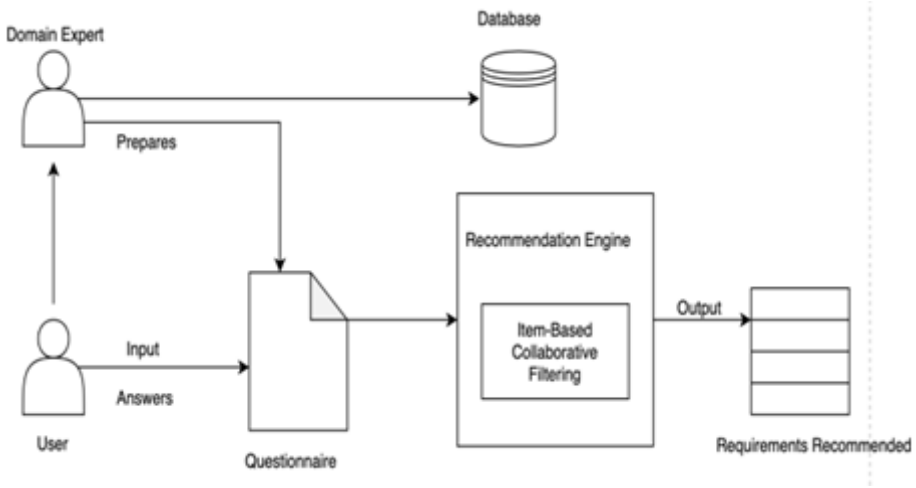


**Fig. 1 System Architecture Diagram**

Our system architecture contains of two users Domain Expert, User. Domain expert's role is to prepare the questionnaire which the user answers. The questionnaire once prepared is also stored in the database for further use. When the users have answered the questionnaire, the inputs are sent to the recommendation engine which users machine learning technique to recommend requirements to the users. To improve the accuracy of the algorithm recommended requirements are stored in the database. Our tool has different modules that govern its major functions entirely.

### Modules

### Questionnaire Module

The questionnaire module is prepared after extensive research and study of various projects in the specified domain. The user answers this questionnaire and the data is stored in the database and it is used to recommend to the user a set of consistent requirements that is of use to him/her.

This questionnaire is a set of twenty questions answered by the user in which their preferred options are given and the output is a list of requirements recommended to them using the recommendation engine.

### Database Module

The database is made with sqllite3 and it stored the information pertaining to the system. The database also contains a csv file containing the data stored from the previous users who have affectively interacted with the system which has hence led to produce tangible reports

### Recommendation Engine

This module is the brains behind the entire system. The rec- ommendation engine uses Item-Based Collaborative Filtering for recommending requirements to the user based on previous user interaction. The answers from the questionnaire serves as the input for the engine and the final derived output is a list if requirements recommended to the user that may further be pursued to generate a Software Requirements Specification Document for the user.

## V. DATASET

The dataset for this project was not readily available at any resources so it was a tedious and time-consuming task to prepare it. A google form was created which consisted of well- researched 'requirements' after going through a number of already existing SRS's of e-commerce projects. The data was collected from this google form asking users for their preferences and needs. This collected data has been used to calculate recommendations for every user in the project.

## VI. GRAPHICAL USER INTERFACE

The GUI module is used to collect what the client needs and these go into the recommendation engine from where requirements are recommended to the user. A questionnaire is prepared of every user which the user answers and these answers decide what recommendations will be recommended to that user based on what users have previously used. In this interface itself will in the end the final SRS will be generated that the user can use for documentation purpose, etc. GUI has been developed using Eclipse IDE, Version: Eclipse IDE 2018- 12.

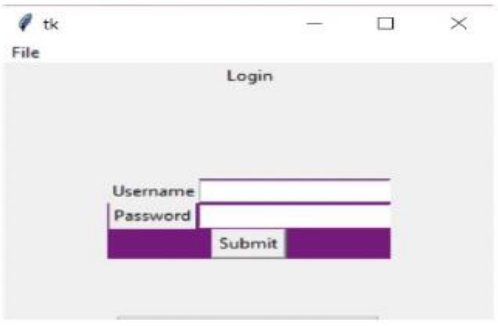


**Fig. 2 Login GUI**

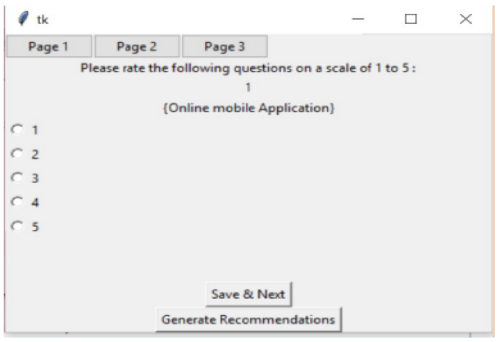


**Fig. 3 Questionnaire GUI**


**Fig. 4 Questionnaire GUI**

## VII. RESULT

After the users answer the questionnaire given to them according to the chosen domain. The answers then are processed through our Recommendation engine which uses a particular item-based filtering algorithm to recommend users requirements depending on the questionnaire answered and the dataset, using this algorithm we can determine actual needs of the user by training the system.

The information when processed a weighted average is taken of the similar items of the present user rating for the respected item creating a matrix and lists the similarity value of each requirement with respect to others. Using machine learning techniques and implementing in the requirement elicitation process will help software developers to elicit requirements automatically.

The requirements (results) generated by the tool works as predicted and with higher accuracy for the detailed set of requirements, which further help all the software developers and also the naïve users without any prior knowledge to work on other process rather investing much time for gathering requirements.

The tool is built with different modules which are indepen- dent providing low ambiguity to prevent from attacks or data breach.

Lastly, this tool will help the requirement engineering process much more standardized, well-defined to elicit requirements from the users as the before processes used for requirement gathering are done by humans, which causes a major risk of human error whereas when the computer algorithm is trained with huge data set it will provide highly accurate and efficient results and most importantly without any error.

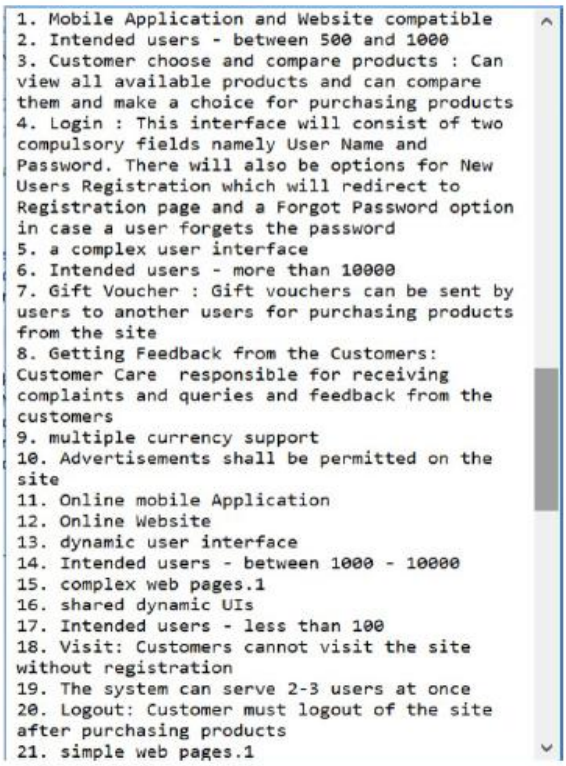


**Fig. 6 Requirement Recommended to the Users**

## VIII. CONCLUSION AND FUTURE WORK

The result of the system is a recommended list of user requirement; user requirements are used to generate an SRS for easier documentation. System also stores the user preferences selected by the user according to the questionnaire, producing accurate results as an output. The system can be maintained quite easily as it has been modularized with each function separately written in a different module efficiently. In future updates, we tend to increase our data set efficiency to provide better results and recommend accurate functional and non- functional requirements based on user results. We also look forward to build our tool more user-friendly to provide ease access and detailed set of requirements.

## REFERENCES

1. Massila Kamalrudin “Automated Software Tool Sup- port for Checking the Inconsistency of Requirements”,2009 IEEE/AM International Conference on Automated Software Engineering.
2. Mohamed A. Abd Elmoniem “A Requirements Elici- tation Tool for Cloud-Based ERP Software”, MECSE 2017, December 2017, Cairo, Egypt.
3. Nor Aiza Moketar, Massila Kamalrudin, Safiah Sidek “An Automated Collaborative Requirements Engineering”, ASE’16, September 3–7, 2016, Singapore, Singapore.
4. Sanjaya Kumar Saxena “Decisively- Application of Quantitative Analysis and Decision Science in Agile Require- ments Engineering”, 2014 IEEE 22nd Inter- national Require- ments Engineering Conference (RE).
5. Madhusudan Singh and Shiho Kim, “Reconcile Secu- rity Requirements for Intelligent Vehicles”, 2017 17th International Conference on Control, Automation and Systems (ICCAS 2017).

6. Ropak Sinha, Sandeep Patil2, Cheng Pang, Valeriy Vy- atkin,, and Barry Dowdeswell, "Requirements Engineer- ing of Industrial Automation", IECON2015-Yokohama.
7. K. Li, R.G. Dewar, R.J. Pooley, "Computer-Assisted and Customer-Oriented Requirements Elicitation", Proceed- ings of the 2005 13th IEEE International Conference on Requirements Engineering (RE'05).
8. Qian Zhang Armin Eberlein, "Architectural Design Of An Intelligent Requirements Engineering Tool" ICSE '04 Proceedings of the 26th International Conference on Software Engineering Pages.
9. Neil W. Kassel and Brian A., "Automation of Require- ment Analysis in Software Engineering", International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169.
10. Sanjaya Pandey, Subik Pokharel, Hassan Reza, "Towards Cyber-physical Requirement Engineering", 2018 TSI Press, Inc.