International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-2, December 2019

Partly-Automated Scheme (PAS) Reinforced Framework to Prioritize Requirements with Abbreviated User Involvement and Verdict through AHP

K Glory Vijayaselvi , ThirumalaiSelvi RABSTRACT

ABSTRACT: ***Extremely significant phase of SDLC is Requirement Engineering . Building of software and its functionalities successfully is exclusively based on the requirements gathered from the user of the project. The accomplishment of the end product has direct relationship with this Requirement Phase. Requirement Prioritization Process (one of the process) in the Requirement Engineering phase supports the engineers to work out and identify the prioritization among the requirements. From the available methods to prioritize the requirement, AHP is viable but not for large size projects. This work primarily concentrated on applying AHP for bigger projects. In this paper, we constructed a framework to prioritize the requirements with AHP considering Implementation Simplicity for Large Scale project, reduced number of comparisons and precise Stakeholder's Participation. The proposed framework has been assessed through an exploratory case study that has fixed number of requirements and the status after the arrival of new requirements to the priority list. This is to know about the actuality of the proposed framework, which has been conducted in a software company. The main findings and lessons erudite from the effort are presented.***

Key Words: AHP, SDLC, Requirement prioritization, Users, comparisons, priority.

I. INTRODUCTION

Software systems plays a vital role in today's technological world. Successful delivery of such software starts from the very first phase of SDLC i.e. Requirement Engineering. From the study it is found out that about seventy percentage of the software delivered unsuccessful because of the failure in this phase.[1]. If the end user is not satisfied, then the purpose of software developed will not be attained. After receiving the requirements, the software will be released in versions.

Revised Manuscript Received on December 30, 2019.
***** Correspondence Author
K Glory Vijayaselvi* , Research Scholar, Research and Development Centre, Bharathiar University, Coimbatore & Assistant Professor, Department of Computer Science (Shift II), Women's Christian College, Chennai, India.
ThirumalaiSelvi RABSTRACT, Assistant Professor, Department of Computer Science, Govt. Arts College for Men, Nandanam, Chennai, India.

Users avoid using some functionalities in software even after implementing it with much time and cost. Hence it is important to find the urgency of requirement. And it is a major encounter for the team in requirement phase. Picking out the stakeholders to rank the requirement is another pilot in this area.

According to Firesmith [1] requirement prioritization is the process of determining the implementation order of the requirement for implementing a software system.

From the review, it is found out that requirement prioritization condenses the development cost of the software system and time taken by forty percentage[1]. The requirement prioritization also upsurges the gratification among the customers and also increases the excellence of a software. Hence this requirement prioritization is very important process in SDLC.

II. FACTORS THAT CAN BE CONSIDERED FOR REQUIREMENT PRIORITIZATION

While prioritizing the requirement, criterion are there to consider. It differs based on the nature of the project. Some common criterion are

- Importance/ urgency of the requirement
- Time taken for the implementation of requirement
- Cost for the implementation of requirement
- Value of the requirement
- Consequences to be faced if the requirement is not implemented
- Others eg. Risks, external factors, etc.,[3]

III. PREROGATIVE AND STUMBLING BLOCK OF REQUIREMENT PRIORITIZATION

The prioritization process makes the user to involve in the development process[1][4]. Since the users involvement throughout the development of the software is considered as very important, the HCC Model [3] is proposed as a research work .

Involving stakeholders in all the phases helps the successful implementation of required requirement at required time. It reduces the software failure[1][5]. This also helps to manage the resources with optimum utilization which is very essential in this current competitive world. But collecting and consolidating the requirement prioritization from different stakeholder is difficult.

Within the specified time, the priority list has to be prepared. Hence human inducement is very much important in this process to make the stakeholders to participate in the work interestingly.

IV. RELATED WORK

1. AHP was more accurate [6] is the statement declared after the test. Based on the test, it is derived that 16 stated that AHP is accurate (69.57%) and 7 stated that CB Rank is accurate (30.43%). According to the criteria, Ease of use, AHP stands little back because of the number of comparisons. Tool was developed for both the methods. The experiment was carried out. As a result, they concluded that AHP produces accurate result but it is not easier to use.
2. AHP is most suitable to prioritize the requirement [7]. AHP is more accurate, faster and helpful in decision making. But AHP is challenging for the larger projects. So tool support is required. But it is defined as the difficult method and it consumes more time than the other. AHP is reliable but when the project is large, comparison will be a problematic one . So tool is required to overcome these drawbacks.
3. AHP gives better result but suffers from scalability problem[8].Because every pair of requirement has to be compared.
4. In the framework proposed, stakeholders will be prioritized by others and then their influence on projects will be calculated[2]
5. AHP is suitable for multicriteria decision making problems. In addition to the end user, inclusion of hierarchical manager can be done. Key users of the requirement can be calculated. Goal of the company should be considered.
6. Requirement selection for a group is one of the main objective[9].Tool support is required. Framework has to be developed to address scalability problem.
7. Diverse views of stakeholders makes it very difficult to identify the priority of the requirement[1]. If the resources are limited, again it will be a difficulty. When there is a change in the priority of the requirement among the users, it will create a problem. AHP is reliable. But it is not feasible with large number of requirements.
8. Improvement is required in the available method[10]. Some of the parameters that has to be considered is scalability, ease of use and reliability of results. Automated requirement prioritization technique should be implemented for efficient prioritization process.
9. More number of comparisons has to be performed [11]. AHP can handle both qualitative and quantitative data.
10. Lack of user and client involvement in software development not only affects product quality but also resulted in user dissatisfaction.
11. From the study, it is concluded that more field study has to be conducted and the tool has to be developed[13]
12. While selecting the method, the developing team should consider the accuracy of result provided and all the related characteristics of the process. Among all the methods available to prioritize the requirement, AHP is feasible but not for large scale projects. The main parameter that has to be considered is ease of use, time and accuracy. [12]

V. AHP METHOD

In AHP, one compares all the probable pairs of requirements to find out well-organized list of the requirements according to their consequence. Usually a range of 1 to 9 is used, where 1 denotes identical importance and 9 represents tremendously more importance. During the progression, if r number of requirements are elicited, $r\times(r-1)/2$ comparisons want to be made, which for the application with large number of requirements results in struggle. The result is a set of requirements prioritized along a scale. The AHP combines multidimensional scales of measurement into a one-dimensional scale of priorities. AHP is highly dependable, since the great level of redundancy in the pair wise comparisons makes the process resistant to comparison errors [9]. Another advantage is the fact that the values assigned in the pair wise comparisons are based on knowledge, perception and real data. Thus, AHP can handle both the qualitative and the quantitative aspects of a decision problem. As a result, the fact that the resulting priorities are related and based on a scale permits useful assessments of the requirements[3] The elicited value k_{ij} is inserted in the corresponding cell of the matrix (k_i, k_j), while the cell (k_j, k_i) is filled with the reciprocal of the value $k_{ji} = 1/k_{ij}$ [7]

Hence AHP method is satisfactory with multi criteria decision making[1]. The steps of AHP are as follows:

1. Listing out the criteria
2. Pair wise comparison of criteria for urgency
3. Construction of normalized comparison matrix
4. Merging the matrix
5. Calculation of AHP score for each criteria

The values used for criterion comparison are

Table1

Value	Description
1	If the two criteria has equal weight
3	If first (row) criteria is important than second (column) one.
5	If first (row) criteria is much more important than second (column) one.
7	If first (row) criteria is significantly more important than second (column) one.
9	If first (row) criteria is definitely more important than second(column) one.

VI. THE PROPOSED FRAMEWORK

This paper proposes new framework which implements only AHP in all its phases. Because of the accuracy of the method, the AHP attracts more in prioritization. Accuracy is an essential parameter. Others stand in the next. So with this AHP, the framework is designed to prepare the prioritized list. In this work, the parameters considered are number of requirements, number of users, number of comparisons and the priority of the requirement.

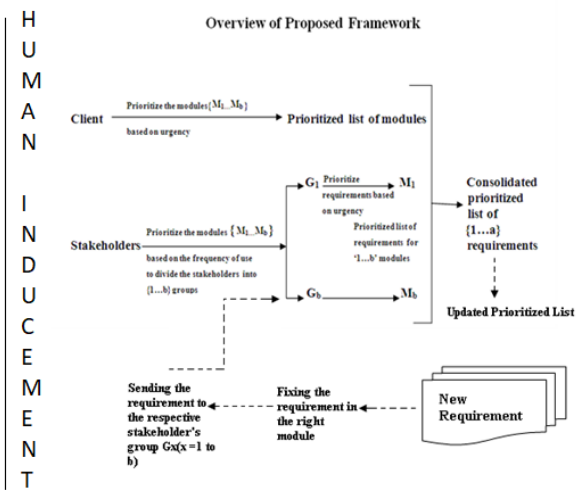

Figure 1. PAS Framework

VII. GOAL OF THE WORK

The aim of this paper is to find the answer for the ensuing questions which aroused as a challenge when the framework was proposed in [13].

RQ1: Still is there a chance to reduce the number of comparisons?
RQ2: Is there a way to reduce the number of users involved in the process?
RQ3: Can the consolidated priority list be updated after the arrival of new requirement?

A. To reduce the number of comparisons

Aim of this work is to reduce the number of comparisons made in AHP to prioritize the requirement. In paper [13], the number of comparisons are reduced from 1518 to 141. Still it is considered as a large number. The solution for this challenge is to select half from the first quartile of stakeholders to prioritize the requirements in modules. This will also definitely reduce the number of comparisons which addresses the second challenge

B. To reduce the number of users involved in the process

Selecting half from the first quartile of stakeholders to prioritize the requirements in modules will definitely reduce the number of comparisons which addresses the second challenge. This will result in selecting the pertinent stakeholder to rate the requirements.

C. Updating the priority list after the arrival of new requirement

The frame work suggested in [13] is to prioritize the requirement collected in the beginning. If any new requirement arises in the during the development of the software, then the list has to be updated. The proposed framework addresses this challenge technically.

According to the proposed framework, after receiving the new requirement from the client, it will be found out that to which group the requirement will belongs to. So, after that, the requirement will be send to that group members to compare it with all the requirements. Then the prioritized list of that respective module will be updated. And then the final prioritized list of the requirement will be updated. Human inducement also required to make the stakeholders actively participate in the process. Points can be given to the stakeholders ,if they complete one matrix.

VIII. THE EXPERIMENT

In this study, the software taken is "Website for Career Guidance", which has 4 modules. 96 users were selected randomly. First, they have to rate the modules (given in table 5.2) based on the frequency of usage of the modules. They were given a short training about how to fill the AHP sheet with the values.

The users are allowed to fill only the cells in the right half above the diagonal. If the modules in row are important than column, then the integer value has to be given. Otherwise, decimal value has to be given as given in the table 5.1.

Table 2 Criteria

Criteria	Row wise important	Column wise important
Equally used	1	1
Moderately Frequent	3	0.33
Strongly Frequent	5	0.2
Very Strongly Frequent	7	0.14
Extremely Frequent	9	0.11

A. Users: Since 4 modules were taken for this research work, the selected 96 users have to make only 3 entries. So totally 96 * 4 =204 entries.
Sample AHP given to each user is given in table 3

Table 3
AHP sheet for users to prioritize the modules based on the frequency of usage

Modules	Aptitude test(M1)	Workshops, Seminar and Conference details(M2)	Job Openings(M3)	Books Available(M4)
Aptitude test(M1)	1	9	9	5
Workshops, Seminar and Conference details(M2)	0.11	1	1	0.33
Job Openings(M3)	0.11	1	1	0.14
BooksAvailable(M4)	0.2	3.03	7.14	1
Total	1.42	14.03	18.14	6.47

After receiving the data of frequent use of modules from the user, the table is normalized (Table 4).

Table 4

Modules	Aptitude test(M1)	Workshops, Seminar and Conference details(M2)	Job Openings(M3)	Books Available (M4)	Tota	Average
Aptitude test(M1)	0.70	0.64	0.50	0.77	2.61	0.65
Workshops, Seminar and Conference details(M2)	0.08	0.07	0.06	0.05	0.26	0.06
Job Openings(M3)	0.08	0.07	0.06	0.02	0.23	0.06
Books Available(M4)	0.14	0.22	0.39	0.15	0.90	0.23
Total	1.00	1.00	1.00	1.00	3.00	1.00

Normalised form of Table 3

Similarly, for the data received from all 96 users, all 96 matrices are normalized and average column for all 96 matrices were derived.

Then the 'average' attribute for the rows M1,M2 and M3 were selected from all 96 matrices (Table 5)

Table 5

USER	1	2	3	...	94	95	96
M1	0.62	0.57	0.59		0.59	0.63	0.65
M2	0.04	0.22	0.22		0.13	0.14	0.06
M3	0.13	0.15	0.14		0.04	0.04	0.06
M4	0.21	0.06	0.05		0.25	0.2	0.23

Average Attribute from all the users for the modules

Users with m1 higher is clustered in to one group. Similarly, for m2, m3 and m4. Then for each group, the differences between the module which is having higher value with the other modules are calculated. The differences represent how frequent the user is using the first module than the second. Higher the difference, higher the usage of first module.

If no user is using the particular module(m4) frequently, then the user is sorted according to the usage of the module. First half from the list is selected. Difference between the usage of that module with other modules will be calculated. Higher difference column will be selected. Median will be derived. First quartile above the median will be selected. If any of the user selected is already selected to rate the requirements in another module(m1), then the particular user will be withdrawn from rating the requirement from that module(m1). And they will be allotted with module m4.

The data given here is for the module1.

Table 6

USER	M1	M2	M3	M4	M1,M2	M1,M3	M1,M4
1	0.62	0.04	0.13	0.21	0.58	0.49	0.42
2	0.57	0.22	0.15	0.06	0.35	0.42	0.5
3	0.59	0.22	0.14	0.05	0.38	0.46	0.54
4	0.58	0.25	0.12	0.05	0.32	0.45	0.53
5	0.59	0.24	0.12	0.05	0.36	0.47	0.55
6	0.55	0.17	0.15	0.14	0.38	0.39	0.41
25	0.62	0.04	0.13	0.21	0.58	0.49	0.42
26	0.57	0.22	0.15	0.06	0.35	0.42	0.5
27	0.59	0.22	0.14	0.05	0.38	0.46	0.54
28	0.58	0.25	0.12	0.05	0.32	0.45	0.53
29	0.59	0.24	0.12	0.05	0.36	0.47	0.55
30	0.55	0.17	0.15	0.14	0.38	0.39	0.41
49	0.62	0.04	0.13	0.21	0.58	0.49	0.42
50	0.57	0.22	0.15	0.06	0.35	0.42	0.5
51	0.59	0.22	0.14	0.05	0.38	0.46	0.54
52	0.58	0.25	0.12	0.05	0.32	0.45	0.53
53	0.59	0.24	0.12	0.05	0.36	0.47	0.55
54	0.55	0.17	0.15	0.14	0.38	0.39	0.41
73	0.62	0.04	0.13	0.21	0.58	0.49	0.42
74	0.57	0.22	0.15	0.06	0.35	0.42	0.5
75	0.59	0.22	0.14	0.05	0.38	0.46	0.54
76	0.58	0.25	0.12	0.05	0.32	0.45	0.53
77	0.59	0.24	0.12	0.05	0.36	0.47	0.55
78	0.55	0.17	0.15	0.14	0.38	0.39	0.41
19	0.62	0.13	0.03	0.21	0.49	0.59	0.41
20	0.62	0.13	0.03	0.21	0.49	0.59	0.41
21	0.59	0.13	0.04	0.25	0.47	0.56	0.41
22	0.59	0.13	0.04	0.25	0.47	0.56	0.41
23	0.63	0.14	0.04	0.2	0.49	0.59	0.41
24	0.65	0.06	0.06	0.23	0.59	0.6	0.41
43	0.62	0.13	0.03	0.21	0.49	0.59	0.41
44	0.62	0.13	0.03	0.21	0.49	0.59	0.41
45	0.59	0.13	0.04	0.25	0.47	0.56	0.41
46	0.59	0.13	0.04	0.25	0.47	0.56	0.41
47	0.63	0.14	0.04	0.2	0.49	0.59	0.41
48	0.65	0.06	0.06	0.23	0.59	0.6	0.41
67	0.62	0.13	0.03	0.21	0.49	0.59	0.41
68	0.62	0.13	0.03	0.21	0.49	0.59	0.41
69	0.59	0.13	0.04	0.25	0.47	0.56	0.41
70	0.59	0.13	0.04	0.25	0.47	0.56	0.41
71	0.63	0.14	0.04	0.2	0.49	0.59	0.41
72	0.65	0.06	0.06	0.23	0.59	0.6	0.41
91	0.62	0.13	0.03	0.21	0.49	0.59	0.41
92	0.62	0.13	0.03	0.21	0.49	0.59	0.41
93	0.59	0.13	0.04	0.25	0.47	0.56	0.41
94	0.59	0.13	0.04	0.25	0.47	0.56	0.41
95	0.63	0.14	0.04	0.2	0.49	0.59	0.41
96	0.65	0.06	0.06	0.23	0.59	0.6	0.41

Difference between M1 with other modules

Then the column with the higher value is selected. The selected column along with the user column is created as a separate table. The records are sorted based on the difference value from higher to lower. This sorts from the more frequent users to less frequent users.

Table 7

USER	M1, M3	USER	M1, M3	USER	M1, M3	USER	M1, M3	USER	M1, M3
24	0.6	**43**	0.59	**69**	0.56	**53**	0.47	**2**	0.42
48	0.6	**44**	0.59	**70**	0.56	**77**	0.47	**26**	0.42
72	0.6	**67**	0.59	**93**	0.56	**3**	0.46	**50**	0.42
96	0.6	**68**	0.59	**94**	0.56	**27**	0.46	**74**	0.42
23	0.59	**91**	0.59	**1**	0.49	**51**	0.46	**6**	0.39
47	0.59	**92**	0.59	**25**	0.49	**75**	0.46	**30**	0.39
71	0.59	**21**	0.56	**49**	0.49	**4**	0.45	**54**	0.39
95	0.59	**22**	0.56	**73**	0.49	**28**	0.45	**78**	0.39
19	0.59	**45**	0.56	**5**	0.47	**52**	0.45		
20	0.59	**46**	0.56	**29**	0.47	**76**	0.45		

Sorted Users Based on Difference Column

The median for the difference column (m1, m3) is calculated. The users from the first quartile above the median is selected to rate the requirements of the module.

Table 8

Median	0.52
No. of users above the median is 24	
First quartile above the median is selected i.e. users	
24	
48	
72	
96	
23	
47	
for the first module	

Selection of users for the first module

The median value for the difference column is calculated. Users from the first quartile above are the median selected to rate requirements from the respective module(module1).

Table 9

Requirements	User 1	User 2	User 3	User 4	User 5	User 6	Total	Average	From Client	Priority
M11	0.18	0.21	0.18	0.18	0.21	0.18	1.15	0.19	0.69	0.13
2	0.1	0.1	0.1	0.1	0.1	0.1	0.62	0.1	0.69	0.07
3	0.04	0.04	0.04	0.04	0.04	0.04	0.25	0.04	0.69	0.03
4	0.68	0.64	0.68	0.68	0.64	0.68	3.99	0.66	0.69	0.45
M25	0.74	0.74					1.48	0.74	0.2	0.15
6	0.19	0.19					0.38	0.19	0.2	0.04
7	0.07	0.07					0.14	0.07	0.2	0.01
M38	0.2	0.2					0.4	0.2	0.06	0.01
9	0.75	0.75					1.49	0.75	0.06	0.05
10	0.05	0.05					0.11	0.05	0.06	0
M411	0.75	0.73	0.71	0.73	0.71		3.63	0.73	0.05	0.04
12	0.06	0.05	0.05	0.05	0.05		0.27	0.05	0.05	0
13	0.19	0.22	0.24	0.22	0.24		1.1	0.22	0.05	0.01

Priority given for the requirements by users

For Module 1, 6 users are selected. 2 users for module 2 and 3 and 5 users for module4 are selected to rate the requirements in the respective module. Average column from the normalized matrices are taken for consolidation.
The average is calculated for the requirements. The average column from the matrix(modules rated by clients) are taken. Then the average column is multiplied with the column rated by clients. The result column is the priority value of the requirements. Then the requirement column and the priority column are selected and sorted based on the priority value.

Table 10

Requirements		Priority
M1	4	0.45
M2	5	0.15
M1	1	0.13
M1	2	0.07
M3	9	0.05
M2	6	0.04
M4	11	0.04
M1	3	0.03
M2	7	0.01
M3	8	0.01
M4	13	0.01
M3	10	0.00
M4	12	0.00

When the new requirements arrived , first the module it belongs to will be determined. Then it will be given to respective group of selected users directly. Along with all the other requirements, the newly arrived has to be compared and rated by the users in that group. After this process, priority list will be automatically updated with the new requirement. For the above experiment with the proposed framework, the project selected has 4 modules. Module M1 has 4 requirements, Module M2 has 3 requirements, Module M3 has 3 requirements and Module M4 has 3 requirements. 3 clients and 96 users were selected to rate the modules based on urgency and frequency of usage respectively. With the above framework, it requires 663 comparisons totally

a. To rate the module

By clients:number of modules: 4
number of comparisons by each client: 6
Total number of comparisons by clients: 4*6= **24**
By users:number of users:96
number of comparisons by each user: 6
Total number of comparisons by users:96*6= **576**
TOTAL : **600**

b. To rate the requirements:

By users

Module1: Number of users selected:6
Number of comparisions by each user:6
Total number of comparisions by users:6*6 =3
Module2:Number of users selected:2
Number of comparisions by each user:3
Total number of comparisions by users:2*3 =6
Module3:Number of users selected:2
Number of comparisions by each user:3
Total number of comparisions by users:2*3 =6
Module4:Number of users selected:5
Number of comparisions by each user:3
Total number of comparisions by users:5*3 =15

63

Total number of comparisons = 663

IX. TO RATE ALL THE REQUIREMENTS BY ALL THE SELECTED USERS THROUGH AHP

All the 96 users provided with all the requirement to compare, totally 13 requirements, (13(13-1)/2=78) comparisons for a single user. For the 96 users and 3 clients ,99*78=7722 comparisons. Since it is difficult to judge who will use which module frequently, a greater number of users has to be selected to prioritize the requirement. Sample sheet rated by the user is given

Table 11

Requirements	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13
R1	1	7	0.14	0.14	0.14	7	7	7	7	7	5	5	5
R2	0.14	1	9	0.14	0.14	5	7	5	0.14	9	5	7	7
R3	7.14	0.11	1	0.11	0.14	0.14	0.2	0.11	0.14	0.14	5	5	5
R4	7.14	7.14	9.09	1	9	9	9	9	9	9	9	9	9
R5	7.14	7.14	7.14	0.11	1	7	7	5	9	9	9	9	9
R6	0.14	0.2	7.14	0.11	0.14	1	7	7	5	3	7	7	7
R7	0.14	0.14	5	0.11	0.14	0.14	1	7	0.14	7	5	7	7
R8	0.14	0.2	9.09	0.11	0.2	0.14	0.14	1	0.14	7	0.14	9	9
R9	0.14	7.14	7.14	0.11	0.11	0.2	7.14	7.14	1	5	0.14	5	7
R10	0.14	0.11	7.14	0.11	0.11	0.33	0.14	0.14	0.2	1	7	5	9
R11	0.2	0.2	0.2	0.11	0.11	0.14	0.2	7.14	7.14	0.14	1	9	9
R12	0.2	0.14	0.2	0.11	0.11	0.14	0.14	0.11	0.2	0.2	0.11	1	5
R13	0.2	0.14	0.2	0.11	0.11	0.14	0.14	0.11	0.14	0.11	0.11	0.2	1
TOTAL	23.9	30.68	62.5	2.39	11.46	30.4	46.11	55.76	39.3	57.59	53.5	78.2	89

Comparison by each user

The next step is to normalize the matrix. Then the average column of all the users was taken to find the priority of the requirement.

Table 12

Users	1	2	...	96	TOTAL
R1	0.104940884	0.103951474		0.105356159	8.336254993
R2	0.083141826	0.081528251		0.081601663	6.507816812
R3	0.046722621	0.046606881		0.026490036	4.056133357
R4	0.254135767	0.252532094		0.25756312	20.37173401
R5	0.155519079	0.153987893		0.153981899	12.40625881
R6	0.074960472	0.074275885		0.074500889	5.840710165
R7	0.05294235	0.051761813		0.052487381	4.04239546
R8	0.045506203	0.044173444		0.044910079	3.53315293
R9	0.073566378	0.074916969		0.072324432	5.845558531
R10	0.039620867	0.039213282		0.039515375	3.044378001
R11	0.048523179	0.056690219		0.077911967	4.059282331
R12	0.012447626	0.012412064		0.007654239	0.898882082
R13	0.007972749	0.007949732		0.025702761	0.987443107
	1	1		1	

Normalization of Table 11

X. ARRIVAL OF NEW REQUIREMENT

After obtaining the priority list, new requirement was added in the third module as its 4th requirement. 3 comparisons per user has been taken place. Hence by the proposed framework, for 2 users, 6 comparisons. Finally, total number of comparisons after this process is 663+6=669.

By using the normal AHP sheet 13 comparisons for each user. 13 * 99 = 1287. Totally 7722+1287=9009

XI. COMPARITIVE STUDY

A. Through proposed framework PAS

The priority list obtained from the proposed framework before the arrival of new requirement is given in table 13

Table 13

Requirements	Priority
4	0.4547
5	0.1469
1	0.1327
2	0.0713
9	0.0478
6	0.0378
11	0.0363

3	0.0285
7	0.0139
8	0.0128
13	0.0110
10	0.0035
12	0.0027

Priority list before the arrival of new requirement

Table 14

Requirements	priority
4	0.454652
M2 5	0.146939
M1 1	0.132682
2	0.071278
9	0.042283
6	0.037841
M4 11	0.036296
3	0.028456
7	0.013941
M3 8	0.013699
13	0.011019
10	0.008189
12	0.002726
10A (new)	0.002337

Priority list after the arrival of new requirement

B. Only through AHP sheet

The priority list obtained from AHP sheet alone for all the requirements from all the users is given in the Table 15

Table 15

REQUIREMENTS	PRIORITY
R4	20.37173
R5	12.40626
R1	8.336255
R2	6.507817
R9	5.845559
R6	5.84071
R11	4.059282
R3	4.056133
R7	4.042395
R8	3.533153
R10	3.044378
R13	0.987443
R12	0.898882

Before the arrival of new requirement

Table 16

Requirement	Priority
R4	20.37173
R5	12.40626
R1	8.336255
R2	6.507817
R9	5.845559
R6	5.84071
R11	4.059282
R3	4.056133
R7	4.042395
R8	3.533153
R10	3.044378
R13	0.987443
R12	0.898882
R10A	0.557509

After the arrival of new requirement

XII. CONCLUSION

The proposed method yields the suitable result. From Table13 and 15, the data is taken and it is proved that 84% of the result is same through Python Environment (Figure 4 and 5) and the chart that shows pictorial comparison is displayed in figure 3.

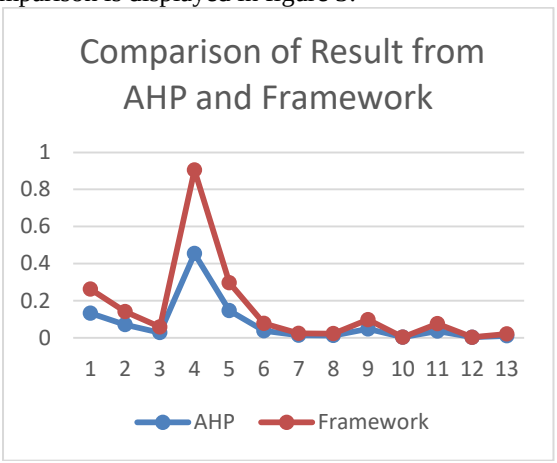

Figure 2 . Comparision of result from AHP and PAS

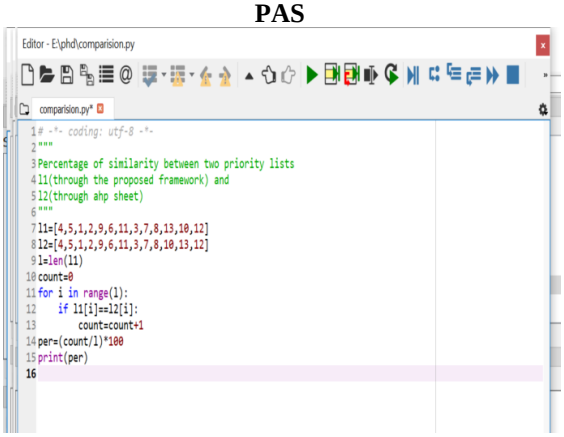

Figure 3. Coding to compare both the result

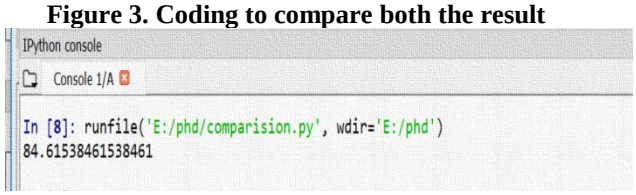

Figure 4 Coding to compare two priority list before adding new requirement

After adding new requirement

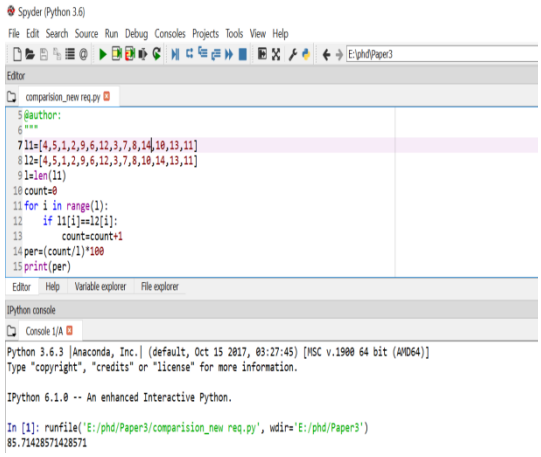

Figure 5. Coding and output (to compare two priority list after adding new requirement)

This study was based on the existing model from the reviewed literature. Beside the model, the proposed framework is suitable to prepare the priority list of the requirements. Hence the framework results in the reduced number of comparison of requirements, reduced number of user involvement, which is suitable for large scale software development.

XIII. FUTURE WORK

The future work of the research is focused on improving the tool for the proposed framework. Some more improvement has to be done to get 100 percent equivalent result with AHP sheet. To find the way to reduce the number of users to rate the requirements based on importance/urgency. Human inducement is required to fill the data. Points can be handled to encourage every responder.

ACKNOWLEDGMENT

First of all, we express thanks to God for granting us this prospect to carry out our research work. We thank our family members for their enormous support. We would also like to show our gratefulness to the reviewers for giving out their pearls of wisdom with us during the review of this paper and for their comments on an earlier version of the manuscript.

REFERENCES

1. An Appraisal Of Software Requirement Prioritization Techniques Iroju Olaronke1, Ikono Rhoda2 And Gambo Ishaya2, 1 Department Of Computer Science, Adeyemi College Of Education, Ondo, Nigeria. 2 Department Of Computer Science And Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria,Vl - 1,Do - 10.9734/Ajrcos/2018/40763
2. Saffron: A Semi-Automated Framework For Software Requirements Prioritization, Syed Ali Asif, Zarif Masud, Rubaida Easmin, Alim Ul Gias, International Journal Of Advanced Computer Science And Applications, Vol. 8, No. 12, Pp. 491-499, 10.14569/Ijacsa.2017.081265
3. An inclusion of human inducement in engineering the system efficaciously.K. Glory vijayaselvi[1], Dr. R.Thirumalaiselvi, International Conference "Human Computer Interaction - Redefining Corporate Paradigms" , ISBN 978-81-92950-45-7
4. The Impact Of Analytical Assessment Of Requirements Prioritization Models: An Empirical Study, Aneesa Rida Asghar,Dr. Shahid Nazir Bhatti ,Atika Tabassum ,Dr. S Asim Ali Shah ,(Ijacsa) International Journal Of Advanced Computer Science And Applications, Vol. 8, No. 2, 2017
5. A survey on prioritization methodologies to prioritize non functional requirements, Saranya B, Subha R,Dr. Palanisamy S, IJCSBI, ISSN 1694-2108, vol 12, April 2014
6. Tool-Supported Requirements Prioritization: Comparing The *Ahp* And *Cbrank* Methods, Ana Preini, Filippo Ricca, Angelo Susi, Information-And-Software-Technology, Volume 51,Issue 6, June 2009, Pages 1021-1032
7. Comparison Of Requirement Prioritization Techniques To Find Best Prioritization Technique, Javed Ali Khan, Izaz Ur Rehman, Yawar Hayat Khan, Iftikhar Javed Khan, Salman Rashid, I.J. Modern Education And Computer Science, 2015, 11, 53-59, Doi: 10.5815/Ijmecs.2015.11.06
8. Tool-Supported Collaborative Requirements Prioritisation,Palo Busetta, Fitsum, Denise, Anna Perini, Alberto Siena, Angelo Susi, Doi: 10.1109/Compsac.2017.243, Conference: Ieee 41st Annual Computer Software And Applications Conferenceunderstanding Requirement Prioritization Artifacts: A Systematic Mapping Study
9. Understanding Requirement Prioritization Artifacts: A Systematic Mapping Study,Rahul Thakurta, Requirements Engineering,Issn: 0947-3602 (Print) 1432-010x (Online), November 2017, Volume 22, Issue 4, Pp 491–526,
10. A Systematic Literature Review Of Software Requirements Prioritization Research, ,P Achimugu, Ali Selamat, Roliana, Mohd Nazri Mahrin, Information And Software Technology Volume 56, Issue 6, June 2014, Pages 568-585
11. Comparing Ahp And Electre I For Prioritizing Software Requirements,Joao M Fernandes, Susana Prozil Rodrigues, Lino A. Costa ,2015 Ieee/Acis 16th International Conference On Software Engineering, Artificial Intelligence, Networking And Parallel/Distributed Computing (Snpd), Sndp 2015,June 1-3 2015,Ieee
12. Chassis to Prioritize the Requirements through AHP with Diminutive Comparisons,K. Glory Vijayaselvi[1], Dr. R.Thirumalaiselvi[2], Jour of Adv Research in Dynamical & Control Systems, Vol. 11, 06-Special Issue, 2019
13. Survey on requirement elicitation technique, Joseph, Amit, AbdulRahman, Abiodun, IJIRCCE, Vol 5, Issue 5, May 2017, 10.15680

AUTHORS PROFILE

Ms. K Glory Vijayaselvi is currently pursuing Ph.D. in Bharathiar University, Coimbatore, India and currently working as Assistant Professor in Department of Computer Science (Shift II), Women's Christian College, Chennai, India. Her area of specialization is Software Engineering

Dr. R. Thirumalaiselvi is currently working as Assistant Professor in Government Arts College (Men), Autonomous, Department Of Computer Science, Nadanam, Chennai, India. Her areas of specialization are Web Engineering and Web Mining (Data Mining)