

# THE INTERNATIONAL JOURNAL OF HUMANITIES & SOCIAL STUDIES

## An Analysis of Library and Non- Library Software Development Problem Factors: A Comparative Study

**Dr. Nader Saboktakin**

Faculty of Humanities, Department of Knowledge and Information Science,  
Persian Gulf University, Iran

**Sedigheh Mohammad Esmaeil**

Associate Professor, Department of Knowledge and Information Science,  
Islamic Azad University, Science and Research Branch of Tehran, Iran

### **Abstract:**

*This research intends to analyze comparatively the challenges in development of computer software with various subjects and library computer software in country. This research is an applied research and through an analytical-survey method, evaluated challenges in producing the library and non-library software. Questionnaire is tool of data collection and statistical society consists of producers of custom software which respectively are about 600. Sample size includes 234 peoples of which 201 questionnaire i.e., 86% have been filled. Interview with experts have been performed in order to determine the reliability, and Cronbach's alpha is 0.8 for the questionnaire. According to test of comparison of two societies' means in library software; challenges of documentation with measured t-statistics is 3.411 more than these challenges in non-library software. Also, the maximum difference is related to challenges of documentation and system requirements. In order to decrease challenges in software producing which means quality promotion and a precise estimate of cost and time; library software companies ought to pay more attention than non-library companies software to steps or procedures of documentation than other phases or areas.*

**Keywords:** Comparative Analysis, Software Development, Non-Library Software, Library Software, Problem Factor, Iran

### **1. Introduction**

Undoubtedly, information science area in Iran considering new relations with science, industry, research and producing has become an area which may be called beating heart of scientific community in the country. In this regard, using information carriers, particularly computer software may be attended by librarians who work in information science area. Software production by information and communication technology industry and software engineering is an achievement that not only attracted public interest, trust and a huge inclination in volunteers entering universities for this major, but also this new information supplement attracted librarians' attention as organizers of information science system in the country. Furthermore, since study and research in relation to computer software development as a key element in development and movement is unavoidable, pathology and study on challenges in production of such software is important and unavoidable too. Therefore, study about software products is very important due to automatization of activities in organizations including libraries and conducting such researches are suppose necessary for librarians and other people working in information science due to their significant role in providing information sources for society. In other words, conducting this research is necessary because from one side, it studies software producing in the country and in other side compares challenges in development of library software toward non-library software packages with various subjects. Evidently, this study tries to find out strong and weak points in library software development which can be used to support strong points and remove weak points.

Information is in a high importance and position which is not possible to imagine a rival for it. A majority part of information which is presented in new carriers has been made by software producers. There are several definitions for software. According to a definition by Institute of Electrical and Electronics Engineering (IEEE); software is computer programs, procedures and possibly associate documentation and data pertaining to the operation of a computer system. Also, software or computer program is defined as detailed programming or process of question solving by computer through clear and successive computing commands. Software is not only a highly innovative and economically important sector in its own right, but it is often also an important element of development in other sectors. Many process and structural development depend heavily on organizational changes that are facilitated by software innovations (Haryani and Gupta, 2016).

Non-library software in this study means software which is provided for a customer as an employer in a mutual contract with a software company as a contractor. Custom software programs are related to various issues and branches as: Educational, administrative, financial, industrial, hospital and electronic health, banking, insurance, transportation, planning and engineering, cultural, religious, legal and judicial, security and military, geographical information, hospitality,

publishing services, advertising artistic, tourism, public information, internet services (web pages designing) electronic commerce etc. The first library software was invented about a half century ago. Library software is actually a computer program which is used to storage, process and retrieves data and is written by a current programming language. However, H.P. Luhn used a computer for the first time in 1961 to create a keyword in context (KWIC) index for chemistry abstract papers (Raad, 2002). By producing software, he provided a list of keywords of papers' titles in chemistry abstracts. At the same time, Douglas Aircraft Company would make a list of letters by a computer (Modaresi, 2005). It was success of Machine-readable cataloging (MARC) plan in the late 1960s in designing a pattern to insert and store bibliographic data showed that facilities of storage and retrieval of library software can be developed through data exchange and implementation of storage standards. So, in addition to borrowing book section, stored data can be used in other sections such as cataloging, ordering and submitting, periodicals and library comprehensive list. Although UNESCO attempted to help third world countries and made an open-source library software under the title of "DS/ISIS" (Computerized Documentation Service / Integrated Set of Information Systems and it became the base of library software companies in the world including Pars Azarakhsh company and NOSA company (Iran Software and Hardware Co), however they developed a Persian version of mentioned software to use by libraries and also have converted it from DOS-based operating system to a windows-based software and promoted it as a web-based program, such various versions of the software were made during 1990s and librarians' advices and critiques had a significant role for promoting library software. At present time, these new versions of library software are compatible with similar foreign products and they have released themselves from limits of original software since many years ago. (Fatahi, Parirokh, 2005). Using programming codes and parts or modules which already had been used by other non-libraries software producers concerning various subjects is a common method to be used by new software producers and purportedly there is no need to reinvent the wheel. However, due to the fact that working procedures in different libraries regardless of library type is affected by organizational culture which is arisen from culture-society, customizing and adding, removing facilities of the software is unavoidable, thus, software companies use various methodologies to make software.

Principally, in comparison with other industrial products, computer software production is more risky. According to Standish Group Chaos Report presented in 1994 concerning 365 managers in IT part of medium companies which is known as Software Crisis; only 16% of software development projects have been completed successfully while 53% has faced with challenges and 31% have failed and abandoned. Results of report for 2004 show that only 29% of projects have been completed successfully, while 53% had problems and 18% of projects had failed. Such report for 2008 shows only 32% of software development projects have been completed successfully while 68% had problems and/or failed (Merchuka, 2012). Success of a software development project depends on precise schedule for completion, performing by exact initial cost survey and considering software quality in development process and achieving facilities which had been estimated at the beginning of project (On time, on budget, with quality). Also, challenge here means a situation in which a project has already been completed, but imposed higher cost, more time and lower quality than initial expectations due to lack of appropriate planning and policies in time, cost and quality management. If a project is stopped or abandoned in any step of its life cycle, it means the project has been failed (Mensah, 2003).

### *1.1. Research Questions*

What are demographic specifications of library and non-library software developers?

What is difference between producers' challenges that develop library software with non-library software developers?

### *1.2. Research Hypothesis*

Library software development challenges (including documentation, system requirements, programming, maintenance personnel resources and process management) are more than non-library software development challenges.

## **2. Literary Review**

Arastoopour (2007) studied feasibility of using the Koha in large and medium libraries in Iran. Considering some features related to main modules of this software, he explains that although Koha is one of the best open source code software, but it's not suitable for huge academic libraries. Sadat Hosseini, Kushki & Asefi. (2009) mention to importance and necessity of using open source code software. They compared two software of Koha with PHPMyLibrary regarding various facilities in different parts of library. Results show that Koha software has more facilities and makes fewer challenges in meeting needs. Also, Sabaghchi, Ghazi Noori and Elahi (2011) studied use of knowledge management tools including groupware or cooperation and interaction among peoples to decrease challenges related to different steps of software development process such as requirements and maintenance steps. Jalilpoor (2012) and also Shafiei and etal. (2013) have studied respectively level of students' satisfaction in Chamran University and librarian's satisfaction in Isfahan University of Medical Sciences from elements related to Pars Azarakhsh software user interface and results show their average satisfaction and necessity of software production based on users' needs. Parirokh, Sadat Afshik and Rajabali Baglu (2013) studied usability as a non-functional requirement in Simorgh software which has been made by NOSA company. Results show that; this software has challenges in use and is not able to meet users' needs. Kumar (2005) in a paper titled "library integrated management system of open source code" studied three open source code software packages. Results show that Koha has the least challenges and obtained the highest point among these three software packages. However, My Library software and open Biblio software are suitable for libraries with small complexes. Chen & Huang (2009) have

studied use of some methodologies in decreasing software development challenges as well focused on necessity of more importance on some steps and processes of software development in order to decrease cost and time of software development. Also, Taher et al. (2010) studied observing standards and procedures in software development steps which entail software quality promotion and more satisfaction in users. Kanabiran & Senkaran (2010) studied some challenges which decrease software external quality. Garoufallou (2012) studied and compared eight web search engines software by 16 librarians in Greece which includes usability in relation to system requirements. According to findings, 66.5% of mentioned librarians expressed their satisfaction or high satisfaction from useful results of searches, while other responders had low satisfaction and/or were completely unhappy. Alhandavi (2013) considered programming challenges related to software development and recognition methods of program source code by programmers. Sfetsos and etal (2014) studied 26 library software packages of which one-half was network-based and the other half included software packages which were used without network. According to their findings, there is a significant relation between library software design and other non-network software regarding usability, flexibility and comprehensibility, while there is a significant relation in other features such as function facility, generalizability and influenceability between the two groups. Lehtinen and etal (2014) studied internal and external factors which ignoring them makes the software development projects fail. Considering necessity of studying country software industry and recognition of challenges which software producers face with and also to promote position of library software industry and evaluation of their challenges with other producers' challenges who develop various software about various issues, thus, present research intends to know some internal challenges related to software development steps (system requirements, programming, maintenance) and some external challenges (documentation, personnel resources, process management) which lack of attention to them leads to decrease in quality and increase in affiliated time and cost. Therefore, this research tries to answer following questions:

### 3. Methodology

Present research is an applied study and quantitative method has been used to conduct it. Questionnaire was used as data collection tool and experts' interview (Delphi panel) was used to determine reliability. Also, using analytical-survey method, library and non-library software production challenges were compared and SPSS statistical software (version 19) in the form of descriptive statistics and inferential statistics was used to test research hypothesis and in order to determine significant or insignificant relation between variables. Statistical society consists of 598 producers and Sample size includes 234 items which was targeted based on Krejcie & Morgan table. Questionnaires were sent to project managers of custom software companies which also work in. Totally, 201 questionnaires were filled and completed by project managers of software development groups that mean more than 86% have been returned. In order to conduct present study; questionnaires which were completed by eight library software companies –Pars Azarakhsh, NOSA, Arya Hamrahe Samaneh, Raye Mehr, Rahkareh Sarzamine Hooshmand, Mohandesi Narmafzari Holu, Gruhe Tejar at Electronic Rade Yazd, Ghangineh Raye Mehr -were compared with those completed by other companies which produce non-library software concerning various subjects. This study has formal and content validity considering questionnaires, interview with experts, literature review and other related sources and texts which were used. Also, Cronbach's alpha value is more than 0.8.

### 4. Findings

#### 4.1. What are Demographic Specifications of Library and Non-Library Software Developers?

Statistical analysis has been used in this part to study demographic specifications and statistical sample distribution extracted from library and non-library software developing companies regarding variables such as number of project team members, projects completion period, mean of project team members' experience and methodology used in projects.

Number of Project Members	Non-Library Software		Library Software	
	Frequency	Frequency Percentage	Frequency	Frequency Percentage
01-Feb	14	7	-	-
03-May	70	36.3	-	-
06-Oct	51	26.4	5	62.5
Nov-20	32	16.6	3	37.5
More than 20	26	13.5	-	-
Total	193	100	8	100

Table 1: Distribution of Respondents Based on Quantity of Project Members Separated By Library and Non-Library Software

Table-1 shows frequency distribution of number of project team members in producing library and non-library software. Of software producing teams related to other subjects; 14 project teams had 1 or 2 members (7%), 70 project teams had three or four members (around 36%), around 51 project teams had 6 to 10 members (26.4%), also 32 project

teams had 11 to 20 members (13.5%) and finally 26 project teams has more than 20 members (13.5%). However, of eight groups of library software developers, five project teams has 6 to 10 members (62.5%) and the three project teams has more than 20 members (37.5%). Overall, of 201 samples, the highest number belongs to group with 3 to 5 members with 70 project teams (36.3%).

Quantity of Project Member	Non-Library Software		Library Software	
	Frequency	Frequency Percentage	Frequency	Frequency Percentage
Less than 6 months	40	20.7	-	-
6-12 months	72	37.3	-	-
13-24 months	47	24.4	4	50
More than 24 months	34	17.6	4	50
Total	193	100	8	100

Table 2: Distribution of Respondents Based On Period of Project Completion Separated by Library and Non-Library Software

Table-2 shows frequency distribution of projects completion period in producing library and non-library software. Of software producing teams related to other subjects; 40 project teams had completed projects in less than 6 months (around 21%), about 72 project teams had completed projects in 6 to 12 months (37.3%), 47 project teams had completed projects in 13 to 24 months (24.4 %) and 34 project teams had completed projects in more than 24 months(17.6%). However of eight groups of library software developers, four project teams had completed projects in 13 to 24months (50%) and the other four project teams had completed projects in more than 20 months (50%). Overall, of 201 samples, the highest frequency belongs to group which had completed projects 6 to 12 months with number of 72(37.3%) project teams.

Quantity of Project Member	Non-Library Software		Library Software	
	Frequency	Frequency Percentage	Frequency	Frequency Percentage
Less than 1 year	1	0.5	-	-
1-3 years	45	22.3	1	12.5
4-6 years	103	53.4	6	75
7-9 years	34	17.6	1	12.5
More than 9 years	10	5.2	-	-
Total	193	100	8	100

Table 3: Distribution of Respondents Based on Average Experience of Group Members Separated by Library and Non- Library Software

Table-3 shows frequency distribution of average of project team members' experience in producing library and non-library software. Averagely, of software developing teams; 1 group had less than 1-year experience (0.5%), 45 project teams had 1-3 years' experience (around 22%), number of 103 project teams had 4-6 years' experience (about 53%), 34 project teams had 7-9 years' experience (17.6%) and finally 10 project teams had more than 9 years' experience (5.2%). However, of eight groups of library software developers, one project team had 1-3 years' experience (12.5%) and six project teams had 4-6 years' experience (75%) and one project team had 7 to 9 years' experience (12.5%). Overall, of 201 samples, the highest frequency belongs to the group members with 4-6 years' experience with 103 teams (53.4%).

Quantity of Project Member	Non-Library Software		Library Software	
	Frequency	Frequency Percentage	Frequency	Frequency Percentage
No Methodology Used	23	11.9	-	-
SPI Models	4	2.1	-	-
Agile	43	22.3	3	37.5
Rup	60	31.1	5	62.5
Other	63	32.6	-	-
Total	193	100	8	100

Table 4: Distribution of Respondents Based on Methodology Used Separated by Library and Non-Library Software

Table-4 shows frequency distribution of methodology use in developing library and non-library software. 4project teams (around 2%) use SPI methodology,43 project teams (around 22%) use Agile methodology, 60 project teams (31.1 %) use RUP methodology and number of 63 project teams (32.6 %) use other methodologies while 23 project teams don't

use a special methodology. However, of 8 groups of library software developers, 3 project teams use Agile methodology (37.5%) and 5 project teams use RUP methodology (62.5%). Overall, of 201 samples, the highest frequency belongs to the project teams which use RUP methodology in software development with 66 teams (32.3%).

#### *4.2. What Is Difference between Producers' Challenges That Develop Library Software with Non-Library Software Developers?*

In order to study challenges in library software companies and comparing them to other no-library software companies, 201 samples were examined of which 8 samples belonged to library software companies and 193 samples belonged to non-library software companies, descriptive statistics indices including mean and standard deviation were measured.

Table-5 shows descriptive statistics of research variables separated by library and non-library software development. As it can be observed, the highest mean difference is related to variable of project team members who not obligated to the project with mean of 1.96, so, library software companies have more challenges in obligating team members to the project in comparison with other software development companies. Also, in variables concerning system requirements, variable of 'system requirements are changed frequently' and 'There isn't any managerial supports and policy in software development process' with mean of 1.45 has the highest rate, thus, library software development companies ought to try more than other software companies in requirements step concerning employers' requirements. Also, library software challenges related to variable of 'programming and project control are inefficient' and 'project team members has no human resources management and time management' respectively with 0.02 and 0.06 have obtained less mean difference with mean of non-library software development challenges. Variables of "changes have not been documented completely" and "various modules have not been allocated in program to be independent with respect to functionality and operationality" and also variable of 'delivered software systems are not testable easily' respectively with mean of 0.12, 0.26 and 0.28 have the least mean difference with challenges related to non-library software development.

By examining production challenges of library and non-library software which are observed in table-6; the highest difference is related to the mean of Documentation challenges so that, mean of challenges related to Documentation in library software development companies with value of 3.88 is more than challenges related to Documentation in non-library software development companies with value of 3.33. Also, respectively, means of challenges related to system requirements, personnel resources, and programming in library software development companies are more with respect to non-library software development companies, but challenges related to maintenance and process management in library software development companies respectively with means of 2.65 and 3.07 are less than mean of non-library software development companies respectively with values of 2.78 and 3.13.

#### *4.3. Research Hypothesis*

Challenges related to library software development including documentation, system requirements, programming maintenance, personnel resources and process management, each one are more than mentioned challenges in non-library software development.

Test of comparison of two means: This test was used in order to compare challenges related to non-library software development with challenges related to library software development. To perform this test, at first, hypothesis of variance equality for both groups should be tested. Table-7 shows equality of variances for variables of the study. Also, null & alternate hypothesis were formed as follows regarding 5-point Likert scale:

##### 4.3.1. Hypothesis Testing

- H0: library software challenges are not more than non-library software challenges.
- H1: library software challenges are more than non-library software challenges.

Indices	Challenges	Quantity	Mean	Variance Difference	Mean Difference
1.Project has been documented in invalid or obscure manner	Documentation	193	3.16	1.099	-1.22
		8	4.38	0.518	
2.There has no document or they were poor		193	3.23	1.147	-0.77
		8	4.00	0.000	
3.Tracking the past documents related to design details and users' requirements in the project is difficult		193	3.29	1.080	-0.34
		8	3.63	1.061	
4.Canges have not been documented completely	System Requirements	193	3.61	0.984	-0.27
		8	3.88	0.991	
5.Documents are not comprehensive and consistent		193	3.37	1.029	-0.13
		8	3.50	0.756	
6. System requirements have been misdiagnosed.		193	2.83	1.192	-0.30
		8	3.13	1.246	
7.System requirements have been identified poor or obscure	Programming	193	3.23	1.096	-0.15
		8	3.38	1.188	
8. System requirements have been identified unrealistic or contradictory		193	2.50	1.090	-0.13
		8	2.63	0.916	
9. Requirements related to software quality have not been regarded		193	2.86	1.102	-0.64
		8	3.50	1.104	
10. System requirements are changed frequently	Maintenance	193	3.52	1.104	-1.11
		8	4.63	0.518	
11. Requirements are not according to programming standards		193	2.58	1.054	0.33
		8	2.25	0.886	
12. Comments related to source code are not complete		193	3.24	1.140	-0.26
		8	3.50	0.756	
13. In the program, different modules are not divided somehow to be dependent operationally and functionally	Personnel Resources	193	2.74	1.116	-0.14
		8	2.88	0.835	
14. The program has many complications and its reconstruction is not possible.		193	2.99	1.152	0.36
		8	2.63	1.188	
15. Improper use of programming techniques reduced the ability of source code comprehension		193	2.77	1.031	-0.73
		8	3.50	1.195	
16. Delivered software systems are not comprehensible and analyzable easily	Process Management	193	2.74	0.988	-0.64
		8	3.38	1.188	
17. Delivered software systems can't be optimized or changed easily.		193	2.89	1.045	0.39
		8	2.50	0.926	
18. Delivered software systems are not stable and they are not resistant against unexpected effects of changes		193	2.84	0.990	0.46
		8	2.38	0.518	
19. Delivered software systems are not easily testable.	Personnel Resources	193	2.74	1.112	0.24
		8	2.50	0.926	
20. Delivered software systems are not easily maintainable.		193	2.69	1.024	0.19
		8	2.50	0.926	
21. Frequent substitutions happen in project team.		193	2.95	1.169	-0.93
		8	3.88	0.835	
22. Members of project team don't have sufficient experience or skills.	Process Management	193	2.80	1.054	0.55
		8	2.25	0.707	
23. Members of project team have not obtained appropriate educations		193	2.98	1.070	-0.52
		8	3.50	1.195	
24. Members of project team don't have human personnel management and time management.		193	3.05	1.239	-0.20
		8	3.25	0.886	
25. Members of project team are not obligated to the project.	Process Management	193	2.55	1.141	-0.20
		8	2.75	1.165	
26. There isn't any managerial supports and policy in software development process.		193	3.04	1.220	-0.59
		8	3.63	1.188	
27. programming and project control are inefficient		193	3.04	1.058	0.54
		8	2.50	1.195	
28. There isn't proper estimate of cost and schedule of project execution	Process Management	193	3.33	1.048	-0.55
		8	3.88	0.835	
29. Configuration management is inefficient for changes control in the software.		193	2.97	1.087	-0.56
		8	2.38	0.744	
30. Quality control audits are inefficient to ensure the quality level	Process Management	193	3.26	1.013	0.26
		8	3.00	1.195	

Table 5: Descriptive Statistics of Questionnaire Questions Separated by Library and Non-Library Software

Variables	Mean		Standard Deviation	
	Non-Library Software	Library Software	Non-Library Software	Library Software
Challenges related to Documentation	3.33	3.88	0.862	0.413
Challenges related to System Requirement	2.99	3.45	0.845	0.707
Challenges related to Programming	2.87	2.95	0.785	0.665
Challenges related to Maintenance	2.78	2.65	0.799	0.438
Challenges related to Personnel Resources	2.87	3.12	0.801	0.623
Challenges related to Process Management	3.13	3.07	0.818	0.632

Table 6: Descriptive Statistics of Research Variables Separated by Library and Non-Library Software Development

According to the test results of equality of variances (table-7), it can be inferred with 95% confidence that variances of both groups (library and non-library software) are equal in variables of system requirements, programming, personal resources and process management, while they are not equal for two variables of documentation and maintenance. According to these results; value of T was measured to compare two means. If significance value of test (T-coefficients) in test of comparison of two means is more than 1.96, then null hypothesis will be rejected and alternate hypothesis will be confirmed and vice versa.

Levene's Test for Equality of Variances		T Statistic Test for Equality of Means				
Challenges	F statistics	Significant Level	T Coefficient	Degree of Freedom	Significant Level (One tailed)	Mean Difference
Documentation	6.396	0.012	3.411	9.741	0/0035	-0.541
System Requirement	0.416	0.520	1.518	199	0/0655	-0.460
Programming	1.864	0.174	0.301	199	0.382	-0.085
Maintenance	4.806	0.030	-0.771	9.066	0.2305	0.127
Personal Resources	1.111	0.293	0.905	199	0.183	-0.260
Process Management	1.002	0.318	-0.183	199	0/4275	0.053

Table 7: Tests of Equality of Variance and Means Compare

As it is observed in table-7; considering test of comparison of means of two societies and comparison of challenges related to non-library software development with those in library software development, challenges of documentation in library software with calculated t-statistics of 3.411 and freedom degree of 9.741 are more than these challenges in non-library software, therefore, alternate hypothesis is confirmed. But challenges of System requirement related to library software with calculated t-statistics of 1.518 and freedom degree of 199 are not more than these challenges in non-library software; therefore, null hypothesis is confirmed. Also, according to the study results; challenges of programming, maintenance, Personal resources and Process management with respectively calculated T of 0/301, -0.771, 0.905 and -0.183 are less than 1.96, thus, null hypothesis is confirmed and these library software challenges are not more with respect to these challenges in non-library software, in the other words, library software challenges are equal and/or less than non-library software challenges. It should be mentioned that; significance level for each variable has been divided in two parts and was compared with 0.05 in order to determine significance level with 95% confidence interval. Table-8 shows confirmation or rejection of research hypothesis.

Row	Hypothesis	Significance Level	Confirmation or Reject
1	Documentation challenges in library software development are more than mentioned challenges in non-library software development	3.411	confirmed
2	System requirement challenges in library software development are more than mentioned challenges in non-library software development	1.518	rejected
3	Programming challenges in library software development are more than mentioned challenges in non-library software development	0.301	rejected

Row	Hypothesis	Significance Level	Confirmation or Reject
4	Maintenance challenges in library software development are more than mentioned challenges in non-library software development	-0.771	rejected
5	Personnel resources challenges in library software development are more than mentioned challenges in non-library software development	0.905	rejected
6	Process management challenges in library software development are more than mentioned challenges in non-library software development	-0.183	rejected

Table 8: Study on Confirmation or Reject of Research Hypothesis

## 5. Discussion and Conclusion

According to demographic specifications, about one-fourth of project group members of non-library software developers which produce software about various subjects have experiences about 1-3 years and about 75% have experience for six years or less. However, 75% of members in library software producers group have 4-6 years experiences that is to say, software producers in country are young. It seems that one reason for challenges in software development is the very age of young producers and lack of obtaining the proper educations and this is reason why project managers are selected among most experienced members, therefore, proper educations leads to software quality promotion. Also, 58% of non-library software development is done during or less than a period of 12 months, while half of library development teams had completed projects in 13 to 24 months (50%) and the other half of library development teams had completed projects in more than 20 months (50%). Also, regarding test of comparison of two means; challenges in documentation was more than these challenges in development of other software, therefore, in order to decrease challenges that means quality promotion and have a precise estimate of cost and time of project completion, library software producers ought to pay more attention to mentioned steps and procedures in comparison with non-library software producers. The results show that system requirements challenges in library software producers is not more than these challenges in non-library software development with various subjects. Also in the study by Sfetsos and etal (2014) concerning library software, in usability which is related to system requirements and software quality; no significance relation has been confirmed between library software and other similar software which are used without connecting to a network. But according to Parirokh, Sadat Afshik and Rajabali and also Jalilpoor and etal results as they explained that Pars Azarakhsh and NOSA companies as the oldest producers of library software in country have to pay more attention to their needs due to poor production of appropriate user interface which is observable after installation and implementation and shows lack of proper recognition of beneficiaries' requirements including users. Also, Sadat Hosseini, Kushki & Asefi (2009) in comparing two software preferred Koha because of more facilities and less challenges in meeting needs. Also, in evaluating various software, Garufalo inferred that specification of usability related to system requirements is rather useful.

According to the study results; challenges of programming, maintenance, Personal resources and Process management with respectively calculated T of 0/301, -0.771, 0.905 and -0.183 are less than 1.96, thus, null hypothesis is confirmed and these library software challenges are not more with respect to these challenges in non-library software. However considering importance of coding, Taher, Ahmad and Kaziran (2010) explained that 77% of software development companies in their study use standard coding for programming in the projects and all software engineers in those companies have obtained necessary education on mentioned item. However, they explained that 19% of them use standard coding occasionally while 4% of companies are not aware of importance of standard coding and said they use standard coding in programming rarely. A study by Alhandawi (2013) shows that challenge related to programming strongly affect the software quality and since about 100 billion lines of related commands to current software have problems and include non-structured, poorly documented and patched codes, recognition of them has been challengeable; therefore, it takes a long time of programmers to recognize and read them. Meanwhile sometimes frequent changes is result of continuing education of project group, particularly according to the findings, most of project group members of the library software have experience equal or less than six years, that is to say, members in teams are mostly youth, thus they will leave the company. In a study by Letinen et al. (2014) the part which is more related to reasons of failure in software development projects is concerned to personnel with 29% which includes lack of experience, obligation and responsibility more than other personal issues.

## 6. References

- i. Alhendawi, N. (2013). Supporting Source Code Comprehension During Software Evolution and Maintenance. PhD Dissertation in Software Engineering, Kent University.
- ii. Arastoopoor, S., (2007), Military Software with Open Source Code: A review on Library Integrated System Modules and Abilities (Koha), No, 2(7), P.: 20-37
- iv. Chen, J. C.; Huang, S. J. (2009). An Empirical Analysis of the Impact of Software Development Problem factors on Software Maintainability. The Journal of Systems and Software, 82, 981- 992.
- v. Fatahi, R. & Parirokh, M. (2005), Library Software in Iran: Library and Information Science Encyclopedia, Tehran: National Library of I.R. Iran



- vi. Garoufallou, E. (2012). "Evaluating search engines", Program, Vol. 46, Iss 2 pp.182 – 198.
- vii. Haryani, S. and Gupta, v.b. (2016). Factors Affecting Innovation Capability Of Indian Software Firms (With Special Reference To Indore City). International Research Journal of Engineering and Technology (IRJET). Volume: 03 Issue: 12.
- viii. Jalilpoor, P., (2012), Analysis of Prominent Elements and Characteristics in Pars Azarakhsh Library Software Interface and Determination of Shahid Chamran University Students' Satisfaction from Using the Software, Systems & Information Services, Quarterly Magazine, 1<sup>st</sup> Year, No.4, P.:47-61.
- ix. Kannabiran, G.; Sankaran, K. (2011). Determinants of software quality in offshore development: An empirical study of an Indian vender. Information and software technology, 53, 1199-1208.
- x. kumar, V. (2005) Free/ Open Source Integrated Library Management Systems: Comparative Analysis of KOHA, PHP My Library and Open Biblio, Available at: <http://eprints.rclis.org/8074/01/FOSS-ILMS-ppt.pdf>
- xi. Lehtinen, T. O.A., Mantyla, M. V., Vanhanen, J., Itkonen, J. & Lassenius, C. (2014). Perceived causes of software project failures – An analysis of their relationships. Information and software technology, 56, 623-643.
- xii. Marchewka, J. T. (2012). Information Technology Project Management. Newyork: John Wiley and Son. p. 5. xii. Modaresi, F., (2005), Library Software in one Sight, Payame Baharestan, P.: 5-8.
- xiii. Parirokh, M., & Sadat Afshik, S., & Rajabali Beiglu, R., (2013), Simorgh Software Usability in Users' View: Necessity of Retrospect in Library Software Design Paradigm, Librarianship and Information Science Quarterly Magazine, Vol.16, No.2, P.62-71.
- xiv. Raad, Iraj (2002), Recognition of Desirable Characteristics of Library Software, Monthly Book of Generalities (February), P.: 109-113.
- xv. Sabaghchi, S., Ghazi Nuri, S. & Elahi, Sh., (2011), Selecting Knowledge Management Tools in Software New Products Development.Management Improvement, 5<sup>th</sup> year, No.2, P.: 80-100.
- xvi. Sadat Hosseini Kushki, Z., & Asemi, A., (2009), Evaluation of Open Source Code Software of Library Management: Comparative Analysis of and, Information Science Studies & Public Libraries, No.15 (4), P.: 103-123.
- xvii. Sfetsos, P.; Ampatzoglou, A.; Chatzigeorgiou, A; Deligiannis, I; Stamelos. I. (2014). A comparative study on the effectiveness of patterns in software libraries and standalone applications. International Conference on the Quality of Information and Communications Technology, 9<sup>th</sup>.
- xviii. Shafiei, S., Nuri, R., Rahimi, A., Shafiei, P. & Ghazavi, R., (2013), Features of Pars Azarakhsh Library Software in Isfahan University of Medical Sciences and their Utilization., Health Information Management, 10<sup>th</sup> Period, No.1
- xix. Tahir, A.; Ahmad, R.; Kasiran, Z. M. (2010). An Empirical Study on the Use of Standards and
- xx. Procedures in Software Development Projects.International Conference on Software