

Resistance Factors in the Implementation of Software Process Improvement Project in Malaysia

Mohd Hairul Nizam Md. Nasir, Rodina Ahmad and Noor Hafizah Hassan
Faculty of Computer Science and Information Technology, University of Malaya (UM), 50603,
Kuala Lumpur, Malaysia

Abstract: For the past two decades, improving the quality of software has become an important agenda in the software industries as they have been assessed according to standards such as the CMM, CMM-I, ISO 9000 SIX-SIGMA and etc. As the result, software process improvement project implementations have been the main targets by most software companies. However, many initiatives are facing variety of problems and difficulties due to lack of guidance and experience. Hence, this research attempted to identify and analyze main resistance factors which influenced the implementation of the software process improvement project specifically companies operated in Malaysia including local and multi-national companies. The findings helped other software companies to manage future projects through the use of preventive actions or proper planning which intended to lessen anticipated problems during software process improvement projects implementation. This research used a survey instrument to gather data from 29 companies operated across Malaysia with the total of 174 business and software professionals responded. Average of 4 to 8 questionnaires were distributed to each company with the objective of getting wider views on each SPI project. The questionnaires were mainly distributed to professionals who are directly involved in SPI projects. The results showed that the most critical resistance factor is lack of adherence and participation of the entire individual involved in SPI projects. This result is similar with the result gained by Brietzke and Rabello which they have conducted it in Brazil and corroborated the research findings experience in SPI project.

Key words: Software process improvement, resistance factors, SPI implementation

INTRODUCTION

Inspired by the efforts of Deming^[6] and Juran^[9], the software engineering community has realized that high-quality software development processes will produce high quality products. It has been generally agreed upon that, the quality of a product is largely governed by the quality of the process used to build it^[2]. Therefore, to obtain a high quality software product, it is essential to have a high quality process. Similarly, continuous improvement is necessary to achieve an acceptable high quality process. Subsequently, many software researchers and practitioners have refocused their efforts on the process dimension of software engineering. After two decades of research into productivity and quality from applying new software methodologies and technologies, industrial and governmental organizations begin to realize the benefits that can be gained from managing their software process. As an organization grows in maturity, its

software process becomes better defined since the organization institutionalizes its software process through standard, policies and organizational structures.

As a result, the current focus moved from the work products to the organizations that produced the work products. Various software process improvement (SPI) models such as Software Process Improvement and Capability determination (SPICE)^[8], Bootstrap^[16], ISO 9000^[18], Six Sigma^[13], the Capability Maturity Model (CMM)^[18] and the Capability Maturity Model Integration (CMMI)^[4] have been proposed to assist organizations to achieve more predictable results by incorporating proven standards and procedures into their software process. Organizations that have made use of these standards advocated in ISO 9000 and CMM have usually shown excellent improvements. For example, by improving its development process according to CMM maturity, Hughes Aircraft improved its productivity by 4-1 and saved millions of dollars^[15].

Corresponding Author: Mohd Hairul Nizam, Faculty of Computer Science and Information Technology,
University of Malaya (UM), 50603, Kuala Lumpur, Malaysia

Brietzke and Rabelo^[3] conducted a survey in Brazil to identify resistance factors influencing the implementation of a software process improvement projects. Essentially 36 respondents from 29 companies were collected and analyzed by them. The research contributes in identifying main resistance factors perceived as critical to the implementation of SPI. Since the survey has been conducted only in Brazil, it is obviously useful to replicate the study in other part of the world. The replication will aid to verify the factors as well as to compare whether there is any major difference in other country such as Malaysia. Subsequently, this research extracts the identified factors from Brietzke and Rabelo^[3], then, a set of survey question was developed and distributed to 29 software companies which have been involved in SPI projects. 4-8 questionnaires were distributed to each company with the objective of getting wider views on each SPI project. The questionnaires were mainly distributed to software professionals who are directly involved in SPI projects.

OVERVIEW OF SOFTWARE PROCESS IMPROVEMENT

Software process improvements: The processes for large-scale software development can themselves be quite large and complex which may involve many software engineers, programmers and designers. As a result, they are often hard to define, difficult to understand and sometimes even harder to establish or standardize. Software process improvement (SPI) emerges to tackle these process-related difficulties. The underlying assumption behind SPI and the many SPI frameworks is that quality process will influence the quality of product. We should therefore focus our improvement efforts on the software process in order to improve software quality. The following definitions are given to describe quality and software process:

- Quality-Kitchenham mentions that quality is hard to define, impossible to measure, easy to recognize^[10]. Most SPI frameworks specify a measurable quality indicator for the process.
- Software process-a set of partially ordered process steps, with sets of related products, human and computerized resources, organizational structures and constraints, intended to produce and maintain the requested software products^[12].

At present, software models for improving the quality of software through management of the software process have become significant in the

software industry. Many companies are now being assessed according to de facto standards such as the Capability Maturity Model (CMM), SIX SIGMA or International Standards Organization (ISO) 9000, which have brought substantial profit to the companies that utilize them to improve the quality of software products. This is mainly the motivation for the development of software process improvement models.

Organizations that make use of the standards advocated in CMM, PSP, CMM-I, ISO usually show excellent improvements. Besides, Ferguson *et al.* reported that there is a schedule estimation improvement and strong quality improvements in the developed software when software engineering groups from three different companies, namely Advanced Information Services, Motorola and Union Switch and Signal using Personal Software Process (PSP) as their software process improvement model^[7]. CMM-I helps organization such as IBM Global Australia Application Management Services to help in reducing a cost effectively. The result shows that on-budget delivery improved from over 90 percent to nearly 100 percent as the organization moved from SW-CMM maturity level 3 to CMMI maturity Level 5^[17]. Kumar reported that Six Sigma is a compelling method for breakthrough improvements for delivering world-class processes with a defect rate of less than 3.4 parts per million^[11].

ANALYSIS OF THE RESISTANCE FACTORS IN SOFTWARE PROCESS IMPROVEMENTS PROJECT

Based on software process improvement (SPI) literature, there are various factors which are influencing the implementation of SPI project.

This study has categorized all those factors according to Beecham's research^[2] as presented by Brietzke and Rabelo in their research^[3]. All those factors have been broadly divided into 2 main categories which are 1) Organizational factors and 2) Project factors. The organizational factors related to problems within the scope of the organization and are usually under senior manager's responsibility as reported in^[2,3], meanwhile project factors related to problems regarding the software project management such as planning, activities, resources among others^[5,11].

Organizational factors: Software processes play an important role in coordinating different teams in large or small. These are the factors which are related within the scope of the organization and are usually under senior managers' responsibility as presented by

Table 1: Description of each organizational factor

	Description	Key Resistance Factors
Human	According to Abrahamsson without commitment from all organizational levels (human) to support SPI, the initiative will most likely fail or the results are not far reaching ^[1] . The experience of senior management with an SPI project will give positive impacts to the improvement process. Consultation support such as advice and training of SPI action teams and staffs is one critical aspect in ensuring the success of SPI project. Beecham <i>et al.</i> in stated that organizational issues (especially the human element) are important contributing factors to the success of SPI initiatives ^[2] .	Lack of commitment in all levels of the organizations Lack of adherence and participation of all the individual involved in SPI projects Lack of professionals experience and skill Lack of leadership and backup by top management level Lack of adequate training
Political	According to Wheeler and Duggins the political factor is important to the construction for a department to SQA ^[22] . The establishment of quality policy which is one of the political issues comes after the commitment of senior management. Quality policies and standards for SPI efforts describe the organizational goals and objectives related to the quality.	Lack of the establishment of organizational policies Lack of the establishment of Quality Policy
Cultural	Taylor and McGraw proved that in order to ensure success in a cultural change program, a champion who can build, deploy, drive, and own each initiative going forward must be properly decided ^[19] . However, every cultural change program requires good cooperation from both management and tactical technical staff, improvement programs will fail if either group is left out or underemphasized.	Lack of expertise in implementing cultural changes.
Goals	According to Weigers's research, if the goals, deadlines, and expected results by the managers are impractical, the effort towards SPI may be unsuccessful ^[21] . Clarke and Osterweil found a reason why software process needs to have indefinitely ongoing processes ^[17] . It is essential that clear goals need to be specified earlier, so that progress towards those goals can be continually monitored, and so that revisions to either goals, of processes, or both can be made persistently.	Lack of consistency between software processes improvement project and the organization's strategic objectives Absence of focus on the organization's most urgent needs. Unrealistic expectation towards the SPI project.
Change Management	Initial analysis needs to be conducted to determine whether the SPI initiative apt with the organization's objectives and interests. This is also discussed and supported by Statz <i>et al.</i> ^[18] . Weigers in conceived that the SPI project's team be used to actively facilitate the efforts toward changes on the part of the project teams rather than simply check the situation of the ongoing process in order to report a long and depressing list of findings ^[21] . Miler and Górski highlighted that in order to have a successful software process improvements, risk from configuration and change management which is not explicitly defined will lead to unsuccessful business process ^[14] .	Insufficient and ineffective assessment of the current software process

Brietzke and Rabelo^[3] and others as stated in^[1,14,17,18,19,21,22]. There are 5 factors which are categorized under organizational factors namely 1) human 2) political 3) cultural 4) goals and 5) change management Table 1 provides a description states the key resistance factor for each one identified.

Project factor: These are the factor which reflects the resistance factors on ongoing project which contributes and gives impact during the software process and it involves contribution from all level of personnel management as described by Weigers^[21] and others^[2,18,20]. There are 4 factors which are categorized under project factors namely 1) budget and estimates 2) documentation 3) quality and 4) tools and technologies. Table 2 provides a description states the key resistance factor for each one identified.

MATERIALS AND METHODS

The main objective of this research is to replicate the survey performed by Brietzke and Rabelo in a different country^[3]. Simultaneously, the focus will be

on verifying the relevance of the identified resistance factors as well as comparing the similarities and difference between the studies. Essentially, we perform the initial literature review on SPI, looking at the broader context of SPI, key success factors and the difficulties highlighted in various studies. In order to acquire the overall picture of software process improvement standards, some subjects related to the software process aspects, software qualities and software process improvement standards itself are reviewed. Mostly, the review is focused comprehensively on the past published experience on the implementation of software process improvement project by the industries as well as the resistance factors during implementing such project. All the information above is collected using on-line search via the internet specifically on the online databases namely ACM, IEEE, technical reports published by Software Engineering Institute (SEI), academic textbooks, magazines, online articles and others.

Secondly, we focus on abstracting key resistance factors from Brietzke and Rabelo and deriving the questionnaires^[3]. The resulting questionnaires were

Table 2: Description of Each Project Factor

	Description	Key Resistance Factors
Budget and estimates	According to Weigers lack of progress in improvement plans is frustrating to those who really want to achieve progress and this put down the importance of time and costs in the process evaluation [21]. Further research support by Statz stated that the senior management often overlook on the manageability of SPI project size issue and sufficient budget size for SPI project issue[18]. The first year is the most difficult period for a SPI program. Resistance to change will be at its peak, the costs are likely to be higher than in the 17 following years and due to the steep learning curve, the first year goals and targets can easily be missed.	Current budget and estimates exceeds planning. Lack of understanding by top management level that the software processes improvement project is a long-term return on investment process. Lack of visibility about the ongoing software processes improvement project activities.
Documentation	In SPI project, documentation is a must in order to provide proof and dissemination throughout the organization in a formal way. Therefore, it is helpful to have an infrastructure for documentation, since it is a mandatory practice throughout the organization. According to Beecham <i>et. al</i> , the documentation is also gaining importance in the list of problems associated to SPI[2]. It includes data measurement, proceedings register, coordination and management of the documentation, data collecting the operational framework forms the relationships and dependencies between what is to be done, by whom, and how to do it.	Excessive documentation and formality. Lack of infrastructure and of a documentation management. Lack flexibility in the use of the documentation in projects of different types and sizes.
Quality	Quality refers to the state of the software as it was released or delivered to customers. In order to achieve higher level of quality assurance is by creating consent about how all the requirements definition processes has to be performed and which information should be provided to ensure the successfulness of the SPI project.	Lack of involvement of top management in the relationship between the project teams and the person or group of quality assurance. Lack of treatment to guarantee process conformity in instances of hiring and/or dismissal of skilled professionals.
Tools and Technology	The problem associated refers to the implementation of new tools and technologies, amount of work and pressures that hinder the use of new tools. According to Umarji, complexity of SPI tools and technologies need to ease with which developers can adapt to changes in work practices caused by SPI[20]. However it has several acceptance issues because it often involves learning new technology, changes in work practices and an additional workload. Also, SPI involves collecting data about projects, resources and deliverables and often practitioners are not keen on sharing this type of data.	Automation of not well-defined processes. Lack of training on the support tools and technologies defined as support. Pressure and absence of planning concerning the adaptation period.

distributed to software companies which have been undergoing software process improvement project. Twenty-nine companies have been identified around of the 9 state in Malaysia. There are 174 professionals taking part in this survey. The result of this survey will be presented in the Result section meanwhile based on the data that have been analyzed the resistant factors will be ranked accordingly as demonstrated in the graph illustrated below.

RESULTS AND DISCUSSION

Survey results are described according to the section divided below: Demography Information: This section presents an analysis of the profile of the respondents and companies taking part in this survey. The demography information has been organized in the first section in the questionnaires which comprises several multiple choice questions. The respondents' profiles captured in this first section are role in the organization, education level and academic area, working duration in software development area, period of time working in

Table 3: Number of respondents according their roles in their organization

Roles	Numbers
Business Person	13
Project Manager / Quality Manager	19
It Consultant	15
System Analyst	26
Software Engineer /Developer	64
System Administrator	6
Designer	3
Others(technical background)	28

software process improvement project and expertise level on the area of software process improvement.

Table 3 demonstrates number of respondents according to their roles in the organization which represent the individual who is taking part in the survey.

In Fig. 1, it shows the years of involvement of respondents in software development area whereby 64 respondents (36%) have involved in software development area between 1-3 years, 43 respondents (25%) have involved between 3-5 years and 34 respondents (20%) have 5 years and above involvement

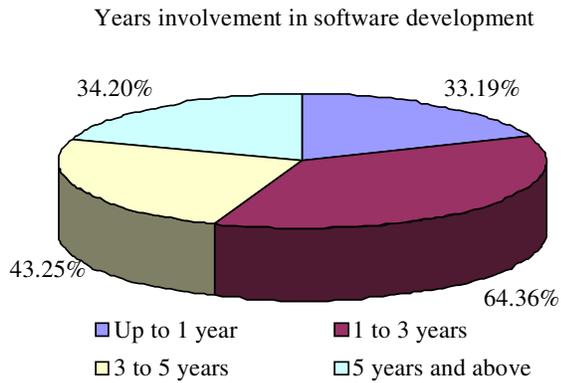


Fig. 1: Number of respondents according to years of involvement in software development area

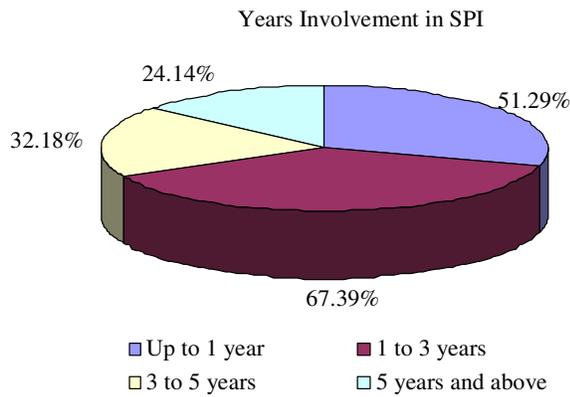


Fig. 2: Period of time working in software process improvement project

in software development area. 33 respondents (19%) have only up to 1 year involvement.

Figure 2 shows from 174 respondents, 51 of them have of up to 1 year experience in Software Process Improvement. Moreover, 67 out of 174 respondents have an experience between 1 to 3 years; meanwhile 32 respondents have an experience between 3-5 years. Also, there is 24 respondents have more than 5 years experience in this SPI area.

Respondents also were asked regarding their level of expertise in the SPI using ordinal scales start from little up to excellent level as illustrated in Fig. 3 below. It shows that, 85 from the respondents have an average of expertise level, 57 rated themselves have high level of expertise and 13 respondents rated themselves as an excellent level. Based on the results, it demonstrates that the respondents have a good and sufficient knowledge of SPI and also can provide reliable input to this survey.

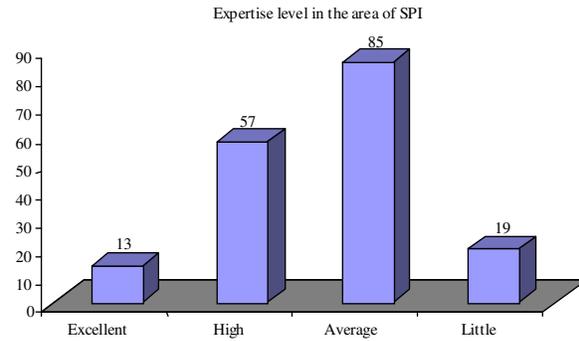


Fig. 3: Expertise level in software process improvement

Table 4: Total influence level in SPI area for the organizational resistance factors

	Influence level					Total of influence level score, T
	1	2	3	4	5	
F01	9	21	43	47	54	638
F02	5	17	42	68	42	647
F03	8	24	57	57	28	481
F04	4	31	40	66	33	615
F05	2	12	75	64	21	612
F06	6	35	59	56	18	567
F07	9	45	56	43	21	544
F08	11	38	63	53	9	533
F09	4	32	61	56	21	580
F10	10	24	61	56	23	580
F11	3	26	67	55	29	621
F12	3	26	55	75	15	595
F13	4	31	80	45	18	576
F14	6	22	62	66	18	590

Resistance Factors: Section II in the questionnaires cover the resistance factors that may influence and contribute to the delay or failure for the implementation of Software Process Improvement. The questionnaire uses the scale of 1-5, ranging from the least influential to the highest influential factor. Ordinal scale is used whereby the user need to choose to rate the influence level of resistance factor from 1 until 5.

Total Influence Level for the Organizational Resistance Factors: Table 4 summarizes the total of influence level score for each organization resistance factor according to the formulae below:

$$T(f_n) = \sum R(f_n) \cdot W(f_n) \quad (1)$$

$T(f_n)$ = The total of influence level score attributed to factor (f). It is a sum of the score rated by the respondent multiplied by weightage score according to influence level

$R(f_n)$ = The score attributed to factor (f) as rated by the respondent according to the influence level

Table 5: Value of influence level and weightage score

Influence Level, R	Weightage Score, W
1	1
2	2
3	3
4	4
5	5

$W(f_n)$ = The weightage score attributed to factor (f) according to influence level as assigned in Table 5

f_n = Refers to the factor number

Table 5 indicates the value of influence level (R) and weightage score (W) for the formulae above.

Legend:

- F01 Lack of commitment in all levels of the organizations
- F02 Lack of adhesion and participation of all the individual involved in SPI projects
- F03 Lack of professionals experience and skill
- F04 Lack of leadership and backup by top management level
- F05 Lack of adequate training
- F06 Lack of the establishment of organizational policies.
- F07 Lack of the establishment of Quality Policy
- F08 Lack of expertise in implementing cultural changes.
- F09 Lack of consistency between software processes improvement project and the organization's strategic objectives
- F10 Absence of focus on the organization's most urgent needs.
- F11 Unrealistic expectation towards the SPI project.
- F12 Insufficient and ineffective assessment of the current software process
- F13 Existence of a software processes improvement project team not focused on orientation and technical support.
- F14 Simultaneous focus on many improvement areas

All the results gained in Table 6 are then plotted in a graph format as illustrated in Fig. 4. It can be observed that the most top 3 organizational resistance factors are factor number 2 which is lack of adhesion and participation of all the individual involved in SPI project, followed by factor number 1 which is lack of commitment in all levels of the organizations and the third one is factor number 11 which is unrealistic expectation towards the SPI project. All these top 3 resistance factors are categorized under people factor as referred in Table 2. The three lowest of organizational

Table 6: Total influence level score according to project resistance factors

Influence	Influence Level					Total of Level Score, T
	1	2	3	4	5	
F15	7	20	49	57	41	627
F16	6	30	61	54	23	580
F17	3	24	55	70	22	496
F18	6	20	61	54	33	610
F19	9	36	52	54	23	568
F20	5	28	82	52	7	550
F21	8	23	65	56	22	583
F22	7	36	55	63	13	561
F23	8	27	69	51	19	568
F24	4	23	62	64	21	597
F25	0	22	70	59	23	605

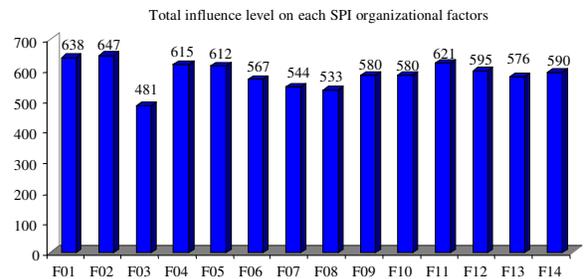


Fig. 4: Total influence level score according to organizational resistance factors

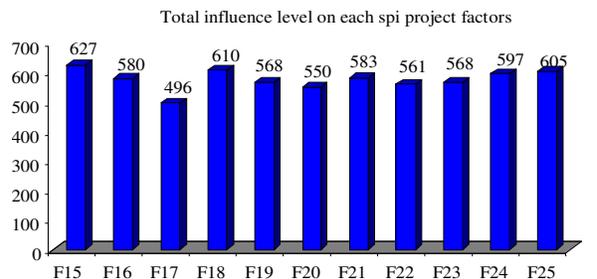


Fig. 5: Total influence level score according to project resistance factors

resistance factors are and lack of professionals experience and skill, lack of expertise in implementing cultural changes and Lack of the establishment of Quality Policy .

Total Influence Level for the Project Resistance Factors: Table 6 summarizes the total of influence level score for each project resistance factor according to the formula given .Then, the results gained are plotted as illustrated in Fig. 5 below.

Legend:

- F15 Current budget and estimates exceeds planning

- F16 Lack of understanding by top management level that the software processes improvement project is a long-term return on investment process.
- F17 Lack of visibility about the ongoing software processes improvement project activities.
- F18 Excessive documentation and formality.
- F19 Lack of infrastructure and of a documentation management.
- F20 Lack flexibility in the use of the documentation in projects of different types and sizes.
- F21 Lack of involvement of top management in the relationship between the project teams and the person or group of quality assurance.
- F22 Lack of treatment to guarantee process conformity in instances of hiring and/or dismissal of skilled professionals.
- F23 Automation of not well-defined processes
- F24 Lack of training on the support tools and technologies defined as support.
- F25 Pressure and absence of planning concerning the adaptation period.

As illustrated in Fig. 5, the total of influence level score for each of the project resistance factor is very close to each other with standard deviation of 29.7, not much different with organizational factors. It can be observed that the most top 3 project resistance factors are factor number 15, Current budget and estimates exceeds planning. Then followed by factor number 18 which is lack of training on the support tools and technologies defined as support and then the third most is 25, pressure and absence of planning concerning the adaptation period.. Meanwhile, the three lowest of project resistance factors is lack of visibility about the ongoing software processes improvement project activities, followed by lack flexibility in the use of the documentation in projects of different types and sizes. The third least is lack of treatment to guarantee process conformity in instances of hiring and/or dismissal of skilled professionals which are identifies as factor number 17, 20 and 22 respectively.

Based on the survey findings reported previously, total influence level score for both organizational and project resistance factors are being merged to gain overall results in order to determine the most and the least influence resistance factors.

According to the results survey, the most critical resistance factor is lack of adhesion and participation of the entire individual involved in SPI projects. This result is similar with the result gained by Brietzke and Rabelo^[3] and corroborates the research findings experience in SPI projects. The second most critical resistance factor is lack of commitment in all levels of the organizations. This factor is directly influenced by

the size or hierarchy of the company, the larger size or hierarchy of a company, the more time needed to get a commitment from all levels of the organization. The third most critical resistance factor is unrealistic expectation towards the SPI project. It is essential that clear expectations and goals need to be specified earlier, so that progress towards those goals can be continually monitored and so that revisions to either goals, of processes, or both can be made persistently.

All these top 3 resistance factors are classified under organizational factor as described in details under analysis and Table 1 which are related within the scope of the organization and usually fall under senior managers' responsibility. The results gained in this research is in accordance with the survey findings conducted by Brietzke and Rabelo^[3] whereby both human factors which are lack of adhesion and participation of all the individual involved in SPI projects and lack of commitment in all levels of the organizations are perceived to be the main determinants in the success of SPI projects. Moreover, all the three factors which have been perceived to be critical are obviously considered to be the most difficult elements which can be taken out of organizational staff. Participation, commitment and reasonable expectations are the end result which should be manifested by the organizational staff, if they are willing to contribute to the SPI project. This finding strongly indicates that organizations implementing SPI projects should spend more effort to create awareness and gain full participation and commitment from their staff to ensure successful implementation of SPI project.

Meanwhile the 3 least resistance factors identified in this survey are lack of visibility about the ongoing software processes improvement project activities, lack of professionals experience and skill and lack of expertise in implementing cultural changes. These 3 factors might have been considered less critical due to the background of most of the respondents. 67% of them have at least 1 to 3 years experience in SPI projects. Based on these, they might perceived that having experience, expertise and vision are not the most critical components which may contribute to the successful implementation of SPI. However, one of the top most resistance factors identified in survey performed by Brietzke and Rabelo^[3] is lack of expertise in implementing cultural changes. The difference in the perceived importance of this factor might be due to the background of the respondents. More than 80% of the respondents of this survey are from non-managerial background. These may indicate that they are not really considering the managerial views of getting people to understand and absorb the SPI practices as a cultural

change. Rather, the views are focusing more on understanding how the typical staff in the organization should react and cooperate to ensure successful implementation of SPI projects.

Furthermore, several respondents mentioned that the SPI project implementation result is also defectively affected if SPI schedule mix up with the ongoing software development project in their companies. The respondents are suggesting that proper and synchronized planning should be done to ensure that the SPI implementation schedule can be carried out harmoniously with the ongoing software development project.

CONCLUSION

This study has identified and analyzed crucial resistance factors which influence the implementation of the software process improvement project specifically software companies operated in Malaysia. The top three and the bottom three least resistance factors have been listed. It concludes that organizational factors specifically human factors playing an important role in determining the success of the SPI project. Participation and commitments from all individuals across the organization are vital and imperative to ensure success for SPI initiative.

One of the study limitations is in only using a set of questionnaire to gather all data required. Moreover, the data sample only covers 29 companies in 6 states of Malaysia. In this case, the degree of validity of the data maybe limited and may not be applicable and reflect for all companies operated in Malaysia. However, it is believed that the characteristics of the IT companies in Malaysia are quite alike. Our future work intends to increase the number of participating companies and uses additional data gathering techniques with the objective of getting wider and more accurate picture of the implementation of SPI.

Based on these findings, we hope to facilitate other software companies to consciously manage future projects through the use of protective actions or proper planning which can reduce the anticipating problems during SPI projects implementation.

ACKNOWLEDGEMENT

First and foremost we would like to express our gratitude to almighty that gave us the possibility to complete the research work successfully. Secondly, we would like to forward our deepest thank to my colleagues, lecturers and technical staffs from the

Department of Software Engineering for their endless assistance, technical advice and co-operation.

REFERENCES

1. Abrahamsson, P., 2001. Commitment development in software process improvement: Critical misconceptions. *Software Eng.*, pp: 71-80.
2. Beecham, S., H. Tracy and R. Austen, 2003. Software process improvement problems in twelve software companies: An empirical analysis. *Empirical Software Eng.*, 8: 7-42.
3. Brietzke, J. and A. Rabelo, 2006. Resistance factors in software process improvement. *CLEI Elect. J.*, 9: 4.
4. Mellon, C., 2002. Software Engineering Institute, 2002. Capability Maturity Model Integration (CMMI), Version 1.1.
5. Clarke, L.A. and L.J. Osterweil, 2000. Continuous self-evaluation for the self-improvement of software. *International Workshop on Self-Adaptive Software*, pp: 27-39.
6. Deming, W.E., 1986. *Out of the Crisis*. MA: MIT Press, Cambridge.
7. Ferguson, P., W.S Humphrey, Khajenoori, S., Macke and S.A. Matvya, 1997. Introducing the personal software process: Three industry cases. *IEEE Comput.*, 30: 24-31.
8. ISO/SPICE, 2008. Available at <<http://www.isospice.com/categories/SPICE-Project/>>.
9. Juran, J.M. and F.M., Gryna, 1988. *Juran's Quality Control Handbook*. 4th Edn., McGraw-Hill Book Company, New York.
10. Kitchenham, B.A, 1996. Evaluating software engineering methods and tool-part 2: Selecting an appropriate evaluation method-technical criteria. *ACM SIGSOFT Software Eng. Notes*, 21: 11-15.
11. Kumar, G.P., 2005. *Software Process Improvement -TRIZ and Six Sigma, Using Contradiction Matrix and 40 Principles*.
12. Lonchamp, J., 1993. A structured conceptual and terminological framework for software process engineering. *2nd International Conference on the Software Process*, IEEE-CS Press, Los Alamitos, CA, USA., pp: 41-53.
13. Mccarthy, B. and R. Stauffer, 2001, Enhancing six sigma through simulation with igrafx process for six sigma. *Winter Simulation Conference*, pp: 1241-1247.
14. Miler, J. and J. Górski, 2004. Risk-driven software process improvement-a Case Study, Euro SPI.

15. Pfleeger, S.L. 1998. *Software Engineering: Theory and Practice*. Upper Saddle River, NJ: Prentice Hall.
16. Russell, C. and H. Cheng, 1995, *Bootstrap Methods in Computer Simulation Experiments*, Winter Simulation Conference, pp: 171-177.
17. Robyn N. and C. Colin, 2005. Carnegie Mellon, Software Engineering Institute, *Software Process Improvement Journey: IBM Australia Application Management Services*
18. Statz, J., D. Oxley and P. O'Toole, 2007. *Identifying and managing risks for software process improvement*.
19. Taylor, D. and G. McGraw, 2005. *Adopting a software security improvement program*. 3: 88-91.
20. Umarji, M. and C. Seaman, 2005. *Predicting acceptance of software process improvement*.
21. Weigers, Karl E., 1996, *Software Process Improvement: Ten Traps to Avoid*, Software Development.
22. Wheeler, S. and S. Duggins, 1998. *Improving software quality*. Southeast Regional Conference, pp: 300-303.