

A Framework of Tools for Managing Software Architecture Knowledge

Rusli Abdullah, Zainab Mohamed Shah & Amir Mohamed Talib

Faculty of Computer Science and Information Technology

University Putra Malaysia, 43400, UPM Serdang, Selangor, Malaysia

E-mails: rusli@fsktm.upm.edu.my & ganawa53@yahoo.com

Abstract

Software architecture (SA) process consists of several activities, which involve complex knowledge intensive process. The knowledge produced and consumed during this process needs to be shared and reused among different stakeholders, and across different life-cycle phases. Therefore, software architecture knowledge needs to be managed for improving organization architecture capabilities. It is the way knowledge management (KM) plays an important role in the SA process. This paper utilized SA evaluation to analyze SA and used Architecture Tradeoff Analysis Method (ATAM) to support a disciplined architecture process. With this approach, it gives support to provide or manage the knowledge required or generated during the SA process. The effective tool support is needed and become important to capture and manage architectural knowledge (AK) consumed or generated during SA process. If not captured and managed, this critical knowledge is implicitly embedded in the architecture, become tacit knowledge which erodes as personnel on the project change. To cover these issues, this paper developed a framework of tools for managing SA knowledge. The tool prototype designing and implementing a web-based knowledge management system (KMS), which is offer a hybrid architectural KM approach.

Keywords: Architectural knowledge, Architecture tradeoff analysis method, Knowledge management, Knowledge management system, Software architecture and software development life cycle

1. Introduction

Software Architecture (SA) is an essential part of software development life-cycle (SDLC). It is a discipline that focuses on the design and specification of overall structure of a software-intensive system. It bridges the gap between the requirements and the detailed design phases of the SDLC (Bass, 2003) and is the first step for designing a system which needs to fulfill a collection of desired quality attributes (Shaw, 1996). Therefore, SA is defined as the high-level structure of a software system, captured in terms of components and connectors.

Besides that, knowledge produced and consumed in SA are architectural design decisions. The collection of architectural design decisions and the resulting design together constitute AK. AK can mainly be classified in two categories, namely technical (such as patterns, tactics, and quality attribute analysis models) and contextual, also called Design Rationale (DR), (such as design options considered, tradeoffs made, assumptions, and design reasoning) (Ali-Babar, 2006). AK need techniques and tools support for effectively manage the knowledge. If not appropriately managed, knowledge underpinning the key architectural decisions is lost, and hence is unavailable to support subsequent decisions to maintain and evolve the SA of a system (Bosch, 2004; Tyree, 2005).

Several methods has been developed to support a disciplined architecture process, such as a general model of SA design (Hofmeister, 2007), the Architecture Tradeoff Analysis Method (ATAM) (Clements, 2002), and architecture-based development (Bass, 1999). These approaches support to provide or manage the knowledge required or generated during the SA process.

In SA process it involves a complex knowledge-intensive task such as designing, documenting, evaluating and maintaining it systems (Robillard, 1999) and also its involving many different stakeholders. As the size and complexity of systems increases, more people get involved, and architecting means collaboration. There often is not a single all-knowing architect. Instead, this role is fulfilled by more than one person. To be able to take well-founded decisions, all stakeholders need to have the relevant AK at the right place, at the right time. Sharing AK is crucial, in particular for reusing best practices, obtaining a more transparent decision making process, providing traceability between design artifacts, and recalling past decisions and their rationale.

Besides that, generally, the documenting the set of design decisions is not done. Much of the rationale behind the solutions is usually lost forever, or resides only in the head of the few people associated with them, if they are still around. Therefore, the reasoning behind a solution is not captured. At a later stage, it then becomes difficult

to trace the reasons of why things are the way they are. Also it is costly if the architectural decisions need to modify. By the ways, KM approaches been introduced in order to solve the problem in capture and manage the architecture knowledge. And also by developing effective KM approach will make everyone can share; collaborate against the knowledge all around them.

The primary focus of our system is to acquire knowledge from SA evaluation, which is evaluating the architecture of our system early in the development process. It is needed because the architecture represents an enormous risk in a development project. Making bad architectural decisions may substantially slow down our system and even lead to failure. Additionally, if some problems are identified, it is easier and cheaper to change the architecture in early stages of the development process.

In this paper, we introduce the extensions of the requirement of a framework of tools for managing SA knowledge. Section 2 presents a discussion of related works. Section 3 provides an overview of our research methodology. Section 4 highlighting the system architecture design and development. Finding and results discusses in section 5. Lastly, section 6 presents some concluding remarks.

2. Related Works

Knowledge is considered to be a crucial resource of an organization, and then it should be carefully managed. Historically, organization knowledge has been stored on paper or in people's mind. Unfortunately, paper has limited accessibility and it is difficult to update (O'Leary, 1998). Knowledge in people's mind is lost when individuals leave the company. Furthermore, in a large organization, it can be difficult to localize who knows some matter. So, knowledge has to be systematically collected, stored in a corporate memory, and shared across the organization. KM entails formally managing knowledge resources in order to facilitate creation, access and reuse of knowledge, typically using advanced technology. So that, KMS are tools to effect the management of knowledge and are manifested in a variety of implementations (Davenport et al. 1998) includes document repositories, expertise databases, discussion lists, and context-specific retrieval systems incorporating collaborative filtering technologies.

The architecture process aims to solve a mix of ill- and well-defined problems, which involve processing a significant amount of knowledge. Architects require topic knowledge (learned from textbooks and courses) and episodic knowledge (experience with the knowledge) (Robillard, 1999).

One of the main problems in the SA process is the lack of access to the knowledge underpinning the design decisions and the processes leading to those decisions (Ali-Babar, 2006; Bosch, 2004). This type of knowledge involves information like the impact of certain middleware choices on communication mechanisms between different tiers, why an API is used instead of a wrapper, and who to contact to discuss the performance of different architectural choices.

The SA community has developed several methods (such as ATAM (Clements, 2002) and PASA (Williams, 2002) to support a disciplined approach to architectural practices. Some of these do emphasize the need for KM to improve reusability and grow organizational capabilities in the architecture domain. Except for (Clements et al, 2002), there is no approach that explicitly states what type of knowledge needs to be managed and how, when, where, or by whom. Also, none of the current approaches provides any conceptual framework to design, develop, and maintain a suitable repository of architecture knowledge. Because of lack of suitable techniques, tools, and guidance is why architecture design knowledge is not captured.

(Abowd et al, 1997) proposed two broad categories of SA evaluation approaches: questioning and measurement. The former category includes techniques like scenarios, questionnaires, and checklists. The latter category consists of metrics and simulation. Scenario-based methods (such as ATAM, SAAM, and CBAM) are considered the most mature and well-known. Though there are differences among these methods, the five common activities by comparing four main SA evaluation methods. These five activities make up a generic process of SA evaluation process (Ali-Babar, 2004).

In architecture evaluation, we use scenarios to represent a concrete quality attribute. (Bass, 2003) provides a model to describe a scenario. As in his model six elements are adopted to normalize various scenarios into a standard form.

CBAM introduces cost-benefit into the consideration of decision making during architecture design. This method is bridging two domains in software development the architecting process and the economics of the organization.

During architecture evaluation, sponsors or organizations have adequate reasons to expand their economic gains and avoid risks as much as they can. CBAM is adding the costs (and implicit budgets or money) as quality

attributes, which need to be considered among the tradeoffs when a software system is going to be planned. SAAM and ATAM primarily considered the design decisions with respect to architectural quality attributes like modifiability, performance, availability, usability, and so on. CBAM is claiming that costs, benefits and risks are as important as the other quality attributes and they are relevant to be considered when the architectural decisions are being made.

Architecture Tradeoff Analysis Method (ATAM) is a scenario-based architecture method for assessing quality attributes such as: modifiability, portability, extensibility, and integrability. Besides the assessment of quality attributes, ATAM explores the quality attributes interaction and their interdependencies highlighting trade-off mechanisms and opportunities between different qualities. ATAM method consists of four phases: presentation, investigation and analysis, testing, and reporting. Each phase is a collection of steps. The presentation phase involves exchanging information through presentations. The investigation and analysis phase concerns the assessment of the key quality attribute requirements versus the architectural approaches. The testing phase compares the results of the previous phase to the needs of the relevant stakeholders. Finally, the reporting phase summarizes the ATAM results.

3. Research Methodology

Selecting the right methodology in developing software is the most important and crucial part. This is based on the assumption, subject to endless debate and supported by patient experience, that a methodology approach to software development results in fewer defects and, therefore, ultimately provides shorter delivery times and better value. The challenge in selecting and following a methodology is to do it wisely to provide sufficient process disciplines to deliver the quality required for business success, while avoiding steps that waste time, squander productivity, demoralize developers, and create useless administrator. Proposed steps are listed in the flowchart in Figure 1 (Note 1).

The initial stage of this project methodology is Literature Analysis whereby related journal and article that related to KM and SA knowledge has been studied, reviewed and analyzed. Besides that, the analysis helps the author to get the ideas, concepts and methods of SA Portal on how to works on it and the last author was identified the prime requirement in developing KM in SA. The analysis provides the author an idea on how to prepare the plan for the system to be developed. Furthermore, it is understood that proper and well defined plan can help the author to produce a quality system.

After that, author designed questionnaire (Note 4) in order to collect requirements from users (various backgrounds) in term of GUI and performance of the SA Portal and proceed to design framework SA portal based on questionnaire analysis. Then survey has been done during this phase to gather information and user requirement on the system to be developed.

Here, we began to develop a Portal prototype. By having the prototype, let the same group of users having the post-test and let them to test whether they were satisfied with the system or not. Lastly, conclusion will be made based on the result in term of quality perspective – data accuracy, reliability, efficiency and usability.

4. System Architecture, Design and Development

4.1 System Architecture

Architectural design represents the structure of the program components that are required to build a computer-based system. It provides the overall picture of a system. (Pressman, 2001) stated architecture as, "At the most simplistic level, it is analogous to the architecture of a building. It is the manner in which various components of the building are integrated to form a cohesive whole". From the problem statement and the literature review (LR), the author had applied the system based on web based architecture as describe in Figure 2 (Note 1).

From the LR, it was said that KMS should have at least three tiered architecture (Chua, 2004). The services provided should supported by KM technologies. They are infrastructure services, knowledge services and presentation services. The architecture is illustrated in Figure 3 (Note 1). By implementing this architecture, it can compelling the need for KM in organization by a host of social, economic and technological factors including the shortening of product life cycle, fluidity of workforce and the prevalence of work arrangements. Furthermore, when used in tandem with an appropriate KM strategy, technology is a powerful enabler of organizational success. Besides that, through the simplicity of a three-tiered architecture, the KMS architecture can help the organization to understand the KM technology.

4.2 System Design

In this stage, the physical design of the system is implemented to accomplish the specification from the system requirement. After collecting and analyzing the system requirement it is necessary to identify the detail of the system to be developed so that necessary task is performed and constructed according to user requirement. More specifically, the system design phase is focused on the interface construction and the database construction.

System design and the arrangement of each server can assist in creating a good system and achieve the objective and target of the software development. The system design research is started by showing the system at high level down to the lowest level. Activities undertaken in this phase were: User Interface design, Apache Web Server, PHP and MySQL Database.

4.3 System Development

This section discusses the development process and the design of the KM in SA Portal. It covers the flow of the system and interface used beside that SD is developed based on the KMS architecture which covers the infrastructure services, knowledge services and presentation services.

4.3.1 System Diagram

This section explained the diagram used in this system. Figure 4 (Note 1) shows the use case of the system. From the use case, it will show the actor involved in the system. In this system, the actors involved are Administrator, User, Project Manager, Developer, Architect and Lead Architect. Besides that, through the use case diagram the roles of each actor can be depicted. Furthermore, it also shows the system function that will be performed by which actor.

4.3.2 Interface Design

4.3.2.1 Welcome Screen

The welcome screen is an application start up as shown in Figure 5 (Note 2). The welcome screen contains information such as the application's title, the login section, announcement/news section, words of wisdom section, express link and also calendaring. The screen will allow the particular user of the system to use the application. There are six category of user that can use the system. They are User, Administrator, Project Manager, Developer, Architect and Lead Architect.

4.3.2.2 Main Menu

The main menu is displayed after the registered user of the application login to the system. It is permanently located at the top of the screen as shown in Figure 6 (Note 2). The registered users just click the menu, when he/she want to access each of the modules. The registered users can create scenario, evaluate scenario contribute an ideas thru Forum menu and also performing online vote after login to the system.

5. Result and Discussion

From the questionnaire survey (Note 4), all the respondent agreed that KM in SA should be applied in an organization since this facility is not provided in their agencies. The respondent also highlight that there is a need for SA Portal during SD process or evolution to capture and manage knowledge which is involve in SA process. This is because if not appropriately managed, knowledge underpinning the key architectural decisions is lost, and hence is unavailable to support subsequent decisions to maintain and evolve the SA of a system. 70% respondent believed that the management of AK is vital for improving an organization's architectural capabilities. Furthermore, 60% respondent understood that AK need to find a systematic way to capture and document the decision already made so that an accurate architectural representation can be made.

Apart from the need of SA Portal to be implemented in the organization, respondent also has been asked whether KMS architecture has been applied in the system. From the survey conducted, 60% respondent agreed and 40% respondent strongly agreed that the system should have collaborative environment. Additionally, in terms of knowledge services characteristics layer which consist of knowledge creation, knowledge sharing and knowledge reuse, 10% respondent agreed and 90% respondent strongly agreed that the system had covered this layer. For presentation services, most of the respondent agreed that when visualizing the system they can understand what the system is trying to do. The percentage that agreed with this layer is 60%. Overall, 70% respondent agreed and 20% strongly agreed that KMS architecture has been applied in the system. Figure 7 (Note 2) illustrated the respondent acceptance on KMS Architecture.

From Figure 7 (Note 2), it can be viewed that the KMS architecture's layers that are repository, application and interface must be included in implementing KMS. In the survey respondent were asked on whether the system

has the infrastructure services that is collaborative environment, knowledge services which comprise of knowledge creation, knowledge sharing and knowledge reuse, and lastly presentation services which cover the visualization characteristic.

Figure 7 (Note 2) is also shown that, the developed system has applied the KMS architecture as proposed by (Chua, 2004). Collaborative environment means that the system should provided with synchronous meeting or asynchronous discussions forum. In SA Portal provides a discussion forum where every user can communicate and discuss among user regardless they are from other project. So, every user can exchange information and idea among them. Additionally, knowledge services cover the knowledge creation which responsible to capture and codify knowledge held by experts. Besides that in knowledge services it covers the knowledge sharing and knowledge reuse. Knowledge sharing means that the user has a platform to share knowledge with on another. Knowledge reuse, user can have an inter-disciplinary perspectives and facilities cross referencing. By including those KMS architecture layer, the system will cover the basic requirement to implement KMS architecture in SA Portal.

For infrastructure service that is for storage, 90% respondent agreed that the system fulfill this layer. This is illustrated more in Figure 8 (Note 2).

Figure 8 (Note 2) shown that the system have the repository layer. Whereby, it has the storage technology. The knowledge repositories can be stored as data warehouse or as knowledge server. This means that the stored data or knowledge can be used for future data analysis when top management wants to make any decision in their future project.

In terms of usability, 80% of the respondents agreed that the system is easy to find, the navigation within the system is easy and the layout is as what they expected. For efficiency, 100% respondent agreed that the retrieval time for information is reasonable. For functionality, 70% respondent agreed that the system content as good, 50% agreed that the system can help them to find solution and 60% satisfied with the information provided by the system. However, 40% respondent responds that the system navigation still needs an improvement.

Apart from that, 80% respondent agreed that the system can be used as a channel to find solution and help them to identify issue for future development. In terms of reliability, 60% respondent agreed that the information stored the system is reliable. While for portability, 80% respondent agreed that the system can be used at different platform. The quality characteristic is illustrated in Figure 9 (Note 2).

Figure 9 (Note 2) explained that in terms of usability, the system is easy to use and navigate. That means the system do not have too many resources to train the user to use the system. For efficiency, the system retrieval time is efficient. For functionality which is the function provided by the system, users said that basically the developed system covers the basic KMS characteristics. For portability, which is the system capability to be adapted to other environment consisting of different hardware and different OS the developed system fulfilled this factor.

Figure 10 (Note 2) illustrated the reliability of the system and functionality in terms of the need to improve the system. In general, 80% agreed that as a new comer at the organization, SA portal can be used as source to get more information and also as a place to share knowledge and experience. Moreover, 90% agreed that the reward system introduce in this system can encourage the architect to give solution to the task assign to them.

Overall, all of the respondents were comfortable and satisfied with the system in handling typical task scenario of KM system in Software evaluation. From the survey 80% respondent can accept the developed system. The frequency chart of the system evaluation results is illustrated in the Figure 11 (Note 2).

6. Conclusion

Knowledge is a crucial resource for organization. It allows companies to fulfill their mission and become more competitive. The management of knowledge and how it can be applied to SA process has received little attention from software engineering research community so far. However, software organization generates a huge amount of knowledge for different activities of SA process that should be captured and processed. In this way, they would obtain more benefits from it.

The main contribution of this research is the application of KMS architecture in SA environment which considers the processes of a knowledge life cycle such as creating, maintaining, sharing and distributing knowledge. Furthermore, the study also in charge of storing and managing information and lessons learned which are consumed and generated during the SA process. The system facilitates the management of explicit and implicit AK generated during the architecting process. Thereby, the costs in time and effort should decrease.

Additionally, in this study, the author has applied the architecture proposed by (Chua, 2004) which has the interface, application and repository layer.

The project implementation has initiated some ideas in setting up the KMS in software evaluation, as a knowledge repository for the architect or lead architect who is involved in a lot of decision making during software architecting for a project in an organization. It is because it involves a lot of decision making. Several dozens of tradeoffs need to be made to satisfy current and future requirements from a set of stakeholders who may have completely vested interests in architecture decisions. Therefore, the management of AK needs to be supported by appropriate tools and technologies to improve business processes and practices by utilizing individual and organizational knowledge resources.

References

- Abowd, G., et al. (1997). Recommended Best Industrial Practice for Software Architecture Evaluation, *Tech Report CMU/SEI-96-TR-025*, Software Engineering Institute, Carnegie Mellon University.
- Ali-Babar, M., Gorton, I., Kitchenham, B. (2006). A framework for supporting architecture knowledge and rationale management. In: A.H. Dutoit, R. McCall, I. Mistrik, B. Paech (eds.) *Rationale Management in Software Engineering*, pp. 237–254
- Ali-Babar, M. and Gorton, I. (2004). Comparison of Scenario- Based Software Architecture Evaluation Methods, *Proceedings of the 1st Asia-Pacific Workshop on Software Architecture and Component Technologies*.
- Ali-Babar, M., Zhu, L., and Jeffery, R. (2004). A Framework for Classifying and Comparing Software Architecture Evaluation Methods, *Proceedings of the 15th Australian Software Engineering Conference*
- Bass, L., Clements, P. and Kazman, R. (2003). *Software Architecture in Practice*, 2nd ed.: Addison Wesley/Pearson.
- Bass, L., Kazman, R. (1999). Architecture-based development. *Tech. Rep. CMU/SEI- 99-TR-007*, Software Engineering Institute (SEI), Carnegie Mellon University, Pittsburgh, USA.
- Bosch, J. (2004). Software architecture: The next step. In: European Workshop on Software Architecture.
- Chua A. (2004). Knowledge management system architecture: a bridge between KM consultants and technologies, *International Journal of Information Management* 24, 87-98.
- Clements, P., Kazman, R., Klein, M. (2002). *Evaluating Software Architectures: Methods and Case Studies*. Addison-Wesley.
- Davenport, T. H., De Long, D. W., and Beers, M. C. (1998). Successful Knowledge Management Projects, *Sloan Management Review*, pp. 43-57.
- Hofmeister, C., Kruchten, P., Nord, R.L., Obbink, H., Ran, A., America, P. (2007). A general model of software architecture design derived from five industrial approaches. *Journal of System and Software* 80(1), 106–126.
- O’Leary D. E. (1998). Enterprise Knowledge Management, *IEEE Computer*, vol. 31, no. 3, pp. 54-61.
- Pressman, R.S. (2004). *Software Engineering – A Practitioner’s Approach*, 5th Ed., McGraw-Hill, pp. 271, 335, 365, 401, 402.
- Robillard, P.N. (1999). The role of knowledge in software development. *Communication of the ACM* 42(1), pp. 87–92.
- Shaw, M and Garlan, D. (1996). *Software Architecture: Perspectives on an Emerging Discipline*, Prentice Hall.
- Tyree, J., Akerman, A. (2005). Architecture decisions: Demystifying architecture. *IEEE Software* 22(2), 19–27.
- Williams L.G., Smith C.U.: PASA. (2002). An architectural approach to fixing software performance problems. In: *Proceedings of the International Conference of the Computer Measurement Group. Reno, USA*.

Notes

Note 1

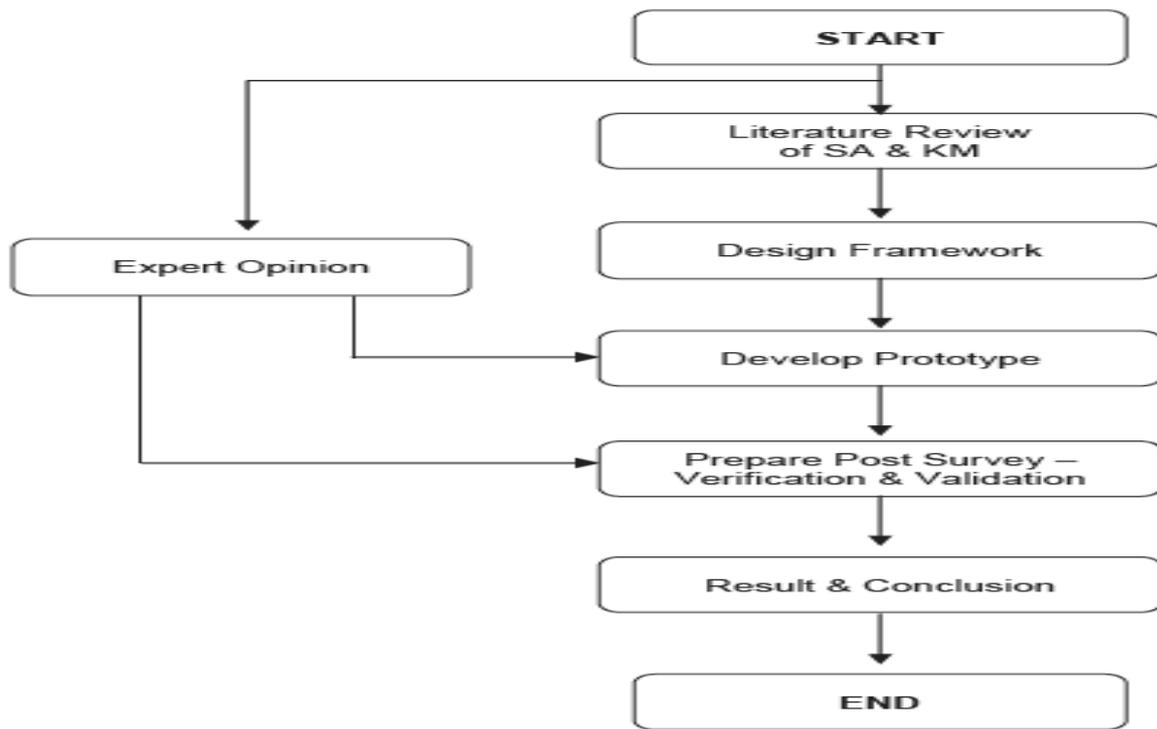


Figure 1. Proposed Methodology

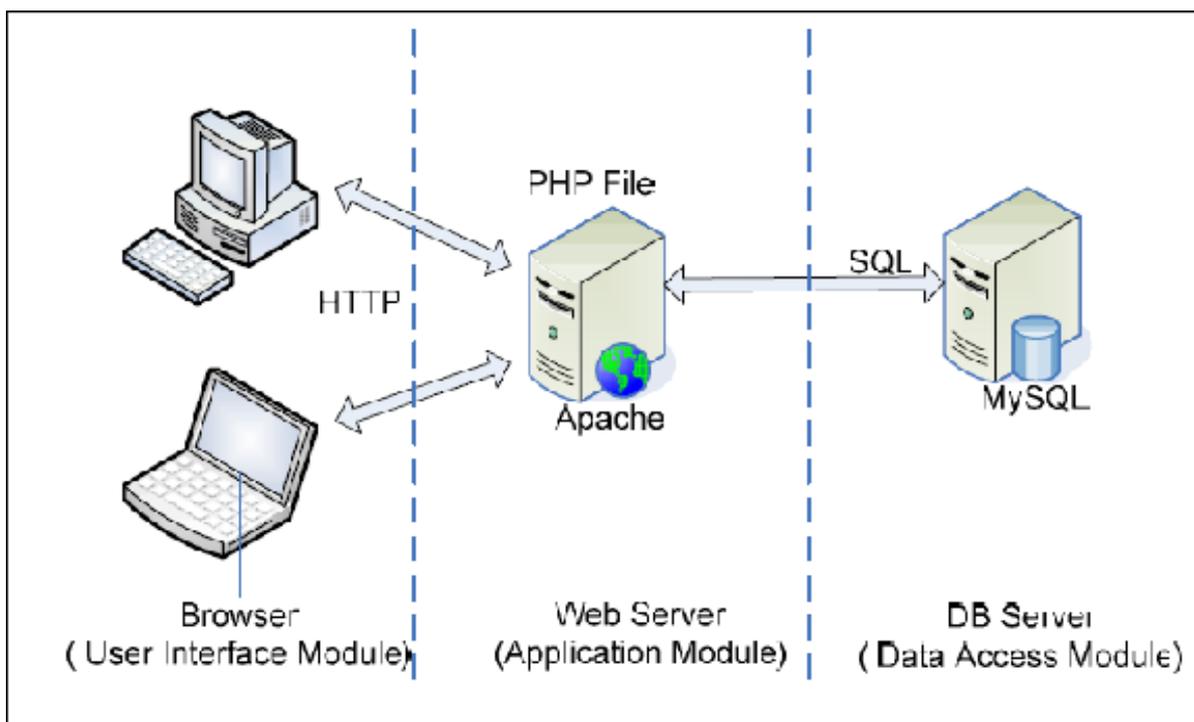


Figure 2. System Architecture

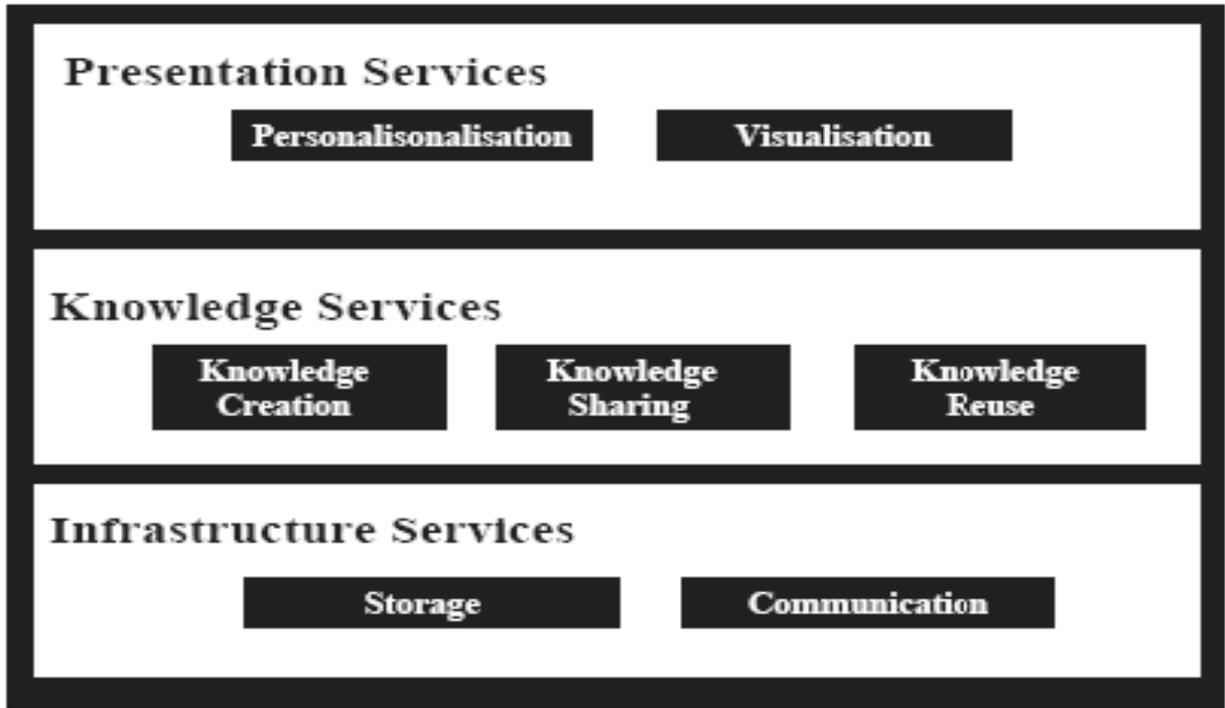


Figure 3. KMS Architecture

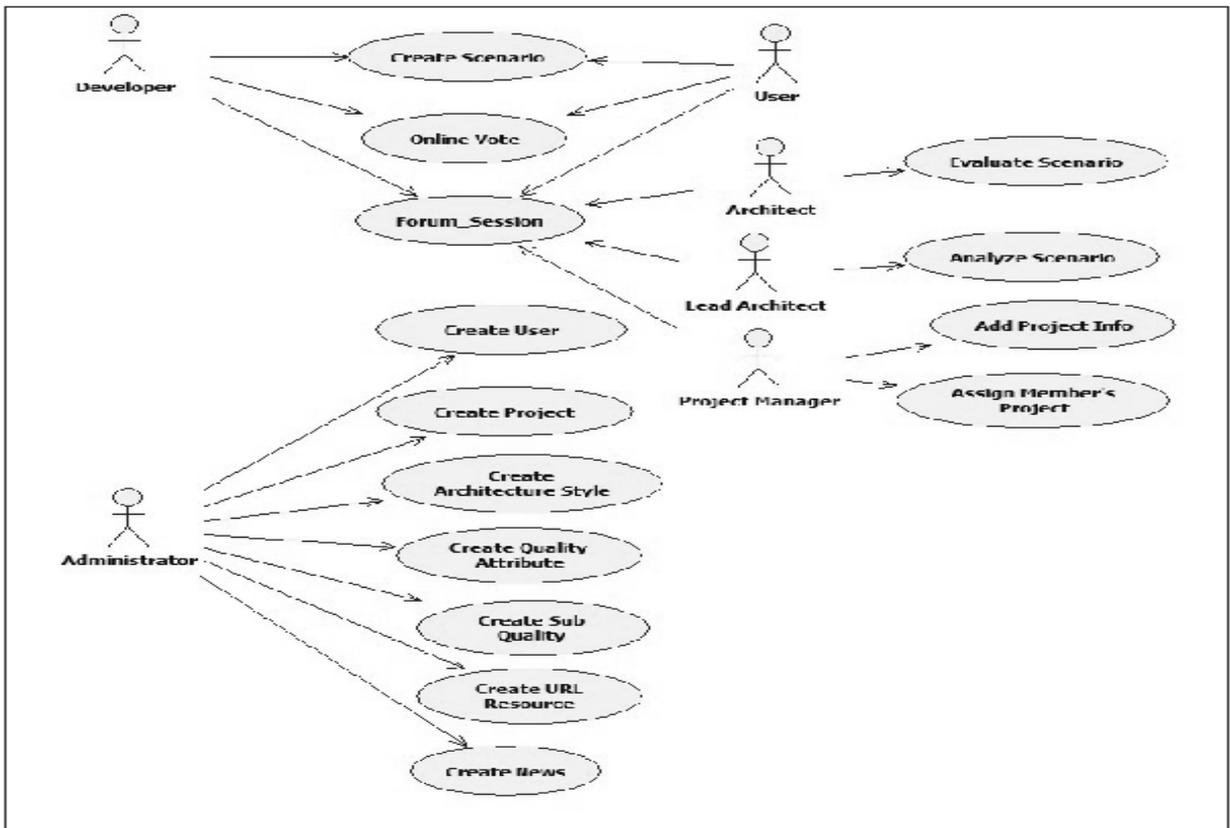


Figure 4. Use Case Diagram

Note 2

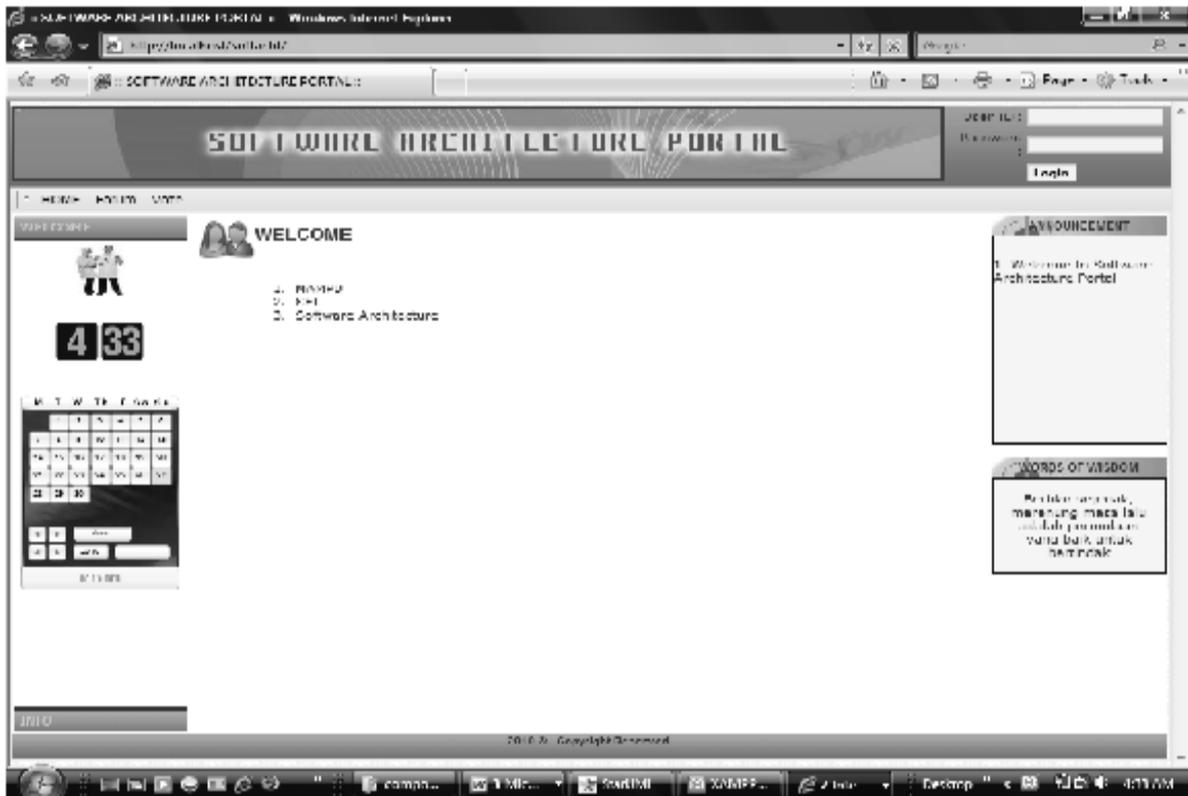


Figure 5. Welcome Screen



Figure 6. Main Menu Screen

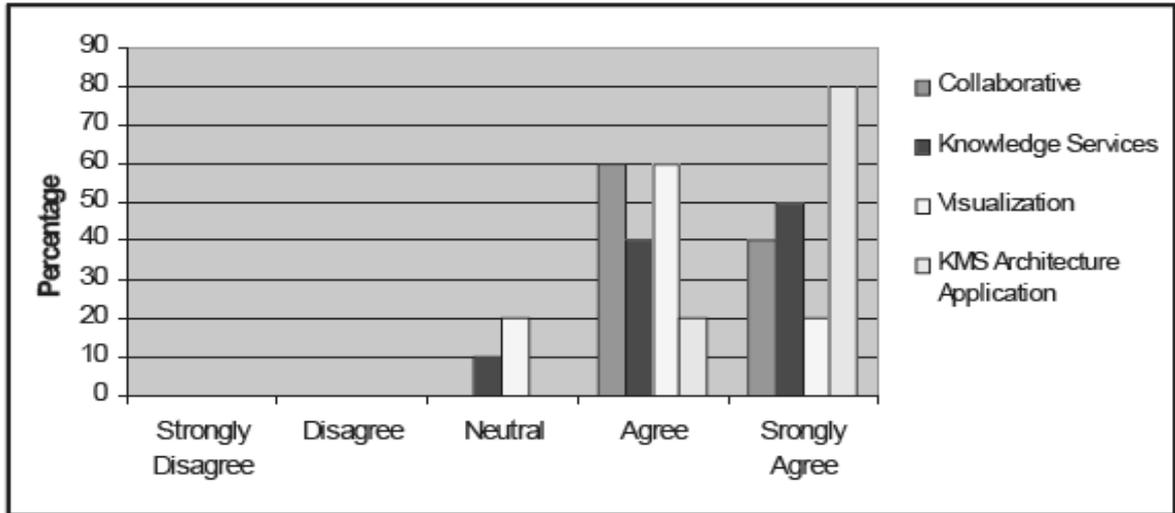


Figure 7. KMS Architecture Acceptance

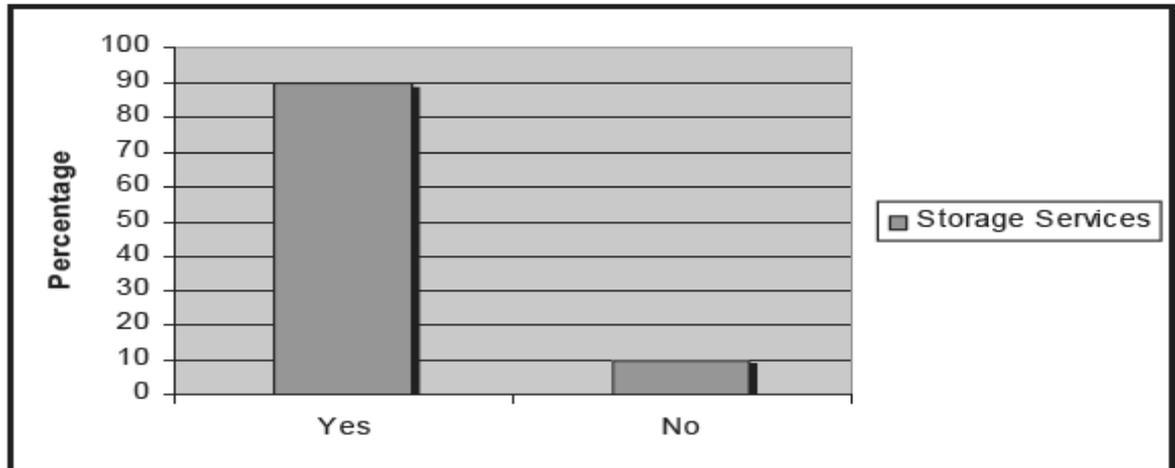


Figure 8. Storage Services

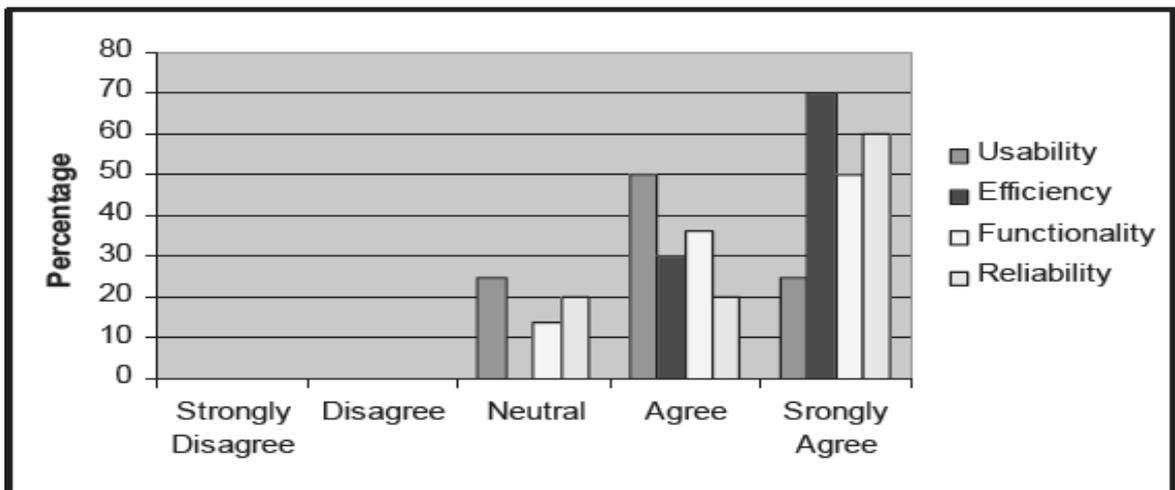


Figure 9. Quality Factors

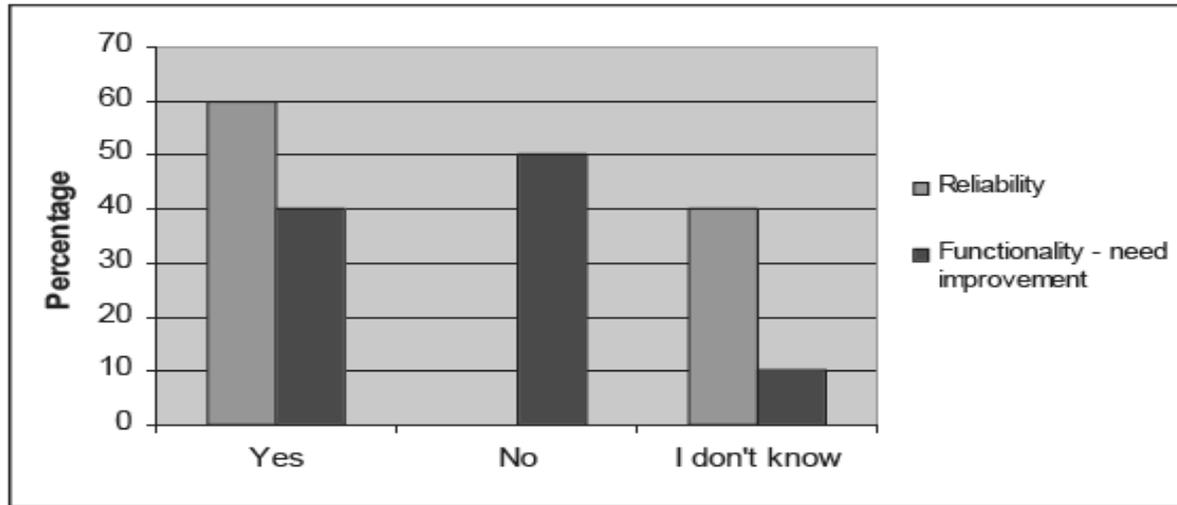


Figure 10. System Reliability and Functionality

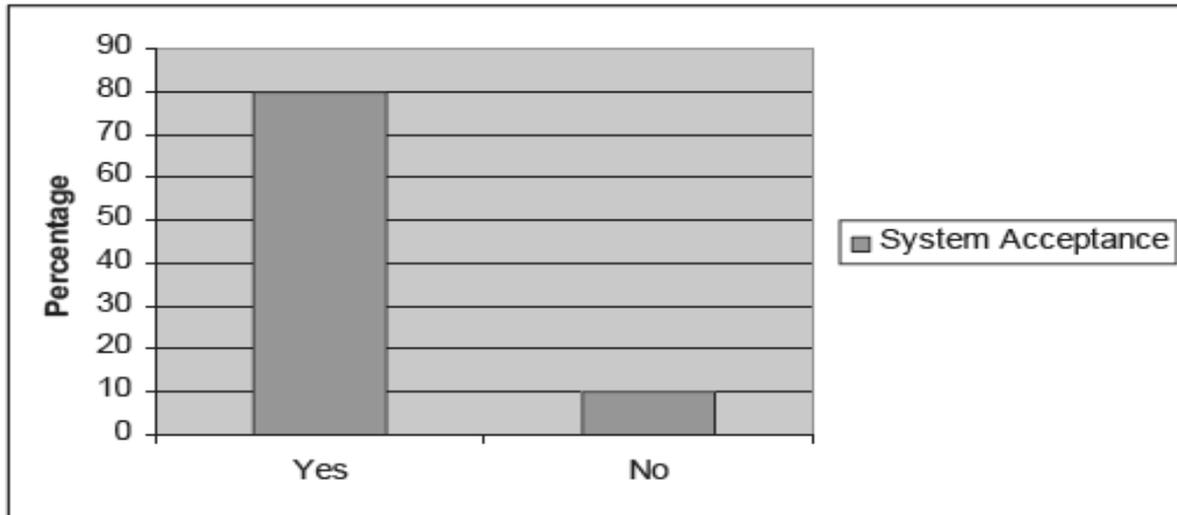


Figure 11. System Acceptance

Note 4

QUESTIONNAIRE CONSENT FORM

This research questionnaire is designed to gather information on:

- (a) To identify requirements of the proposed component of the system framework.
- (b) To verify the components of the proposed framework.
- (c) To get requirements to develop the system prototype based on the proposed framework.

Please answer all questions in the questionnaire. Identify of each respondent will remain anonymous and confidential. The information gathered will be solely used for the research purposes only. It will be destroyed after the research.

Thank you for the time spent to answer this questionnaire. Your cooperation is highly appreciated.

Instruction: Please choose the answer.

Section A: DEMOGRAPHIC

1. Gender:

A. Male B. Female

2. Age:

A. 25 – 30 years B. 31 – 35 years C. 36 – 40 years D. more than 30 years

3. Education Level/Qualification:

G. Certificate H. Diploma I. Degree J. Master K. Doctorate L. Others

4. Role in Software Development Process

A. Head of Department B. Head of Division C. System Analyst D. DBA E. Programmer F. Others

5. Working experience:

A. 5 – 10 years B. 11 – 15 years C. more than 15 years

GENERAL

6. Is software architecture portal exists in your organization?

A. Yes B. No

Note: If Answer 6 is yes you may proceed to Question 7 else proceed to Question 8.

7. Do you satisfy with the portal for your usage?

A. Not satisfy B. Satisfy but need enhancement C. Very satisfy

8. Does your organization needs a software architecture portal for software development process?

A. Yes B. No

9. Do you agree that software architecture portal plays an important role in capturing and managing information to project team members when needed?

A. Yes B. No

Note: The following questions are based on participant's feedback on the software architecture knowledge portal hands on:

USABILITY

10. What do you feel about the organization of the information for the portal?

A. Very Poor B. Poor C. Neutral D. Good E. Very Good

11. Do you find the navigation within the portal is easy?

A. Very Poor B. Poor C. Neutral D. Good E. Very Good

12. How do you rate the layout of the information on the portal?

A. Very Poor B. Poor C. Neutral D. Good E. Very Good

EFFICIENCY

13. What do you think of retrieval time of information?

A. Very Poor B. Poor C. Neutral D. Good E. Very Good

FUNCTIONALITY

14. How do you rate the content of the web pages for each phase?

A. Very Poor B. Poor C. Neutral D. Good E. Very Good

15. Do you think the knowledge in this portal is satisfied with software architecture?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly Agree

16. What is your satisfaction level on the information provided?

A. Very Poor B. Poor C. Neutral D. Good E. Very Good

17. Do you think navigation of the portal needs improvement?

A. Yes B. No C. I don't know

18. If you developer, do you agree this portal will make easy to create scenarios of the project?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly agree

19. If you are architect or lead architect, do you agree this portal will satisfy during do evaluate or analyze the scenarios?

F. Strongly Disagree G. Disagree H. Neutral I. Agree J. Strongly agree

20. If you are project manager, do you agree this portal will help you to capture the information of the project?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly agree

RELIABILITY

21. Do you agree that the document provided in the portal is reliable?

A. Yes B. No C. I don't know

PORTABILITY

22. Knowledge management system in software architecture portal has the same performance regardless of the operating system of your computer (Microsoft Windows, LINUX, Macintosh etc). Do you agree with this statement?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly Agree

SUMMARY

23. If you are one of the software development team members, do you agree this portal is very useful in searching information and sharing experience?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly Agree

24. If you are new in the organization, do you agree that this system is useful in helping you to get more information of the previous project?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly Agree

25. People get feedback on what they have contributed and thus motivated to contribute more. Do you agree that an organization that emphasis on sharing knowledge should provide reward for the staff who be the knowledge contributor of the month?

A – Yes B – No C – I don't know

26. Overall, do you satisfied with the knowledge management system in software evaluation?

A. Yes B. No C. I don't know

27. Do you agree that the system has a storage characteristic?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly Agree

28. Do you agree that the system should have the collaborative environment?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly Agree

29. Do you agree that the system cover the knowledge services characteristics that are Knowledge Creation, Knowledge Sharing and Knowledge Reuse?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly Agree

30. In terms of Presentation Services do you agree that by visualizing the information in the system, you can understand what the system is mean for?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly Agree

31. Overall, do you agree that the system covered the Knowledge Management System architecture which is Infrastructure Services, Knowledge Services and Presentation Services?

A. Strongly Disagree B. Disagree C. Neutral D. Agree E. Strongly Agree

32. What is your suggestion to improve this system?

*****THANK YOU*****