Design Framework of KSPI Model (5iKSPI)

Bharti Sharma Student, M-tech Department of CSE Punjabi University Patiala Neeraj Sharma Lecturer Department of CSE Punjabi University Patiala Neeshu Sharma Student, M-tech Department of CSE PTU, Jallandhar

1. ABSTRACT

Several contemporary models have been developed to assist organizations to evaluate and to improve their software development processes and capabilities. There is substantial evidence that a successful software process improvement (SPI) program will result in improvements in productivity, quality, schedule, and business value. In this paper work, we have analysis several SPI models like: BOOTSTRAP, CMM, ISO, PSP, SPICE, TSP, and Trillium, and Knowledge based models Pillars Of Knowledge Management Model, Core like: Capabilities And Knowledge Building Activities Model, Organizational knowledge Management Model, Model Of Knowledge Organization, Model OF Knowledge Management Stages, Intellectual Capital Model, Spiral Of Organizational Knowledge Creation, KPMG knowledge Management Process Model. Here we design a new SPI model(5iKSPI) that based on Knowledge management techniques.

2. LITERATURE SURVEY

According to Suliman Al-Hawamdeh [10] the majority of those interested in knowledge management, the key drivers are organizational efficiency, maximizing organization's potential, competitive advantage, building a learning organization and managing intellectual capital. However, implementing knowledge management is also not that easy. Organizations wanting to implement knowledge management have to grabble with issues such as strategy, technology, organizational culture and knowledge organization.

In other research from Espen Frimann et. al [5] we found that employee participation in the improvement process has a positive effect on the success of Software process improvement. Another research from Bridget Meehan et. al. [2],the finding of established that four knowledge management activities : knowledge creation, knowledge storing, knowledge sharing, knowledge leverage. Even with the best graduate education, software developers - especially software engineers - will need to periodically update their skills and their mastery of new technology Eduardas Bareiša et. al [6]. Christensen [1] explores knowledge management in practice through 10 managers from industry and compares his results to the results of the theoretic study. The main conclusion is that KM theory does reflect, in generic terms, the practices that support KM activities, but the challenge is to observe this practical application of generic drivers, which often is difficult to observe in practice.

3. Introduction:

Phases of 5iKSPI (knowledge Software Process improvement)

- Incipient
- Investigate
- Improve
- Implementation
- Interpretation

3.1 Incipient Phase

This is initial phase for starting any project, whole project is defined in this phase, it defines in following sub parts:

3.1.1 Identification Part:

- Meeting with the enterpriser: In a meeting, two or more people come together for the purpose of discussing a (usually) predetermined topic such as business or community event planning, often in a formal setting.
- Project goal & metrics: The desired end result of the project which must be verified. Project analyses identify weaknesses and optimization potential and should also be carried out during the course of the project.
- Understand the need of improvement, if project is already developed
- Establish Sponsorships
- Time & cost estimation

3.1.2 Create:

- Project Design Specification: A written statement on the project's objectives, methods, timetable and resources, project technology, baseline scenario, use of the additionally tool; providing the framework for the execution of the project, set out in sufficient detail to be quantified, costed, implemented and monitored.
- Design of the SRS: A Software Requirements Specification (SRS) is a complete description of the behavior of the system to be developed. Use cases are also known as functional requirements.

- Re-construction or Re-designing of the project, if it already done
- Design the test plans: The plan typically contains a detailed understanding of what the eventual workflow will be. There are three major elements that should be described in the test plan: Test Coverage, Test Methods, and Test Responsibilities.
- Design test cases: A test case in software engineering is a set of conditions or variables under which a tester will
- •
- determine whether an application or software system is working correctly or not.

3.2 Investigate Phase:

In this phase, it basically give the analysis of the current project and define the requirement of the project.

3.2.1 Analysis:

• Acquire Tacit knowledge : Tacit knowledge (as opposed to formal or explicit knowledge) is knowledge that is difficult to be transferred to another person by means of writing down or verbalizing it[3]. In the field of knowledge management,

the concept of tacit knowledge refers to a knowledge which is only known by an individual and that is difficult to communicate to the rest of an organization. The tacit aspects of knowledge are those that cannot be codified, but can only be transmitted via training or gained

through personal experience. Tacit knowledge has been described as "know-how" -- as opposed to "know-what" (facts), "know-why" (science), or "know-who" (networking). It involves learning and skill but not in a way that can be written down.

- Collect Exiting Literature: The word "literature" has different meanings depending on who is using it. In this data is collected from old data repository[13], which is collection of data from previous project results
- **3.2.2 Requirement:** Here we define the need of the project, [11]it include the following sub-parts:
- Resource Requirement
- Tool, technology & people requirement.

3.3 Improvement Phase:

This phase deals with the those activities which are needed to improve the quality of the software.

3.3.1 Activities:

- Hire experts & consultancy: Selected participants with scientific and technical expertise are invited to provide input and feedback on the development of government policies, guidelines, research, programs, etc.
- Packing needed knowledge from data repository.

Survey: Surveys are usually descriptive in nature, yet can also be used to provide casual explanations or explore ideas. A survey can be conducted using different information-gathering techniques such as mail-out questionnaires, in-person interviews and telephone surveys

• Knowledge Maps: Knowledge map is a visual representation of knowledge. They can represent

explicit/tacit, formal/ informal, documented/undocumented, internal/external knowledge.

- Motivation of the employee for better performance
- Engineering techniques : Successful engineering managers typically require training and experience

in both general management (which may include business administration) and the specific engineering disciplines that will be used by the engineering team to be managed

3.3.2 Lesson learn program

The experience may be positive, as in a successful test or mission, or negative, as in a mishap or failure. A lesson must be significant in that it has a real or assumed impact on operations [12], valid in that is factually and technically correct; and applicable in that it identifies a specific design, process, or decision that reduces or eliminates the potential for failures and mishaps, or reinforces a positive result [1]

- **Research & Development** Section: Research and development is one of the means by which business can experience future growth by developing new products or processes to improve and expand their operations.
- **Workshop:** Workshops are meetings where participants are involved in-group discussions and are normally organized around one or more theme areas[2]. Workshops allow participants

with differing values and priorities to build a common understanding of the problems and opportunities confronting them

• **Training** : The term training refers to the acquisition of knowledge, skills, and competencies as a result of the teaching of vocational or practical skills and knowledge that relate to specific useful competencies

• Joint Venture With Other Organization

3.4 Implementation: Implementation is the realization of an application, or execution of a plan, idea, model, design, standard, algorithm, or policy. In computer science, an implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system, it includes following two sub-parts:

3.4.1 Schedule:

- A *daily* schedule.
- A weekly or monthly schedule for employees
- Assigning job to each employee.
- Working Shifts of the employee.

3.4.2 Apply:

• **Development**: The process of translating from a detailed design to statements in a programming language[8]

- **Testing:** Software testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. The purpose of testing can be quality assurance, verification and validation, or reliability estimation.
- **3.5 Interpretation:** Interpretation phase is the final and essential phase. During the interpretation phase, the collected data are used for answering the stated questions and in that way for identifying whether the goals are achieved, it includes the following sub-parts:
- **3.5.1 Measurement:** In other words, results of the measurements are discussed and conclusions made in terms of measurement results. If hypotheses and conclusions are consistent the goal has been achieved. The following tasks should be performed during this phase:
- Verification & validation: Verification and validation is the process of checking that a product, service, or system meets specifications and that it fulfills its intended purpose Verification and validation often is carried out by a separate group from the development team; in this case, the process is called "Independent Verification and Validation".
- Alpha & Beta testing : testing of an application when development is nearing completion; minor design changes may still be made as a result of such testing.

Typically done by end-users or others, not by programmers or, testers.

3.5.2 STORE:

Whenever knowledge is created, this needs to be identified. Knowledge can then be evaluated to determine how it should be stored or distributed via the processes of "Sharing of Existing Knowledge".

- Acquire Explicit Knowledge: Explicit knowledge is knowledge that has been or can be articulated, codified, and stored in certain media. It can be readily transmitted to others Explicit knowledge is relatively easy to capture and store in databases and documents[15]. It is shared with a high degree of accuracy.
- Create New Hybrid knowledge Map : This Map basically design the complete project based on knowledge creation, knowledge storing, knowledge sharing, knowledge using.
- Create New Knowledge Repository:
- After a package has been designed and codified, it needs to be integrated to the repository. Thus, changes need to be identified and analyzed from the point of view of their impact on existing knowledge.

4. CONCLUSION:

After designing of framework for 5iKSPI Model, it need to be have empirical research on it, only after that we can say how mach its effect to improve the quality of the software. However, there are many issues that can only be answered through practical use and further investigation like: Cost, Effectiveness, and Ease-of-use.

5. **REFERENCES**

[1] Christensen, P. H.. 'The Wonderful World of Knowledge Management. In Challenges and Issues in Knowledge Management." A. F. Buono and F. Poulfelt (Eds). Information Age Publishing. 337-364., **2005**

[2] Daniela Damian. Stakeholders in Global Requirements Engineering: Lessons Learned from Practice. IEEE Software, 24(2):21–27, **2007.**

[3] Bridget Meehan , ita Richardson "Towards Managing Knowledge for Software Process Improvement" Sep 2000

[4] Ekkart Kindler, Vladimir Rubin, and Wilhelm Sch"afer. Process Mining and Petri Net Synthesis. In Proc. of Business Process Management WSh., pages 105–116, **2006.**

[5] Espen Frimann Koren, Siw Elisabeth Hove "Using a Knowledge Survey to plan Software Process Improvement Activities" **Jan 2004** a Case Study

[6] Eduardas Bareisa, Eimutis Karciauskas, Virginija Limanauskienė, Romas Marcinkevicius, Kęstutis Motiejunas "Software Engineering Process and its its Improvement in the Academy" **2005**, Vol.34, No.1

[7] Nicolas Anquetil, K'athia M. de Oliveira, Kleiber D. de Sousa, and M'arcio G. Batista Dias. Software Maintenance seen as a Knowledge Management Issue. Inf. Softw. Technol., 49(5):515–529, **2007**

[8] Raghavendra Rao Loka. Software Development: What Is the Problem? IEEE Computer, 40(2):112–111, Feb 2007 [9] Secchi, P. (Ed.) (1999). Proceedings of Alerts and Lessons Learned: An Effective way to prevent failures and problems (Technical Report WPP-167). Noordwijk, The Netherlands: ESTEC

[10] Suliman Al-Hawamdeh" Knowledge management: rethinking information management and facing the challenge of managing tacit knowledge" Information Research, Vol. 8 No. 1, **October 2002**

[11] Systems and software engineering - recommended practice for architectural description of software-intensive systems. Technical report, **2007**Thomas R. Cowles, "Criteria for Lessons Learned (LL)", 4th Annual CMMI Technology Conference and User Group November 15-18, **2004**