Software Process Improvement in Very Small Entities:
An investigation of Software Development Knowledge Management and Team issues in maintaining and evolving software process and process improvement.

Shuib Bin Basri

A dissertation submitted in fulfilment of the requirements for the award of

Doctor of Philosophy

to

School of Computing

Dublin City University

Supervisor: Dr. Rory V. O’Connor

2010

This thesis is based on the candidate’s own work, and has not previously been submitted for a degree at any academic institution

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of PhD, is entirely my own work and has not been taken from work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed : ID No: 55156231

(SHUIB BASRI)

Date:

ABSTRACT

The focus of this thesis is on how software process and Software Process Improvement (SPI) is practiced within the Very Small Entities (VSEs) among Irish software development companies. In particular the research is concerned with the process of software development knowledge management and team issues in supporting the software process and process improvement. In addition the study has put an emphasize on the issue of software quality standards to understand the reasons and factors that contributes to their low acceptance and implementation in VSEs. Using a mixed method research methodology which is dominated by the qualitative approach, this study has utilised in depth the interview technique at both management and team level in software development VSEs. These methods assisted the researchers in examining the attitude and perceptions of practitioners towards the research issues. The researchers also made use of survey questionnaires in VSEs in order to gain more input and to validate the qualitative data. The findings from the first stage analysis (qualitative analysis), in which the content analysis and grounded theory coding approaches were used, show the pattern and detailed categories that influence and are related with the software process and process improvement in VSEs. These categories are related to each other and allow the researchers to produce and validate the studies theoretical model. Likewise the second stage analysis (quantitative analysis) assisted the researchers in conforming and enhancing the first stage findings. This investigation shows that SPI programmes in VSEs are being undertaken in a very informal manner and also in indirect ways. The primary reasons identified for the informal nature of VSE SPI are due to cost, time, customer and company size, which give a higher priority to the product rather than process. In relation to teams, the small team size coupled with the working and management style have lead VSEs to be more informal in their knowledge management process and team organization. Moreover VSEs are largely ignoring the best practice SPI models. The reasons and acceptance criteria for this are discussed. This research also confirmed that SPI does not solely depend on technology but also the contributions of human aspects have a strong emphasize, especially in VSEs. Therefore a contribution of this research is to provide an extended knowledge and understanding of SPI research area in general and within VSEs domain in particular.

ACKNOWLEDGEMENT

No research would have been possible without the assistance and willing cooperation of many individuals and groups. I am taking this opportunity to record my deepest gratitude to all individuals, directly or indirectly, who have contributed towards the completion of this thesis.

First, I would like to express my heartfelt gratitude to my supervisor Dr Rory V. O'Connor for his expert guidance, confidence and stimulating suggestions throughout my research process. I would like also to express my profound appreciation to Dr. Ita Richardson from University of Limerick and others research group member who given their direct or indirect supports over time in finishing this study.

Thanks are also due to my employing institution, Universiti Teknologi PETRONAS, Malaysia (UTP) and LERO (Irish Software Engineering Research Centre) who have granted me the financial assistants through out my study. Without this commitment and generosity, it would not have been possible for me to embark on this research project. I also would like to express my appreciation to all the practitioners who gave their time so willingly and generously, and for sharing their experiences of software development with me.

I am greatly indebted to my parents, Basri Yusoff and Mariam Md Rawi, my lovely wife, Hartini Isahak and four sweet children, Alif Farhan, Amal Husna, Anas Kauthar, and Abdah Iman. The research project seems to have a way of life for imposing sacrifices on my family. I would also like to thanks all of my friends, in particular, Paul Clarke, Aakash Amir, Izwan Ismail and wife, and Dr Vera Sheridan for their kindness, help, support and warm friendship.

Last, but not least, to all members of faculty that given a full support and assistance in many ways along my study periods in DCU.

December 2009
SHUIB BASRI

List of Tables

List of Figures

Chapter 1 INTRODUCTION

1.1 Introduction

This thesis is concerned with software process, Software Process Improvement (SPI) and software process maintenance and evolution. A software process can be described as the way a company develops its software products and the steps that are followed at each phase of the software lifecycle. There are a number of software process standards in the market with the same basic objective, to help software companies in managing their software development activities in order to produce a high quality product. These standards list steps that a development organization should follow in each stage of the software production. It also provides assistance in making software project estimation, development plan and measuring quality of software development product.

It is commonly accepted that by having a better software process, a better software product will be produced and that software development productivity will increase (Zahran, 1998). This belief has lead to a focus on SPI in order to help companies in realizing the above benefit. To gain this benefit is not an easy task; substantial planning is required to analyse the resources that have to be invest. In addition, even though several methods and guidelines (e.g. Moprosoft (Oktoba, 2005), CMMI (Chrissis et al., 2003)) have been produced by several organizations in order to achieve their specific objective in development project, there are still a lot of challenges and obstacles that have to be manage (Laporte et al., 2008b; Mishra and Mishra, 2008). Moreover, in today's business environments; where user requirements keep changing, technologies are becoming more advanced and methods are more complex; focusing on improving current development processes are not enough without planning for maintaining and evolving current development processes.

Therefore this study is focused on SPI issues and mainly concentrates on software process maintenance and evolution. These processes could help software organization to always be ready to face every changes and challenges in today's software business. Process maintenance is the process involved in order to sustain software process momentum without allowing the team applying the old and not updated software development process and, to follow current development process strategies. This situation is known as process erosion problem. This problem must be addressed in order to ensure that people will always follow the updated and improved processes in order to align with current business objectives. Furthermore, software process evolution is process change that will evolve overtime in an organisation due to the influence of either external or internal factors, especially in business experiences and environments. Moreover, since software development projects by their nature is a teamwork effort and involve knowledge intensive exchange, the influence of software development teams dynamic and well organized software development knowledge could help organization prevent from the knowledge atrophy problem, the situation where important knowledge that related to software development project are loss due to the leaving of key development people. Nevertheless, the presents of these two elements will also enhance process improvement, maintenance and evolution of current development process more effectively and efficiently.

The significance of this research lies in the fundamental ability to understand and recognize the factors that support process maintenance and evolution in software development process. This research shows that in order for VSEs to survive and stay relevant in current software industry, the improvement, maintenance and evolution of the current software development process is pertinent. The research also shows that the significant of these processes is two-fold: (1) an organize of the software development knowledge and (2) the effectiveness of the software development teams within the organization. In addition to these two important factors, the emerging of a new process improvement standard, ISO/IEC 29110, which has been specifically developed by ISO for VSEs, is also being seen as having a potential to encourage and support VSEs in assessing and improving the current software process.

Therefore based on the research context and research issues above, the study has set out to explore the following research question:

> “*How can software development teams and software development knowledge facilitate Irish Software VSEs in maintaining and evolving software process and process improvement?*”

1.2 Research Agenda

1.2.1 Research Context

The context for this research has been limited in scope only to the software product companies whose primary business is software development. Software product companies are companies whose primary business is software development and performed task by a group of software developers. As a software developer, they would be familiar with the several software development process and considerable aware about the process development models. The context also has been decided to confine the study to Irish Software product companies. The reasons are based on the geographical location of the researcher, practicality and ease of access to those software companies and comparability of research data due to companies same jurisdiction, same economic and regulatory regimes governing their operation.

The Irish software industry can be divided into two main sectors; the Multi-National Corporation (MNC) and the indigenous sector (Acs et al., 2007). Even though these companies are different in terms of revenue and total exports, the numbers of people employed have been distributed evenly between the MNC and the indigenous companies with the ratio of 54% and 46% (Enterprise Ireland, 2005a). In additional the latest Enterprise Ireland publication (Enterprise Ireland, 2009) has cited that the indigenous software industry nowadays consists over 500 companies with over 10,000 people employed and exporting around €1.4 billion of products and services. Based on the important contribution by indigenous companies within the Irish economy and the challenges they deal with in producing a high quality software product, there is a need to assist them a way to sustain relevance in today’s software industry. However since the indigenous software companies’ have a different number of employees, the research context need to be further narrowed down.

Based on a European Software Institute (ESI) report which stated that, in Europe, 85% of the Information Technology (IT) sector companies have 1 to 10 employees (Laporte et al., 2008a). In the context of Irish software firms, 61% of indigenous Irish software firms which employed 10 or fewer, with the average size of indigenous Irish software firm being about 16 - 22 employees (Coleman and O'Connor, 2008; ISA, 2009; ICT Ireland, 2009). This has shown that most of the Irish software companies belong to Very Small Entities (VSEs) category. VSE has been defined as "*any IT enterprises, departments and projects with between 1 to 25 employees*" (Laporte et al., 2006). Therefore this study will concentrate on this type of indigenous software companies and will refer these groups of indigenous companies as an Irish Software VSEs.

1.2.2 Research Problem

The main issue of this study is that, in practice, most of large software companies are aware or well understand about the importance of improving their software process in current business situation (Zahran, 1998; Pino et. al., 2008). However, improving the current process only is not enough to sustain relevance in today's dynamic software business environment, without maintaining and evolving current software process. Moreover, several recent studies (Valtanen and Silvonen, 2008; Laporte et al., 2008b; Habra et al, 2008) regarding SPI showed that researchers have started focusing on small and medium companies instead concentrating on very large company.

However most of those studies are still focused on how to improve the software process but very little research explored on maintaining and evolving software process in software organization. According to Aaen et al., (2006) most of the researchers in the software discipline are more focused on software development dynamics issues but less focused on software organization and SPI dynamics issues. Beydeda and Grunh (2004) also argue that it is important for software process to evolve and change during their enactment in order to be more competitive and to decrease process risk. In addition, the existence of knowledge management process and an effective team in any organization will support organization improving their decision making, increase innovation and performance, and sustainable organizational competitive outcome (Rhodes et al., 2008 and, Faraj and Sambamurthy, 2006)

Moreover with the development of the new process lifecycle standard (ISO/IEC 29110) by the International Organisation for Standardisation (ISO) with the objective to guide the VSEs in assessing and improving their current software process (Laporte et al., 2008a), it is predicted being potentially to encourage VSEs to be more active in improve, maintain and evolve their current software process.

1.2.3 Research Questions

Therefore based on the main research question, research context and research issues above; the study has set out the specific focus of the investigation and problem addressed by this thesis that can be summarised by the following research questions:

RQ1: What is the current status of Software Process Improvement among Irish software VSEs?

RQ2: How software knowledge in Irish Software VSEs should be managed in order to maintain and evolve software process?

RQ3: How should software teams be organised in order to support software process improvement in Irish Software VSEs?

RQ4: How can knowledge and teams could support each other in order to prevent process erosion and knowledge atrophy in software development process within Irish Software VSEs?

RQ5: Does the emerging of ISO/IEC 29110 standard could encourage Irish Software VSEs in improving, maintaining and evolving their software development process?

1.2.4 Research Objectives

The research objectives of this study can be stated as:

1) To explain the role of software development knowledge management within Irish software VSEs, in maintaining and evolving software process.

2) To explain the role of software development teams within Irish software VSEs, in maintaining and evolving software process.

3) To identify critical factors in software development teams and software development knowledge management in Irish Software VSEs that could help them in maintaining and evolving their software process.

4) To investigate how software development knowledge management and software development teams could facilitate Irish Software VSEs in preventing process erosion and knowledge atrophy problems.

5) To explore the potential acceptance of the emerging of ISO/IEC 29110 standard among the Irish Software VSEs.

1.3 Structure of Thesis

The thesis is organized in three parts:

Part 1 contains four chapters. Chapter 2 details the role of the software industry in Ireland especially in Irish indigenous software companies. Chapter 3 contains the background information on software process literature, knowledge management literature, software development team literature and their relationship. In addition, this chapter also present research context model that will guide and drive the whole present study. Furthermore, research hypothesis also presented in chapter 3. Chapter 4 defines the detailed research methodology undertaken to complete this research. It describes the steps of the process and rationale behind the decision taken to combine the qualitative and the quantitative research methods.

Part 2 contains three chapters. Chapter 5 discuss the analysis and results of first stage of the research analysis outcome. Chapter 6 present the analysis and results of the second stage of the research analysis outcome. Chapter 7 elaborates in detail the combination of findings between stage one and stage two of the research findings.

Part 3 contain one chapter. Chapter 8 summarizes the contribution of this research, address its limitation and also explores directions for future research.

Chapter 2 THE IRISH SOFTWARE INDUSTRY

2.1 Introduction

This chapter discusses the Irish software industry and role of the indigenous software sector within. It explains the differences between the Multi-National Corporation (MNC) and the indigenous companies, categories of companies which thrive in the Irish software industries. This chapter also highlights how indigenous companies may be being classified and segregated. Finally, this chapter will explain the companies' characteristic relation with the study research context.

2.2 The Irish Software Industry Background

The Information Communication and Technology (ICT) sector has become a dominant industrial sector in the Irish economy. This sector is not only valuable in Irish economy but also a key component of the national economy. McIver (1998) has defined the Irish Software Industry as "*companies whose main business is in software or software intensive products that develop or modify software in Ireland*". Arora and Gambardella (2004) reported, in 2002 Irish ICT industry reached $13.9 billion in total sale. Barry and Curran (2004) added that 60% of all software sold in Europe originates in Ireland. In addition, Enterprise Ireland (an Irish economic development agency) estimated that at the end of 2005, the Irish ICT industry consist more than 900 companies which 140 of them are foreign owned companies with 24,000 people employed and exporting over €23 billion worth of products and services (Enterprise Ireland, 2005b).

Enterprise Ireland (2005a) reported that the success of Irish industry can be traced to the decisions made by Irish Government in 1970s to attract high value industries including software to invest and operate in Ireland. In the 1980s the focused moved towards the bespoke software service, limited export and low profit. The software sectors real growth was in the 1990s when the entrance of the MNC such as Intel, Symantec, Novell and Sun Microsystems. Such firms have a higher level of integration in local economy as compared to the previous entrants' software multinationals companies' (Hanratty, 1997; Tallon and Kramer, 1999). According to Crone (2002) the employment in the Irish software industry increased at an annual rate around 15% during 1990s compared to before which just under 8000 peoples. Heeks and Nicholson (2004) in their analysis of the software export experience of India, Ireland and Israel have point out five main success factors:

- Demand
- National vision and strategy
- International Linkages
- Software industry characteristics
- Supply factors and infrastructure.

The software industry sector in Ireland can be divided into two categories namely the MNC and the indigenous sectors (Acs et. al., 2007). Referring to Investopedia (2007), MNC is defined as *"a corporation that has its facilities and other assets in at least one country other than its home country. Such companies have offices and/or factories in different countries and usually have a centralized head office where they co-ordinate global management. Very large multinationals have budgets that exceed those of many small countries"*. Giarratana et al., (2003) cited that MNC in Ireland existed before the local industry started to grow and that the main factors which attracted the MNC to invest and stay in Ireland are varied. Flood et al., (2002) and Arora et al., (2001) claimed that the attractive grant and tax incentives given by the Irish government are the primary reasons that attracts MNC to invest in Ireland. These were supported by the many skilled and talented young workers with the low cost and the closeness with the European Union market (Arora et. al., 2001). According to Giarratana et al., (2003) the existence of MNC in Ireland is vital because they provide skills and reputation to the local firms.

Even though there are huge differences between MNC and indigenous companies (in term of revenue and total export) at the end of 2005, which MNC accounted 92% from the total revenue compared to indigenous companies which only accounted 8%, the number of people been employed have distributed evenly between MNC and indigenous companies (Enterprise Ireland, 2005b). This report also showed that MNC has employed around 54% of software workforce and the remaining 46% of software people were employed by the indigenous companies). In additional the latest Enterprise Ireland publication (Enterprise Ireland, 2009) has reported that indigenous software industry nowadays consists over 500 companies with over 10,000 people employed and exporting around €1.4 billion of products and services.

2.3 The indigenous Irish Software Industry

Coleman (2006), in his thesis has defined Irish software indigenous companies as "*those founded in Ireland, and whose parent company resides there, and whose main business is in the development or modification of software or software-intensive products*". According to Acs et. al., (2007) the number of indigenous companies and firms in Ireland has increased tremendously from 1991 to 1998. Crone (2002) argued that indigenous software companies in Ireland comprise of a number of sub-sectors. He segregates these companies that into two categories; product focus and service focus companies. However this segregation is not clear because some of the service focus companies are also heavily involved in software development (Coleman, 2006). Arora et al., (2001) added that even though 44% of indigenous software companies are mainly focus on software development products, they also offer service product elements such as installation, training and customisation. Therefore the indigenous software sector basically a cross functional companies.

Enterprise Ireland (2009) in their report cited there are five main characteristics that support the success of the indigenous Irish Software Industry nowadays:

- Software industries have been obsessed by the young, educated and highly-skilled workforce.
- The Irish government has largely increased the investment in research and development which involved the third level educations and the industry. As a result there is an increase in third level research and development and business expenditure.
- The software industry has excelled in corporate sectors like telecommunication, finance and e-learning by providing and developing expertise in this domain. This strong network of companies has a great

potential for i) leading edge university based research and development and ii) foundation for future growth.

- The present of multinational leading companies such as Google, Microsoft and others have contributed significantly to the local skills based directly or indirectly in software engineering.
- The location within the European Union and closed ties with the US has given the advantages to the software industries.

According to Mulligan et al., (2003) the micro, small and medium-size enterprises (SMEs) provide 99.8% business activity in Europe and account for 68% of employment while business turnover account for 63%. The majority of these SME's employ less than 10 people and small business has become a major contributor to the private sector employment and output. The European Commission (2005) have defined three levels of SME as being: Small to medium – "*employ fewer than 250 persons and which have an annual turnover not exceeding 50 million Euro, and/or an annual balance sheet total not exceeding 43 million Euro";* Small – *"which employ fewer than 50 persons, and whose annual turnover or annual balance sheet total does not exceed 10 million Euro"* and Micro – *"which employ fewer than 10 person and whose annual turnover and threshold of 2 million Euro"*.

The majority of indigenous firms in the Irish software industry fall under SMEs (Green et al., 2001). Crone (2002) reported the indigenous software sector in Ireland in 1990s was "*characterized by the particularly high start-up rate as well as low closure among established software companies*". Crone claimed in his report that in 1998, only 10 companies or 1.9%, out of a total of 630 indigenous software companies, employed more than 100 employees while the others are employed between 10 to 99 employees (Crone, 2002). Irish Software Association (ISA) stated that according to Ireland's Strategy for Science, Technology and Innovation (2006-2013), the indigenous software sector employs over 8,000 people in 400 companies (ISA, 2009). ICT Ireland (2009) added that the indigenous software sector employs 15,000 people in 660 companies. Therefore, that the average size of indigenous companies in Ireland is about 20- 22 employees. Coleman and O'Connor, (2008) in their research supported, that the average size of indigenous companies in Ireland is about 16 employees. HotOrigin the venture capital group has report on the state of indigenous software industry in Ireland claimed that indigenous software firms can be categories across three stage company development, ‘Start-up’ (1-25 employees), ‘Build’ (26-75 employees) and ‘Expansion’ (75+ employees) (HotOrigin, 2004). This report also claimed that 75% of the indigenous software firms are fall into a start–up category, 9% of the companies are in the expansion category and the remaining 16% fall under build category.

2.4 Very Small Entities (VSEs)

In relation with the section 2.3.1, according to Laporte et al., (2008a), most of the software industries especially in Europe, Brazil, and Canada are fall under Very Small Entities (VSEs) which employed less than 25 employees. VSEs is been defined by Laporte et al., (2006) as “*any Information Technology (IT) services, organization and project with between 1 and 25 employees*”. Laporte et al., (2008b) reported that in Europe 85% of the Information Technology sector have only 1 to 10 employees. While in Montreal area, Canada, 78% of the software development companies have less than 25 employees and in average 50% of them have less than 10 employees (Laporte et al., 2005). Furthermore in Brazil, small IT companies represent 70% of the total number of the software companies (Laporte et al., 2008b). In the context of indigenous Irish as been discussed in section 2.3, which shows that the majority of the Irish indigenous software firms are employed between 10 to 99 employees (Crone, 2002), indigenous software sector employs over 8,000 people in 400 companies (ISA, 2009) and average size of indigenous companies in Ireland is about 16-22 employees (Coleman and O’Connor, 2008; ISA, 2009; ICT Ireland, 2009), has shown that the majority of the Irish indigenous software company are fall into the VSE category.

In relation to the above, several literatures have detailed up the size of software VSEs. Laporte et al., (2005) found out that VSEs companies could be segregated into VSEs and VVSE. In this context, they defined VSEs as a software companies have less than 25 employees but more than 10 employees. Meanwhile, very, very small entities (VVSE) were defined as a company that consists of less than 10 employees. Orci and Laryd (2000) also have proposed a different classification of VSEs. In this classification, 3 main size that constitute VSEs, the extra, extra small

(XXS), which companies whose had less than 3 employees, the extra small or XS, which companies whose had between 3 - 16 employees and small or S, which companies whose had between 16 to 50 employees.

Sapovadia (2006) stated that micro enterprise including VSEs whose have limited resources, particularly in financial and human resources, are practicing unique processes in managing their business. These unique characteristics and unique situations have influenced VSEs in their business style compare to large companies (Mtigwe, 2005). In addition, their constraints in financial and resources also give an impact to companies' process infrastructures (Kaltio and Kinnula, 2000; Sapovadia, 2006) such as limited training allocation, limited allocation in performing process improvement, low budget to response the risk and many other constraints. Moreover due to the small number of people involved in the project and the organization, most of the management processes are performed through an informal way which most of decision-making, communication and problem solving have been discussed orally and less documented. This situation shows that human-oriented and communication factors are very important and significant in VSEs (Valtanen and Sihvonen, 2008; Laporte et. al., 2008b). For example, compared to small companies, very small companies often do not have a regular project meeting (Laporte and April, 2006). Laporte et al., (2008b) has table some characteristics that show the differences between the small and large companies as in table 2.1:

Table 2.1: Characteristic differences between large and small companies

Characteristic	Small Firm	Large Firm
Planning Orientation	Unstructured/operational	Structured/strategic
Flexibility	High	Structured/strategic
Risk Orientation	High	Medium
Managerial Process	Informal	Low
Learning and knowledge absorption capacity	Limited	High
Impact of negative market effects	More Profound	More Manageable
Competitive advantage	Human capital centred	Organizational capital centred

Despite constraints in resources, different in business style and diversity in level of software development process among VSEs, there are some common characteristics in VSEs software development processes (Habra et al., 2008):

- The software development lifecycle is often highly simplified and some of the development phases (e.g. analyzing, implementation and testing) are not formalized.
- The maturity levels of processes within the same company can be mixed up between very good processes and low level processes.
- In general, quality control procedures are not very formalized.
- In general, most of the project management and planning practices in VSEs are not standardized across organization and always depend on the project, clients, teams and project manager.
- The resources allocated to training and human resources are very limited because of strict financial allocation.
- Most of the software projects are driven by a short term strategy and rarely driven by a long term strategy.
- Due to the size, VSEs have difficulties to impose a standard methodological approach in their software project.
- In general, the issue of risk management is less important and not taken seriously. This is due to the short term strategy in VSEs software development project.
- Quality issues are not addressed explicitly with an actual involvement of management.

Moreover from the above discussion, we could understand the background, characteristics and important contributions of VSEs in Irish software industries in specific and others in general. Furthermore due to the some limitation exist in VSEs, especially in terms of time, people and financial, have lead them to manage their software development process in different way as compare to large companies which are more organize and formal. Therefore we believed that the focus to VSEs as our research context is very significant and will add a valuable research contribution in SPI area.

2.5 Summary

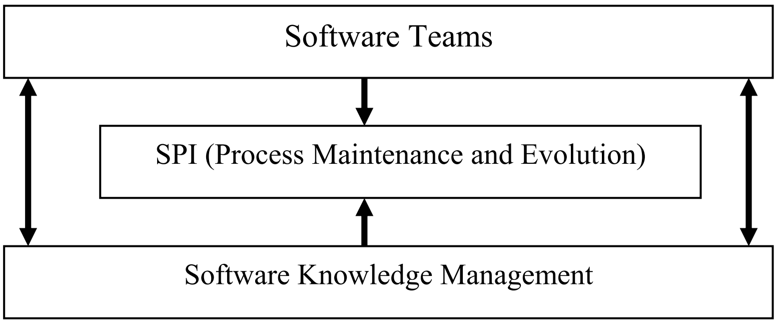
This chapter has presented a summary of the Irish Software Industry. This industry is divided into two major categories the MNC and the indigenous sector. The indigenous sector which has different in terms of revenue and the size of export, have been employed nearly the same amount of people as in MNC. Most of the indigenous companies in Ireland are in VSEs categories, have different style of business due to their unique environment and characteristics. Moreover, this chapter also describes their different and common characteristics in doing their business. The research background study focuses on SPI, Team and KM are of the next chapter.

Chapter 3 BACKGROUND STUDY

3.1 Introduction

This chapter presents concepts related to SPI, Software Knowledge Management and Software Team, and discusses background and relationship each of these concepts as illustrate in figure 3.1. This chapter also will elaborate the importance of process maintenance and evolution in SPI, which are the main issues of this research study. Moreover, this chapter also discusses the relationship between these three concepts and how they support process maintenance and evolution in SPI. Lastly, this chapter will propose the overall research study framework and hypothesis that will drive the present study.

Figure 3.1: Research Relational Concept Model


Software Teams
SPI (Process Maintenance and Evolution)
Software Knowledge Management

3.2 Software Engineering

It is commonly understood that software engineering has been defines as "*an engineering discipline that is concerned with all the aspect of software production from early stages of system specification to maintaining the system after it has gone into us*" (Sommerville, 2004). This definition has been clarified in details by other researchers (Chow et al., 2003; William, 2004) and claimed that software engineering is concerned in all aspects of software development process such as:

- the whole life cycle of the software development process (Maciazek and Liong, 2005),
- the software process model and software development process in general (Scacchi, 2002) and

- the project management, knowledge, human, method and theories to support software production (Chau et al., 2003).

Therefore software engineering can be referred as a collaborative development process that bring together domain expertise with technological skills and process knowledge in the production of software.

3.3 Software Process

Zahran (1998) defines software process as "*a set of activities, methods, practices and transformations that people use to develop and maintain software and associated product (e.g. plan, documentation, code, test case and user manual)"*. Therefore software process can be elaborated as a development process has followed by organizations to develop and produce a software product.

The software process has four distinct roles (Krunchten, 2000):

i. To present a guidance as the guideline of the activities to be undertaken.

ii. To specify the artefact that should be developed and when.

iii. To direct the task of the development team.

iv. To offer ways of monitoring and measuring a project progress and outputs.

The activities in the first role above are related to four common fundamental activities in software process (Sommerville, 2004):

i. Software specification where functionality and constraints of the software product is defined.

ii. Software design and implementation where software is designed and developed.

iii. Software validation where the software is checked and screened to ensure the result followed what user need.

iv. Software evolutions where the software must be maintained and modified to meet the changing customer need.

The process and the activities in the software process need to be represented and documented clearly as it is implemented within the organization. This document or process guide will become a standard procedure during development. To simplify the understanding and to create a general framework for an organization to adapt, software process should be represented in a model or a framework (Canfora et al., 2005).

3.3.1 Software Process Model

A software process model is a simplified, abstracted description of a software development process (Canfora et al., 2005; Sommerville, 2004; Scacchi, 2002) and is prescriptive in that it indicates how software should be developed. The primary purposes of a software process model is to determine the order stages involved in software development and to established the transition criteria for processing from one stage to the next stage (Boehm, 1998). According to Sommervile (2004), there are three major general categories of software process model namely the waterfall, evolutionary and component-based development models. The waterfall model was derived from the general software engineering process (Royce, 1970). It represents the fundamental development activities of specification, development, validation and evolution as separate process phases. Each of these phases must be completed before starting the other phases. The evolutionary development model replicates each of the fundamental steps as used in waterfall with in a several small development cycles in each development phases. This model was claimed more effective than the waterfall approach in producing the system that meets the immediate needs of customers (Sommerville, 2004). Component based development is a model which based on reusable software component that has been previously tested in software development (Thomas, 1995). This process model is more focused on integrating the reusable components into a system rather than developing component from scratch. The main advantage of this approach is faster delivery of the software and reducing cost and risk (Sommerville, 2004).

3.3.2 Software Process Improvement (SPI)

SPI has gained increasing importance in software engineering. According to Zahran (1998) the main aims of SPI are to understand the software process used in the organization and to guide the implementation of changes of that process in order to achieve specific goals such as to improve software development time, on budget and with the desired functionality. According to several authors, SPI has a close link between the quality of the development process and the quality of the product developed using the process. Zahran (1998) stated, "*it is widely accepted that the quality of a software product is largely determined by the quality of the process used to develop and maintain it*". Humphrey (1995) also claims that "*to improve your product, need to improve your process quality".* Herbsleb and Goldenson (1996) reported that research shows that SPI will increase product quality, improve customer satisfaction and reduce risk at the high maturity companies. SPI itself denotes the changes implemented to software process that bring about improvement (Olsen et al., 1989). The SPIRE handbook (Sanders, 1998), which focused on the small companies, also promotes improving software process arguing that software process is like a business process and both are related, which mean process efficiency will lead to business efficiency and better product. Ward et al., (2001) also claimed that small companies become more successful in producing quality product if software process improvement fundamental are observed closely. Other authors that support this relation between the process and output, also report the success of SPI including (Ahern et al., 2004; Borjesson and Mathiaseen, 2004; Goldenson and Gibson, 2003).

Hall et al., (2002), argued that the SPI influencing factors can be divided into 4 categories namely the economic, people, organization and implementation factors:

- *Economic factor* - Bucci et al., (2001) and Hall et al., (2002) claimed that high costs and inadequate resources have been found to be the greatest impact to SPI success. According to Batista and de Figueiredo (2000), whose studied a very small team which implemented SPI, found that over 12 months, monthly cost decrease by 33% while monthly benefits increased by 17% of their monthly value. Moreover, Solligen (2004) suggested that investment in SPI should be treated as any other investment and stress the need for measuring the return of investment (ROI) to secure continuing improvement programmed. Debou and Kuntzmann (2000) claimed that late impact of SPI programs on projects as a very important issue. They stressed that action time frames must plan carefully in which an assessment should cover a three to five months in average. They added that planning that have a longer time period will made a management tends to lose patience and practitioners lose their interest in SPI.

- *People factor* – The contribution software development team in software development project and SPI have been discussed seriously in literature. Komiyama et al., (2000) argue that the success of software project and process is determined by the interest of software team on the project and process itself. Horvart et al., (2000) argue in small firms are in greatly rely on key individual. Therefore a human factor is particularly important to them. This factor has been shown in MAP (Management, Approaches and Perspectives) framework for SPI (Aean et al., 2001) which viewed SPI is a combination between social and technical intervention. Knauber (2000) added in detail that in small software organization, the influences of key individuals, such as the company founder or a talented are not enough without sufficiently educated developer. He comments that the reluctant of the key player in small establishments can effectively sabotage the chance the success of improvement scheme. In conjunction of that, Sommerville (2004) claimed that technology and people factors also affect the quality of software product. De Marco and Lister (1999) also stressed that the sociological rather than technological factors were the main cause of failure and human issues must be given high priorities in software development. Staff participation also is essential in improvement activities and should be involved in improvement initiative because they have detailed knowledge and first hand experience of strength and weakness of the current process (Stelzer and Mellis, 1998; Dion, 1993). Niazi et al., (2006) in their research on critical success factor for SPI with 34 SPI practitioners supported that staff involvement, training and mentoring are important criteria in determining SPI success.

- *Organisation factor* - The ability to manage organizational change as a prerequisite for meeting competitive demands is very important. Several authors emphasize that organizational change management is one of the key issues to achieve success in software process. Cattanoe et al., (1995), Clark et al., (1997) and Herbsleb and Goldenson (1996) claimed that SPI needs effective change management otherwise the culture and politics within the organization can create a barrier to SPI adoption. Management commitment is important in order to ensure the success of organization change. Niazi et al., (2006) and Mathiassen et al., (2005) found that developer opinion and management commitment plays a vital role in the implementation of the SPI programs. Wiegers (1998) argues that the lack of adequate management commitment is the first trap to avoid when starting to improve software process. Even in software project the lack of management commitment to the project is considered to be risk number one (Keil et al., 1998).

- *Implementation factor* - There are a variety of implementation factors that can cause the failure of a well planned SPI initiative. According to Weigers (1998), an action plan is needed after the assessment and SPI should be treated as a project. It is also important to ensure updated software development processes are institutionalized in organization (Stelzer and Mellis, 1998). This action will help to prevent the developer turning back to the not updated process (Stelzer and Mellis, 1998; Wiegers 1998; Butler 1997). Niazi et al., (2006) agreed that there is a need to create a SPI implementation methodology that guides SPI practitioners in implementing SPI initiatives. In addition, SPI initiative also should have realistic objectives which can be achieved in the foreseeable future (Stelzer and Mellis, 1998). It is important that improvement objectives should be ambition and the SPI goals were well understood (El Emam et al., 2001). According to Kaltio and Kinnula (2000) from their experience at Nokia, the critical elements for successful deployment of process are a well organised and institutionalized support infrastructure. Moreover organisations also are urged to set up SPI steering groups, Software Engineering Process Group (SEPG) or control boards made up of representatives from different projects and functions. This practice is important in order to promote SPI and to provide coordination on critical issues (Hall et al., 2002; Herbsleb and Goldenson, 1996).

- *Communication factor* – Beside the four main issues above, one main focus issue that has been discussed in detail in SPI literatures is communication. Stelzer and Mellis, (1998) and; Kautz, (1998), claimed an effective communication in an organization is important in order to ensure SPI been accepted by the people. They added to overcome resistance to SPI, it is important to establish and maintain effective communication and feedback channels in organization. They also claimed that a strong communication effort is needed before and during SPI implementation. This situation could also enhanced shared experiences culture among teams in the organization (Stelzer and Mellis, 1998; Kautz, 1998). In small software organizations, Johanssen (2004) in her master dissertation found and agreed that small software organizations have the advantage in enhanced and open communication between management, developers, contractors, client and users, compare to a large enterprise. She added that a flat organization structures in small software organization help organization radically to reduce the coordination and communication administration. In addition the short distance between the decision maker and the developers in flat organization structure will enable early detection and tackling of problems that otherwise will get out of control (Ward et al., 2000). Therefore SPI also will be more successful when employees time and resources dedicated to SPI are effective and excellent (Baddoo and Hall, 2003; Kaltio and Kinnula, 2000).

3.3.3 SPI Models and Standards

There are a number of SPI models and standards developed by the international organizations, industry consortia, large software purchasers and software developers. Capability Maturity Model Integration (CMMI) (CMMI Product Team, 2006), BootStrap (Haase et al., 1994) and Trillium (Zahran, 1998) are among the SPI models that have been produced. In addition, The International Organisation for Standardisation (ISO) also embarked on the programme to create a software process assessment standard. The ISO 9000 was developed by ISO, is a series of standards used to certify the quality system used by an organization (ISO, 2008). These standards provide organizations with guidance on managing quality system. Moreover, the ISO/IEC 15504 was developed by ISO for the software process improvement under the SPICE (Software Process Improvement and Capability dEtermination). SPICE was established as a project to develop draft standards, trial the developing standard and to promote awareness of developing standard (Rout, 2004). Furthermore, these standards include guidelines for developing assessment instruments and provide a reference models and rating scheme (Zahran, 1998).

- The Capability Maturity Model Integration (CMMI)

According to CMMI Product Team (2006), CMMI provides two views of capability: a staged view and a continuous view. The staged view gives 5 levels of evolution towards organizational maturity. The levels are Initial, Managed, Defined, Quantitatively managed and Optimizing. The continuous view provides six levels of process capability. The levels are Incomplete, Performed, Managed, Defined, Quantitatively managed and Optimizing. The SEI claims the CMMI model and CMMI best practices enable organizations to do the following:

- link management and engineering activities more explicitly to business objectives,
- expand the scope of and visibility into product lifecycle and engineering activities to ensure that the product or service meets customer expectations.
- incorporate lessons learned from additional area of best practice
- implement more robust high- maturity practices
- more fully comply with the relevant international standards such as ISO9000 and ISO/IEC 15504 (Chrissis et al., 2003; CMMI Product Team 2002; CMMI Product Team, 2006).

- ISO 9000

ISO 9000 was developed by the International Organisation for Standardisation (ISO). ISO does a series of standard used to certify the quality systems used in an organisation (International Organisation for Standardisation, 2000). According to ISO, the standards are intended to provide generic core of quality system standard applicable to a broad range of industry and economic sector. The ISO series are referring the ISO family which is a set of recognised standards of a quality management which include ISO 9001, ISO 9002, ISO 9003 and ISO 9004. Moreover in latest ISO standard shows that all three standards 9001, 9002, and 9003 are combines into one, called 9001 (ISO 9001:2000, 2001). According to ISO (ISO 9001:2000, 2001), ISO 9001, 9002 and 9003 are model for the external quality assurance. They specify the set of requirements and also as a based for certification or registration for the organization which demonstrate conformance to these requirement. Furthermore, the ISO 9004 series consist of standard of internal use by the organisation (Stelzer et al., 1996). They added that there are two different situations that the ISO series mainly used:

- to provide quality for quality management and
- as a framework for third party certification or registration.

Like CMMI, ISO 9000 series standards have a common concern of quality and process management. ISO 9001 is more concern with software development and maintenance and identified minimum requirements for quality systems (ISO 9001:2008) but CMMI is more concerned about the need for continuous improvement (Coleman, 2006). Paulk (1994) also added that the ISO 9001 have a broader scope that focus on hardware, software, processes material and service as compare to CMMI which is focuses strictly on software (Chrissis et al., 2003; CMMI Product Team, 2006). Unlike CMMI, ISO 9000 does not provide a road map for improvement beyond the quality of adherence to quality management document (Coleman, 2006).

• ISO/IEC 15504 (SPICE)

There are two dimension of the ISO/IEC 15504 (SPICE) reference model, the process dimension and the process capability dimension (Zahran, 1998). Process dimension is characterized by process purposes. It includes measurable objectives for each process and relates to the process life cycle model (ISO 12207). The ISO/IEC 15504 technical report divided the process dimension into five categories: customer-supplier, engineering, support, management and organisation. Process capability dimension is characterized by a series of process attribute. It is measured by capability levels. According to (Zahran, 1998), "*capability level is a set of attribute(s) that work together to enhance capability to perform a process. Each level provides a major enhancement of capability in the performance of the process*". ISO/IEC 15504 specifies process capability dimension is divided in to six capability levels. The level is from incomplete through performed, managed, established, predictable to optimising. These capability levels represent milestones along the road to software process improvement.

- ISO/IEC 29110

In general most of SPI standards and models are more prone to a larger organization compare to medium and small organization (Hauck et al., 2008). In addition, some of them have developed and created their own software development process model, but most are adapting and referring to the commercial standard such as CMMI and SPICE (Laporte et al., 2008a; Von Wangheim et al., 2006). However these situation are different in the majority of small software organization which are not adapting any standard and perceived that those models as being oriented toward the large organization (Laporte et al., 2008; Von Wangheim et al., 2006). Some studies have shown that this negative perception on the software process model was drove by a negative view of cost, documentation and bureaucracy (Laporte et al., 2006; Pino et al, 2008). Moreover, it has been reported that the small software organization found that it is difficult to relate the existing standards (e.g. CMMI, ISO 12207) to their business need and justified the international standard in their business operation (Laporte et al., 2006) . Laporte et al., (2008a) added that most of the small software organization which have few employees cannot afford to establish and follow the software process as defined by the current standard.

Therefore to overcome the above issues, ISO is currently developing a new lightweight standard in order to encourage small software organization to actively assess the development process (Laporte et. a., 2006). This lightweight process and practice have been proposed in the ISO/IEC JTC1/SC7's international software engineering standards (Laporte et al., 2008b). This new lightweight standard has specifically developed for VSE is currently being developed and has been form as ISO/IEC 29110. The main objective of ISO 29110 is to assist and encourage small software organization (which has less than 25 employees) in assessing and improving their software process (Laporte et al., 2008a).

- Agile Methods

According to Beck (2000) and Cockburn (2002) agile software development method or agile method was evolved in the late 1990s as part of a reaction against a complex rigidly methods within the existing process and process improvement model. It deals with the unstable and volatile requirements by using a number of techniques (Huo et al., 2004). They added that there are four characteristic in agile methods: simple planning, short iterative, short release and frequent customer feedback. Larman (2004) and Larman and Basili (2003) argues that agile methods are subset of iterative and evolutionary methods. It also based on iterative enhancement (Basili and Turner, 1975) and opportunistic development processes (Curtis, 1989). In all iteration product, each iteration process is a self contained, mini project with activities that span requirement, analysis, design, and implementation and test (Larman, 2004).

Agile methods emphasize real time communication such as face to face rather than written documents (Sfetsos et al., 2006). Most agile team are collocated and include all necessary people in order to finish the project. Viscontil and Cook (2004) emphasized, working software is a primary measure of progress in agile methods. Some agile software development methods are including Extreme Programming (XP) (Beck, 2000), Scrum (Schwaber and Beedle, 2002), Feature Driven Development (FDD) (Palmer and Felsing, 2002), Crystal (Cockburn, 2000) and Adaptive Software Development (ASD) (Hughsmith, 2000).

Therefore from this part, researchers could understand that there are several different models and standards that could apply in SPI. The literatures also have shown that in general the purpose of the models and standards is to guide software companies in producing a quality system. In addition, the CMMI and ISO are the main official standards that shape the software quality standard in software organization and prone to the large companies. CMMI is provides organizations with the essential elements of effective processes that ultimately improve their performance. While ISO standards are intended to provide generic core of quality system standard applicable to a broad range of industry and economic sector Moreover, the existing of new ISO standard (ISO/IEC 29110) is predicted to overcome this gap and help small companies to seriously involve in assesses their current development process which usually follows the agile software development method.

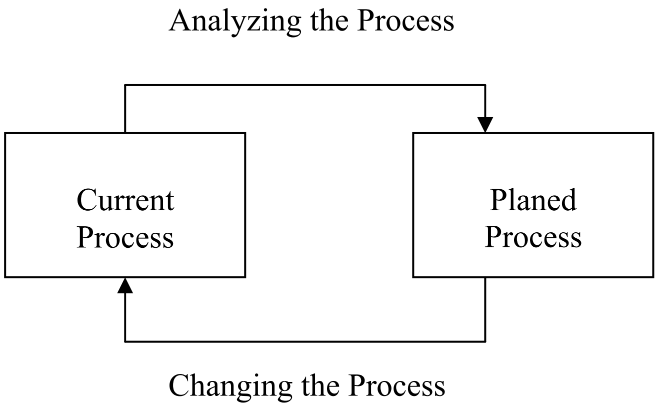
3.3.4 SPI Lifecycle

Software developments go through different phases in software lifecycles and so does SPI. According to Cook and Wolf (1998), software process and software product development are related to each others. They argue software process as a set of sequence task leading to design, development and maintenance of a software product. Therefore to improve current organization software process, it has to follow a lifecycle process. Fuggetta, (2000) argued that two most SPI lifecycle always been referred in the literature are the ISO and SEI based improvement method which is call the SPICE and IDEAL. Stelzer and Mellis (1998), added that most SPI lifecycle produce by ISO 9000, was built based from the Shewart four steps improvement cycle for planning, executing and managing improvement program. The lifecycle is divided the process into four phases:

- PLAN – planning from the improvement effort,
- DO – execute the plan,
- CHECK – measuring the improvement and
- ACT - implement the plan according to the check phase.

Stelzer and Millis (1998) added that the most important level in SPI lifecycle is changing software processes and it consists of two phases. The first phase is an analysing the process phase and changing the process the process is the second phase. The output from the first phase is a planned process for the second phase. The result of the second phase is become the new current process and this process will be continuously. The processes are shown in figure 3.2.

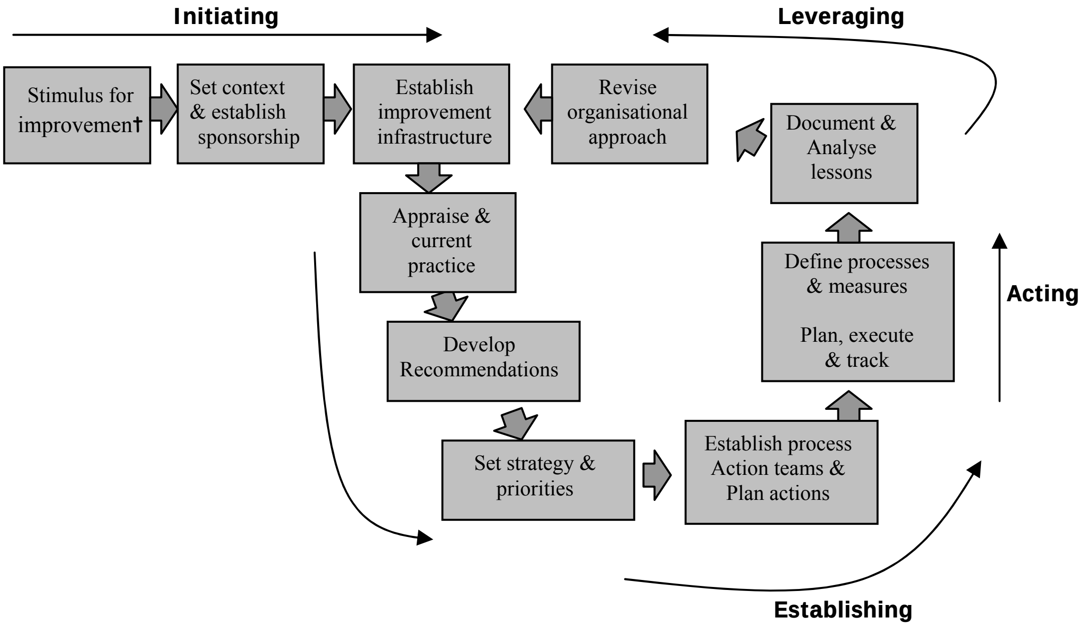
Figure 3.2: SPI Change Process



3.3.5 The IDEAL Model

The IDEAL Model is a framework which can be use to guide the implementation of SPI program (McFeeley, 1996). IDEAL provides a practical and understandable approach to continuous improvement by providing necessary activities to establish a successful improvement program. Gremba et al.,(1997) supported that the IDEAL model also provide a disciplined engineering approach for improvement, focuses on managing the improvement program and establishes the foundation for a long term improvement strategy. The IDEAL model presents a five phase cyclic approach to SPI as shown in figure 3.3:

Figure 3.3: The IDEAL Model

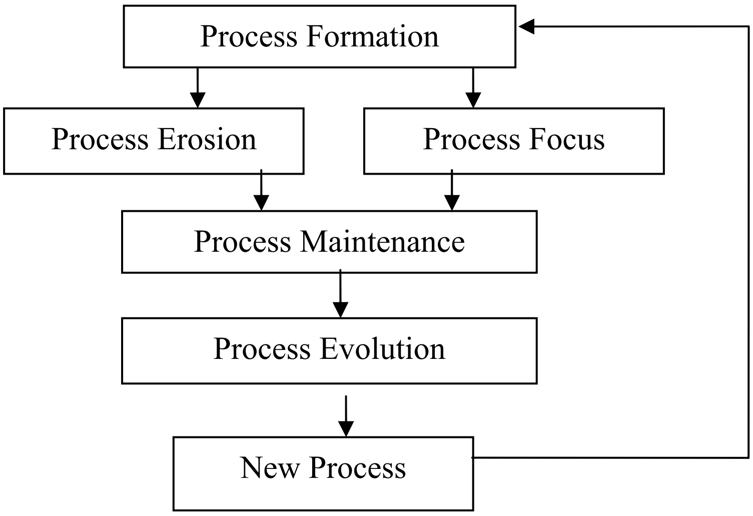

Initiating
Leveraging
Stimulus for improvement†
Set context & establish sponsorship
Establish improvement infrastructure
Revise organisational approach
Document & Analyse lessons
Appraise & current practice
Define processes & measures
Plan, execute & track
Acting
Develop Recommendations
Set strategy & priorities
Establish process Action teams & Plan actions
Establishing

The model defines 5 phases: the Initiating, Diagnosing, Establishing, Acting and Leveraging phase. Each phase consists of several task and some tasks comprise of several sub tasks. According to Casey and Richardson (2004) the IDEAL model is based on SEI's experiences with their governmental and industry customers, which are usually very large organizations. They added since most of software companies belong to the category of small and very small enterprise, the use of IDEAL is still limited.

3.3.6 Process Maintenance and Evolution (PME)

Coleman (2006) in his thesis on software process usage in Irish owned software firms has developed a SPI network diagram to show how software process have been formed, maintained and evolved over the time. In summary the whole processes can be depicted as in figure 3.4.

Figure 3.4: SPI Life Cycle Diagram



The root process of the lifecycles is started with a *process formation*. The main objective of process formation is to establish the process improvement in the organization. In general, process formation was initiated by two main important process occurrences in organization namely *process erosion* and *process focus*. Process erosion was happened when the established and improved development process have not been followed by the people due to the cost of process and the loss of the related knowledge. These problems were lead to a minimum process, where development processes is not aligned with the latest business process. Meanwhile process focus happened when there are changes in business events or the market requirements which could be a positive change or negative change. According to Humphreys (1989) the negative changes can also introduce an entropy problem to the process improvement. Process entropy is a process where the people using the inefficient development process or an old development process, which are not updated and not improved. While customer complaints, low quality (Coleman, 2006), low management support, low staff involvement, less experience staff, low staff awareness and less resources (Niazi et al., 2006) are among the negative changes. While positive changes can be classified as the entire positive event as opposed to the negative changes that could encourage process improvement. Therefore in order to sustain current development process performance, *process maintenance* need to be done. In this case which studied by Coleman (2006), process maintenance was be done by hiring of expertise to study the whole process of improvement and identify the factors that affect the current process such as the business event was reflects the process maintenance happened. The last phase in practice SPI lifecycle is the *evolution process*. Over time, the software development process experiences procession erosion and focus. This situation will leads organization to the weakness or insufficiencies in software process that will prevent software companies from producing a quality software product.

3.3.7 The importance of PME

Beside SPI life cycle, the importance of process maintenance and evolution also needs to be emphasized. Software processes are complex entities that may last for a long period of time and are carried out through the interaction of humans and computerized tools. Like software system, software process, once deployed need to be continuously maintained and evolve in order to cope with different kind of changes or customization both in organizational and technological issues. It used to support software production activities and also prevent from process entropy and erosion problems. The ability of handling these changes and problems are important in order to ensure that process improvement will continue (Aaen et al., 2007).

According to Borjesson and Mathiassen (2005), in current business situation where changes happened such as market change, technology innovation and customer requirement, software organization need to organize, manage and execute their software process improvement in a way that allows the process to react and response to these changes environments. Holmberg and Mathiassen (2001) added that it is important that organizations have a strong focus on improvement initiatives and always aware of business environment changes. Baydeda and Grunh (2004), argue that the development of software process is always based on a certain assumptions related with customer, market, competition and others. All these assumption will influence all the processes involved in software development. Therefore it is important for organization to correct the assumption by change and evolve the process in order to be more competitive and reduce process risks.

According to Stelzer and Millis (1998), one of successful factor in software process improvement is stabilizing changed process. Stabilizing changed process means continually supporting maintenance and evolution at a local level. This effort is required because the SPI initiatives do not have along lasting effects and the benefit are often short-lived. Therefore it is necessary to prevent an improve software process turning back to the old level and to prevent an employee fall back to the old habits. These situations call process entropy (Humphrey, 1989) or process erosion (Coleman, 2006). Continuous feedback, motivation, recognition, reinforcement (Wohlwend and Rosenbaum, 1994), guidance and support to the staff (Paulish and Charlton, 1994), management commitment, opinion leader and change agent to which influence in providing continuous feedback, and motivation (Trienekens et al., 2007), are important factors in managing process erosion from happened.

Hardgrave and Armstrong (2005) in their industrial research on software process improvement claimed that organizational SPI is a continuous process. They argue that either the SPI initiatives are successful or not, companies should take an appropriate plan and action. These approaches need a process change and evolve in order to suit with the dynamic business environment requirements. Nierstrasz (2004) added that among the important factors to be success in software projects are continuous change and increased complexity. Continuous change is important to ensure that the program is following the environment changes. As the program evolve and be more complex more resources need to preserve to simplify the structured. Programs changes and evolvement must be followed by the changes and evolvement of software processes because software process and software development are related each other (Sommerville, 2004). Therefore the proposed issue that process maintenance and evolution should be at the center of software process.

3.4 Knowledge Management (KM)

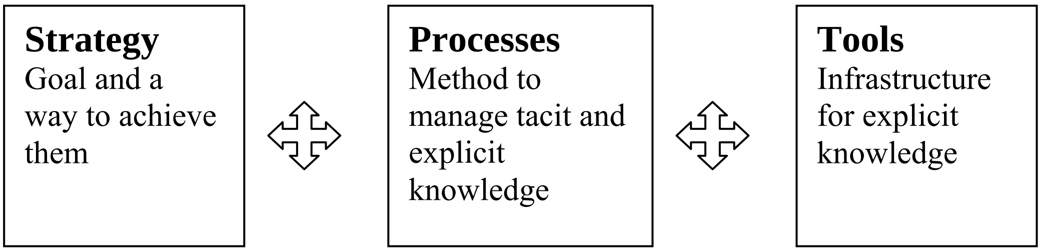
This section elaborate in details the concepts related to KM in software development process. Furthermore, this section also will discuss on knowledge atrophy problem, one of the main problem in this research study that have to be aware in software development project. The relation between KM and SPI, then will be discussed and concluded.

3.4.1 KM Background

KM is a discipline that crosses many areas such as economics, informatics, psychology and technology. Garavelli et al., (2004) have defined KM as a "*set of the interdependent activities aimed at developing and properly managing organization knowledge*". This definition have been clarified by Nonaka, (1991) and Nonaka and Takuechi, (1995), claimed that KM is a process of creating value from ideas and making this value available for entire organization. Furthermore, KM also could accommodate critical issues in organizational adaptation, survival and competence in face of increasingly discontinuous environmental change (Maholtra, 2000). Zhang and Zao (2006) have listed three main important criteria on KM namely strategy, process, and technology. These criteria are required in order to select, organize, share and leverage business critical information in an organization (Desouza, 2004). In addition KM is the important in order to help organization in improving decision making, increase organization innovation and performance and help organization to sustain their competitiveness (Rhodes et al., 2008; Bolingger and Smith, 2001). Therefore from the literature, KM is seen as a strategy that creates, acquires, transfers, consolidates shares and enhances the use of knowledge in order to improve organizational performance and survival in a business environment.

In additional, KM also is an iterative process of continuous development. In order to understand the effective of KM, Dingsoyr and Conradi (2002) have investigated KM approaches in eight case studies; proposed with a framework of KM program that must exist in the companies. The framework is depicted in figure 3.5:

Figure 3.5: A framework of the components of a KM


Strategy
Goal and a way to achieve them
Processes
Method to manage tacit and explicit knowledge
Tools
Infrastructure for explicit knowledge

In this framework Dingsoyr and Conradi (2002), define that KM program consist of three parts; strategy, process and tools.

- *Strategy* - is referred to the company goals and strategies in KM and how the company does plans to achieve them. These strategies can be divided into an informal strategy and formal strategy. Informal strategy is concern with the spontaneous conversation and unstructured mechanism rather than formal channel such as structured meeting or a written communication. Desouza (2003), in his studies found that informal communication, facilitate the knowledge exchange. Nonaka and Takeuchi (1995) also supported that socialization which is very informal way of distributed knowledge contributes to knowledge creation and sharing (Nonaka and Takeuchi, 1995).

- *Processes* - are defined as activities in order to facilitate KM. It is refer to methods for collecting and distributing knowledge and also an organization, project manager and software developer activities in relation to KM.

- *Tools* - referred to KM tool software system. The tools must able to update and maintain the information and knowledge as well as supplying new knowledge into system. The knowledge can be represented in databases, web pages or file.

Moreover, Kukko et al., (2008) stated that in order to help software team in renewing software development process, a well defined KM process which include the technology, strategy and process (including culture) must be well and clear defined. Chan and Chao (2008), in their study of 68 SMEs organization in Hong Kong which employed fewer than 200 employees, reported that in order to be success in KM program in SMEs organization, 2 main capabilities must be present: the infrastructure capability and the process capability. The infrastructure capabilities which are the prime elements are similar with what Dingsoyr and Conradi (2002) three main elements for KM framework as been described above. Meanwhile process capabilities which are an integration of organization resources in creating an organization knowledge consists 4 capabilities; Acquisition, Conversion, Application and Protection; are important in supporting the prime elements.

3.4.2 Knowledge

Knowledge is not accurate information but it is a product of human intercourse. Zyngier (2002) stated human intercourse is a "*process as applied to a given context. Until it has been synthesized into knowledge, the best information is of limited value"*. According to the dictionary (Merriam-Webster, 2001), knowledge has been defined as "*the fact or condition of knowledge with familiarity gained through experience or association; acquaintance with or understanding of a science, art, technique; the fact condition of being aware of something*". In software development organizations, Davenport and Prusak (1998) described knowledge as "*a fluid mix framed experience, values, contextual information, and expert insights and grounded intuitions that provide a framework for evaluating and incorporating new experience and information, it originates and applied in minds of the knower. In software organizations, it is often becomes embedded not only in the documents or repositories, but also in organizational routines, processes, practices and norms*".

There are three levels of refinement that relate to knowledge, the lowest one is data, followed by information and the highest level is knowledge (Stenmark, 2002; Belingger, 1997). Data is the raw material of information and it can be a qualitative and quantitative data. A single piece of data has no meaning unless the context is understood. Information is the patterns and relationship in the data and these are point out and discussed. According to Rus et al., (2002) the data is made informative and must be put into a context and linked like data. Knowledge is a multifaceted concept with multi-layered meaning. It is broader than data and information and requires understanding of information. Knowledge also not only contains of information but the relationship of information, its classification and its meta-data (Alavi and Leidner, 2001).

Knowledge can be divided into two classes, explicit and tacit (Polanyi, 1966). Explicit knowledge also known as a codified knowledge, correspond to the knowledge that can be expressed in words and numbers and shared in a form of data, scientific formulae, product specifications, manuals and universal principles (Scarbrough et al.,1999). Explicit knowledge is easier to communicate, transmit and reuse across an organization (Desouza, 2003). Tacit knowledge is highly personal knowledge that is gained through experience and largely influenced by beliefs, perspectives and values embedded in the individual experiences of workers (Argesti, 2000). He added that tacit knowledge is deeply rooted in action, commitment and involved in a specific context. Nonaka (1991) also claimed that tacit knowledge is in the head of individual, conversation, unwritten procedure and process in organizational culture. According to Hosbond and Rtoft (2003) the distinction between tacit and explicit knowledge is very general in software development professional work. Therefore, they proposed a second dimension of knowledge in order to detail up the distinction of knowledge in these area. The two new categories of knowledge are informal and codified knowledge. According to Hansen et al., (1999) informal knowledge is referring to knowledge at the specific condition or occasions and codified knowledge is organized according to the rules, code and convention (Hosbond and Rtoft, 2003).

In software engineering, knowledge can be divided into 4 types of knowledge depending on the set of software engineering activities which knowledge concerns (Lindval et al., 2003; Bharadwaj and Tiwana, 2005):

- *Organizational knowledge* – This concern more on organizational knowledge and the organizational issues such as how to run the software development project, what are the business objectives and other related organizational issues. It can be further up divided into different levels of organizational knowledge such as individual, group and inter-organisation (Stenmark, 2002).

- *Managerial knowledge* – This concern more on project planning, staffing and management.
- *Technical knowledge* – This refers to development knowledge and skills such as programming skill, using specific tools and methods, technical writing and all related technical knowledge in software engineering.
- *Domain knowledge* – This relates to application domain and specific system to which the software pertains.

3.4.3 Individual Learning

Individuals (e.g. software developers) are the most important actors who perform tasks for achieving goals that been set by the organizational level. Through social and collaborative among the people in an organization, process knowledge is created, shared, amplified, enlarged and justified on organizational setting (Nonaka, 1994). According to Argyris (1977), "*the individual learning activities, in turn, are facilitated or inhibited by an ecological system of factors that may be called an organizational learning*". Weick (1995) added that knowledge is about action-outcome and the effects of the firm environment. Therefore either individual or organizational, knowledge is created through a conversion between tacit knowledge and explicit knowledge (Nonaka, 1994). There are four phases in knowledge creation as explained by Socialization, Externalization, Combination and Internalization (SECI) model (Nonaka and Takeuchi, 1995). The phases of SECI models are as below:

- *Socialization* (from tacit to tacit) - the conversion of tacit knowledge through interaction between individual through social interaction and shared experience among organizational members.
- *Externalization* (tacit to explicit) - the expression of tacit knowledge and its translation into comprehensible forms that can be understood by others. This can be made through communication, documentation, drawing, teaching or giving presentation.
- *Combination* (explicit to explicit) - refers to the creation of new explicit knowledge by merging, categorizing, reclassifying and synthesizing the existing explicit knowledge (Alavi and Leinder, 2001). Combining previous knowledge

that has been received from others and merger with individual experience and understanding creates a new knowledge.

- *Internalization* (explicit to tacit) - refers to creation of tacit knowledge from explicit knowledge. Internalization is the process of understanding the information putting with individuals own existing knowledge and therefore convert the information into knowledge.

3.4.4 Knowledge Creation and Sharing

Knowledge creation is a continuous process whereby individuals and groups within the organization and between the organizations share tacit and explicit knowledge. Bloodgood and Salisbury, (2001); Nonaka et al., (2000) and Nonaka and Toyama, (2003) added that it is necessary to the organisation to keep creating and updating their knowledge continuously. Organization capability to create knowledge is the most important source of organization sustainable competitive advantage (Nonaka et al., 2000; Parent et al., 2000). Nonaka and Takeuchi (1995) purposed that organizational knowledge ongoing creation process involves four modes of knowledge creation processes: socialization, externalisation, combination and internalisation. According to Bergman et al., (2004) knowledge creation process is started when an individual recognize the related and useful data and information and then able to transform it into a new knowledge that brings a future value to an organization. In term of organization, knowledge creation process need to facilitate and manage in a structured way to enhance knowledge creation within and between an organization (Gupta and Govindarajan, 2000; Nonaka et al., 2000).

Organizational knowledge is not only created within the organization but also can be acquired externally and this can be done through knowledge sharing. The knowledge sharing process may be viewed from several perspectives. Grant (1996) refers knowledge sharing as the transferability of knowledge between firms and within firms. Nonaka and Takeuchi (1995) explain the socialization and external process that transfer knowledge to another organizational member. Awazu, (2003) defines knowledge sharing as activities of transferring or disseminating tacit and explicit knowledge from one person, group or organization to another. However, Nonaka and Takeuchi (1995) claimed that tacit knowledge is personal, context specific and therefore hard to formalize and communicate. While explicit knowledge can be described as knowledge that is transmittable in formal and systematic language. Polanyi (1966) comments that the only way to learn tacit knowledge is through apprenticeship and experience.

The important of knowledge sharing and knowledge creation in any organization will help organization to continuously innovate and help organization to sustain their competitiveness (Rhodes et al., 2008). These activities are usually facilities by a social network within an organization and through the development between departments in an organization link. This relationship among individual and business unit in organization must be established before the process of knowledge creation and sharing can be happened (Szulanski, 1996). In addition Turner and Makhija, (2006) added that in sharing and creating a knowledge, trust plays an important role in how individual transferring and sharing their knowledge with others; organization controls which are use to manage knowledge can significant influence on how this situation will happened.

3.4.5 Knowledge Atrophy

Knowledge is vital for every organization because it is needed to perform a work in organization and when necessary need to change them. According to Hendricks and Vriens (1999) an organization cannot survive and sustained their competitiveness without knowledge. Therefore knowledge needs to be managed to ensure that the right knowledge gets into the right place and so increases the innovation power of organization and its knowledge worker.

Similar to software process as discussed in section 3.3.7, knowledge in organization also will be eroding over the time and will contribute to loss of knowledge in organization. The erosion or atrophy of knowledge is often implicit and its loss is often not recognizing until too late. According to Shaw et al., (2003) knowledge erosion is referred as the loss of knowledge resulting from people leaving an organization or changing jobs within it. Several author claimed that knowledge erosion became one of the main problems as the organization expanding over the time. Shaw et al., (2003) argue that loss of knowledge, expertise and people are among the negative impact in KM when the organization is expanding in the future. According to Lintern et al., (2002), the problem of knowledge erosion has become pervasive in modern organization where there are tendency towards modularisation of expertise and lean staffing. This issues supported by Bjornson and Dingsoyr (2005) in their study on small companies, claimed that the lacking of resource and time in small company in implement KM will introduce a knowledge erosion situation through employee retirement and resignation.

Limited staff exposure and departure of key staff members (through resignation, retirement, and organization downsizing) leaving the organization with knowledge gap is a main factor which contributes to knowledge erosion (Lintern, 2002; Mullin, 1999; Probst et al., 1999; Bjornson and Dingsoyr, 2005; Shaw et al., 2003). This situation also will costs to the organization are even greater because it take ages to combine all the education, training and experience received by the employees become one (Hall et al., 2008). In addition this scenario could weaken the competence and competitiveness of the organization, and also can have a serious impact on organization culture, norms and value in a time period (Xu and Quaddud, 2005).

In such an environment it is vital to share and maintain knowledge to prevent knowledge erosion. Several literatures propose methods in preventing from knowledge erosion. According to Bjornson and Dingsoyr, (2005) mentoring program can effect in leveraging personal knowledge and sharing between projects. Armour and Gupta (1999) argue that mentoring can often be more effective than training and written documentation. In the relation with mentoring program, Singh (2008) stressed that the role of leadership must be taken into consideration for those program to success. He added that leadership style in software organization has been directive in nature where the utmost concern is closely regulating job behaviour of fellow knowledge workers. Anquentil et al., (2007) proposed two techniques in handling knowledge erosion; knowledge organization via ontology and capturing knowledge via post mortem analysis method. Ontology is an explicit specification of a simplified, abstract, view some domain that we want to describe, discuss and study (Gruber, 1995). Post mortem analysis is an analysis that gathering all participants view, respond and report from project that is on going or just finished (Stalhane et al., 2003). Shaw et al.,(2003) in their research on maintaining KM practices reported that there are four criteria that need to improve in sustaining KM practices in an organization; the staff development, team building, communication of role and function, and formal continuous process improvement. Hansen and Kautz (2004) suggested the adoption of knowledge map technique in software process improvement could help in identifying and analysing knowledge flow in an organization. Lintern et al., (2002) believed that the most effective way to create and maintain organizational knowledge primarily through collaborative and social processes.

3.4.6 KM and SPI

In nowadays business environment where software development project is becoming more complex, the greater reliance upon the knowledge processes to resolve problems are really important. Several researchers reported that software development process have always been knowledge intensive (Aurum et al., 2003; Dingdoyr et al., 2005; Robillard, 1999). Bjornson and Dingsoyr (2005) stated in their review that proper managing of organizational knowledge is important in SPI efforts and it is a major factor for success. Mathiassen and Pourkomeylian (2003) in their survey on practical usage of KM to support innovation in a software organization claims that KM and SPI are very close related. They added that KM is used to update practices within software organization generally and SPI specifically. These situations have lead to considerable interest in how organization can effectively respond to changing environment or agile environment. Aaen et al., (2007) supported that to be successful in agile situation, software organization must have agility characteristics. Organization agility characteristic is referred as "*the ability to manage and apply knowledge effectively, so that an organization has the potential to thrive in a continuously changing and unpredictable business environment*" (Dove, 2001). This definition required software organization to have good response ability and also proper KM process. Sirvio et al., (2002) claimed in order to succeed in the future, software organizations must manage and use knowledge more effectively at individual, team and organizational level. As such according to Kess and Haapasalo (2002) software process are essentially knowledge processes and structured within a KM framework. Therefore KM has been used within software organization in general and software process initiative in particular.

Software process is not standardized in all software projects and, it must be updated and improved frequently in order to cope with any environment changes (Borges and Falbo, 2003). Several researchers have stated how to improve and update the software process. According to Hansen and Kautz, (2004) the lessons learned during the project process definition should be shared with and an effective KM is required in supporting software process definition and activities (Sirvio et al., 2002). Hansen and Kautz (2004) added that SPI could strengthen KM abilities for software development organization. Meehan and Richardson (2002) in their investigation on software process in three small software development companies argued that KM is core to software process improvement model. They argued that SPI model support all information about process that have been defined, standardize and be available to all employees. This process needs KM activities such as knowledge creation, storing and sharing.

Mathiassen et al., (2002) and, Bjornson and Dingsoyr (2008) emphasized that knowledge creation and sharing are among the important principles that must be adopted by an organization in order to succeed in SPI. They argue that the relationship between SPI and organizational learning are very strong. Arent and Nobjerg (2000) added that in learning perspective, the knowledge gained in the improvement effort must become embedded in the individuals, groups, and organization. They added that both tacit and explicit were important for improving practice and improvement requires on going interaction between learning process. In addition in term of managing project team knowledge, Kettunen (2003) in his study emphasized that with appropriate KM methods, problems that could impact process improvement such as possible lack of competencies, missing work instructions and others imperfect process could be identify. Therefore KM and SPI have strong relationships.

3.5 Software Development Team

This section elaborate in detail the concepts related to software team in software development process. In addition this section also will discuss on the main software development team characteristics and relationship. Furthermore, this section will explain the relationship between software team and SPI and also between software team and KM.

3.5.1 Background

Software developers do not work in isolation. They work together to develop products that they could not develop by working as individual. According to Rosen (2005) software development is a combination of two basic processes; social process and technological process. This is also known as a socio-technological process. Social process is a process related to the human factor and technological process is related to technology. Sawyer and Guinan (1998) argue that software production is more effected by social process rather than technological process. People are not only claimed as the greatest asset in a software organization (Sommerville, 2004) but also critical to software development success (Rosen, 2005). Software is always developed in a group rather on the individual basis (Rosen, 2005) and the basis of every software project is a team (Cohen and Bailey, 1997).

3.5.2 Team

According to Faraj and Sproul (2000), teams are the main resources in accomplishing the task. Katzenbach and Smith (1993) defined a team as "*a small group of complimentary skill, who are committed to a common purpose, performance goal for and approach, for which they hold themselves mutually accountable*". Basically teams are established to solve some problems or to accomplish some task which forms the basis of team mission. According to Barnum (2000), the benefit of creating a team is to increase speed, productivity, problem solving activity and organizational learning. Team which have good characteristics such as goal orientation, interdependence and interpersonal interaction, perception of membership, structured relationship, mutual influence and individual motivation, will give an impact to the organization and the project in order to achieve the project goal. (Levi, 2001)

According to Cohen and Bailey (1997) in an organization, teams can be divided into four different types; work team, parallel team, project team and management team. Work teams are teams that continue producing product or service. This team is usually stable, fulltime and well defined. Parallel teams are teams that consist of different people from different back ground that pull together in order to solve or improve certain problems. Project teams are teams that have been established for a specific project only in a short term. Project teams just produce one product only at one time and not repetitive nature of project. It also used all the knowledge, judgment and expertise of the experts to solve or improve the project or product. Management teams are responsible for the overall performance of the business activities. They provide guidelines, direction, and control over the entire sub-business unit.

3.5.3 Team Characteristics

According to Demirors et al., (1997), the effective team characteristic can be divided into four main characteristics; size, structure, composition and processes.

- *Team Size* - In general small teams are believed to function better rather than a large team. According to Demirors et. al., (1997), in a small team, issues like lack of communication, less attention, lack of commitment or low motivation are not obvious compared to a large team. Gorla and Lam (2004) argue most of the research in software development teams has focused on large software team rather than small team and the result is not applicable to small software development team. Von Wangenheim et al., (2003) points out that, most of software products are produced world wide by small and medium sizes enterprise (SMEs) ranging from 1 to about 50 employees. According to Staples and Cameron (2005), small teams will create a problem in solving the task because of lack of resources but large teams will need a higher coordination cost. Sheard and Kakabadse (2004) argue that team size is important in ensuring team performance. Hackman (2002) has considered that four to five peoples in a team as an optimum number of people. However, according to Belbin (2000) teams of six members are still considered but after that the contribution to the task will be uneven and more likely will dominate by one person.

- *Team Structure* - Effective team structure will lead to a smooth flow of information which is related to a communication structure and flow of authority which is related to decision making structure (Taylor, 2000). According to Faraj and Sambamurthy (2006), an effective team structure will bring all the important skills to software organization such as the business and the process knowledge, required design skills and programming skills. Wagner (2000) outlines that team structure in an organization can be segregate into two structural dimensions; centralize and decentralize teams which have different levels in term of rights and decision making. In a small scale, team structure in an organization can be divided in to three subcategories; functional team (Johnson et al., 2006), self-directed team (Janz, 1999) and cross function team (Desanctis and Jackson, 1994). Functional team structures have hierarchical structure where each team member is a specialist or expertise on its own and not shares with other team member (Johnson et al., 2006; Robinson, 1984). Self-directed teams have a flat structure and are responsible for the whole product and service (Albert and Fetzer, 2005). Cross functional teams have a flat structure, contains group of people working toward a common goal and made of people with different functional expertise (Desanctis and Jackson, 1994). Walczak (2005) urged cross functional team will enhance the knowledge sharing activities by transfering their knowledge to the original functional areas for improvement.

- *Team Composition* - According to Demirors et al., (1997) the collective of the behavioural factors such as knowledge, experience, skills and problem solving strategy will reflect a team composition in an organization. Martin et al., (2004) added these behavioural factors will impact the team's ability to carry out the task. Gorla and Lam (2004) investigated social interaction, as measured the degree of extroversion on the Myers-Briggs type indicator (MTBI), argue that personality composition and social interaction was strongly related to team performance that can affect project performance (Ryan, 2005). Guinan et al., (1998) whose studied of

66 software development teams in one site at the requirement stage, proposed two factors that affected team performance; behavioural and technical factors. Behavioural factors dealing with experience spread, team skill and managerial involvement. Meanwhile technical factors covered the structured method, production technology and coordination technology. Experience spread is important in team especially in decentralized team. According to Kiesler (1978) decentralized teams could give a negative impact to the internal communication, cohesion and coordination. While Guinan et al., (1998) suggested that distributed team would help in problem solving if been managed carefully. They added that direct managerial involvement will increase team performance. Therefore a stable team composition in software development are require because if the turn over of staff very high it will impact the performance of the project (Umarji and Seaman, 2005).

- *Team Process* - The success of a project depends not only on the technical aspects but also on how effective an individual function in a team process as a whole. Effective teams are observed to perform individual members and accomplished their goals to the satisfaction of all involved (Demirors et al., 1997). According to Barnum (2000) effective team started with team building process that to go through four main stages namely forming, storming, norming and performing. Morever Demirors et al., (1997) stated team process reflects the way team task is accomplished. Team process is related to activities, goal and producing high quality products. Hackman (1987) added that team goal must be clear, measurable and accomplishable. All team members that want to join the team must dependent each others and have same understood of the team goal (Urdaneta, 1999). Team interdependency in performing a team task is a critical factor in team process (Hackman, 1987). According to Marks (2004) with team interdependency all team member have equal roles and responsibility in ensuring quality product. The role of an individual in the project team must be clearly defined in project team. Ayman (2000) defined team roles as "*an expectation of each team member based on their position and function*". According to Belbin (2000) team roles can be divided into functional, professional and work roles. Staples and Cameron (2005) argue that equally responsibility among the team member will leads to a good coordination in team work and clear communication among team member (Salas et al. 2000). Faraj and Sproul (2000) argue that team process can facilitate the social mediation of initially unshared knowledge.

- *Team Communication* - Beside four main issues above, one main focus issue that has been discussed in detail in team development issues is communication. In the team process; team communication plays an important role in exchanging information. According to Ayman (2000) team communication involved verbal and non verbal communication. The content, the medium and sequence of communication determine the process of the team dynamics. The number of one way communication links is *n*(n-1)*, where *n* is a group size. In a group with 7 or 8 members some will rarely communicate with one another (Sommerville, 2004). Kraut and Streeter (1995) investigated communication and coordination in 64 projects in a large software company in the United States. These authors outline four interacting factors that effect communication and coordination in software development. The factors were scale, uncertainty, and interdependence and communication type. According to Ayman (2002), team process involved in three stages; input, process and output. Input defined as "*team factors that present in the beginning*", process is defined as "*factors that impact the team members' interaction*" and output is defined as "*the result of the process*". In the perfect condition a quality input will produce a quality out but sometime the output is not meet the standard because of process loss (Guzzo and Shea,1992). According to Wittenbaum (2003), process loss can be define as "*a situation where a group under-performs relatives to potential performance*". Fiore et al., (2003) added that process loss is the situation that the team expert cannot produce a right solution to the problems. According to Steiner (1972), process loss could occur in two ways: team member reduce their efforts while working in a team and team member cannot coordinate their task and action properly. Marks (2004) in his study on team monitoring argue that feedback in the form of verbal suggestions and corrective behaviour could help team member in focusing back on track and help avoiding process loss. Fiore et al., (2003) added that a KM is needed in avoiding process loss.

- *Team Dynamics* - Team dynamics are the hidden strengths and weakness that operate in a team between different peoples or groups. According to the dictionary (The Concise Oxford Dictionary, 1998) dynamics has been define as "*the motive forces, physical or moral, affecting behaviour and change in any sphere*". Therefore referred to the above team definition and dynamics definition, team dynamics could be refer as an interaction between team member subject to many forces both internal and external. Scarnati (2001) stated that team dynamics could effect how team reacts, behaves or performs and the effects of team dynamics are often very complex. Ayman (2000) argued that there are various forces which could influence team dynamics including nature of the task, the organizational context and team composition. McCarty (2005) in her dissertation on dynamics of successful software team identified four characteristics of team dynamics; positive, negative, internal and external team dynamics. Positive team dynamics is referred as positive forces that could lead the team to create a high performing successful team. According to Triplet (1998) the present of social relationship in a team could increase team productivity. Katzenbach and Smith (1993) found that involvement as a team could enhance social and interpersonal skill. Levi (2001) added that working with different skill an expertise people in a team could also enhance the employee skills. Gorla and Lam (2004) argue that social interaction skill dimension can divide a team member to extrovert or introvert. Extroverts' team member is a people oriented, sociable and enjoys interaction with others. While introvert person is a type of people who like to work alone and less social interaction. Meanwhile, Singh (2008) believed that the positive mode of leadership (such as well focus directive, well plan and others) in software organization could enhance the positive team dynamics.

Negative team dynamics is referred as negative forces that could lead the decrease of team performance and preventing people from contributes with their full potential (McCarty, 2005). According to Sommerville (2004), from management point of view, in software development organization people are required three types of needs that have to be fulfilled and satisfied; social, self esteem and self realisation needs. Social needs are related to social interaction and communication, self esteem is referring to recognition and reward given by the organization and self relations needs are related to task responsibility and type of work that could broaden people skills. The lack or ignorance of these needs will give a negative impact to organization because people feel unsecured, low job satisfaction and decrease their motivation (Sarma and Van, 2004). These will stop them from giving a full commitment and cooperate in their work as a team.

Internal team dynamics are referring to the forces that exist within the team itself (McCarty, 2005). According to Furumo and Pearson (2006), team member will not cooperate if they do not feel that that are a part of the team. Ayman (2000) argued that within a team, roles and norms must be clear. Roles are referring to position and expectation of the team member. Norms referring to codes of conduct that been agreed by the team members directly or indirectly. Levi (2001) found that internal social interaction between people could build team cohesion. Carron (1982) has defined team cohesion as "*a dynamic process which is reflected in the tendency or a group to stick together and remain united in the pursuit of its goal and objectives*". Littlepage et al., (1989) argued that cohesiveness is essential for an effective team performance. Shaw (1981) claimed that communication between team members will enhance in cohesive team. According to Gist et al., (1987) a cohesive team will freely challenge each others and easily sharing new knowledge with other team members. Beside that an organization also could increase their performance with the present of cohesive team (Kozlowski and Ilgen, 2006). Therefore these team processes are important to improve team learning and more effective team performance (Hackman and Morris, 1975).

External team dynamics are referring to the present of external forces that beyond the team control and could impact the team performance (McCarty, 2005). According to Kirkman et al., (2004) the intrinsic and extrinsic factors in projects may motivate team. Intrinsic factors are the internal factors that consist in the task and team activity itself. Extrinsic factors are external factors that influence team from the outside such as reward and recognition, feedback from the organization and customer, team member pressure and the working environments. Kayworth and Leinder (2001) argued that the physical present of the leader also could motivate team extrinsically. Lee and Miller (2004) added that change of policies and strategies during project development could impact schedule and cost directly and people in project indirectly. In addition a better working environment also could enhance job satisfaction among team member (Javed et al., 2004).

3.5.4 SPI and Team

Software processes are related to software development and is very dependent on people in making decision and judgement. According to Rosen (2005) in software development, human factors are not the only important to be consider in the process but they are also a determiner in project success. Fuggetta (2000) argued that software development is not just creating an effective programming and tools, but also depends on people, organization and procedure. He added that software development also is carried out of teams of people involved in a highly creative activity. Stelzer and Mellis (1998) added that people involvement in improvement activities is important because employees must adopt process innovation in their day to day activities. Dyba (2005) in his quantitative survey of 120 software organization stated that employee participation and the way people are treated are the important factors in organization development and change. He found that employee participation is the strongest influence on SPI success. The lack of involvement will disturb the improvement process because if employee did not commit themselves to all the propose change activities, the aim of process improvement will be fail (Oestreich and Webb, 1995). Stelzer and Mellis (1998) added that the strengths and weaknesses of the current process are inside the staff hands and knowledge. Dyba (2005) argue that peoples are the main factor in software process improvement that needs to be encouraged and support in an organization. Moreover, according to Beaver and Schiavone (2006) in their analysis of the effect of software development team on the software product quality claimed that even though people are the main driver for software quality but the processes have been given more attention. Therefore the involvement and full commitment from teams in process improvement is critical in all time.

Hall et al., (2007) argue that the dynamic performance of software project which involved many processes is always depend on team especially in quality of communication within team and between teams. They added that the communication can be applied in many ways not only in verbal but also in term of documentation form such as version control, guidelines, reports and many more. Phongpaibul and Boehm (2005) points out that the level of communication in software process was depended on the size of software project. They claimed that for a small project the interaction between team members is adequate but for a larger project a mix interaction between team member and specification are required. Hoegl and Proserpio (2004) argued the communication also has a related impact with the team proximity. They argue that the increase distance from one team to another could effected the team dynamics in which it will interrupt team communication, coordination, mutual support, effort and cohesion. Steiner (1966) also claimed that the link between team member also becoming more difficult with the increase of the team member and this will impact the team dynamics. Brodbeck (2001) in his research of 21 software development teams shows that that they are relation between internal-task related communication and performance of software projects.

Therefore in order to be success in SPI, organization must have a solid support from the software development and management team. The development and management team must be able to work together, share the knowledge and able to communication one another effectively. This is because the essence of software development is good relationship, effective communication and high esteem of teamwork among software development and management team.

3.5.5 Team and KM

Software development teams do not work in isolation and always need to work in a cross functional environment. Aurum et al., (2008) argue that in software development environment effective knowledge sharing among software developer is a critical factor and KM becomes an enabler of organizational learning. According to Politis et al., (2003), KM practices including acquisition, creation, sharing and exploitation are combination process between human resource management and information management. He added that human resources are the important factors for effective KM. Sapsed et al., (2002) argue the dynamic interaction between the team members in an organization is important in knowledge creation process. This is important in software development where with systematic knowledge creation process could supply correct information and knowledge to development team in order to complete a task or making a decision (Rus and Lindvall, 2002)

In the same situation, Ramesh and Tiwana (1999) believed KM could help in managing all relevant information and knowledge that the team members contribute especially in a new product development. It is also believed that a proper KM process could help a team when facing process erosion (Fiore et al., 2003) and knowledge atrophy problems (Ramesh and Tiwana, 1999). Salas et al., (2000) argued that in a team which have a high turn over, updating and sharing the knowledge with others is very important. This knowledge will forms a basis to complete task in any given situation and team can reuse this knowledge for subsequent project or solving the project problems. Rhodes et al., (2008) added that cross function teams may facilitate the formulation of a knowledge map for employee to use to find the appropriate knowledge. In term of knowledge sharing, Rus and Lindvall (2002) argues that by having an efficient knowledge sharing process such as systematic post mortem, could enhance team experience, identify potential improvement need and increase job satisfaction. Newell et al., (2004) added that the levels of knowledge sharing within the projects groups are depended on the strength of the cohesiveness between team members. According to Kozlowski and Ilgen (2006) a high strength of team cohesion will improve the team performance where members have to coordinate their knowledge, skill and effort in complex and highly interdependent workflows.

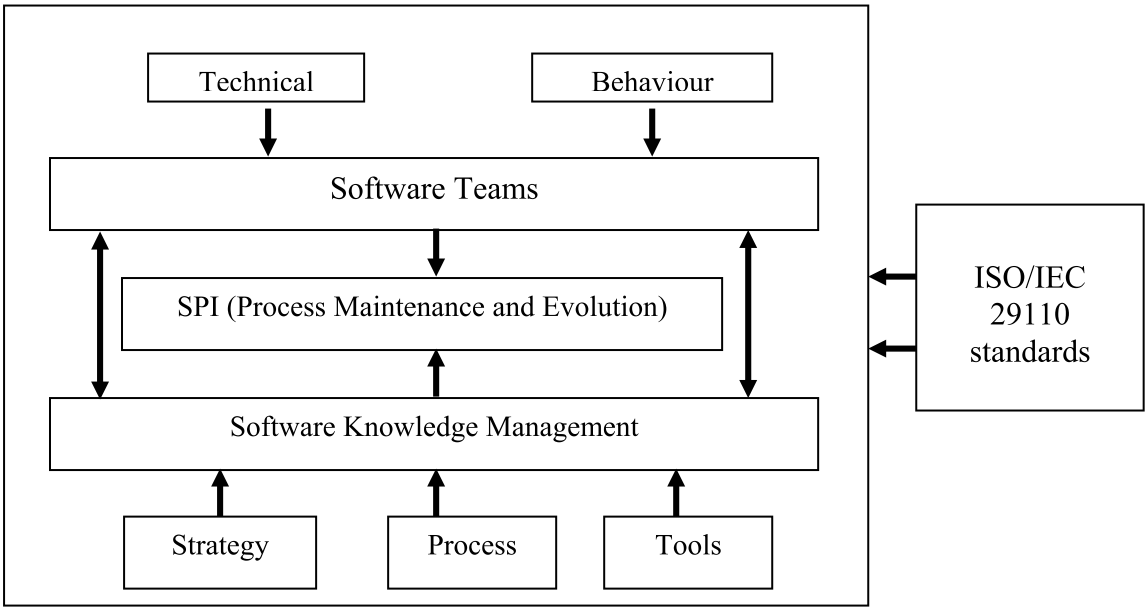
According to Hall and Goody (2007) culture is the main determinant of the success of KM especially in knowledge sharing within and between team in promoting KM in an organization. Plesssis (2007) added team which was practicing KM as a team culture will easy to share knowledge, collaborative relationship and personal responsible in creating and sharing knowledge. Haven and Knapp (1999) supported that KM could reshape the attitudes and behaviour of people in order to ensure that both personal and organizational knowledge are always available. Several researchers also argue that the performance of the organization is dependent to the value of culture that widely shared among the team members (Knapp, 1998; Kotter and Haskett, 1992; Denison 1990).

Therefore, from the discussion above, it can be view that KM and software development team are much related each others. The proper KM process helped software development team in maintaining the development knowledge in an organization and help in prevent knowledge loss problem. In addition, an organization also could ensure the progress of software development activities are always on the right track and team are being update with related development knowledge especially in software development process.

3.6 Research Context

From the discussions in the above sections, a research model for this study has been depicted as in figure 3.6. The figure, illustrates how two main factors; teams and KM, their characteristics and relationship with SPI. The diagram shows that it is important for the software organization to have a KM process with a clear strategy, right process and correct tools. Researchers believed that with an organised knowledge process, a software process will always be updated and improved frequently in order to cope with any environment changes. Meanwhile, an organised software development team that have a balance characteristic between the technical and behaviour will help software organization in creating an effective software development team and team dynamics. These characteristics will encourage team involvement and give a full commitment in retaining, maintaining and evolving software organization software process. From the diagram, it is also shown that the software teams and software KM are related to each other. This relationship is vital in preventing knowledge atrophy and process erosion problems. According to Salas et al., (2000) effective and dynamics team could enhance knowledge in improvement process and, with a proper KM process could help teams become more effective in performing team task and making a decision (Kettunen, 2003). Aean et al., (2002) added with an appropriate knowledge creation and sharing process could provide team members with clear SPI goals and sustain their interest. Lastly, this research study propose that the role ISO/IEC 29110 is very significant in this research due to the main objective of these standard is to assist VSEs in assessing and improving their software process. Researchers believe with the clear guidelines and procedures that have been created in the standard have a potential to encourage VSEs seriously in improving their current software process.

Figure 3.6: The Detailed Research Model



3.7 Research Hypothesis

From the above discussion, the current research variables and its relationship towards the SPI have been identified, explained and elaborated. The explanations have indicated that all variables are related and supported each other especially in the issues of process maintenance and evolution. In addition, the issue of software quality standard also indicated that the new ISO standard, ISO/IEC 29110, would positively encourage VSEs in seriously improve, maintain and evolve their software development process. Therefore the research hypothesis for this study has been established as below:

H1: There is a positive relationship between the organised management of software development knowledge, and software process and process improvement in Irish software VSEs.

H2: There is a positive relationship between organized software development teams, and software process and process improvement in Irish software VSEs.

H3: The organization of software development knowledge and software development teams will assist to prevent Irish Software VSEs from process atrophy and knowledge entropy problems in software process.

H4: The adoption of the proposed standard ISO/IEC 29110 could positively encourage Irish Software VSEs in improving their current software development process

3.8 Summary

This chapter examined the role of software process, SPI, KM and software team in software development. The important of process maintenance and evolution and the relationship of software process, KM and team were explained and analysed. The overall research framework and research hypothesis also have been explained and presented. The next chapter presents the research methodology chosen for the study and the reason for its selection.

Chapter 4 RESEARCH METHODOLOGY

4.1 Introduction

This chapter presents an overview of the various philosophies of research and the role of quantitative and qualitative research methods. It explains a number of various quantitative and qualitative methodologies, and its applications in software engineering research. A detailed explanation and discussion of present study research method and its justification also will be elaborated in this chapter. Furthermore, this chapter also presents the overall research design and processes that have been followed in this study. Lastly, the data collection and analysis processes are explained and discussed.

4.2 Research Philosophy Paradigm

All research is based on some underlying assumptions about what constitutes valid research and which methods are appropriate. The most pertinent assumption is those which related the underlying epistemology which guides the research. According to Oates (2006), epistemology is an assumption about knowledge and how it can be obtained. Three categories that based on the underlying research epistemology are: Positivists, Interpretivist and Critical.

- *Positivists* - Positivists research are the oldest of the three paradigms in research philosophy (Oates, 2006). According to Levin (1988), positivists see that the reality is in order and it can examine and explain from an objective perspective. Easterby-Smith et al., (1991) argue that positivist view the world is in ordered, regular and can investigate its objective. They also belief that observer is not part of what it is research and science is not driven by human interest.
- *Interpretivist* – Interpretivist research is generally attempt to thorough understand how people perceived their world. According to Oates (2006) and Kaplan and Maxwel (1994) interpretive approach are more focused on identify, explore and explain the full complicity of human sense making as the situation emerges rather than defined hypothesis. They added that interpretive researchers aims for the plausibility rather than proof as in positivists. According to Goulding

(2002) both of the positivist and interpretivist have their own advantages in generating valid and valuable knowledge in research process. Chen and Hirschheim (2004) argue that positivist paradigm is more on generalisability on the previous research while interpretive paradigm could enhance the depth of understanding of the phenomenon in question.

- *Critical research* **–** Critical research focuses on the opposition conflicts and contradiction in social aspects, finds a way to solve and release a problem (Orlikowski et al., 1991; Galliers, 1991; Bryman, 2001). Oates (2006) argues that critical research is more concerned with '*identifying power relation, conflicts and contradictions and empowering people to eliminate them as sources of alienation and domination*'. Chua (1986) claims, in critical research, the main task are more on social critique which focusing on the restrictive and alienation condition of status quo.

4.3 Research Method and Methodology

The uses of research method and research methodology terms have been used very frequently and interchangeably in literatures, so it is useful to understand the differences between these two terms. According to Mingers (2001) research methods are related to the way or techniques to collect the data such as survey and interview, while research methodologies refer to the overall structured set of guidelines or activities that support the research activities in order to generating valid and reliable research result. According to Creswell (2003), quantitative and qualitative are terms that have been used to differentiate the research paradigm. Quantitative methods are based on the positivism paradigm and qualitative methods are based on an interpritivism paradigm (Firestone, 1987). Gallier (1991) has categorized research methodologies and group them according to the positivist and interpretivist paradigms which related to the quantitative and qualitative research methodologies as shown in table 4.1:

Table 4.1: Taxonomy of Research Methodologies

Positivist	Interpretivist
Lab Experiments	Subjective/Argumentative
Field Experiments	Reviews
Surveys	Action Research
Case Studies	Case Studies
Theorem Proof	Descriptive/Interpretive
Forecasting	Future Research
Simulation	Role/ Game Playing.

4.4 Quantitative Research Methods

The quantitative research method is the systematic scientific investigation of quantitative properties and phenomena, and their relationship. According to Creswell (2003), quantitative research generally designed to collect data in a form suitable for statistical analysis and should also be objective, non-reactive, representative, and collected using standard measures. Punch (1998) added that quantitative research is an empirical research where data are informed by numbers. Myers (1997) argues that the quantitative research method widely used in natural/social science and the goal of these methods is to determine whether the generalisation of a theory hold true. Bryman (2004) has interpreted quantitative research as research that emphasize on the deductive approach, more on natural scientific model, positivists research and view of social reality as an external objective reality.

Furthermore according to Creswell (2003), quantitative research design can be divided into three main types, experimental, quasi-experimental and, descriptive and correlational. Experimental and quasi experimental are designed to study cause and effect. Descriptive and correlational study the research variable in the natural setting without imposes any treatments. Bryman (1992) added there are five ways collecting data in quantitative research method:

- Social Survey which have a capacity for generating quantifiable data on large number of people who are known to be a wider population in order to test theories or hypothesis.
- Experiments which involve at least two groups; the control group and uncontrolled group.

- Analysis of previously collected data.
- Structured observation where the researchers record all the observation data in accordance within the time given and measured the resulting data.
- Content analysis which performing quantitative analysis or the content of media.

Burns (2000) added that quantitative research has four main characteristics:-

- Control - Control is implemented in experiment in order to provide definite and solid answer to the research question.
- Operation definition - The description of how the variable will be measured in the study.
- Replication - To repeat the study with a similar population using a similar methodology with the same results would be obtain.
- Hypothesis Testing – A process on how researchers create research hypothesis systematically and then prove or disprove it subject to an empirical test.

According to Locke et al., (1998) quantitative and qualitative researches are different in paradigms due to the different assumption about the nature world, truth and function of research. Myers (1997) claimed that the advantages of quantitative methods are more on data precisions and control, which associates with positivist research. However, quantitative method limitation are obvious when the research focusing on social, cultural and human behaviour where it is more on qualitative method (Denzin et al., 2000).

Creswell (1994) has generalised that quantitative research has a different process compare to qualitative research. Quantitative research is an investigation into an identified problem, based on the testing theory, measured and analyzed using number and statistical techniques. The main objective of this method is to determine whether the predictive generalisation of a theory hold true. While qualitative research is more on process to understand a social or human problem from many different perspectives. It has been run in a natural setting and involves building a process of building a complex and holistic picture of the research situation.

Moreover the validity and reliability issues need to stress in the quantitative research methods. According to Golafshani (2003) validity and reliability are the key characteristics in quantitative research that refer to quality and rigour in design. Validity is referred to the accuracy and truth of the data and finding that are produced (Joppe, 2000). He added that validity is refers to the concepts that being investigated such as people or object that are being studied, methods that have adopted in collected data and finding that are produced. Furthermore, the validity process could be divided into internal and external validity (Basham et al., 2009). Meanwhile a reliability issue is referred to consistency and dependability of a measuring instrument (Joppe, 2000). Kirk and Miller (1986) added that reliability is an essential pre-requisite for validity and have identified 3 types of reliability in quantitative research ; (1) the degree to which a measurement, given repeatedly, remains the same, (2) the stability of a measure over time and (3) the similarity of measurements within a given time period. Furthermore among other techniques for measuring the reliability of the quantitative data are the Cronbach's alpha, Spearman-Brown formula, Kendall's tau and Cohen's Kappa (Basham et al., 2009).

4.5 Qualitative Research Methods

Qualitative research established itself as a method of inquiry for the study of human and group life. According to Patton (1990) qualitative method have detail information on human, social and cultural phenomena that could reduce generalisability as in quantitative method. Blaxter et al., (2002) added that qualitative research is more concentrated at collecting and analyzing the non numeric data, with the focus on exploring information in depth rather than breath. Qualitative researchers also deploy a wide range of interconnected interpretive practice hoping to get a better understanding of the subject matter at hand (Denzin et al., 2000). They added that the word 'qualitative' itself means the qualities of entities and the process. It also reflects that qualitative approaches are more emphasize word and not experimentally examined or measure in term of quality, amount, intensity or frequency (Bryman, 2004; Denzin et al., 2000). Bryman (2004) also argue that qualitative research method is more an inductive approach, rejected practice and norm of quantitative and represent social reality viewpoint.

Excluding survey that can be adopted in both quantitative and qualitative research work, there are several number of research methodologies that have been used in qualitative research (Oates, 2006; Creswell, 2003).

- Phenomenology
- Ethnography
- Case studies
- Action Research
- Grounded Theory

4.5.1 Phenomenology

Phenomenology is sometimes considered a philosophical perspective as well as an approach to qualitative methodology. It is involved in social research disciplines including psychology, sociology and social work. Phenomenology may include events, experiences, situation and concept. Phenomenology methodologies focus on people's subjective experiences and interpretations of the world.

4.5.2 Ethnography

Ethnography is a methodology to study about people and culture, which originated in anthropology. As anthropological based, ethnography methodology involves extensive fieldwork and now has move to social science activities. This methodology using multiple data generation method such as interviews, observation, document and the most important is the personal field notes about what researcher see, feel and experience (Oates, 2006; Stahler and Cohen, 2000). According to Patton (1990) the main advantage of ethnographic methods is ability to gain a detailed and depth information from research participant feedback that is not constrained by predetermined categories.

4.5.3 Case Studies

According to Verschuren (2003) both qualitative and quantitative can apply case studies as their research methodologies. Case studies methodologies focus on in depth analysis of one or groups of units such as an organisation, a department, group of companies or an individual person. According to Yin (2003) case study is '*an empirical inquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between phenomenon and context are not clear*'. A variety of data generation method can be used in case studies methodologies such as interviewing, questionnaires, observation or/and document analysis. The aim of case studies is to generate a detailed insight of the case, its complex relationship and processes.

4.5.4 Action Research

Adopting action research means that researchers are trying to generate new knowledge about a social system and at the same time trying to improve and change it. According to O'Brien (2001) action research is a learning and experimental process which involved problem identification, problem solving, solution evaluation and repeated process if the result is not satisfied. The aim of action research is to contribute and improve researcher strategies, practices, and knowledge of the case within which the researcher practice (Gerald and Roger, 1978).

4.5.6 Grounded Theory

Grounded theory (GT) is a qualitative research where the intension is to do the field research and then analyse the data to see what the theories emerges so that the theory is grounded in the field data. GT is an inductive approach where the theory is developing first then the field evaluations are done. According to Straus and Corbin (1998) theory that captured from the data are more helping to resemble what is actually going on compared to series of concept based on speculation or experience. According to Oates (2006) questionnaires, interviews, and observation are the structured form of data generation in GT. The major difference between GT and other methods is its specific approach to theory development.

4.5.7 Mixed-Methods or Triangulation Methods

Mixed-methods or triangulation methods are methods which combine quantitative and qualitative approach are mixed in a single study. It is a third major research paradigm after the quantitative and qualitative research methods. Onwuegbuzie and Johnson (2004) argue that the application of mixed-method could support the weakness existed in quantitative and qualitative research methods. This combination has been referred as triangulation (Denzin, 1970) and mixed-method (Creswell, 2007). According to Onwuegbuzie and Johnson (2004) the goal of this combination is more to expand people understanding rather than to search for corroboration. Seaman (1999) and Mingers (2001) added that understanding and the result of the research are more fruitful of real world compare to the isolation method. Creswell (2007) explained that the implementation of quantitative and qualitative methods involved data collection that divided into two approaches; sequential or concurrent with the priority to one method over the other or having a same and equal status. Two types of data are integrated at several stages in the process of research: data collection, data analysis, data interpretation or combination of research process places. Creswell (2003) also listed six major categories in mixed-methods strategies:

- Sequential exploratory strategies
- Sequential explanatory strategies
- Sequential transformative strategies
- Concurrent triangulation strategy
- Concurrent nested strategy
- Concurrent transformative strategy

4.6 The Present Research Method and Justification

The main method selected for this research study is a mixed-method approach, which is a mix between the qualitative and quantitative research methods. This method according to Creswell (2007) will either first conduct the qualitative research phase of a study or followed by quantitative research or vice versa. The mixed-method can also be use to differentiate types of data, investigator, theories and methodologies (Denzin 1989). It also could be categorised in two main categories, within method triangulation and between method triangulation (Denzin, 1978).

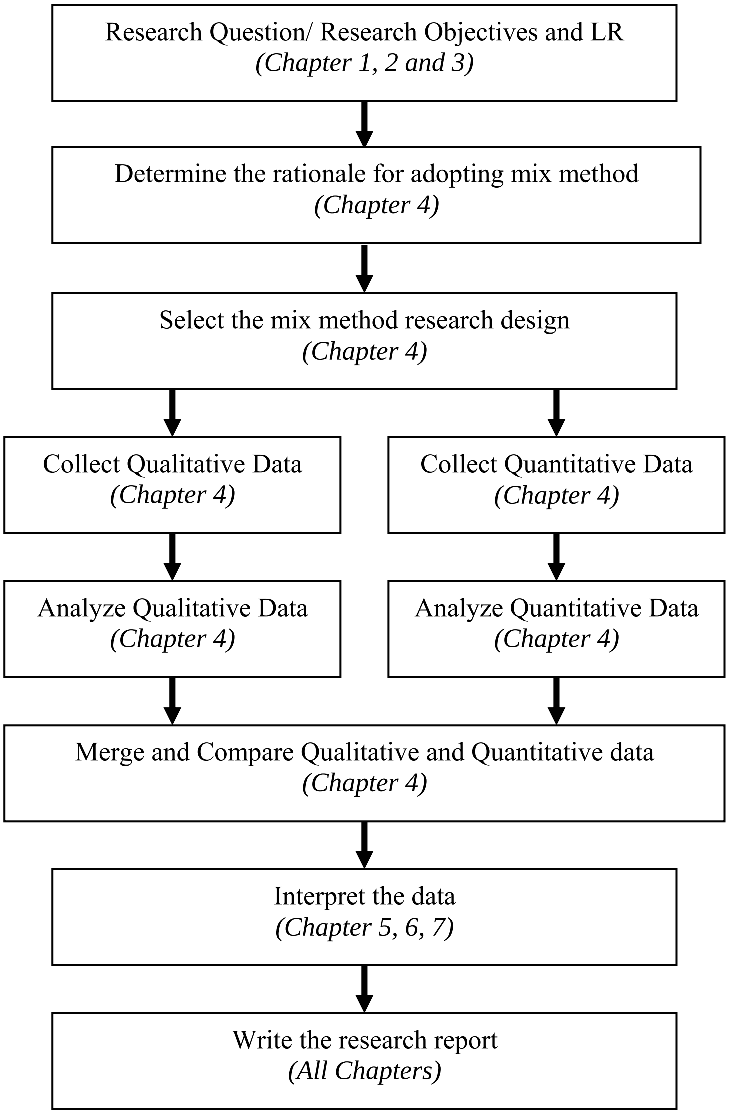
Mathiason (1988) argues that the implementation of mixed-method will help to prevent from biased problems either related to the researcher data sources or methods. Moreover according to Benbasat (1987) using multiple method increases the robustness of the results because finding can be strengthened through triangulation. McGrath (1982) added that the mixed-method provides an additional insight because a problem is approached from differing perspective and introduces both testability and context into the research (Kaplan and Duchon, 1988). Seaman (1999) supported that the mix of the technical and human behavioural aspects in software engineering area provides itself to combining quantitative and qualitative method in order to take advantage of the strengths of both. In addition, several researchers agreed that mixed methods have a lot of advantages compare to others research methods (Jick, 1979; Sieber, 1973; Rosman and Wilson, 1985, Tashakkori and Teddie, 1998). They claimed that mixed-method has a higher confidence result, an effective research design and process, stimulate a creative collection of data collections and gathered rich and detail data. Furthermore these advantages have been categorised into four main characteristics; participant enrichment, instrument fidelity, treatment integrity and significance enhancement (Collin et al., 2006).

The types and categories that adopted for present study is the investigator mixed-method (Denzin, 1989) and the triangulation design (Creswell, 2007). In investigator mixed-method, investigation is more focus on multiple case studies rather than a single case study. This involved both types of research method, quantitative and qualitative. The multiple type of data source also could increase the confidence in research findings and could also make findings that could not be made in using a single data source (Brathal and Jorgesen, 2002). Meanwhile, the adoption of triangulation design approach (Creswell and Clark, 2007) could help researchers in detailed up the present research investigation process. In this approach, researchers adopted the concurrent triangulation strategy (Creswell, 2003), where both qualitative data and quantitative data are collected and analysed almost simultaneously. At the end of data analysis processes as discussed in part 4.7.3, researchers have validated the research results with the all the variables that have been found in the literatures that support our study research model as depicted in figure 3.6. The integration of both data has been done at the interpretation phase, where the overall result from all datasets are interpreted and discussed by corroborate and cross validate the study findings in order to confirm and have thorough understanding of the research results (Creswell and Clark, 2007). According to Morse, (1991) the purpose of this design is "*to obtain different but complementary data on the same topic*". Patton (1990) added that this approach was intended to combine the strength between both methods and bring together non overlapping weakness of both methods. Creswell and Clark (2007) added that several advantages pursuing the concurrent triangulation strategy method such as:

- The design makes intuitive sense especially for researchers who are new to the mixed-methods approach.
- It is an efficient design since this both data types are collected during one phase and roughly at the same time.
- Compared to sequential approached, the concurrent approached will help in improving the data collection and analysis timing in any research (Clark et al, 2008).

Therefore based on the advantages justified from the literature and its suitability with the present research study, researchers have chosen the mixed-method as a research method for the present research study. Figure 4.1 shows the overall present research mixed method process and indicates its associated chapters.

Figure 4.1: Present Research Mix Method Process


Research Question/ Research Objectives and LR
(Chapter 1, 2 and 3)
Determine the rationale for adopting mix method
(Chapter 4)
Select the mix method research design
(Chapter 4)
Collect Qualitative Data
(Chapter 4)
Collect Quantitative Data
(Chapter 4)
Analyze Qualitative Data
(Chapter 4)
Analyze Quantitative Data
(Chapter 4)
Merge and Compare Qualitative and Quantitative data
(Chapter 4)
Interpret the data
(Chapter 5, 6, 7)
Write the research report
(All Chapters)

4.6.1 Case Study Method

The main research method driven this research is the case study research method. According to Verschuren (2003) case study method which apply both qualitative and quantitative research methodologies, is also focused on in-depth analysis of one or groups of units such as an organisation, a department, group of companies or an individual person. Yin (2003) stated that a case study can generate a rich description, theory testing and theory building as similar with the GT research method compare to no other research method. A variety of data generation methods have been applied in present research study, personal and focus group interviews and survey questionnaires. Feagin et al., (1991) pointed out that case studies are multi-perspective analyses. This means that the researcher considers not just the voice and perspective of the actors, but also of the relevant groups of actors and the interaction between them. This aspect is a salient point in the characteristic that case studies possess. They also added that case study is also known as a triangulated research strategy where the triangulation can occur with data, investigators, theories, and even methodologies. Moreover, case studies could generate a detailed insight of the case, its complex relationship and processes (Yin, 2003). Furthermore, Oates (2006) claimed that a case study will help the researcher produce data in a situation where the researcher has little control over the events compared to action research and ethnography.

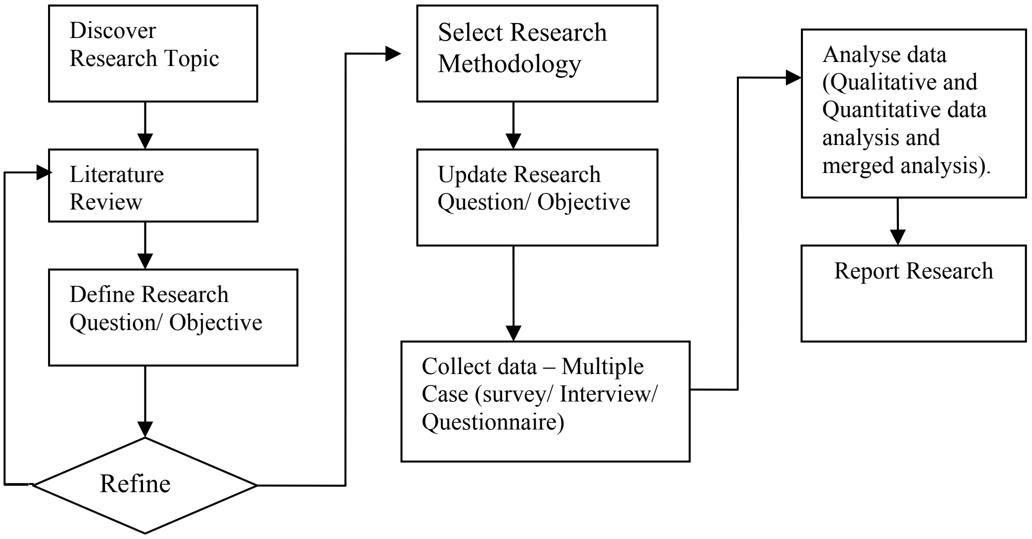
Since the present research aims is to study and evaluate the research situation rather than being involved in solving specifics solution to the situation, the case study method is most suitable in this situation. Yin (1994), added a pure GT method objective is more on developing theory rather than a theory testing. Moreover, the objective of the present study is more focused on creating a detailed description rather than creating a theory, pure GT method is not applicable. Yin (1994) also suggested that every investigation should have a general analytic strategy, so as to guide the decision regarding what will be analysed and for what reason. He presented some possible analytic techniques: pattern-matching, explanation-building, and time-series analysis. Therefore pure GT method is not suitable in this present research but only GT coding process will be use in order to assist researcher in analysing present study data**.**

According to Tashakkori and Teddie (1998), the case study research method helped researchers in understanding of the research study phenomenon more details. They stated that this is due to at the research investigation that can be done in many stages and indirectly helped researcher acquired a richer understanding. Alison and Merali (2007) added that the case studies method is suitable to capture the knowledge and views of the practitioners. This objective is match with the research objective which understands the current situation from the practitioners. Yin (2003) reported that case study design is suitable with a research question that start with ‘how’ and ‘why’ question is being asked about a contemporary set of event over which the investigator has little or no control. Because 'how' and 'why' question are explanatory and usually should be studied over time in replicated case studies or experiments. According to Benbasat et al., (1987) the case study is suitable in software related research which is more focused on organizational factors rather than technical issues. In addition, Sjoberg et al., (2007) argue that in software area and industry, the case study method is useful in evaluating software engineering methods and tools. They argue that case study could help in avoiding scale up problem that happen in experiments method.

4.7 Present Study Research Process

The research process starts with discovering a research topic and is followed by a detailed literature study for the purpose of understanding the scope and research area topic in detail. The outcome from the literature study helped researchers to define the main research question and present research objectives. This process is an iterative process until the final and solid research question and main objective are determined. Other than that, the literature study also assisted researchers to identified all the important variables that will be used as a main subjects for this research and this phase is more an exploratory study phases. The overall present research study processes is depicted in figure 4.2.

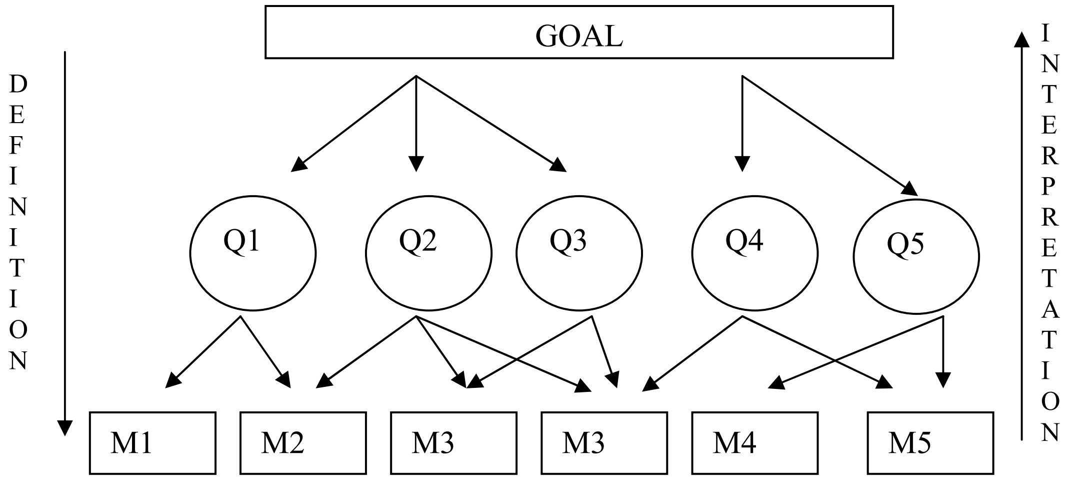
Figure 4.2: Overall Present Research Processes


Discover Research Topic
Literature Review
Define Research Question/ Objective
Refine
Select Research Methodology
Update Research Question/ Objective
Collect data – Multiple Case (survey/ Interview/ Questionnaire)
Analyse data (Qualitative and Quantitative data analysis and merged analysis).
Report Research

4.7.1 Goal, Question and Metric (GQM) Process

A GQM (Goal, Question and Metrics) (Basili et al., 1994) approach has been adopted in present research study with a modification in the last GQM activity. According to Solingen and Berghout (1999), GQM is a systematic approach for integrating goals to models of the software processes. Basili et al., (1994) argue that in GQM information is identified that relevant to solving specific problems (goal) and that can be represent in a practical, applicable and interpretable way. They added that GQM define a specific goal. The goal is refined into questions that usually breakdown to several major categories. Each category will then refine into a metrics that should provide information of these questions. Figure 4.3 illustrate the GQM paradigm (Basili and Weiss, 1984)

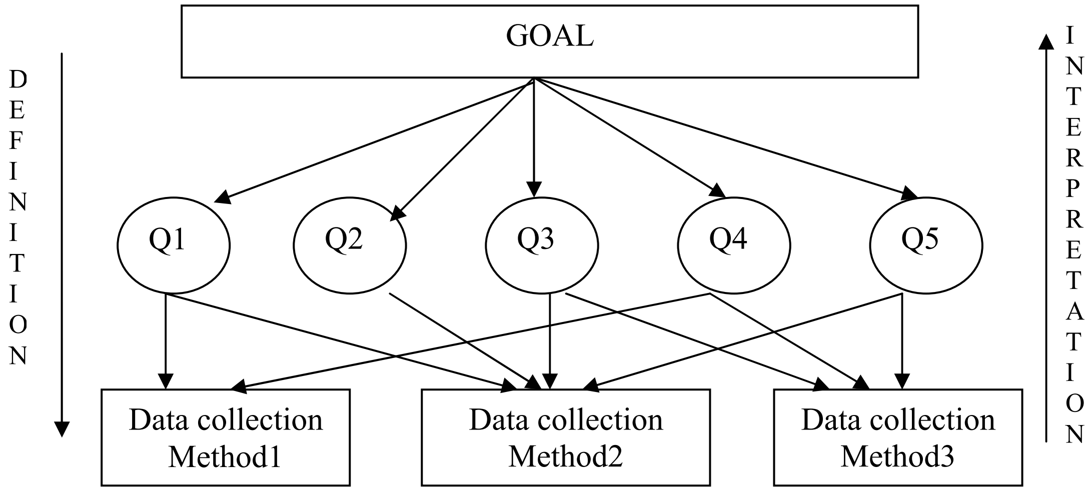
Figure 4.3: GQM Paradigm



In the present research study, the first two main GQM processes are been followed thoroughly. These two processes are the identification of present research goal and the segregation of research questions. The present research goal has been set up after the details of literature and document reviews have been done. Moreover, the research goal has been visualized into several present research objectives. This process followed by interpreted the research goal into a general research question that will drive the whole present research study. In order to be more specific and more systematic, the general research question has been narrowed down into a several small and precise research questions that will guide researcher in running the present research.

The different between present research GQM process and the original GQM process, happened in the third phase of GQM process namely a metric phase. In this present research study the metric phase is replace with a methodology phase. The main purpose of this tailoring process is to help researcher to identify the suitable data collection method for research questions and able to achieve the present research study goal. The present GQM process for this study is as in figure 4.4:-

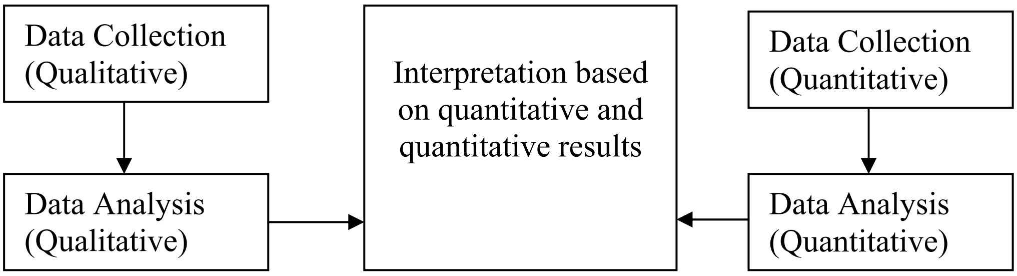
Figure 4.4: Present Research GQM approach



4.7.2 Research Data Collection Process

The present research used a multiple case studies research method. Yin (1994) points out multiple case studies will help researchers prevent from the possibility of misrepresentation and also ensuring validity and reliability of data collection. There are three main methods in data collection process that have been used; face to face interview (interview), focus group interview (focus group) and survey questionnaire. All the data collection processes are done almost simultaneously. For the interview and focus group purposes, researchers have been guide by an interview guide and focus group question guide (Taylor and Bodgan, 1984) which was an output from GQM process. In the same time, survey questionnaires have been mailed to identify respondents. The overall data collection process is shown in figure 4.5

Figure 4.5: Research Concurrent Triangulation Design- Data Collection


Data Collection (Qualitative)
Data Analysis (Qualitative)
Interpretation based on quantitative and quantitative results
Data Collection (Quantitative)
Data Analysis (Quantitative)

In order to help researchers identified suitable companies as research respondents, researchers have browsed Enterprise Ireland website (*www.enterprise-ireland.ie*). The list and information of the software companies in this website were been used as a main reference in searching and identifying suitable research respondents. Beside that several related internet website such as *linked-in* (*www.linkedin.com*), internet search engines such as Yahoo *(www.yahoo.com),* Google (*www.google.com*) and through several IT experts who were contacted to support the identification process.

From the list of the companies being complied and suggested, only companies which have less than 25 employees (ie. compatible with the definition of a VSE) and having software development activities as their main business operation have been selected and evaluated. This process allowed the researchers to list several potential Irish Software VSE companies as our research respondents. In order to be more focused on the research issues, researchers gave higher priorities to companies which have less than 10 employees and this group of companies is then labelled as *Micro-VSE*. This process was then followed by the companies that have more than 10 employees but less than 25 employees and this group of companies is then labelled as *Larger-VSE*. The similar division process also has been made by Laporte et al., in developing a new international standard for VSEs (Laporte et al., 2008). All suitable and potential companies had been emailed or phoned to ask their willingness to participate in this research study. From the feedbacks, only 4 companies are agreed to involve in face to face and/or focus group interviews and 10 companies are willing to fill in the questionnaires.

In overall, the data collection process took around 8 months period from August 2008 to April 2009 to complete, which include identifying suitable companies' process, contacting and confirming potential respondents' process, conducting individual and focus group interviews process and distributing and receiving questionnaires process. Meanwhile, the data analysis processes which involved the qualitative and quantitative data have been done in parallel with the data collection process. All the received data were consolidated and analysed according to the specific methods as discussed in part 4.7.3. In general, the data analysis process has taken around 8 months (February 2009 - October 2009). Appendix A shows the list of all potential and involved companies for the present study.

- **Interview**

The face to face interview approach was used in this study in order to discuss topics in depth, to get respondents' candid discussion on the topic and to be able to get the depth of information of the study situation for the research context. According to Kvale (2007), interview reveals both direct and indirect data. Direct data are responses that subjects provide to direct questions, they are spoken responses. Indirect data are the less obvious or hidden information conveyed by gestures, body language, or a lack of eye contact. In an interview, interviewers can follow-up on half-answered questions; they can probe for deeper responses. He added that an interview data can be used to develop objective questions or closed-ended questions for a questionnaire. Flick (2001), stated that there are three types of interview formats that can be followed; structured, unstructured and semi-structured interview. A structured (focused) interview means that the questions are developed a head of time with some opportunity to ask pre-planned, open-ended, probing questions. This way, there are few variations, and the questions are asked in a specific order. As with observers, the interviewer is practiced on how to ask the questions, and how to probe the subject for depth. An unstructured interview can be used to explore alternative opinions, attitudes, or beliefs. It can help to identify new types of information and define areas of importance that might not have been thought of ahead of time. A semi-structured interview includes the open-ended and specific question. It has been designed to gather not only the information foreseen, but also unexpected type of information (Li, 2006). In this study a semi structured interview approach was used in order to gather information from the target respondents.

For this data collection phase, the researchers have interviewed 11 respondents from 4 identified Irish software companies which fall under VSEs categories. This process was divided into 2 types of interviews namely personal interviews and focus group interviews. Table 4.2 show the list of the companies involved and type of interviewed method that have been applied.

Table 4.2: List of involved companies

No	Company	Interviewee	No of Staff(s) Involved	Personal Interview	Focus Group
1.	Company B	CEO	1	X	
	Company B	Development Team	3		X
2.	Company C	Software Development Manager	1	X	
	Company C	Development Team	2		X
3.	Company AE	Joint-CEO	1	X	
	Company AE	Joint-CEO	1	X	
4	Company AF	Chief Architect	1	X	
	Company AF	Senior Software Developer	1	X	

The interview processes have been done in different places and time. The respondents have been divided into 2 categories; the managers and the development team. For the managers researchers have applied a face to face interview method and for the software development team, researchers have adopted the focus group interview method. All interviewed session were happened approximately 40-90 minutes and has been recorded with the respondents' permission. The details of the focus group approach are explained in the next bullet point. Furthermore, in order to guide the interviewer to gather a data during an interview session, an interview guide (Appendix B) has been developed (Taylor and Bodgan, 1984). The interview guide consists of close and open ended question, and some related notes about the direction in which to drive the interview under difference circumstances. Lastly, in order to ensure the interview questions do not divert from the research objective, the interview questions were prepared following the GQM (Lethbridge et al., 2005) principle technique as been explained in section 4.7.1

- Focus Group Interview

Powell and Single (1996) have defined focus group as "*a group of individuals selected by researchers to discuss and comment on, from personal experience, the topic that has been a subject of the research*". The focus group interview approached was used in this study because team members develop the software and the existence team interactions helped to release inhibitions amongst the team members. In addition this method could activate forgotten details of experiences and also generate data better through wide range of responses. Focus group interviews were also chosen because it was the most appropriate method to study attitudes and experiences; to explore how opinion were constructed (Kitzinger, 1995) and to understand behaviours, values and feelings, (Patton, 2002). According to Powell and Single (1996), the advantage of focus group is the ability to help the researcher in identifying quickly a full range of perspectives held by respondents. They added that focus groups expand the details that might have been left out in an in-depth interview. Kruger and Casey (2000) supported that focus group technique is a proper way to understand and explore how people think and feel about the issues. They also added that focus groups also elicit data that allows a better understanding of the difference between groups of people. Furthermore, in order to gather data and guide interviewer in the focus group interview, a focus group interview guide as in Appendix C have been developed (Morgan et al., 2008). The focus group interview guide started with a broad and open question that more toward the participant perspectives before narrow the question into the research interest. The entire focus group interview questions have been gathers from GQM process.

- Survey Questionnaire

The third data collection method for this study is survey questionnaire. Survey questionnaires were used in this study because it could facilitate the data collection and analyse the substantial data quickly. Moreover this method could help this study avoiding bias problems when interpreting qualitative data (Pfledger and Kitchenham, 2001). There are several advantages and disadvantages about applying survey questionnaires. The main advantages of using survey questionnaire are cost and time effective and also could easily collect a substantial data for the research study. While lower percentage of return rate of the filled questionnaires is the main disadvantage of this method (Lethbridge et al., 2005). Appendix D shows the survey questionnaire sample for this research.

The questions in the survey questionnaire were divided into 5 main parts: the background information, the questions on people issues, questions focussed on the software development process, questions focussed on the software development knowledge and questions focussed on the software development team issues. The survey questionnaires have been sent to substantial number of VSE companies by email and post mail in order to get a good response rate. Table 4.3 shows the list of the companies that have been involved in this exercise.

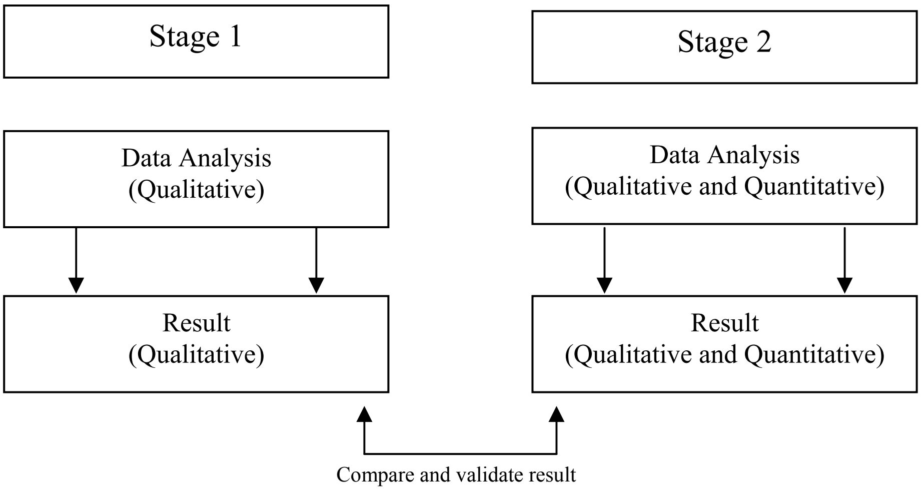
Table 4.3: List of companies – survey questionnaire

No	Company	Survey Respondent	Survey Questionnaire
1.	Company D	Owner/COO	X
2	Company E	Managing Director	X
3.	Company F	Director	X
4	Company G	CEO	X
5.	Company H	Development Manager	X
6	Company AC	Practice Director	X
7	Company AD	Managing Director	X
8	Company AG	CEO/CTO	X
9	Company AG	Senior Software Developer	X
10	Company AH	Senior Software Developer	X

4.7.3 Research Data Analysis Process

The study data analysis process is divided into 2 main stages. In stage 1, all qualitative data that gathered from individual interviews and focus group interviews were analyzed and coded. This process involves the development of the codes, code-categories and inter-relationship of categories which is based on the GT process and coding strategy. The details process of this stage will be discussed in part 4.8.1. Meanwhile in stage 2, the qualitative and quantitative data from the received questionnaire have been tabulated and analysed. The results from this stage were used to validate the analysis result in stage 1. The detail explanations of this stage will be discussed in part 4.8.2. The overall processes of current research data analysis is illustrated in figure 4.6.

Figure 4.6: Research Triangulation Design Data Analysis


Stage 1
Stage 2
Data Analysis (Qualitative)
Data Analysis (Qualitative and Quantitative)
Result (Qualitative)
Result (Qualitative and Quantitative)
Compare and validate result

- Qualitative Content Analysis

The challenges of analyzing the interview and focus group interview data lies in making sense of the substantial amount of data, identifying significant patterns and construction of a framework to communicate the essence of what the data reveals (Denzin et al., 2000). Therefore in order to analyze these data researchers had applied quantitative content analysis approach (Elo and Kyngas, 2008). Elo and Kyngas (2008) claimed that content analysis is a method that suitable to analyze the written verbal or visual communication transcripts. Hsieh and Shanon, (2005) in their explanation regarding qualitative contents analysis have defined qualitative content analysis as "*the subjective interpretation of the content of the text data* *through the classification process of coding and identifying themes or pattern*". Therefore for the current research, the qualitative data from the interview and focus group were transformed into transcripts and organized according to the pattern emerges during the analysis. These data were used as an input to the coding procedure in order to refine the abstract constructs and define the concept and categories. In order to assist researchers in analyzing qualitative data, three coding techniques proposed by GT methodology: open coding, axial coding and selective coding (Straus and Corbin, 1998) have been applied. These data analysis methods also have been recommended in qualitative data research (Denzin et al 2000; Patton, 2002) in order to guide researchers in analyzed the qualitative method more systematically. The *Atlas.Ti* software also was used to help in coding the interview text and linking this code on the semantic network. The decision was making from these transcripts, vague at first and increasingly explicit and grounded (Strauss et al., 1990). Social scientists (Miles et al., 1994; Patton, 2002) acknowledge that data collect and analysis in qualitative inquiry are integrative, iterative, synergistic and interactive in nature. The applied coding processes for the data collection are as below:

- Open Coding - From each transcribed interview transcripts, researchers have analysed the text using line by line or incident by incident coding before allocating an open or initial codes to the text. For this activity researchers have followed Charmaz (2006) initial codes approached which was done by using gerunds as this process will help researcher to detect the process and stick to the data. She also recommended to consider the following questions in order to guide researchers to create an open code:
    - What is the data a study of?
    - What does the data suggest?
    - From whose point of view?

After open code have been assigned and created, lists of open code then are sorted into categories based on how different codes are related and links. These emergent categories are used to organize and group initial codes into a meaningful cluster. This process involved the breaking down interview data and focus group data into discreet parts, close examined and compared for similarities and differences. Open codes that was found to be conceptual similar or related was group under more abstract categories based on their ability to explain the SPI, knowledge, team and standard issues which are the main unit of analysis (Elo and Kyngas, 2008) as in this research conceptual diagram as described in figure 3.6. Then all these open codes were then linked and grouped based on similar issues on the broad categories that represent the unit of analysis. Some of the open codes allocated in this way are known as an "in vivo" code. In-vivo codes are especially important in that they come directly from the interviewees, do not require interpretation by the researcher, and provide additional ontological clarification or context-description. Appendix E and J shows how the initial codes have been created and groups.

- Axial Coding – Axial coding is the process of relating codes (including categories and properties) to each other into subcategories (Strauss and Corbin, 1998). In this process all the general categories in open coding process were grouped under higher ordering heading. The purposes of grouping data were to reduce the number of categories by merging those similar and dissimilar into broader higher categories. In addition the merging process provides a mean for describing the situation to increase researcher understanding and to generate more knowledge. The process was continued with the abstraction process (Kohlbacher, 2006). The purpose of abstraction process is to detail up the categories by identifying the subcategories and how it link to another's. Subcategories with similar occurrence and incidents are grouped together as categories and categories are group as core categories. The abstraction process is an iterative process and continues as far as it is reasonable. Appendix F shows the example of the axial coding and abstraction process.

In general this activity's termed axial because coding occurs around the axis of a category linking categories to subcategories at the level of properties and dimensions. This involves documenting category properties and dimensions from the initial coding phase; identifying the conditions, actions and interactions associated with a phenomenon and relating categories to subcategories.

- Selective coding – The third coding process in the analysis of qualitative data is the selective process. Selective coding is the process of selecting the core category, systematically relating it to other categories, validating those relationships and filling in categories that need further refinement and development. In this process, the first step is to identify the main or 'core' category that related to the collected data. The core category acts as the hub for all other identified categories. In this part, the researcher using the *Atlas.Ti* tools in creating a network diagram based on the abstraction process result as in axial coding phase. The network diagrams were isolated in the beginning and merged at the end of the process. The merging network diagram helped researchers to produce an inter-related network diagrams that represent as a theoretical network diagram for the current research study. Appendix G and H shows the individual network diagram on selective issues and the theoretical diagram.

- Atlas. *TI* - Atlas. *TI* is a qualitative analysis tool which was designed specifically for use with GT (Coleman, 2006). It also allows researchers to link, search and sort the data. The tool is also capable to manage the interview transcripts, create a code, and store the quotation and memo. Furthermore its capability to create a category, link the categories and produce a network diagram from the categories helps researchers to under more about the current research issue. This tool helped researchers to upload the interview scripts, identify the code, create categories and link the categories in order to represent the overall picture of the current research issue as explained in axial and selective coding process (Straus and Corbin, 1998).

- Quantitative Data Analysis

Data gathered from the survey are tabulated and this process is using statistically analysis software (SPSS). Due to the small number of research quantitative data gathered, researchers have determined to use 3 basic statistical methods in order to analyse the quantitative data; Descriptive statistics, Mean analysis and Frequent analysis. Appendix I show all the statistical analysis processes for the collected data.

- Descriptive statistics - Descriptive statistics are numbers that are used to summarize and describe data. The word "data" refers to the information that has been collected from the research questionnaire survey. Descriptive statistics are just descriptive and not involve generalizing beyond the data at hand. According to Pallant (2005) the descriptive procedure is ideal for obtaining an overview of the distributional properties of numeric variables.

- Means Analysis - As for most statistical analyses, the mean is the most often used measure of central tendency. The mean is used most often, because of its relationship to the variance statistic. The mean is also important in the sampling distribution, which is formed from the distribution of all possible individual sample means, and has as its center, the mean of the population. The mean is affected by the presence of extreme scores (outliers) which may not be typical of the sample (or population) as a whole. The mean is preferred when a distribution is symmetric and interest is centred on a score that represents all scores (Pallant, 2005). Therefore this method been followed to understand the means population for the survey data.

- Frequency Analysis - The frequent analysis process provides additional information about the nature of each variable distribution (Pallant, 2005). All values are shown for each variable, as well as label, frequency, percent, valid percent and cumulative percent. This analysis involves constructing a frequency distribution. The frequency distribution is a record of the number of scores that fall within each response category. The frequency distribution, then, has two elements: (1) the categories of response, and (2) the frequency with which respondents are identified with each category.

4.8 Summary

This chapter presented a number of suitable methodologies that could be applied in a research study. It also has presented the differences between quantitative and qualitative research methodologies. This chapter have discussed in details the mixed-method research methodology, the methodology chosen for present research study. A detail explanation on the present research process, starting from the overall study research process design, data collection process and data analysis process are also elaborated and presented. The next chapter will discuss the findings and results from the data analysis process.

Chapter 5 ANALYSIS AND RESULTS - STAGE 1

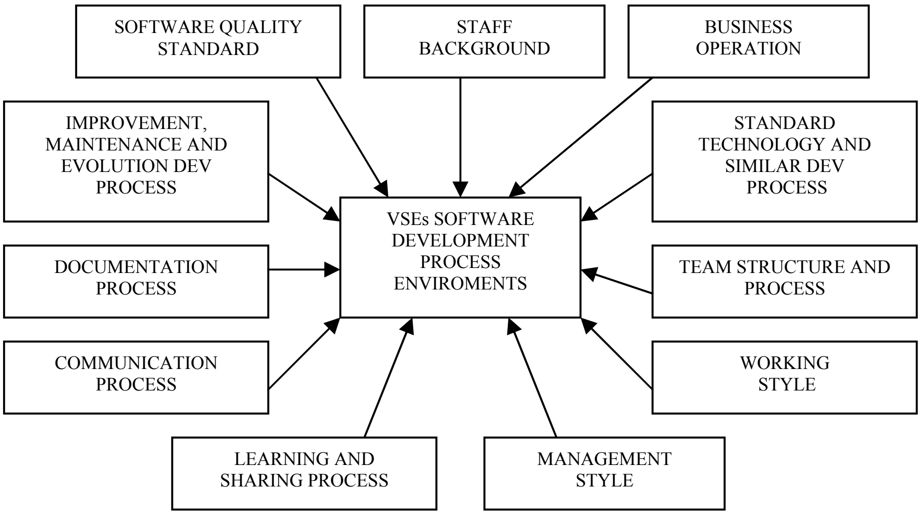
5.1 Introduction

This chapter presents the findings and discussions of the stage one of the data analysis process. This analysis process is mainly based on the personal and focus group interviews data. The results that are discussed in this chapter are the output from the qualitative content analysis, the main qualitative data analysis method process which was adopted using the GT coding approaches for the present study, as detailed in chapter 4. The flow of the discussion in this chapter begins with an overview of the overall related category diagram which indicates the main categories that influence the software development process environment in VSEs. This will be followed by a detailed description on individual or combination categories in order to explain and understand the current research issues. The integration of all the categories is then being discussed. Theoretical and relational diagrams are presented at the end of the chapter in order to show the overall software development process practices in VSEs.

5.2 Main Categories Diagram

Based on the detail analysis process as shown in appendix E and F, which adopted the qualitative analysis method, GT coding approaches and using an Atlas.Ti tools, as explained in part 4.7.2, we have produced and identified 11 main related categories that shape up the software process improvement environment in VSEs. Figure 5.0 illustrates the categories which consist of: staff background, business operation, working style, management style, team structure and process, learning and sharing process, communication process, documentation process, development process, technology and development method and software quality standard. Based on the analysis, these categories are the main categories and variables that have an influence to the software development process environments in VSEs. The details of the main categories are presented in a table as appendix H, which indicates all the subcategories and categories that are related to the main category. In this table, we have grouped and listed in detail the important variable that gathered from the open and axial coding in the analysis process as in appendix E and F.

Figure 5.0: The overall main category diagram


SOFTWARE QUALITY STANDARD
STAFF BACKGROUND
BUSINESS OPERATION
IMPROVEMENT, MAINTENANCE AND EVOLUTION DEV PROCESS
STANDARD TECHNOLOGY AND SIMILAR DEV PROCESS
VSEs SOFTWARE DEVELOPMENT PROCESS ENVIROMENTS
DOCUMENTATION PROCESS
TEAM STRUCTURE AND PROCESS
COMMUNICATION PROCESS
WORKING STYLE
LEARNING AND SHARING PROCESS
MANAGEMENT STYLE

5.3 Organizational Background and Business Operation

In order to understand the whole software development process in details, we started an analysis of the companies' environment particularly regarding the VSEs staff background and the companies' business operation. These two main categories are the overall variables that gave an influence to VSEs software development process activities and practice as a whole. These issues are discussed in detailed by Coleman (2006) and, Rapponen and Lyytinen (2000). Coleman (2006) has clarified that software development people background having direct or high influence in software development process activities in an organization. Meanwhile Rapponen and Lyytinen (2000) whose have done an empirical study on software development risks stated that the business environment and operations are among the important variable that influence the success of software development process. From the abstraction process in content analysis as in appendix H which involved the coding process, we have identified the categories and subcategories which explained in details the variables for this main category. Table 5.0 shows all the categories for this part.

Table 5.0: Staff Background

Sub Category	Category	Main Category
IT Experience - Manager	People Skills and Experiences	Staff Background
IT Experience - Team		
Expert		
IT related qualification/degree		
High Experience/ Skill full		
IT Background	Background of founder	
Non IT Background		
Proactive	People Criteria and Qualification	
High Interest and creative		
IT related qualification/degree		
Average Abilities		
Adaptable stress		

From the analysis as in table 5.0, the staff background main category could be divided in to 3 important categories which are people skills and experiences, background of founder and, people criteria and qualification category. In people skills and experiences category indicates that in overall all staffs in VSEs have a substantial skills, experiences and qualification in software development process and field as the following interview extracts indicate.

> *"The company was formed between me and my brother. We both are independent contractor providing software development service to no. of different companies…. Obtained electronic engineering from UCD and brother in computer application from DCU"*

> *"I was graduated in Spain and then I was working for almost 3 years in different environment. I was working in COBOL in bank environment and I was working in dot net environment and I finally went to Ireland, there in software design and almost 2.5 years."*

> *"I join company X traffic management and for quite time working in development department developing system like traffic recognition for controlling crossing, speed, tunnel and etc. I came from assembler, ADA, Pascal and eventually more to a new operating system, new development environment like Visual Basic or Visual Studio now."*

> *"I am doing electrical and electronic engineering for 2 year and move to degree in computer application. I join this company 10 years ago and haven't worked with other company before".*

> *"I have a degree in theoretical physic and have a master degree in computer science. I work a long time at 7 years after leaving college at the JET project which is a nuclear fusion research facility in UK".*

Meanwhile, the analysis also indicates that in VSEs the company founders' backgrounds could be categorised as two different types; Information Technology (IT) and Non IT. The analysis also indicates that those founders with IT background usually directly managed and involved in the development process and project as compare to non-IT background founders, which appointed a senior IT staff to manage the development process and projects.

> *"The company was formed between me and my brother. We both are independent contractors providing software development service to number of different companies. We always anticipate that at one stage to form business together. To do that we kind have a special client base and to do so we set up a company. Rather than trade as an independent contractors and trade as a single entities and to find obviously to find grows to the business."*

> *"I couldn't find a job in Ireland in traffic management, found job with this company. This company was a small, tiny family company, is about 15-20 peoples at the stage,. This company is started with conventional manual punch card system"*

> *"I am the founder with person X and CEO of this company just over 10 years."*

> *"I am a CTO of this company. All technical staff is located in this building and the management departments are located in Dublin"*

In the same issues, table 5.0 also indicates the list of staffs' personality criteria that exist in VSEs. These criteria have been identified from the interviewee responds regarding the company hiring criteria as in appendix B. The results have indicates that those high academic qualifications and technical skills are not the main criteria that were looking by the companies in employed new staff. The interviewee also expressed and emphasizes other criteria such as proactive, high interest and creative, adaptable stress, high motivation and reasonable working experiences are more important in VSEs working environments. This issue is important because these elements basically reflect the overall type of people who work in VSEs companies which contribute to the working environment in VSEs. Below answer quotes indicates the above situations.

> *"A good degree is not a concern to begin… I have a questionnaire that I will run with the candidate and specific question and tailor with the specific platform that I looking*

> *or working of them. General questions are to see on how they think rather than how they work"*

> *"We really don't care about qualification; the important is your passion, we looking for people who are self taught and high motivation. We look for people with experience but more important is passion"*

> *"We would generally look for self starters, people above average abilities, and good interest in computer software. We interested above beyond just the academic interest and do not use agencies for recruiting people.*"

> *"I normally try to get people who are proactive, trying to thinking ahead, people who also willing to work to a tight schedule and that also quite difficult."*

Table 5.1 shows the categories and subcategories for the business operation main category. This main category explains how the company business operation works, specifically in the software development function.

Table 5.1: Business Operation

Sub Category	Category	Main Category
Family Business/ Company / Owner - Brothers	Company Nature	VSE Business Operation
Non Family based - Partner		
Small Companies/department	Company Location	
Small office space		
Remote development team/ GSD operation	Development operation	
Collocate development team		
Remote Office		
Specific office		
Frequent Visit		

From the analysis as shown in table 5.1, we identified 3 main categories that support the VSEs business operation main category: the company nature, company location and software development operation categories. In company nature category, we found that business operation in VSEs could be divided into two subcategories; the family and non family based company. The family based company is basically initiated by a member of family which have experience in the business and also managed and worked in the company. While the non-family business based company in not involving family members in managed or worked in the company. The result from analysis also explained that VSEs has established in certain years and being located as a company or as a small department in an organization. Appendix E has detailed up the research findings. Below answer quotes indicates the above situations.

> *"The company was formed between me and my brother. We were both independent contractor providing software development service to a number of different companies. We always anticipate that at one stage to form business together.… Rather than trade as independent contractors, we trade now as a single entity and try to grow the business"*
>
> *"I am the founder, director and developer along with my business partner. I started this company 10 years ago and haven't worked with other company before."*

In addition table 5.1 also indicates how the software development activities/process operates in VSEs. From the analysis, the researchers found that the software development work has been done in both centralized and distributed environments. The centralized process is where all staff is located in one place in a specific building. While distributed process is where some staff have been hired from other or same countries and have an office and work remotely from their country or location. These staff will come and visit the main office on a periodic basis. From interview and focus group session with 2 of the companies, we found that the remote employees were started their work in Ireland for a certain period. Then they went to their home country, set up the office or work station and kept working remotely for the same companies. The communication tools such as internet, phone, email and others have kept them in contact with the main office in Ireland. This process indicates that the GSD (Global Software Development) process has also been practiced in VSEs. Below answer quotes explain the above situation.

> *"All technical staffs are located in here in this building"*
>
> *"We have this main office, but now person X works at home in Spain and person Y is at home in South Africa"*
>
> *"At the moment I have 1 person who is working with me in our office. I also have one person who is working in Poland and one in Latvia"*

This part indicates the overall background and business operation of software Irish VSEs companies. It also points out the criteria of people working in companies and how the development operation has been practiced in VSEs. This understanding of the issues has leaded the researchers to understand the VSEs development and business environments.

5.4 Team Structure and Process

The third main category that has been identified is the team structure and process category. The analysis result shows that the team environment in VSEs could be divided into 2 categories; the organization and team structure category and the team process category as tabulated in table 5.2.

Table 5.2: Team Structure and Process

Sub Category	Category	Main Category
Team Size - Small	Organizational and Team Structure	Team Structure and Process
Organizational and Team Flat Structure -		
Team Role - informal	Team process	
Team Involvement - direct		
Team Culture – informal		

The organizational and team structure category indicates that due to the small number of people working in the organization, the team size is also small and this leads to a flat team and organizational structure. From the interviews analysis results indicate that all interviewees admitted that the companies have no formal team structure or a team structure only exists occasionally as maybe required for a particular project.

> *"There are about 15 employees' works in this company at the moment. Software development department consist of 3 employees. Before this we have 4 people but we need to scale one down and we probably scale another one down. So we end up with 2 or 3, I don't know. It just depends."*
>
> *"Since 2, 3 people in development plus the manager we have a very direct contact with them, with the manager which is quite good regarding this."*
>
> *"Basically is a team of one. We can have a larger team of 2 or 3. But no formal team structure as every body equal within the team"*
>
> *"There are 5 developers including me and peter. No we don't have a formal team development structure at the moment, we all have the same skill and it is very flat."*

In additional during the analysis researchers found that due to the small number of employees, flat organization and team structure and informal environment, interviewees perceive that all people in the company are at the same level. In addition the analysis show that they have the similar level of working experience, skills and very much depends on each other in performing their task. Besides the close working space or area, high frequent and informal communication are also influences this perception. All these criteria have leaded VSEs in narrow down the gap between the management and the team development. An interview answer quotes below best represents the above situation.

> *"We don't have that [formal team structure] but I can see in a large company where might have that. In small company I think it is a bonus we know each other very well"*
>
> *"There are really 2 levels; the level above me is IT manager and General management. But its such a small company almost like family here, so that not really a divided there just like a structure in place inside."*
>
> *"Since 2, 3 peoples in development plus the manager we have a very direct contact with them, which is quite good"*
>
> *"The management and staff relationship is very close. It is probably because we are in the similar age and similar interest. Nobody works in this company that not interested what we do in this company."*

The second category is the team process category which indicates the team role, team involvement and team culture issues. The analysis shows that the staff role which includes the role in team and the task they perform in development process is very informal and very general. This could imply that the development staff could work or be assigned a different role at any time in organization development project. In addition they also can work with others or different people and different position as and when they are required. These situations have explained that team involvement process in VSEs is direct and informal in development activities. Therefore from these two categories a network diagram can be depicted as shown in figure 5.1. An interview answers quote below represents the above situation.

> *"I mean usually either be face to face between 2 developers or over skype through 2 developer remotely communication. In general the developer kind of work independently sometimes have a project, we have a sole responsibility for the project. Other time they assist each other in strategic and help each other for the output for a single project."*

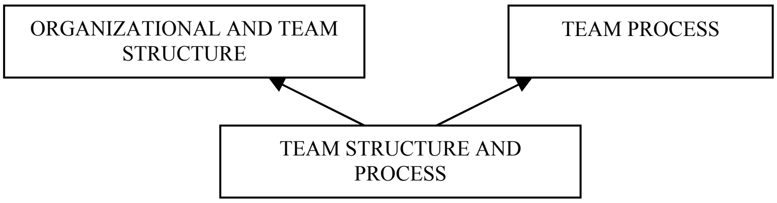
> *"It doesn't mean the notion of team isn't clearly set out in our company but just because of our size."*

> *"As a CEO, I am not sure how others there probably seen myself and brother either same as others or not. My brother and I have done a lot of development work as well. One senior staff probably had seen himself as the head of new staff due to experience in company. But actually they is no real title and rank in this company"*

> *"One developer worked under one senior and will be work with others when needed. So we are interact each other all the time and this is the way we work"*

From the above analysis, we could plot a network diagram as in figure 5.1 that shows the relationship between the main category and category. The details of the network diagram could be referred in appendix G.

Figure 5.1: Team Structure and Process



5.5 Standard Technology and Similar Development Process

The fourth main category is the standard technology and similar development process category. From the analysis 2 subcategories; standard technology/tools and similar development process/method have been identified as list in table 5.3.

Table 5.3: Standard Technology and Similar Development Process

Sub Category	Category	Main Category
Standard Technology	Applied standard technology and similar development method	
Similar development Process		

The analysis found that type of development tools and development process being used and adopted are various in VSEs. These depend on companies' main technology platform and framework in developing their software product. However the analysis as in appendix F has indicated that VSEs are adopted similar development technology, tools and methods in developing company software product. Based on below the interviews answers quotes, researchers have identified the reasons why VSEs adopt the same technology or tool and development methods in developing software system in their project as below:

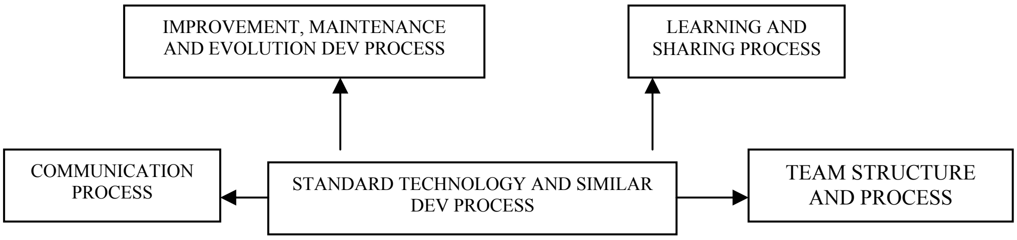
- Low taking risk in adopting existing technology and process.
- Dependent to the proof and efficient development tools and method in order to avoid conflict which could impact the project delivery.
- More focused on delivery time and profit rather than experimenting with tool and development process.
- Limited number of employees working in the companies.

> *"We have not changed the process that impacts the delivery of the process. Delivery of the project is very important thing so we will not change the process that will impact the delivery time"*
>
> *"I have couple projects that I have looked after at moment, there in old technology… [new technology] I will definitely suggest if we have that now, it just happened we might find the project it that suit us and it definitely go in and use it"*
>
> *"Our entire environment would be set and so that our environment are all same. There is no different machine, same naming, and sort off convention. So when you get the project you not changing a shared document."*
>
> *"Yes, the development processes are quite identical and small changes depend to situation but overall the development process is same."*
>
> *"It is purely because if you want to get done quickly then what you need is focusing to the output not the process."*

The analysis of the interviews data also indicates that developers in VSEs follow the agile development philosophy and approaches in developing systems. This could be identified with the method and process they are adopting in performing software development task. The details of this process will be explained in later parts that are related with this issue.

In overall this part explains the issue of development technology and method that have been adopted in VSEs. The analysis has explained that VSEs are using the same development tools and method in developing the system. This is due to a few reasons that have been stated above. The analysis in appendix F has indicates that the combination of both issues has given an impact to the other processes namely the documentation, communication, learning and sharing and SPI processes as shows in figure 5.2. The details relationship between these categories will be discussed in details in each of the above identify category sections.

Figure 5.2: Standard Technology and Similar Development Process


IMPROVEMENT, MAINTENANCE AND EVOLUTION DEV PROCESS
LEARNING AND SHARING PROCESS
COMMUNICATION PROCESS
STANDARD TECHNOLOGY AND SIMILAR DEV PROCESS
TEAM STRUCTURE AND PROCESS

5.6 Working and Management Style

During the study, it shows that the small team size and flat team or organization structure that exist in VSEs gave an impact on VSEs working and management style. The analysis process has indicates that staff have autonomy on their work which make them more self dependent, self responsibility, work independently and self learning as in table 5.4.

Table 5.4: Working Style

Sub Category	Category	Main Category
Autonomous work	Working Style	
Autonomous Communication		
Work independently		
Strategic area		
Sole Responsibility		
Self Learning		
'Team of One'		

The result from the analysis emphasises that people in VSEs working style is more individualistic with tasks being assigned according to expertise. This situation has been defined as '*team of one*' by one of the interviewee. The formal interactions of between the team member is more on the strategic area only such as problem solving or knowledge sharing in particular issue that related with the software development issues. But most of other interaction or communication are more indirect, casual and very informal. This situation gave researcher an indication that notion of team work in VSEs only appears or happened in informal way or periodic basis. The interview answer quotes below presents the above situation.

> "*I think being a small company will all given probably a lot more of responsibility and what would be given may be in larger company because you do work autonomous and you do there is a lot of autonomy.*
>
> *"We are not a huge team and everybody doing something different in their work. We are responsible for what they are doing… we are responsible all of bits of work on at the end bring all at together the final product is delivered."*
>
> *"Basically is a "team of one" essentially or larger team of 2 or team of 3. Basically no formal team structure and every body equal within the team"*
>
> *"It depends and because we are small company sometimes we have big project but most of it is small. It depend the scope of the project if big we might do it in team and again most of it is individual basis. Basically people work quite autonomous and specific."*

In relation to autonomous working, the analysis also indicates that the people in VSEs also exercise an autonomous communication style in performed their works. Informal communication, less structure and direct communication, self learning and explore, frequent informal guidance, and informal meeting code that produce from the analysis indicate the autonomous communication process happened in VSEs. The detail regarding the communication process will be discussed in details in the communication process sections in later parts. Several interviewed quotes below explain the autonomous communication issues in VSEs. Appendix F presents the details indicator on VSEs working style.

> *"They tend to communicate when they want a problem answered. They sense do are very autonomously any kind of communication would be either for direction or ambiguity or needed and assistance to particular problem. Everybody is tense to solve their own problem."*
>
> *"Our manager is involved in mostly designing… we are consulted regarding all his decision regarding changes in the product. So I can feel like personally involved in directly management because every major decision is consulted with me."*
>
> *"We have daily stand up meetings as in agile development practice, that happen approximately 10-15 minutes and generally that they way we discuss the day and previous day issues and from there the people break off and discuss in detail. So the communication process is more toward informal rather than formal."*
>
> *"One developer worked under one senior and will be work with others when needed. So we are interact each other all the time and this is the way we work. We also work very direct, loose and flexible."*

In additional, the analysis process as in appendix F also indicates that there are similar management styles adopted within VSEs. During the study, it shows that the small team size elements in VSEs are also gave an impact on the management style in the companies. Table 5.5 indicates how the management style has been adopted in VSEs.

Table 5.5: Management Style

Sub Category	Category	Main Category
Trust	Management Style	
Family and Flexible Environment		
Loose PM		
Open Environment		

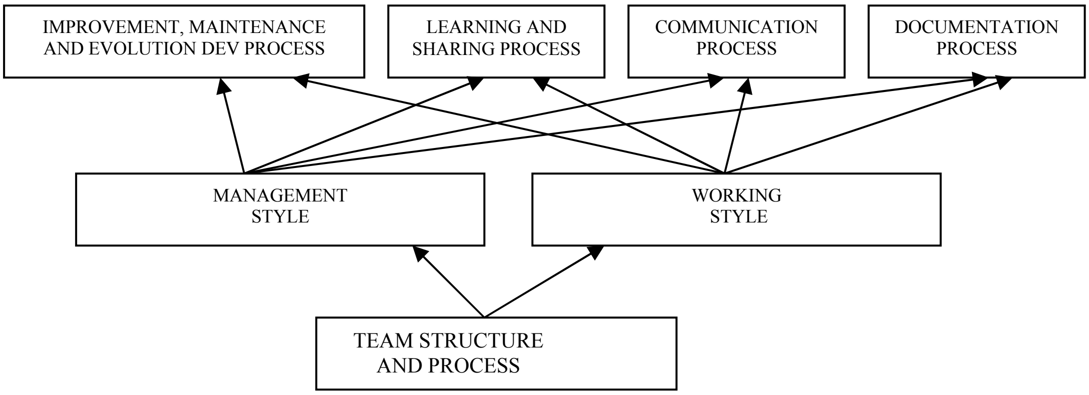
Trust, family and flexible environment, loose project management and open environment are the subcategories that indicate the based management style in VSEs. This type of management approaches is defined as 'Embrace and Empower' (Coleman and O'Connor, 2008) regime as similar to McGregor's (1985) 'Theory Y' style. In this context the idea and opinion from all subordinate have a value and have been adopted in the development process and policy. There are also indicators that the element of trust in development team and their ability to carry task with less direction. This could be identified in table 5.3 which identifies the working style on how the development staffs perform their task. Several interview quotes below indicate the management style issues in VSEs.

> "*I don't bother so much on what people activities as long we meet our objectives. This is because staffs need to be honest and you need to trust other people in performing their task or you don't get their cooperation.*"
>
> *"Project Management are very loose and we are good in estimate about our project… and we not necessarily have to dedicate the project manager. In here people are aware where we are in the project in one time and everybody knows where we are at any one time."*
>
> *"I think one of our strength is that we are not to pushy and we are not really tough on the programmer… maybe one of the reason why people is stay with us because is a family environment."*
>
> *"Is very informal, very flat structure not huge hierarchy and have a freedom to implement what they think is working to the task. We just control very loosely over time."*

> *"Generally people work according to he specific tasks and skills and they are responsible for the delivery and their work. I will observed them from time to time and is very informal and macro."*

In summary of this part, the analysis has indicated how the management styles and working styles have been practiced in VSEs. The analysis in appendix F indicates that the combination of both issues have given an impact to the other processes namely the documentation, communication, learning and sharing and SPI processes as showed in figure 5.3. The details relationship between these categories will be discussed in details in each of the above identify category sections.

Figure 5.3: Management and Working style


IMPROVEMENT, MAINTENANCE AND EVOLUTION DEV PROCESS
LEARNING AND SHARING PROCESS
COMMUNICATION PROCESS
DOCUMENTATION PROCESS
MANAGEMENT STYLE
WORKING STYLE
TEAM STRUCTURE AND PROCESS

5.7 KM Process

In order to understand VSEs KM process, the researchers have combined the 3 main categories namely the communication process, learning and sharing process and documentation process into one section. The objective is to understand the relationship between these 3 main categories in order to support KM process in VSEs.

5.7.1 Communication Process

During the interviews sessions, the researchers have asked several questions on communication issues as in appendix B and C, in order to understand this issue in VSEs. From the analysis, the researchers could divide the communication process in VSEs into 2 categories namely open and informal communication category and online communication category. The analysis also shows that the communication process in VSEs is influenced by the companies team structure and process as been explained in part 5.4 and the working and management style as explain in part 5.6. Table 5.6 shows the details communication process categories produced from the analysis.

Table 5.6: Communication Process

Sub Category	Category	Main Category
Team Structure and Process Working and Management Style		Communication Process
Open Communication	Open and Informal Communication	
Informal Communication		
Communication tools	Online and Electronic Communication	
Internet/ Electronically		

In the open and informal category, the researcher has identified 24 interviews extracts that represent the category as shown in the network diagram in appendix G where people are more towards informal and direct/casual communication. This can be identified in the ways meeting are conducted, which are more informal, 'stand up', periodic and individual. In addition, the interviewees also agree that their day to day communication between staff is very direct and autonomous, due to the working environment in their company which have been discussed in part 5.6. This situation is confirmed by the interviewees, stating that because of the small team size that exists in the organization and the working style culture which is more toward autonomous work have create this situation. Below answer quotes represent the above situations.

> *"Formal meeting not very often because we always communicate each others so formal meetings isn't here, we do might have a kick off our project and to the project and generally people discussing at different all of the time. Very open communication because we are very small everybody knows what every body doing or working on."*

> *"We have a formal meeting once a while but most of it is more informal. It is informal when we discuss development stuff like over the coffee. We usually share our code esp. with peter and he will look at it and share the idea. Later we will introduce to others and ask for feedback. We have informal meeting for a few minutes just to inform others regarding process before we start our tasks."*

> *"We have a daily stand up meeting and we have an iteration planning meetings but very fairly loose. Generally communication is very informal on daily basis."*

In addition the analysis also indicates the relationship between staff in the company also influences the communication process in VSEs. The analysis as in appendix F shows that the family and flexible environment, frequent social interaction between people and flat organization structure have given an impact on communication process in VSEs. Beside that the closeness people working space or area and high frequent of sharing activity have contribute to the communication process in VSEs. Below answer quotes have explain the above situations.

> *"I see a very open, very congenial very friendly and professional environment… I see people on the equal sourcing, openly discussing,. There no very rigid formal hierarchy. The team easily talk to management as we sit side by side. "*

> *"We work very close, meet for morning coffee. We always mix together and are very dynamic because we are small and easy to communicate each other."*

> *"Programmers are friendly and they socialize each other. They get on pretty well each and can easily exchange ideas"*

The second category in this part is online and electronic communication category. From the analysis, the researchers found that the use of communication tools such as email, phone, blog, skype and internet are very active in VSEs. Such communication tools are vital to the company that have a staff member working in different locations. From the analysis researchers found that the main purpose using communication tools beside to communicate between staff members, it also the tools that could close the gap between remote and collocated staff. The analysis also indicates that the use of communication tools is to allow staff to share and document all work related information or knowledge in informal way. Below quote extracts explain this.

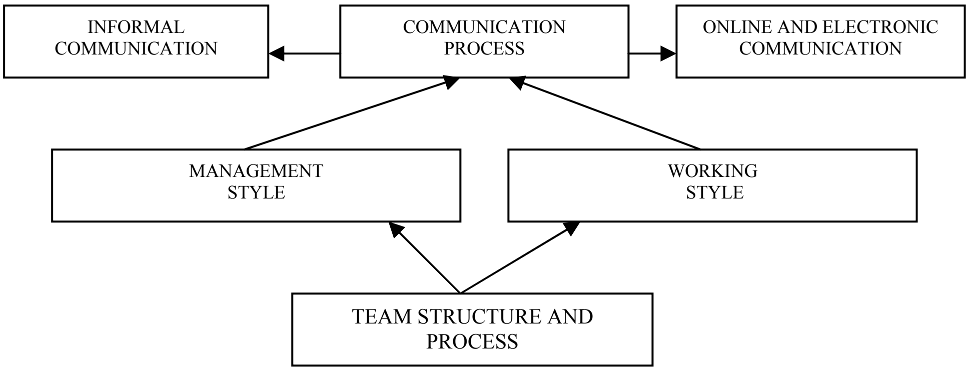
> *"We always skype with and other tools chat message, VPN, blog and others. We have company internal blog to share the information among us"*

> *"Yeah, email is obvious… skype and telephone communication is our 90% communication tool. We work on VPN connection so we can share our LAN… I build a model that they can work with us without any problem"*

> *"Basically we used our own application – teamwork. We using all the time in our entire project. Milestone, tasks that need to achieve is added and it assign to somebody and when they complete it there is a notification. Beside that we used email and sometime people talk using other tools using phoned, Skype and IM"*

In overall, this part is focused on the communication process issues in VSEs. The analysis has shown that in VSEs 2 categories involved in the communication process, the informal communication and online and electronic communication. The analysis also shows that these have process happened because of the influenced by the team structure and process and management and working environments in VSEs. Figure 5.4 shows the communication process network diagram and its relationship with other categories.

Figure 5.4: Communication Process



5.7.2 Learning and Sharing Process

The interview data analysis elaborates how the learning and sharing process happens in VSEs. The analysis in appendix F shows that the learning and sharing process main category could be detailed up into 2 important categories namely self learning category and sharing category as in table 5.7.

Table 5.7: Learning and Sharing Process

Sub Category	Category	Main Category
Communication Process Working and Management Style Team Structure and Process		Learning and Sharing Process
Training	Self Learning	
Self Learning		
Continuous Guidance		
Internal Training	Sharing	
Meeting		
Document		

In the self learning category, the analysis shows in VSEs there are no formal training given to employees in enhancing their knowledge or skills. In the analysis informal training has been defined as internal training, sharing and self learning. The analysis also has explained that people in VSEs are more dependent on self learning in mastering the technology or process that is used in the organization. Besides self learning, the analysis also shows on the job training, self exploring and continues guidance from expert with in the companies are the main process that frequently been practised in enhanced staff knowledge and skills. The following extracts are illustrative of this point.

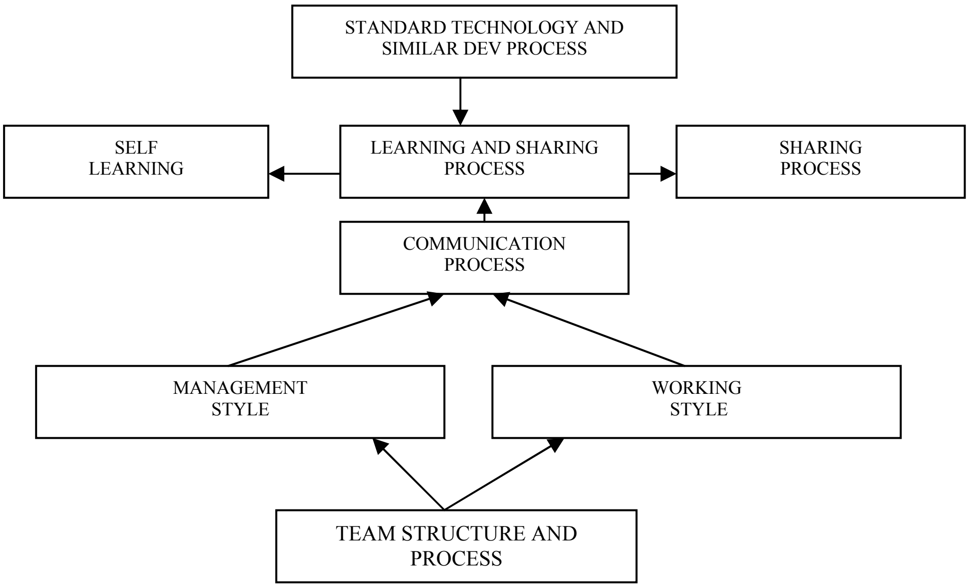
> *"We haven't done any formal training but we do give our employee an opportunity to attend various courses and seminars."*
>
> *"It wasn't a formal training… what I mean once you get started you could find out, who to do certain things, someone have experience can show you the way of the main resources or he can read article with your interest you want to carried out certain task. It wasn't a formal training period, I just call training because I actually learn and still learning but now is not as before"*
>
> *"No, they have to do on the job training; they have to educate themselves on the job by doing and using Google. It's more towards self learning but support from other team including me. In start up company we won't spoon feed the staff and I wont bring that situation to them because it is not fair to them and us either."*
>
> *"Training is more informal and in-house.… They [staff] also are encouraged to improve. Either they learn that way or from the other people directly. It is pretty informal structure*."

The second category in this part is the sharing category. The analysis shows that in VSEs the knowledge sharing process happens in 3 ways: informal training, informal meetings and document sharing. Informal training happened through informal and guidance from expert, peer to peer programming process, shared books and others material, internal training, high frequent open and direct discussion with team member and online sharing with others. The informal meeting process happens through an informal stand-up meetings, direct and open discussion and online meetings via email and skype. The last process is through the document sharing process that will be discussed in part 5.7.3. In relation, the analysis results in appendix F also indicate that the learning and sharing process in VSEs is been influenced and shaped by 3 main factors which are VSEs team size and process which are small team size and flat organization structure; working and management style which are more toward autonomous work and macro management process and, communication process which are indirect and informal process. In addition from the interviews data analysis shows that in general knowledge sharing activities either via electronic or personal means are important in maintaining and evolving the current VSEs software development process. The quotes below have explained these situations.

> *"However when you want to do a new things and you want to introduce a new methodology you discussed with the rest of the team, that is good and also we are supporting, if you want to do something but you not sure, you can go to any others who has more an expertise in the same area"*
>
> *"We shared books and we buy books and we pass around. Generally it is informal process just asking question, grasp him and talk. Sometime we did pair programming but not always. Generally it is some kind of informal."*
>
> *"Knowledge also shares sometime in peer programming activities. It doesn't happen a lot but it can happen when problem arise and we sit down to explain and discuss."*
>
> *"We usually share our code especially with peter and he will look at it and share the idea. Later we will introduce to others and ask for feedback."*

In overall, this part is focused on the sharing and learning process in VSEs. The analysis has shown that VSEs more practicing self learning and informal sharing in enhancing knowledge and skill. The analysis also indicates that this process happens because of the influenced of team environment, working and management style and communication process factors that are practiced in VSEs. Figure 5.5 illustrate the learning and sharing process network diagram.

Figure 5.5: Learning and Sharing Process


STANDARD TECHNOLOGY AND SIMILAR DEV PROCESS
SELF LEARNING
LEARNING AND SHARING PROCESS
SHARING PROCESS
COMMUNICATION PROCESS
MANAGEMENT STYLE
WORKING STYLE
TEAM STRUCTURE AND PROCESS

5.7.3 Documentation Process

During the interviews sessions, the researchers asked several questions about the company documentation process as in appendix B and C. The purpose is to understand the issues of documentation process in VSEs which are related with KM process. From the analysis, the researchers identified 2 important categories that could explain the documentation process in VSEs namely the informal and individual documentation category and specific information and procedure category as in table 5.8. The analysis also shows that the documentation process in VSEs is influenced by same development tools and similar development method in every company software development activities as explained in part 5.5 and working and management style as described in part 5.6.

Table 5.8: Documentation Process

Sub Category	Category	Main Category
Standard technology and similar development method Working and Management Style Team Structure and Process		Documentation Process
Informal documentation	Informal and Individual Documentation	
Individual Documentation		
Technical Business related document	Specific Information and Procedure	
Client related Document		

In informal and individual documentation category, the analysis has indicated that in VSEs the documentation processes are very informal process and are individual initiatives. From the interviews open code data as in appendix E indicates that all interviewees agreed that they did not have a formal documentation process in software development activities and process. The analysis also shows in VSEs documentation process are not given high priority because of time constraints and small team size. The results from the analysis also indicate in VSEs most of the information is documented in an electronic format rather than a paper format. The interviewees also admit due to similar technology and development method applied in all development projects, details documentation process is not necessary and important. They claimed that staff is more focused to software development activities rather than the documentation process. Below answer quotes have explain the above situations.

> *“No we haven’t clearly defined and documented it yet. We considered our process to be evolving and are not finished. We started of with formal old style model then we change continuously until we are happy on it. We can improve it any more, so that kind of project of itself.*”

> *“The documentation is only for our internal need, so there no general standard how we documented it.”*

> *“We don’t have a formal documentation process. Basically the requirement gatherers put it in written and distribute to the team. Then they know what the processes need to be”*

> *“I think documentation is more on electronic basis. We are more focusing on programming code mostly. Other than that is more personal or informal documentation. Time is the main constrains and we can’t get a good developer either.”*

In additional from the analysis indicates that due to the autonomous work culture in VSEs, which based on person experience and skills, most of the documentation process in VSEs is individual and personal basis. Beside that, the analysis results as in appendix F, shows that programme source code, technical issues and business procedures are the main documentation in the VSEs which fall under specific information and procedure category. Below answer quotes have explain the above situations.

> *“We have decided that we have a library where we actually pick certain things. We also have a kind of code of conduct. How we actually write code, we put what ever possible explanation what we are doing with the source safe whenever we take something out or put something in, we production and set.”*

> *“I think the most important thing which we have here is kind of source control software which is contain all the changes, even little changes of the software overtime, we produce 20-30 versions of the software of a major product, we need to have all the different changes that a major product, we need all the different changes that want over time, because our customer having different version of software, so we have to mind this.”*

> *“The code is well written and that the documentation. That’s no activities that sit down and write a lot of documentation. Typically time doesn’t allow us to be comprehensive to that regards. But sometime people took their own initiative to document for themselves.”*

In conjunction of the above, the knowledge loss issues also have been explored in order to understand the interviewees’ views and companies’ actions in order to mitigate this problem. Appendix B and C shows the interview and focus group questions that has been asked regarding these issues. The analysis indicates that due to economy situation nowadays, standard technology, similar development and company nature in VSEs, indicates that knowledge loss problem due to the departure of staffs is not an important issue. In additional, the interviewees' claimed that the informal and autonomous work environments creates an atmosphere that people in the company are more willing to share and work closely each others. The extracts illustrate the above situation.

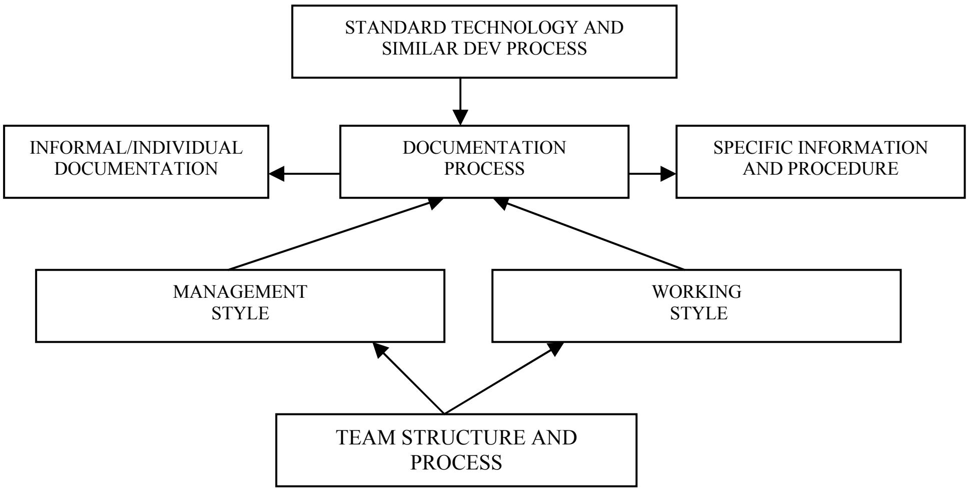
> *"We follow the same tool, process and framework in every project. This really helps to secure the development knowledge in our company….So if anyone left the company, the company would not loss the knowledge seriously"*

> *"We are a small tight team. We actually work well. A lot of people have a limited set of skill of new technology and we been expose them to the new technology. It is good for the people in career progression in experience of that."*

> *"Sometimes we make people working together and ensure that no one exclusively work in one project. But sometimes it is hard to apply due to our small size."*

In overall, this part is focused on the documentation process in VSEs. The analysis has shown that in VSEs documentation process is informal and individual basis. The analysis also indicates only technical and business related issues are documented. Beside that from the analysis, the researchers have identified 3 main categories that relate and influence the documentation process environment in VSEs. Figure 5.6 illustrate the learning and sharing process network diagram.

Figure 5.6: Documentation Process


STANDARD TECHNOLOGY AND SIMILAR DEV PROCESS
INFORMAL/INDIVIDUAL DOCUMENTATION
DOCUMENTATION PROCESS
SPECIFIC INFORMATION AND PROCEDURE
MANAGEMENT STYLE
WORKING STYLE
TEAM STRUCTURE AND PROCESS

5.8 SPI: Improvement, Maintenance and Evolvement (SPIME)

In order to understand the main issues in this research, which is the software process improvement, maintenance and evolution in VSEs, the researchers asked several open questions to interviewees as in appendix B and C. From the analysis as in appendix F, it shows that this main category could be divided into 2 categories which are process status category and development method category as illustrates in table 5.9.

Table 5.9: SPIME Process

Sub Category	Category	Main Category
Standard technology and standard development method Team Structure and Process		Improvement , Maintenance and Evolvement Development Process
Informal/Indirect	Process status	
Small Scale		
Process Loss and Focus		
'Agile' Development Style	Development method	

The first subcategory that exists in process status category is process loss and focus subcategory. In this subcategory, the researchers found that besides the staff background main categories characteristics, as discussed in part 5.3 influenced the software development activities in VSEs, the results from the analysis also indicate the SPIME process started when the process loss and/or process focus happened. From the analysis as in appendix F, shows that process loss happened when the technology change, customer requirement creep, software function creep, and a new idea or suggestion from the staff exist in their business activities. Meanwhile, process focus happened when new customer requirements, market changes, business procedure and requirement upgrade, software module or product update and expert/staff suggestion and idea occurred in their business environment. In addition, the analysis also indicates that VSEs are working very close with the customer in improving the software product and process. The quotes below explain the above situations.

> *"We delivered customer requirement as on going basis….. This is kind of quality improvement process and we had identifying what involved and how it goes"*

> *“We always release one version and for example if the customer has a certain version to upgrade he also need a new functionality if he wishes so if they buy this new functionality, which is the version of our system...”*
>
> *“We loosely base our process base on extreme programming but we use a bit what we want in fact we go, we probably do go over the top in some areas. We start a section of work that might be a light document and that would be a specification and that either we got a customer input or internal input. If it is internal work this is to formulate our thinking enough to point we can show to the customer”*
>
> *“The improvement has been done based on our own framework building block. The building block consists of several small functions. We will improve the building block depends on client requirements. ”*

The second subcategories in this part are the small scale and informal or indirect subcategory. From the analysis process as in appendix F indicated that the SPIME process in VSEs has been done in a small scale but very frequent. This process could identify the analysis code such as organic, natural change, reactive vs. proactive, trial and error, module orientation, minor changes and profitable orientate that extract from the interviews quotes represents the scale of SPIME process in VSEs. Meanwhile from the axial code as in appendix F such as RAD development, frequent change, direct and rapid change and ‘agile’ process are reflected the frequent level of changes in SPIME. The analysis in appendix F also shows that this process happened due to resources constraints especially the small team size and standard technology applied in company’s software product. Below answer quotes have explain the above situations.

> *“we are doing it little by little and that way we going to do we going to improve the process, kind of and we open the discuss ask about what to do next, what is the basic/biggest problem of current process and what we could do to address that”*
>
> *“We always look at the way to improve it because all is under the source control. We got a team work to shared idea for specific framework and we make it changes, inform everybody and discuss it. Usually it is a small change, and very informal. The evolution is very rapid. We discuss it the best way to do.”*
>
> *“You don’t have in small company extensive testing. We cannot do that. But our project quite establishes only certain thing we add to the product has to be tested to certain degree. If you add something you have to add regression test, you actually test the main function all your system again I would do really kept to minimum.”*
>
> *“Yes, the development processes are quite identical and small changes depend to situation but overall the development processes are same.”*

Beside smalls scale and high frequent changes in SPIME process, the analysis also indicates that the improvement, maintenance and evolvement process in software development are performed in informal or indirect process. Not following any standard and guideline, not structure improvement process and informal post mortem process are the indicators that the SPIME process are being performed in informal and indirect way. This situation has been identified in interviewees answer quotes as below:

> *"We would do it a little step. What I haven't done is sit down and written how exactly the process works. And how we want the process work, to large extend I know authority what is the best it is/ for is and no doubt with the changes we will implement almost"*
>
> *"I don't have experience with the large company that I can compare. Here the lifecycles we maintain in this office is enough for me because of there is one step from the lifecycle which it is suitable for small size of the company is we don't have a specific designated testing unit, person, who doing testing, after testing they going back to redesign or re-implement things."*
>
> *"We always try to improve our process if we have time and resources. In term of formal structure not anything yet to adopt it."*
>
> *"Basically, yes! But we improve it in a small scale and informal over time. This practice is kind of rapid application development and a lot of changing in our product. The most important is our project is profitable and that the biggest focus and high priority. So everything must be done fast"*

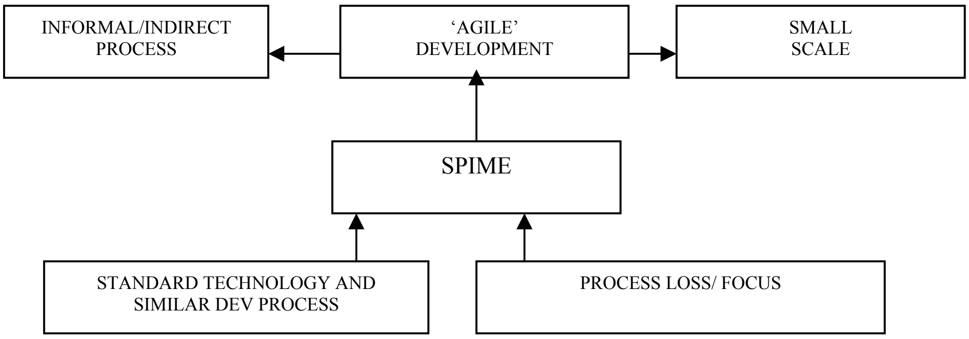
Meanwhile the development method category indicates that VSEs are more likely to follow agile type development approaches in their software development process than other developments methods. This could be identified in the communication process, documentation process, change process and customer collaboration which have been identified and explained above indicates that VSEs have fulfilled the 4 main general characteristic as in Agile Manifesto (Fowler and Highsmith, 2001). Below answer quotes indicate the agile development approaches in VSEs.

> *What we doing at this moment removing to like fully agile development methodology, we are doing it step by step because 2 things: i. continuity from everybody using the system. If we changes to may things at once, you can grind a halt and no one will do any development due basically a new process. ii. we doing it little by little and that way we going to do we going to improve the process kind of and we open the discuss ask about what to do next, what is the basic/biggest problem of current process and what we could do to address that.*

> *“We got a team work to shared idea for specific framework and we make it changes, inform everybody and discuss it. Usually it is a small changes, and very informal. The evolution is very rapid. We discuss it the best way to do.”*
>
> *“We have tool call XPlanner that what promoted by agile development and so in term of the process happen”*
>
> *“We loosely base our process base on extreme programming but we use a bit what we want in fact”.*

In overall, this part is focused on the SPIME process in VSEs. The analysis has shown that in VSEs SPIME process start with process loss and process focus. The analysis also indicates the SPIME process in VSEs has been organized informal way and in small scale basis. Beside that from the analysis also indicates that VSEs are follows an ‘agile’ development method and process. The analysis also found that the standard and similar development methods in every project in VSEs have their SPIME process. Figure 5.7 illustrate the SPIME process network diagram.

Figure 5.7: SPIME Process


INFORMAL/INDIRECT PROCESS
‘AGILE’ DEVELOPMENT
SMALL SCALE
SPIME
STANDARD TECHNOLOGY AND SIMILAR DEV PROCESS
PROCESS LOSS/ FOCUS

5.9 Software Quality Standard: Acceptance and Awareness

The last main category that has been identified is the software quality standard acceptance and awareness in VSEs. The purpose of this main category is to understand VSEs acceptance and awareness level on software quality standards. Several interview questions have been asked and discussed with the respondents’ as in appendix B and C. From the analysis as in appendix F, researchers have grouped the interviews answers in 3 important categories in order to explore this main category in details as in table 5.10.

Table 5.10: Software Quality Standard

Sub Category	Category	Main Category
Low acceptable	Level of Acceptance	Quality Standard Acceptance Level
Less Priority		
High awareness adopting Standard	Level of Interest and Awareness	
Standard Benefit Awareness		
Standard Guideline	New Standard criteria	
Light weight Process		
Business and Technical Process		

The first category in this part is to understand the level of quality standard acceptance in VSEs. From the analysis researchers found that all VSEs companies' are not accredited or following any particular standard in their software development process. This could be identified in below interview answer quotes.

> *"With 12 years in the company, I never had a problem with that, a problem with that [need for standards]… No, not an issue or what so ever. Never came to my attention"*
>
> *"I can't say detail about it but I think we don't have plan regarding future adoption of standards."*
>
> *"We always try to improve our process if we have time and resources. In term of formal structure not anything yet to adopt it."*

During the interviews' analysis the researchers have identified several reasons that have been divided to 2 subcategories in order to understand the problem in adopting standards. The first subcategory is on the low standard acceptable issues. The low acceptable in software quality standard is happened due to the perception that the process involved is overly complicated and too detailed. They also added that adopted software quality standard required additional man hours or specific personnel in fulfilling the listed process in order to meet the standard objective and this will involved an additional cost to the company. In addition, they believe that the processes in software quality standard are not easy to follow and adopt. Other than that due to the small size of people working in VSEs, they believe that current software quality standards such as ISO9000 cannot be adapted and followed. In relation to that, all the interviewees believed that involving or adapting software quality standard in their process will increase the project cost and delay the project delivery. Meanwhile, they argue that the process involved software quality standards are not tailored with the current development process, which are more brief, informal and very light in process. Below interviews quotes explains the above situation

> *"In a company of our size they [standards] would not necessarily, I mean we could be more sophisticated in kind of process direction we taken than a larger company. Our company do some basic documentation sign off aspect that will be it just do project work based on that."*
>
> *"Too much documentation and you need somebody to think that understand the software process. Because the developers are busy with coding, so documentation is the last thing they did. We would need somebody to take in charge for that"*
>
> *"I think that ISO and CMMI are serving a large company with huge resources but now like agile development would work here because the team were smaller because we can consider a mile stone, not have to be bared down getting the right procedure in place and in right format."*
>
> *"I am pretty sure we would have to allocate somebody over an extend of period of time to responsible for ISO and to make sure what ever we are required to do is it implemented follow and changes. it comes own to resources and cost."*
>
> *"We all should work toward about standard of delivering software. But I do think that sometime is hard to convey and make afford that actually has to put be put in place to actually meet certain standard. To say that you know I think we nearly reaching standard you might as well aspect a certain amount of cost to be associated with that. You know they know what got to pay, that not going to be cheapest to if came to the smaller work company"*

In additional, the analysis also indicates that the lack of requirement from the market in general and their customer in particular has contributed to low acceptance of such standards. During the interviews it was also shown that accreditation in software quality only the important when companies involved or plan to work with the government body or agency. Contributing to this is the fact that most of the VSEs clients are private, small or individual companies which do not emphasise having standards accreditation. Below interviews quotes explains the above situation

> *"We had never had a problem selling our stuff or not selling our stuff because there no ISO standard. Software is not bought by ISO or any other standard. Is bought by on a Window standard, if really follow window standard then it nice, they want to have it but ISO…. Who cares!!"*
>
> *"I never heard anything from sale that we couldn't sale anything because of ISO standard… I never heard it here. We have here also problem sometime like 48 hours rules standard, that the different standard... Even that one is not an issue here. It is probably Irish way."*

> *"Allocating resources and no requirement from of the product that satisfied customer because what more important is the quality. The ISO and CMMI are more towards the service rather the software product. So we don't that thing."*

The second subcategory in this part is on the low priority issues. The interviews analysis as in appendix F indicates that a software quality standard issue is a low priority task in software development process and activities in VSEs. The interviewees have explained several reasons which indicate the above situation. Not compulsory or low demand of the standard from their client is the main reasons and agreed by all the interviewees. Higher quality of code and delivery time are more important that the evaluation of the development process and rather than adopting any quality standard that has been seen as 'sale tool' only. The interviewees also responds that current software quality standard objective such as ISO are more toward on the management and services of the software development process rather than a software technical issues and product. In relation they also believed that the software quality standards are built for the big companies rather than for VSEs. Below interviews quotes explains the above situation

> *"If you want to get done quickly then what you need is focusing to the output not the process."*
>
> *"A lot of process in quality standard is non sense. Some ISO standards tell you to do XYZ steps but they may be not being beneficial to our business*."
>
> *"We do informal research if we found something cool article I will try to followed to improve our process. But seriously standards quality is not on my list."*
>
> *"Standard is just seen as a sale tool. Is a good sale tool and for small company probably a lot of their client are more looking on ISO 9000 before they looking at u. Probably not interest and it a resources things and all time required to company with ISO 9000. We can do it internally and ask somebody to do it."*

The second category is level of interest and awareness category. This category explains VSEs level of interest and awareness regarding software quality standards. Even though in the first category they have shown low acceptance and priority level regarding software quality standard, this analysis has also shown that there is an indicator that VSEs are interested and aware about adopting and benefit having a quality standard especially the ISO standard. Leading to a quality product, create consistency, improve company image, create consistency in development work, improve work process and good for business are the main points that the interviewees given, which indicates VSEs high awareness and interest about the benefit of having software quality standard. Even company C have explicitly expressed that the company had a planned to adopt the ISO 9000 but due to the several constraints as have been discussed above made the plans have put on hold. This situation shows that VSEs have an interest and aware about the benefit adopting software quality standard. Answers quote below indicates the above situation.

> *"Yes we do plan too, but one of things we done since started growth to quickly, when we haven't , u know' after pushing ourselves we spend time to learning how we want to do this and how we don't want to do this. We started to put those processes in place so when we decide plan to growth we have a good platform, easy to growth, easy to sustain the growth, train people what your process and what your system is"*
>
> *"They [software quality standard] are nice. It would be great to have them in order to have a consistence software up and running that, you always can relied for exampled code and readable fast of people"*
>
> *"Yeah we have that beneficial, yes. I so say yes."*
>
> *"I have mix feeling on that. It can make the organization more organize and do it right and automatically benefit to the company"*

In order to understand more about the software quality standards in VSEs, the researchers asked the interviewees the criteria they considered important in a software quality standard as in appendix B and C. The purpose here is to understand in detail the criteria that should be exist in future software quality standard in order to encourage VSEs seriously involved standard quality process. From the analysis, researchers found several criteria have been proposed by the interviewees. The results also help researchers to categorise the criteria into subcategory as in table 5.10, namely standard guideline, lightweight process and business and technical process subcategory.

In standard guideline subcategory, interviewees have proposed several criteria as below

- Align with current development process style
- Provide detailed guidelines and assistances
- Provide clear templates

- Provide workshop and/or training on how to actually apply it

Below interviews quotes represent the above points:

> *"Small it will basic sources control quality control, security, testing you need certain boxes ticks in those area need to be. All these best practice need to be set out. A company have need to seen be implementing or ready to it."*
>
> *"So what ever function, feature, what ever is going be on that, that is clear and lay out because you do have tendency would be taught of to do that, I thought to have this and thought and special reporting and come out from the system"*
>
> *"So provide some workshop would be good because we don't have much time to investigate and searching a good standard. Prepared some solution for this type of company that each to follow is a nice way"*
>
> *"Suppose to be less documentation, align with our current process, and easy process to follow and tailor with our size."*

In lightweight process subcategory, interviewees have proposed several criteria as below:

- Minimum documentation requirement
- Easy to administer
- Less change from current development process
- Minimum overhead in terms of cost and resources

Below interviews quotes represent the above points:

> *"Should be light administration overhead. And clients also look into the small cost for the project. This is the way the small company compete and doing business."*
>
> *"It must be less time consuming especially involving developers. Must have an easy structure and process and could help people in how it place and structure."*
>
> *"Suppose to be less documentation, align with our current process, and easy process to follow and tailor with our size."*

In business and technical process subcategory, interviewees have proposed several criteria as below:

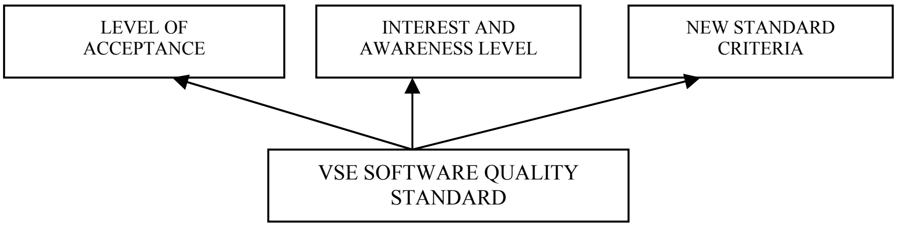
- Align with company existing business and development process.
- Align with others specific software technical standard and process.

Below interviews quotes represent the above points:

> *"The development process should cover the key things in the process to develop high quality software. It also should focus completely on code artefact produce compare to others activities"*
>
> *"The development must align with current development process such as test, functionality, bug track, basic project planning and specification. More on low level issue rather than high level process and more towards software engineer rather than management team."*
>
> *"I think it must align with the process that we have now, Documentation process not so heavy and must be reasonable."*

In summary, this part has explained the acceptance and awareness issues of software quality standards with in VSEs. From the interviews data we have identified 3 main issues or categories that elaborate in details the above issues. Figure 5.8 shows the network diagram for this main category.

Figure 5.8: Software Quality Standard



5.10 Overall Theoretical Diagrams

From the overall discussion above and based on the data analysis we present an important diagram which is the theoretical diagram which illustrates in figures 5.9. This diagram is the result from merging process of all the individual network diagrams and main categories that have been discussed in details in above sections. In details, Appendix G and H, which shown an individual and relational network diagrams, have illustrated the flow of this merging process. In order to align with the research objectives, the discussion in this part will be focused on the software process and process improvement issues.

Figure 5.9: Overall Theoretical Diagram

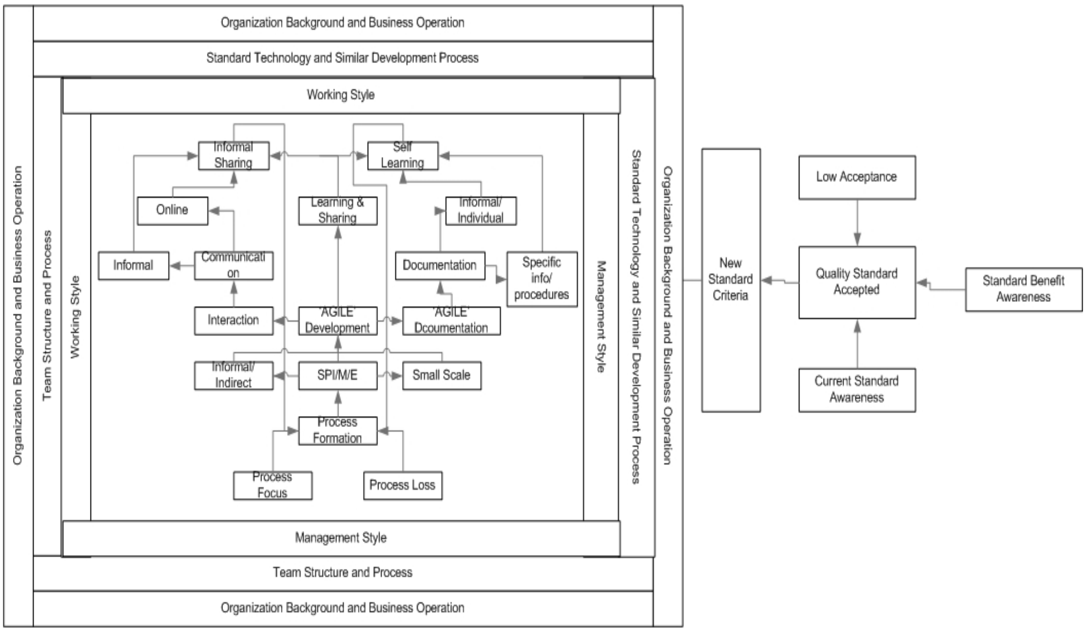

Organization Background and Business Operation
Standard Technology and Similar Development Process
Working Style
Informal Sharing
Self Learning
Online
Learning & Sharing
Informal/ Individual
Informal
Communicati on
Documentation
Specific info/ procedures
Interaction
'AGILE' Development
'AGILE' Dcoumentation
Informal/ Indirect
SPI/M/E
Small Scale
Process Formation
Process Focus
Process Loss
Management Style
Team Structure and Process
Organization Background and Business Operation
Organization Background and Business Operation
Team Structure and Process
Working Style
Management Style
Standard Technology and Similar Development Process
Organization Background and Business Operation
New Standard Criteria
Low Acceptance
Quality Standard Accepted
Standard Benefit Awareness
Current Standard Awareness

Figure 5.9 shows that the software process improvement including the maintenance and evolution processes are started from process loss or process focus which was influences internal and external variables as discussion in part 5.8. The process formation is created and will indicate the process that need to improve, change or upgrade. As discussed in the same part, the improvement, maintenance and evolvement of the software development process in VSEs are done in informal, indirect and small scale at one time but in a high frequent. The analysis in this part also shows that VSEs followed or adopted the agile development approaches which involved a lot of interaction or communication either with the customers or the developers, high focus of the development process and having a minimal documentation process in the organization.

Due to small team size, flat team and organization structure, staffs or management geographical location, autonomous working style and macro project management as explain in part 5.4 and 5.6, the communication process are more become informal and autonomous. Beside that the uses of the communication tools are also being used extensively among peoples in communicate and shared their knowledge.

In term of documentation process, the analysis indicates that in VSEs the documentation processes have been practiced either informally or at individual level as discussed in part 5.7.3. The analysis also shows that due to macro project management style, autonomous working style, standard technology and similar development process have lead to the these situation. Moreover the interviewees admitted that only the issues related to business procedure and technical specification are being formally documented.

The analysis also indicated that due to the informal communication, informal documentation and autonomous work have created the informal and personal organization learning and sharing process as discusses in part 5.7.2. Therefore from learning and sharing process a new idea and weakness of the area that need to improved, change and upgrade. This process will start back at the process formation and iterate.

Continuing from the discussion above, the analysis also indicates the relationship between the software quality standard and software process improvement. Based from the analysis, it shows that the level of acceptance in VSEs is low and no VSEs have accredited or followed software quality standard in their development process. The main reasons are varied, including low requirement from client and market, small team size, time and cost constraints and business priority reason as explained in part 5.9. However, their level of awareness about the benefit of software quality standards is acceptable based on the answered given in the interviews and focus group session. The analysis also indicates the some criteria as indicate in part 5.9 needs to consider by the software industry in order to encourage VSEs in seriously adopted software quality standard.

In overall in this chapter, we present and discuss the relationships between the individual main categories and the network diagram from the above parts. These processes have created theoretical and relational diagrams for the present research study. These diagrams show the flow of improvement process happened in VSE and variables which influence the whole software development process. Beside that diagram also indicate the relation between SPI and software quality standard issues in VSEs.

5.11 Summary

This chapter presents the findings and results of the first stage analysis for the current study. It has been discussed in detail the important categories that influences the VSEs in order to improve, maintain and evolve the software development process. The findings also indicate a pattern on how the development process have been practiced in VSEs and explained on how the knowledge in development process is managed. Beside that, the findings also indicated the positive relationship between KM, SPI and Team as an indication to our research questions. This could be understood from how development team are interacting and sharing each others in mastering and performing their job which related to development process. Furthermore the results also elaborated the acceptance level of software quality standard and its important criteria that should exist in order to encourage VSEs seriously involve and adopt software quality standard. Moreover, the combinations of all categories and criteria have been shown in a theoretical diagram that presents the overall process happened in VSEs. The next chapter discusses the findings and results from the stage two data analysis. The purpose is to enhance researchers understanding on the research issues/questions and also to validate some of the research findings in stage one analysis in order to support our research hypothesis.

Chapter 6 ANALYSIS AND RESULTS - STAGE 2

6.1 Introduction

This chapter present the findings and discussions of the stage two data analysis process. The main analysis process of this stage is based on the survey questionnaire that filled by our research respondents. The results are classified into two types of data which are the qualitative data and the quantitative data. Both data types are analysed and integrated in order to answer all the important variables or issues for the present research study. The discussion on this chapter will begin with the discussions and results on the communication process, sharing and learning process, documentation process, KM issues, SPI issues, software quality standard issues and development team issues

6.2 Communication

The first issue that the researchers want to explore is on the communication process in VSEs. In order to understand this issue, researchers have grouped all related communication questions in the questionnaire into one section and the detail quantitative analysis process has been presented as in appendix I. The purpose of this section is to understand the pattern of the communication process in VSEs. In details, researchers would like to understand how the meeting, feedback, people communication and level of communication have occurred in these companies. Table 6.0 has tabulated results of this issue.

Table 6.0: Communication Process

Employee Size Group		Clear Comm.	Good Team Comm.	Regular Meeting	Regular Feedback	Effective Comm. Channel	Regular Formal Comm.	Regular Informal Comm.
Micro VSE	Mean (Var)	4.80 (0.447)	5.00 (0.000)	2.20 (0. 447)	4.40 (0.548)	4.80 (0.447)	2.20 (0.447)	5.00 (0.000)
Larger VSE	Mean (Var)	4.40 (0.5480	4.80 (0.447)	2.40 (0.548)	4.40 (0.894)	4.40 (0.548)	3.00 (0.707)	4.60 (0.548)
Total	Mean (Var)	4.60 (0.516)	4.90 (0.316)	2.30 (0.483)	4.40 (0.699)	4.60 (0.516)	2.60 (0.699)	4.80 (0.422)

Table 6.0 illustrates that all respondents agreed that the companies have a good communication process but these processes have been carried out in informal ways. The table also shows that the companies did not have a regular formal meeting and practice an informal formal communication in their business operations. These were reflected in column 5 and 8 with the total means results of 2.3 and 2.6, which indicate the low level of formal communication process. This issue has been supported by the last column in the table that shows a strong agreement of the informal communication process with a mean result of 4.8, which indicate the highly informal communication process situation in VSEs.

Furthermore comparison between company employees size, gave researchers more detailed indicator that employee size has influence the formal communication process level in their VSEs daily business operations. In addition, table 6.0 also point out that that Micro-VSEs are more prone to practice informal communication process compare to Larger-VSEs. This issue is reflected in regular meeting, regular formal communication and regular informal communication columns in table 6.0.

Referring to all answers quotes in appendix J, the informal communication process issues have been continuously discussed indirectly by the respondents. This issue could be identified on communication process *(B2, D1, D2 and E1)* answers. In question about training *(B2)*, the respondents' answers indicated that most of the training is not given formally. They claimed that the staff enhanced their skills through self learning, sharing between team member, informal feedback, internal training and 'on the job training' process. The results also support the quantitative analysis results above that communication process in VSEs are mainly through informal and indirect process. Below are the samples of answers quotes that illustrated the communication process in VSEs:

> *"No requirement for same has been identified to date. We have some simple documents defining coding practices and the tools we use. From there we manage or train through feedback as progress made"*
>
> *"We are doing more on self learning and sharing among us"*
>
> *"Internal, self learning and on the job training"*

The issues of informal communication process in VSEs also have been raised in questions that related to learning and sharing process *(D1 and D2).* The answers given have indicated and explained in details the way of communication being practiced in VSEs. The answers quotes below showed that instead of face to face communication, the online communication between people is actively been used in VSEs.

> *"We Informal sharing and changing knowledge… We documenting electronically and having Informal transferring and knowledge sharing"*
>
> *"We all talk all the time about the work we do, the problem we have and what solutions we can use. People must talk all the times to exchange information"*
>
> *"We talk a lot and we use email as well. We did look at using blog, but that didn't really work out well for us"*
>
> *"Informally through ad-hoc conversations, sharing and some code review"*

Meanwhile question about feedback *(E1)* showed that 90% of respondents are agreed that in development projects, they are regularly receiving feedback from the project stakeholders. However the result showed that this process has been done either in informal discussion, online communication and informal internal feedback. The answered quotes below illustrated how the process has happened:

> *"Online communication, informal feedback, internal discussion, informal communication"*
>
> *"We sit in one office so I talk to them all the time"*
>
> *"Informal meeting, casual communication, report"*

The results of this part of analysis have highlighted a pattern or an indication the way of communication has been practiced in VSEs. It shows that the communication process in VSEs is mainly through an informal communication. The informal communication process happened in the context of indirect, online, internal discussion, casual communication, informal knowledge sharing, informal feedback and self learning activities as been quotes by the respondents. The result of this process has a similarity finding with the stage one analysis. This gave the researchers a clear picture that the communication process in VSEs has mainly been done in an informal or indirect way.

6.3 Learning and Sharing

In general learning and sharing process in VSEs is determined and influenced by the communication process in VSEs. Therefore in order to validate this issue, researchers have developed a group of questions that related to learning and sharing activities questions as in the research survey questionnaire. For the purpose of the analysis, all the related questions are grouped into one section and the detail quantitative analysis process has been performed as presented in appendix I. The purpose of this section is to understand the pattern of how people are learning and sharing their development knowledge in performing their tasks. Table 6.1 has tabulated the results from the analysis toward this issue.

Table 6.1: Learning and Sharing Process

Employee Size Group		Learning and Sharing- Exploit Exist Org Knowledge	Learning and Sharing - Learn Past Experience	Learning and Sharing –Collect Past Experience
Micro VSE	Mean (Var)	4.00 (0.707)	4.20 (0.837)	4.00 (0.707)
Larger VSE	Mean (Var)	4.40 (0.894)	3.80 (0.837)	3.40 (0.548)
Total	Mean (Var)	4.20 (0.789)	4.00 (0.816)	3.70 (0.675)

From the table 6.1, it is clear that all respondents agreed that their development team is always sharing their knowledge and experience with others in the organization. All the questions results in this groups obtained more than 3.00 point in mean analysis. This represents an indicator that in VSEs companies, they always utilize the knowledge and experience within the organization in performing their tasks. This analysis also found out that there are no big differences in terms of company size in utilizing existing knowledge and experience in company.

According to data analysis results in appendix J, the learning and sharing activities in VSEs are being done either informal, self learning or informal sharing among the development team. This has shown in question on how the employees enhance their skills *(B2)* which resulted 90% of the respondents agreed that no formal training were given to the staff in enhancing their skills. Moreover the details explanation of this situation could be found in questions on learning and sharing issues *(D1, D2 and E2)* answers quotes as below.

> *"Informally through ad-hoc conversations and some code review"*
>
> *"Informal sharing knowledge/Changing/documenting electronically/Informal transferring and knowledge sharing"*
>
> *"Ensuring that no single member of staff has any exclusive knowledge by using a mentoring/buddy system."*
>
> *"Ensuring everyone talks and exchanges information about projects on an ongoing basis we can mitigate against leaving the company or forgetting knowledge"*
>
> *"Having equal right in companies share and decision/Having active sharing knowledge in development process/Teams are companies' main stakeholder"*

The result of this part of analysis gave researchers a pattern and indication that in VSEs people are always exploiting and referring to experience and knowledge that available in the organization in performing their task. The analysis also shows that the learning and sharing style is influenced by the communication process as discussed in part 6.2 and the relation of these two processes also have discussed in data analysis stage one.

6.4 Software Communication Tools

Software communication tools such as skype, email and others related online sharing tools are among the communication tools that VSEs have always used while performing their development tasks. From the answer quotes in part 6.2 and part 6.3, researchers found that software communication tools issue kept been stated by the respondents in answering how communication, learning and sharing process occurred in their organization. Therefore in order to validate this issue, researchers have grouped all related software communication tools questions in the questionnaire into one section and the detail quantitative analysis process has been presented in appendix I. The purpose of the section is to understand the pattern of how important and active people are in using the software communication tools to enhance their knowledge to perform development tasks. Table 6.2 has tabulated results from the questionnaires analysis in order to show respondents perception toward this issue.

Table 6.2: Software Communication Tools

Employee Size Group		Tools - Good Team Communication	Tools – Enhance Knowledge Propagation	Tools- Software Development Project
Micro VSE	Mean (Var)	5.00 (0.000)	4.60 (0.548)	4.80 (0.447)
Larger VSE	Mean (Var)	4.80 (0.447)	3.60 (0.548)	3.80 (1.095)
Total	Mean (Var)	4.90 (0.316)	4.10 (0.738)	4.30 (0.949)

From the table 6.2 it is obvious that all the respondents agreed that they have a good team communication process in their organization. This is showed in overall mean score which obtain 4.90. Table 6.2 also indicates that people in organization always used the software communication tools either to enhance their knowledge or helping them in performing their tasks in software development. This has been showed in the table that the overall mean score for both issues is 4.10 and 4.30. In addition the analysis shows that the use of software communication tools in performing and enhancing knowledge in software development process is depending on the size of the companies itself. The result also indicated that the smaller size companies are more dependent on software communication and development tools in communicate and enhance their development skills compared to bigger size companies.

6.5 Documentation Process

In this part, the researchers aim to understand respondents' opinions on documentation process and how the documentation process activities happened in their companies. Several questions have been asked in the questionnaire in various places and all related questions have been grouped in appendix I. From the analysis as showed in table 6.3 indicates that in principle respondents are agree that a clear documentation processes and guidelines are important in order to sustain people knowledge in software development. This showed in the third column in the table which shows most of the respondents are agree with the question. Appendix I, presents the details quantitative analysis process that has been performed.

Table 6.3: Documentation Process

Employee Size Group		Formal Doc – Clear Doc Process and Guideline	Formal Doc- Staff Knowledge	Formal Doc- Project Exp and Lesson Learned	Formal Doc – Activities and Progress	Formal Doc – Knowledge/ Experience	Formal Doc – Works Progress and Procedure
Micro VSE	Mean (Var)	3.40 (0.548)	2.20 (1.095)	2.20 (0.837)	2.60 (0.548)	2.20 (0.447)	2.20 (0.447)
Larger VSE	Mean (Var)	4.00 (0.000)	2.80 (0.837)	3.20 (1.095)	3.40 (0.548)	2.80 (0.837)	2.60 (0.548)
Total	Mean (Var)	3.70 (0.483)	2.50 (0.972)	2.70 (1.059)	3.00 (0.667)	2.50 (0.707)	2.40 (0.516)

But in practice, table 6.3 indicates that people knowledge, experience and activities are not documented properly and the documentation process has been done more in informal process. This can be referred to the total mean score for all the other questions in table 6.3 presents that all respondents do not practice a formal documentation process in their documentation activities. Table 6.3 also indicates that number of employees working in the companies give an influence to the documentation formality process in VSEs.

In relation to the above, appendix J has highlighted that only activities that are related to business procedure and technical issues are being documented properly and organized. This could be identified in question on documentation process *(C2)* where 50% of the respondents claimed they felt that they are regularly update their document regularly especially on a specific works and procedures. Moreover the analysis results also showed that small team size issue has hindered VSEs from perform seriously documenting their activities. This could be identified in the answers quotes given by the respondent in the questionnaire as below.

> *"We always document project and work specifications … We constantly updates until such time as the represent the change /work to be complete"*
>
> *"We documented it electronically, and having an equal decision on it"*
>
> *"We are too small to do proper documentation process"*
>
> *"We talk a lot and we use email as well. We did look at using blog, but that didn't really work out well for us"*
>
> *"We did informal documentation process; no standard applied and staff initiative"*

The result in this part of analysis have gave researchers a pattern and indication that in VSEs documentations process are done in two ways; (1) the specific documentation process which is related to business and technical process and (2) informal documentation process which are inclined toward informal, personal and online documentation.

6.6 KM Issues

In this part, researchers would like to explore more direct issues that are related to KM process. Various issues have been brought up in the questionnaire that relates to KM included organization KM strategy, good leadership; project post mortem, training and reward issues. Beside that several additional issues that are closely related to people and management commitment toward KM also have been asked. The objective is to understand the KM process in VSEs and how the peoples' commitment to this issue. Appendix I showed lists of KM related questions that have been grouped from the original questionnaire and presents the details quantitative analysis process that have been performed.

Table 6.4: KM process

Employee Size Group		KM- KM Strategy	KM- Good Leadership	KM-Post mortem	KM-Formal Training	KM- Reward Incentive Knowledge Sharing
Micro VSE	Mean (Var)	3.40 (0.548)	4.60 (0.548)	2.40 (0.548)	1.40 (0.548)	2.60 (0.548)
Larger VSE	Mean (Var)	4.00 (0.000)	4.40 (0.548)	2.00 (0.000)	2.40 (0.548)	3.00 (0.000)
Total	Mean (Var)	3.70 (0.483)	4.50 (0.527)	2.20 (0.422)	1.90 (0.738)	2.80 (0.422)

Table 6.5: KM Commitment

Employee Size Group		KM-Management Commitment	KM-Good working Relationship	KM-Regular Share Opinion & Thought	KM-Regular Share Experience
Micro VSE	Mean (Var)	4.40 (0.894)	4.80 (0.447)	4.40 (0.548)	4.20 (0.837)
Larger VSE	Mean (Var)	3.40 (0.548)	4.40 (0.548)	4.40 (0.548)	4.00 (0.000)
Total	Mean (Var)	3.90 (0.876)	4.60 (0.516)	4.40 (0.516)	4.10 (0.568)

Table 6.4 indicates that in principle respondents agreed they are having a clear KM strategy and a good leadership in their organization is important in manage software development knowledge. These have been shown in the mean score results for these two questions. However the results in table 6.4 indicate that activities related to KM within VSEs have not been performed properly. It is indicated in average total mean row that gained less than satisfied agreement level. Meanwhile, in table 6.5 showed that the management are very supportive in the KM process and people in the organization are always communicating, sharing and having good relationship among them.

In relation to the above, appendix J also highlights all respondents reported that the knowledge created in the organization is not formally documented. This issue could be identified in questions related to KM process *(D1, D2 and E1).* From these questions analysis results, researchers' found that the KM process was done informally through sharing activities and informal documentation such as personal or impromptu process. This could be clearly understood from the answer quotes given by the respondents in the questionnaire as below.

> *"We are doing more on self learning and sharing among us"*
>
> *"Always sharing knowledge informally - since this is family business / having informal regular meeting"*
>
> *"Regular sharing process, internal sharing and team work"*
>
> *"We informal sharing and changing knowledge. We always documenting electronically/ and having informal transferring and knowledge sharing"*
>
> *"Ensuring that no single member of staff has any exclusive knowledge by using a mentoring/buddy system".*

In addition to the above analysis, the answers on knowledge loss issue *(D2)* have indicate that the informal process environment in VSEs helps the companies to mitigate knowledge loss problems from happened. The analysis in this part showed 90% of the respondents claimed did not facing a knowledge loss problem in their company due to the informal process. Below answers quotes reflects the above situations.

> *"Ensuring that no single member of staff has any exclusive knowledge by using a mentoring/buddy system."*

> *"Not a problem since we are using same technology and process in all our project…. We occasionally sharing and transferring knowledge among brothers"*
>
> *"Ensuring everyone talks and exchanges information about projects on an ongoing basis we can mitigate against leaving the company or forgetting knowledge"*
>
> *"Informally through ad-hoc conversations, sharing and some code review"*

The result in this part of analysis produce a pattern and indication on KM process in VSEs. The results showed that in overall it can be determined that KM process such as knowledge creation and sharing are actively been done in VSEs but the method are towards informal rather than formal. This result is aligned with the stage one analysis result.

6.7 SPI - Process Improvement and Assessment

In this part, we would like to explore on the issues of software development process activities in VSEs. The questions are more specific towards the method, improvement, practical and assessment of their software development process activities. Appendix I shows list of software development process related questions that have been grouped from the original questionnaire and presents the details quantitative analysis process that have been performed.

Table 6.6: SPI- Process Improvement and Assessment

Employee Size Group		SPI – Change and Evolve	SPI– Regular Assess	SPI – Follow Update Process	SPI – -'Agile' Type development
Micro VSE	Mean (Var)	3.80 (0.447)	3.40 (0.894)	3.20 (0.837)	4.20 (0.447)
Larger VSE	Mean (Var)	4.00 (0.707)	3.60 (0.548)	3.20 (0.837)	4.20 (1.304)
Total	Mean (Var)	3.90 (0.568)	3.50 (0.707)	3.20 (0.789)	4.20 (0.919)

Table 6.6 indicated that in general respondents agreed that their software development processes rapidly change and evolve overtime. They also claimed that their development process is regularly assessed and staffs always followed or applied the latest development process method. Table 6.6 also indicates that respondents claimed that they are following an agile development philosophy in their development process.

In relation to the above, finding in appendix J also highlighted that 90% of respondents felt that their development process evolve overtime. They stated that following the best practice, client requirement, team size growth, new idea and keep up with the technology change are the reasons for the improvement and evolution of development process. Below answers quotes as in question software development process status *(C1)* gave an indication on how the development process have been improved and evolved.

> *"Software process change is due to growth of the organization. We started out as 2 people 4 years ago and now have 11, so things had to change along the way"*
>
> *"It will evolve as we grow in size and get more applications in production environment"*
>
> *"Not really. We still do the same basic thing in software process; we change some aspects of how we work. It's a little bit ad-hoc... We follow agile method… I suppose"*
>
> *"Our software process will evolve as we grow in size and get more applications in production environment"*

Beside that in question on process loss issues *(C3)* as in appendix J also shows that almost all or 80% of respondents' claimed that their software development processes are not affected by the process loss problem. They claimed that by using standard development tools, similar development process, having frequent guidance and mentoring activities, active in knowledge sharing and proactive coaching could avoid the process loss problems in software development process. Below answers quotes are explained the situations.

> *"As a manager, I don't believe in using the latest and greatest techniques for the sake of it. We'll use something that fits our team dynamics and we'll spurn something that doesn't… whether that counts."*
>
> *"Our document process mostly electronically…we always sharing knowledge informally. Since this is family business, we always having informal regular meeting"*
>
> *"I trust people to do the thing right. If I don't agree with them then I tell them what to do and they will do it. I am a manager after all"*
>
> *"Not really, we still do the same basic things in our software development method. We change some aspects of how we work. It's a little bit ad-hoc... Agile method… I suppose"*

However the respondents also admitted that “laziness” attitudes among the staffs and practicing informal and rapid changes in software development process are among the factors that could lead the process loss problem in software development process.

6.8 SPI – People and Management Involvement and Commitment

Continue from part 6.7, researchers would like to explore on how development team and, management commitment and involvement is in software process improvement activities in VSEs. The questions on this part are stress particularly on the level of team involvement in planning and setting the development process and procedure in the software development projects. Appendix I, shows a list of software development process related questions that have been grouped from the original questionnaire and presents the details quantitative analysis process that have been performed.

Table 6.7: SPI- People Involvement and Commitment

Employee Size Group		Direct Involvement Dev Process Planning	Autonomous Work	Team- SPI Setting Goals	Direct Involve Development Process and Procedure
Micro VSE	Mean (Var)	4.20 (0.837)	4.20 (0.447)	3.80 (0.447)	4.00 (0.000)
Larger VSE	Mean (Var)	4.40 (0.548)	4.40 (0.548)	3.40 (0.548)	4.00 (0.707)
Total	Mean (Var)	4.30 (0.675)	4.30 (0.483)	3.60 (0.516)	4.00 (0.471)

Table 6.8: SPI- Management Involvement and Commitment

Employee Size Group		Management- Guide and Support	Management- Staff High Motivated	Management- Staff Recognition	Management – High Support in SPI process
Micro VSE	Mean (Var)	4.20 (0.447)	4.80 (0.447)	4.40 (0.548)	3.00 (0.707)
Larger VSE	Mean (Var)	4.20 (0.447)	4.20 (1.304)	4.00 (1.225)	3.00 (0.707)
Total	Mean (Var)	4.20 (0.422)	4.50 (0.972)	4.20 (0.919)	3.00 (0.667)

Table 6.7 indicates that the respondents agreed that the level of development team involvement in software development process and planning are very significant. This could be identified with the average mean score for this question is relatively high. Moreover table 6.7 also clarified that even though the development staff working autonomously but they are also actively involved in setting goals, planning and procedures in the company's software development process. Meanwhile, table 6.8 shows the level of management commitment in the improving current software development process. From this table, researchers could indicate that the management has provided their full support in SPI process. This situation is shown in the total mean score for each questionnaire on this issues which more on the positive rather than negative. Therefore, this gives an indicator of the seriousness and high commitment of management in software development process.

From the answers quotes given by the respondents as in appendix J, researchers could understand more details about the above issues. Answers quotes below are explained how serious staffs are involved in software development process.

> *"Explicit requests for input and feedback on any thoughts/ideas for changing the way things are done"*
>
> *"Having equal right in companies share and decision/Having active sharing knowledge in development process/Teams are companies' main stakeholder"*
>
> *"I welcome input from developers on what we are doing and how to make it better"*
>
> *"Direct on the best way to develop, easiest processes"*

The results in this part of analysis gave a pattern and indication that in VSEs development and management team are very supportive and serious in improving their development process in order to produce a quality product.

6.9 SPI - Goal and Planning

In this part, researchers would like to extend our understanding on software development process activities in VSEs. The questions are more specific towards the companies' goal and planning toward SPI. Appendix I showed list of software development process related questions that have been grouped from the original questionnaire and the details quantitative analysis process that has been performed.

Table 6.9: SPI- Goal and Planning

Employee Size Group		Establish SPI Goal	Broad Understand SPI Goal	SPI Goal Align Business Goal	Balance Short and Long Term SPI	Understand Software Development Project Goal
Micro VSE	Mean (Var)	2.00 (1.000)	2.60 (0.548)	2.60 (0.894)	2.60 (0.548)	4.60 (0.548)
Larger VSE	Mean (Var)	3.60 (0.548)	2.80 (0.447)	3.00 (1.000)	2.80 (0.837)	4.20 (0.447)
Total	Mean (Var)	2.80 (1.135)	2.70 (0.483)	2.80 (0.919)	2.70 (0.675)	4.40 (0.516)

Table 6.9 indicates that the respondents are agreed that in general they are clear about the specific goal of the companies' software development projects. This can be identified with the high score in mean analysis regarding these issues. However, table 6.9 also highlighted that VSEs do not have a proper plan and well understand on software process improvement issues. In details, the analysis in table 6.9 shows that all respondents agreed that the companies do not have a proper SPI goal either for short term or long term. They also admitted that the companies SPI goal are not align with their business goals. It is also indicates that the size of the companies gives an influence in setting and planning companies SPI goals and objectives.

In relation, the comparison between table 6.6 and table 6.9 provides a detail indication that in VSEs the improvement process has been done in a rapid way but in a small scale and informal process. It is also showed that VSEs did not have a specific procedures or documented specific plans in improving their development process but more toward informal and direct improvement of the process. These findings also aligned with the first stage analysis which stated that the improvement processes are performed in an informal way or have been done at a small scale but in a rapid process.

6.10 Development Standard Process and Quality Standard

In this part, researchers would like to explore VSEs acceptance and views on software quality standard issues. Appendix J shows that currently all the respondents' processes didn't accredit to any type of quality standards certificate. From the survey researchers found that 60% of them point out that the companies did not have a plan in adopting any quality standard in a near future. No demand from the market and customer, no implied benefit and not important to their business are the mains reasons behind this issues. Therefore, in order to validate these issues researchers have been asked several close ended questions in the questionnaire. The questions have been group in 2 different segments; (I) reason of not adopted quality standard and (ii) the development process standard; in order to have a clear picture on this issue. Appendix I show the list of software development process related questions that have been grouped from the original questionnaire and details quantitative analysis process that have been performed.

Table 6.10: Software Quality Standard

Employee Size Group		Standard- Not Required by Customer	Standard- Lack Support from Mgmt	Standard – Lack Internal Resource	Standard – Lack Support From Govt	Standard Length and Difficult	Standard – Insufficient Time	Standard - Insufficient Guidance	Standard- No Desire
Micro VSE	Mean (Var)	5.00 (0.000)	2.40 (1.949)	3.80 (0.837)	3.00 0.000	4.00 (0.707)	4.00 (1.000)	3.40 (0.548)	3.80 (1.095)
Larger VSE	Mean (Var)	4.40 (0.548)	3.00 (0.707)	4.00 (0.707)	3.00 0.000	4.00 (1.000)	3.80 (0.837)	4.00 (1.000)	3.80 (0.447)
Total	Mean (Var)	4.70 (0.483)	2.70 (1.418)	3.90 (0.738)	3.00 0.000	4.00 (0.816)	3.90 (0.876)	3.70 (0.823)	3.80 (0.789)

Table 6.11: Software Development Standard

Employee Size Group		Dev Standard- Consult Standard in SPI	Dev Standard- Mgmt Knowledge in Quality Standard	Dev Standard- Staff Knowledge in Quality Standard	Dev Standard- Mgmt Ensure Staff Follow Update Process
Micro VSE	Mean (Var)	1.60 (0.894)	3.40 (1.140)	3.00 (0.707)	3.80 (1.095)
Larger VSE	Mean (Var)	2.40 (1.342)	3.20 (0.83)	3.40 (1.140)	3.80 (0.837)
Total	Mean (Var)	2.00 (1.155)	3.30 (0.949)	3.20 (0.919)	3.80 (0.919)

Table 6.10 indicated the main reason of not being adopted to quality standard is mainly because of the lack customer requirement in quality standard. This has been validated by the lower total mean score in the analysis result in table 6.10. Beside that table 6.10 also indicates that the length and difficult process, insufficient time and less internal resources are among other of the important reasons of why the VSEs are not interested in adopting any quality standard. Meanwhile, table 6.11 explains that even though the respondents agreed that the development team and management people in the organization are knowledgeable in development quality standard but they did not refer to any development standard or model in improving the software development process quality.

In relation to the above, appendix J showed that there are some criteria that need to be considered in order to encourage VSEs in seriously take part in adopting software quality standard. Light weight process, low overhead, supplement of training, align with current development process, clear and simple template and rapid assistance from the expert are among the criteria that should be considered by the related body or institution in developing a new quality standard specifically to this size of organization. These requirements could be identified in answers quotes:

> *"Lightweight Process, related to technical, not high overhead, follow current guideline"*
>
> *"Any assistance useful (financial to cover learning cost)"*
>
> *"Training"*
>
> *"External expert"*
>
> *"Less Overhead, Template and Training"*

The results of this part of analysis gave researchers a pattern and indication that the acceptance level of quality standard such as ISO among VSEs are still low even though the staff and management are knowledgeable and aware the benefit about adopting quality standard. The main reasons more related to lack of the customer requirement and the limited resources in the company. Beside that the heavyweight process especially the documentation, cost and not align with current development process are among the reasons why the companies did not plan to adopt it. However from the analysis the companies still interested in quality standard if meet certain important criteria that have a very close related to their tasks.

6.11 Team Dynamics and Structure

In this part, researchers would like to explore respondents' opinions on the companies' software development team status and study people working relationship and team environment in the companies. Appendix I showed list of team relationship and environment issues related questions that have been grouped from the original questionnaire and the details qualitative data analysis process that have been performed.

Table 6.12: Team Dynamics

Employee Size Group		Team Dynamic – Good Working Relationship	Team Dynamic – Regular Share Opinion & Thought	Team Dynamic – Good Social Relationship	Team Dynamic – Good Interpersonal Skill	Team Dynamic –- Closely Located
Micro VSE	Mean (Var)	4.80 (0.447)	4.40 (0.548)	4.80 (0.447)	4.40 0.548	4.40 1.342
Larger VSE	Mean (Var)	4.40 (0.548)	4.40 (0.548)	4.00 0.000	4.20 0.447	5.00 0.000
Total	Mean (Var)	4.60 (0.516)	4.40 (0.516)	4.40 0.516	4.30 0.483	4.70 0.949

Table 6.13: Team Structure

Employee Size Group		Team Structure – Clear Roles	Team Structure Appropriate Size	Team Structure –Diverse Skill Range
Micro VSE	Mean (Var)	3.60 (0.548)	3.20 (1.095)	3.60 (0.548)
Larger VSE	Mean (Var)	3.60 (0.548)	3.40 (0.894)	4.00 (0.707)
Total	Mean (Var)	3.60 (0.516)	3.30 (0.949)	3.80 (0.632)

Table 6.12 indicates that the respondents agree that the team development in their companies have a high level of team dynamics. The results shows that the team have a great working and social relationships, willing to share opinion and idea, having a good interpersonal skill and working closely each other. Results in table 6.13 have details up researchers understand regarding team environment in VSEs. The results show that even though VSEs having a small team and a flat structure but

staffs are clear about their roles, they have enough manpower and skill to do all the development tasks.

Meanwhile appendix J shows that all respondents claimed that their development team are efficient and effective. They claimed that their development team are having all important criteria such as high skills, motivated, dynamic, socialize and good teamwork, open communication, able to meet project deadline and budget, active in sharing and involved in strategic planning. These could be identified in following question on team status *(E3)* answers quotes.

> *"They get well as a social group and communicate regularly/openly. Also the projects we manage are normally 1 to 2 man projects and hence easily manage in an ad-hoc manner by two people that get on and communicate well."*
>
> *"We practice a clear communication and we are active in informal knowledge sharing. Beside that our environment is family culture and, following specific strategic planning... We also actively use communication tools."*
>
> *"They are highly skilled and motivated, great team atmosphere"*
>
> *"We run project on time and, on budget"*

Beside that the result on employee turn over rate *(B3)* question has strengthen the above finding regarding team environment in the VSEs. The result in this question shows that the companies do not have any serious problem with the staff turn over. They claimed that the company environment, management and working styles and team relationships that satisfied the employees have motivated people to stay longer in company. Below are the answers quotations gathered in question B3 which explained the details of the situations.

> *"We handle many varying projects of different sizes and complexities and have a very loose/informal and friendly atmosphere. This means the work is challenging and rarely gets boring while it also being enjoyable here."*
>
> *"We are a young company, so nobody has left us to date. One person went part time to look after her children in afternoons, so I guess being flexible helps."*
>
> *"We have 14 employees. Last one who resigned in was 3 years ago. The reason people stays is we operate in relax and informal environment….We also pay top percentage."*

> *"Staffs are direct involved in any decision in our activities. We work in family and flexible environment, good team environment."*

In overall team environment issue could be identified in most of the answers quotes given by the respondents as in appendix J. These situations give an indicator that all the above parts or processes are much related and depended to the organization team environment, process and culture in the organization.

6.12 Summary

This chapter present the findings and results of the second stage analysis for the current study. It also discusses and presents in details the analysis process and results on the main issues for the study. The results that gathered from the analysis also indicated the agreement results with the first stage analysis. The next section will try to discuss and merging both stage one and two finding in answer the present research questions and hypothesis.

Chapter 7 EVALUATION AND DISCUSSION

7.1 Introduction

This chapter presents the overall findings of the present research. In this chapter the researchers will merge both the finding and discussion from stages one and two data analysis process in light present study research questions and hypothesis. The discussion in this chapter will follow the sequence of the research questions and hypothesis that have been listed and highlighted in chapter 2. At the end of this chapter, we discuss the overall findings in order to further understand the whole SPI processes that have been practiced in VSEs.

7.2 Findings – Research Questions and Hypothesis

7.2.1 Research Question 1

> *"RQ1: What is the current status of Software Process Improvement among Irish software VSEs?"*

The detailed answer and discussion for the above research question have been explained in chapters 5 and 6. In summary, both analyses have explained that the SPI programme in VSEs is done informally, indirectly and reactively. The analysis has identified eleven main categories that relate to and influence the SPI programme in VSEs. This overall environment which has been shown in the categories, categories relationship, details individually categories network diagram and a whole diagram models has represented as the theoretical models for the present research. The analysis has confirmed and is aligned with the previous research, that the improvement process was started with a process loss or focus which is influenced by internally or externally environments. In additional the analysis has confirmed that in VSEs in specific and small company in general, the issues of human or people need to be considered as important as the technology issues. This should be focused on how the people in VSEs are learning and sharing their knowledge process in software development which is very active and highly in team dynamics. Even though both processes have been identified as informal and not organized, this situation has also created a freedom to the staff to contribute ideas and participate actively in software development process. The analysis has also indicated that this culture has helped VSEs to mitigate the problem process erosion and knowledge atrophy problem from happened. Likewise in stage two data analysis process (quantitative analysis), 90% of the respondents believed that their software development process evolve over time This could be identified from the mean score as in table 6.6, 6.7 and 6.8 that shows that they are have a high commitment, regular update and evolve in software development process. However, the results as in table 6.9 have confirmed that the SPI programmes or activities in VSEs are done informally, indirect and reactive. Therefore the results have shown consistency findings with previous study (Coleman, 2006; Niazi et al, 2006; Johanssen, 2004)

7.2.2 Research Question 2

> *"RQ2: How software knowledge in Irish Software VSEs should be managed in order to maintain and evolve software process?"*

In order to answer the above research question, we have completed an analysis on the 3 main important issues in both stages of data analysis which are related to the KM process. The issues are the communication process, learning and sharing process and documentation process. Stage one data analysis has shown a consistent pattern of communication process in VSEs. The result indicates that the communication processes are happened either through direct and informal process, which happened either through 'stand up' meeting, periodic and individual or online communication which used extensively the communication tools such as email, skype and phone. The active usage of online communication has been done especially with staff whose work either outside Ireland or in different locations. These results are validated by the second stage data analysis which indicated that the communication processes are very informal and active as shown in table 6.0. In addition the result also indicated that 90% of respondents' felt that the feedback process from the stakeholder is very active but very informal and in indirect way.

Moreover the analysis in this second stage also indicate that the use of communication tools in communicate and propagate knowledge among staff in VSEs are very significant as shown in table 6.2.

The second issue is on the learning and sharing process. For this issue the results show a pattern that in VSEs learning process are done through self learning which is an informal self-initiative which included self exploring, continuous expert guidance and on the job training. Meanwhile the analysis results indicate that due to small team size, autonomous working style and macro management style, the learning and sharing process also happened in indirect and informal ways either trough peer or online sharing. In relation, the second stage data analysis has confirmed that the learning and sharing process in VSEs are very active as shown in table 6.1 and 90% of respondents' are confirmed that the sharing process are done informal and indirect ways.

The third issue is on the documentation process. Due to small number of staff, autonomous working style, macro management style and applying similar software development method in all software development activities have lead the documentation process being done informally and individually in VSEs. In addition in first stage analysis results, we found that the respondents' stressing that producing high quality product is their main priority compare to having a detailed documentation process in their organizations. From the analysis we also found that the documentation activities are more on specific information and procedures and also depends on individual initiatives. Meanwhile stage two data analysis results have validated the first stage data analysis results which indicate the informal documentation process level in VSEs as shown in table 6.3. The results of stage two data analysis also indicated that 50% of respondents' agreed that they their documentation process is more emphasized on general business procedure and works rather than on software development process. The analysis also indicated that even though the KM process is informal but respondents' perceived that their organization have a clear KM strategy and high commitment in KM as shown in table 6.5. This perception could be measured with 90% of the respondents claimed that don't face any kind of knowledge loss problem in software development process.

Therefore from the results in both data analysis stages, we can map the current result with the KM framework in software development process as defined by Dingsoyr and Conradi (2002), which has been discussed in detailed in part 3.4.5, as below,

- Strategy – The results in both data analysis process indicates that VSEs implement informal strategies which include the informal and indirect communication process, unstructured learning and sharing process and informal and individual documentation process.

- Process – Due to the informal environment, such as autonomous working style, macro management style, small number of employees and similar and standard development process, have lead the KM process to informal environment either in communication, learning, sharing and documentation process.

- Tools – Due to small team size, distributed environment and autonomous working style have lead VSEs in relying on the communication and sharing tools in order to communicate, sharing and enhanced their knowledge in software development process.

The results also have shown an agreement with the previous research works that have been done by several others researchers including Kukko et al (2008), Chan and Chao (2008) and Desouza (2003). Therefore from the above explanation we can indicate that H1 research hypothesis has been supported.

> *"H1: There is a positive relationship between the organised management of software development knowledge, and software process and process improvement in Irish software VSEs."*

7.2.3 Research Question 3

> *"RQ3: How should software teams be organised in order to support software process improvement in Irish Software VSEs?"*

From the first stage data analysis process, we have identified that software development teams issues is one of the important main categories that influence the software process and process improvement in VSEs. The results also indicate that due to small number of employees that the team structure and process are very flat and informal in VSEs. These issues have also determined the formality level of software process improvement activities in VSEs. The results also indicate that these issues also affect other critical main categories which related to VSEs software process and process improvement as discussed in chapter 5. The analysis results also indicates that the small size teams have closer the gap between employees in VSEs have create an active communication either direct or via online. This also have create a close relationship which create a between software development team and indirectly create a high level of team dynamics and knowledge sharing activities in software development activities. Beside that the external environment such as macro management style; autonomous working style active feedback from peers and management and direct involvement of management people in software activities which also have created conducive environment to the software development team in VSEs.

Meanwhile in data analysis stage two, we found that all respondents agreed that the software development team dynamics is very high. This could be identified from the results gathered as in table 6.12. The results in table 6.12 also indicated that the smaller the team in VSEs the higher level of team dynamics will be presents in the organization. In addition, the analysis also have indicated that VSEs staff have all the important criteria such as high skills, high motivated, active in sharing, direct involvement and open communication, which are important in software development process. The results also have shown an agreement with the previous research works that have been done by several others researchers including Dyba (2005), Hall et al., (2007) and Hoegl and Proserpio (2004). Therefore from the above explanation and results gathered, has demonstrated support for H2.

.

> *“H2: There is a positive relationship between organized software development teams, and software process and process improvement in Irish software VSEs.”*

7.2.4 Research Question 4

> *“RQ4: How can knowledge and teams could support each other in order to prevent process erosion and knowledge atrophy in software development process within Irish Software VSEs?”*

In order to answer the above research question, we have combined our analysis on research question 2 and research question 3 above. The analysis has indicated that process erosion and knowledge atrophy problem are not a serious problem in VSEs. From the analysis we found that due to small team size which creates a flat work structure, direct and active communication, close relationship and open environment have encouraged software development teams to share and create knowledge in organization. In addition the analysis in the first stage which indicate that management style in VSEs which is more informal and macro, and working style which more autonomous have helps VSEs mitigate several factors which lead to process erosion and knowledge atrophy. This is shown from the analyses which have indicated that in VSEs knowledge sharing level is high; staff turn over rate is low, high levels of knowledge exploration, continuous guidance from the senior staff and active communication in exchanging idea or knowledge among staff.

Meanwhile in second stage data analysis process indicates that 90% from our research respondents believed that informal process environment in their organization has mitigated knowledge atrophy problem from happened. In addition, the second stage data analysis result also shows that 80% of respondents claimed that their software development processes are not affected by the process erosion problem. They claimed that by using standard development tools, similar development process, having frequent guidance and mentoring activities, being active in knowledge sharing and proactive coaching could avoid the process erosion problems in software development process. The results also have shown an agreement with the previous research works that have been done by several others researchers including Rhodes et al (2008), Aaen et al., (2002) and Kozlowski and Ilgen (2007). Therefore from results gathered from both data analysis process, we have shown a support to H3.

> *"H3: The organization of software development knowledge and software development teams will assist to prevent Irish Software VSEs from process atrophy and knowledge entropy problems in software process."*

7.2.5 Research Question 5

> *"RQ5: Does the emerging of ISO/IEC 29110 standard could encourage Irish Software VSEs in improving, maintaining and evolving their software development process?"*

Several open questions which relate to software quality standard were asked to the interviewees in the interview and focus group session in order to answer the above research questions. The main purpose of all the questions is to understand VSEs acceptance and awareness level on software quality standard which could related with the new ISO standard, ISO/IEC 29110, which is primarily aimed at software development VSEs. The outcome from this exercise could to give an input and an indicator to this new ISO standard in order to ensure its objective could be achieved.

As we discussed in detail in chapter 5, the software quality standard in VSEs issues could be divided into 3 categories which are the level of acceptance, level of awareness and new standard criteria. The first category has prevailed that the acceptance level of any type or model of software quality standard in VSEs is very low and less priority. The reasons mainly related to the low level of customer or market requirement, low of resources and, lengthy and difficult procedures.

However, the analysis also showed that the level awareness of software quality standards and its advantage are high and there are some initiatives or plans to adopt in the future. The third category has indicate the criteria needed or proposed by the VSEs, which include the detail guideline and assistance, less overhead and resources and aligned with VSEs current process, that must be aware in order to encourage or to attract VSEs seriously involved in software quality standards. The details of the criteria are referred to part 5.9.

Meanwhile, in data analysis stage two we found that all respondents' software process did not accredit with any type of software quality certification and 60% of them do not plan to adopt any kind of standard in the near future. As in first stage data analysis result, stage two data analysis result also agreed that customer requirement, length and difficult procedure and insufficient resources are the main reasons of not adopting any kind of software quality standard. In addition, the level of awareness and additional criteria needed results are aligned with the first stage data analysis results.

Therefore from the results gathered from both of the analysis, we have indicated that the above research question (RQ5) and below research hypothesis (H4) is partially support, i.e. that the emergence of the new standard ISO/IEC 29110 could positively encourage VSEs in improving, maintaining and evolving their current software process. This situation can be understood from both data analyses results that indicates VSEs have a different level of agreement on software quality standard acceptance and awareness issues as explained in part 5.9 and 6.10. In overall both analyses have shown that the acceptance and priority on any software quality standard in VSEs is very low. This is due to lack of resources (e.g. time, people and financial), lack customer requirements, detail and difficult current software quality standard procedure and VSEs software development project priority are more towards quality product and delivery time rather than quality process. However, results from both analyses indicate that the level awareness on advantages of having or being accredited with software quality standards and respondents knowledge in these issues is moderate and acceptable. In addition, the respondents have indicated some additional criteria, as discussed in part 5.9 and 6.10, which need to be presented in any software quality standard in order to encourage them seriously participate in software quality standard. The results also have shown an agreement with the previous research works that have been done by several others researchers especially Laporte et al., (2008) and Pino et al., (2008).

Therefore we consider this research question and hypothesis are to be true and success if the related software quality standard body will considered the low acceptance reasons and additional criteria as listed and discussed above.

> *"H4: The adoption of the proposed standard ISO/IEC 29110 could positively encourage Irish Software VSEs in improving their current software development process."*

7.3 Overall analysis

From the above explanations, we have answered all the research questions and research hypothesis. The merging process from both data analysis stages also gave us details and extended our understanding on the research issues. In overall the finding have indicated that the SPI process in VSEs is being done informally, indirect, smalls scale and reactive. In addition the finding also shows that even though VSEs is small in terms of size and lack of resources, they also have a unique KM strategy that could help them in ensuring their software development process keep improve and evolve over time. Moreover the analysis has shown that the small team size in VSEs gave an additional advantage to them. The results have indicates that the small team size allowed staff to be more active and participate in sharing and exploring knowledge, enhance staff relationship that make them easier to work as a team and create an informal management environment and autonomous working style which enhance knowledge sharing and creation environment. Moreover the analysis have shown that all these attributes have helped VSEs in mitigated the process entropy and knowledge atrophy from happened. Moreover, the findings from both analyses have indicated that the acceptance level of any type of software quality standard is very low even though their awareness and knowledge level on software quality standard are moderate. Both analyses have show a similar pattern and results with previous research on lack of acceptance in software quality standard. Moreover both analysis have indicates additional criteria that need to be in any new software quality standard in order to encourage VSEs seriously participate in software quality standard in future.

7.4 Summary

This chapter presented and discussed the merged findings from stage one and two data analysis process. This merged process has helped the researchers to answer and confirm all the research questions and research hypothesis that have been identified for this present research. The answered and discussion that have been explained in this chapter, have extended researchers understanding on SPI issues in VSEs. The next section will summaries all the research processes and findings for the present research. We also will discuss the contribution, limitation and potential future work of the present research and that could be enhanced or pursued in future.

Chapter 8 CONCLUSIONS AND FUTURE WORK

8.1 Introduction

The main objective of this final chapter is to present the conclusions of this research. In this chapter, the researchers also will present the research contribution and examine the limitations of the present study. Finally some future research possibilities are presented which could build upon the present research study.

8.2 Conclusions

In this thesis, we have reported on the works and findings that have been done. The study has focused the present research on VSEs in Ireland. This is because the majority of the software companies in Ireland are fall under the category of VSEs which employs less than 25 people. In order to understand VSEs SPI environment in detail, especially in process of maintenance and evolution, we have investigated the role of KM and team issues in supporting the process. In addition the adoption of software quality standard also has been investigated in order to understand VSEs level of acceptance in improving their software development processes.

It is well known that the above issues have been studied in detail in large and multi-national company setting which have huge resources either in term of financial, employees or time allocations compare to VSEs which are in a different situation. In the same situation, the issues on adoption / accreditation of software quality standards in the literature have reported many success stories especially in relation to large and multi-national companies in improving the success of their software projects. Therefore based on these issues, researchers have taken an initiative to do research in order to understand the details of this situation. The research reported here investigates how VSEs which include the management and software development team perceptions on the above issues and its relationship. Therefore by having both inputs could help us in understanding the issues broadly and holistically.

In accomplishing the aims of the thesis, we have adopted two main approaches as follows:

8.2.1 Qualitative approach

In this present research, the qualitative approach which is the main approach of the research has dominated the data collection and analysis process. The researchers have used two main data qualitative approaches which include the personal and focus group interviews. In assisting researchers conducting several interview sessions, an interview and focus group guide have been prepared and developed. Several identified and agreed Irish VSEs software companies are involved in this process. Several related questions have been asked to the management people in the interview session and also the development staffs in the focus group sessions.

The overall main data analysis process of this research has adopted the qualitative contents analysis approaches which involved the several data coding approaches. In order to ensure the systematic data coding process, researchers have adopted the GT coding approaches which involved the open, axial and selective coding. These processes which is vague in the beginning and become more specific and focus at the end. In addition, GT method helps researchers to gather detailed results, produced the result pattern and enhanced researchers understanding on the whole research situation. Moreover the uses of the systematic coding processes have assisted researchers in producing a result pattern which helped researchers in answering research questions and hypothesis. Overall the processes have supported researchers producing and validating the research theoretical diagrams as identified in early stages of the present research.

8.2.2 Quantitative approach

In order to validate the qualitative finding and increase VSEs participation, we distributed survey questionnaires to a set of Irish VSEs software companies. From the completed questionnaires, we have analysed the data following statistical approaches and using the SPSS software package. We have run several statistical analyses which produced the result that the researchers required which have added and validated the qualitative results

8.2.3 Results

Based on the findings in stage one data analysis process (qualitative) indicates that there are 11 main categories that shape and relate with the software process and process improvement in VSEs. All these main categories have been presented, explained and depicted in a relationship diagram. The findings also helped the researcher produce categories, subcategories and explain the relationship each of category with the software process and process improvement in VSEs. The merging process of all the main categories, categories and subcategories helped the researchers to produce, validate and enhance the initial research theoretical models. In order to validate the findings of the first stage data analysis and enhanced its reliability, the quantitative analysis was completed. The results from both of the analysis processes have confirmed that in VSEs the software process and process improvement are being done on an informal, indirect, small scale and reactive basis. The analysis also indicates that similar development process and practice are being performed in VSEs in every of their software development process in software development and activities. In supporting software development process, the finding have shown that the communication, learning and sharing process have been done actively in VSEs. However due to small team size; most of the processes are done informally and indirectly either through direct or online communication.

It was collectively agreed by the respondents that the documentation process in VSEs is done very informally, individually and specifically. In terms of KM issues, the result showed that all respondents claimed that they have a clear KM strategy in the organization. However the analysis showed that this process are done informally and is not organised. In addition the result show that even though the KM was done informally either in communication, management, working style and team structure in VSEs, 90% of the respondents believed that this environment have lead them to mitigate the knowledge and process loss problem in their organization. Moreover the results also indicated that in overall the size of the company given an impact to all the process that have discussed above.

In terms of process evolution and maintenance of current software development process, the findings indicated that respondents agreed that their software development process frequently change and evolved over time. They also agreed that they regularly assess and update their development processes. However the findings showed that the changed and evolved processes are informal, indirect and very reactive which depends or is linked to customer requirements, developers' initiatives and technology changes. In terms of development process methodology 100% of respondent stated that they had adopted an agile development approach philosophy in the company's projects. This could be identified from the analysis which showed that the development processes are very informal, less documentation, customer oriented and active in communication. Overall although the results showed the high informal and indirect culture in VSE in most of their development activities, the results also indicate that VSEs commitment towards SPI is very high and positive.

The team development issues have indicated that VSEs which employ a small number of employees have a very flat and informal team structure and process. The first stage and second stage analysis showed that the due to the small team size and open working environment which are macro and autonomous, the team dynamics in VSEs are very high. Even though some of the employees are working remotely (abroad) the results show that the staff relationship, socialize, sharing and interpersonal skill level are high. This could be clearer in the second stage analysis process results which obtained high score in data analysis.

In terms of software quality standards, the results also indicate that the level of acceptance and adoption of software quality standard within VSEs is very low. From the questionnaires, researchers found that 60% of the respondents do not have any plans to adopt any kind of quality standard in the companies. Among the main reasons are not required by customer, length and difficult process and insufficient guidance. However from analysis shown that the results have indicates the level of awareness of software quality standard and its advantages are moderate. In addition the results also indicated and listed of requirement that need to emphasize or aware by software quality body like ISO or other researchers in developing software quality standards for VSEs.

8.3 Research Contributions

By systematic and comprehensive analysis of the present research model, as depicted in figure 3.6 and supported with the detail research processes as explained in chapter 4 and results of qualitative and quantitative data that produced from this research as shown in the appendices E, F, G, H, I and J, we have provided several research contribution as follows:-

The first contribution is an improved understanding of software process and process improvement in general and how these issues affect VSEs in particular. The research also presents and explains the factors that influence the way the process has been established and evolved in VSEs. In addition the reasons behind why the software development companies' and VSEs in particular ignored and are not adopting the best practice software process and process improvement has also been discovered and explained.

The second major contribution of this research is an expanded understanding of SPI research area by merging the issues of KM and team development from both a general and VSE specific perspective. Our results indicate that these two factors gave indirect influences to the process of improving, maintaining and evolving current software process and process improvement activities in software development companies. This has been presented in theoretical models that have been depicted in chapter 5.

The third contribution of this research is providing an additional knowledge to the SPI research area focused more on VSEs, which have been least explored by current literature. The research has found the variables that influence the software process and process improvement issues that could be explored individually in further detail in future.

The fourth contribution is the type of strategies used to carry out research methodology work, which is a combination of an interviews (interview and focus groups) and survey questionnaire research approach. The survey questionnaire approach is a familiar approach that often used in the software engineering field. However the interviews data analysis research technique which adopted qualitative contents data analysis and adopted GT coding approach, is rarely been used in the analysis of the software process improvement and in software engineering research in general. Therefore we believe that we are adding to the body of knowledge associated with suitability of the GT research method to software engineering area.

The final major contribution explores why the commercial process models such as ISO 9000 and CMMI are being ignored by VSEs although several researchers or professional body have produced various guidelines and standards. In addition the present study has listed some of the criteria that need future researchers need to be aware of in order to encourage VSEs in seriously adopting or following a process model in their software process and process improvement.

8.4 Limitations of the Study

We have identified some limitation and constraints have been occurred in order to produce a solid and generalise findings and results. Below the limitations and constraints are discussed:

The first limitation is regarding the qualitative data issue. Unlike quantitative studies which produce a specific and measurement data, the qualitative data relies on respondents' opinions and perceptions that cannot be measured and quantified. This situation may sometimes be perception based and at odds reality. In many situations they may be no support evidence to verify the opinion expressed. In addition, it is possible, that the participants may report what they believe what researchers would like to hear especially in small companies like VSEs, or they are not willing to fully disclose that they are not following or adopting any proper practice that something they do not want to explain or share with others. Moreover companies do not wish to publish negative results due to worries that the results will give a wrong image and bad impact to the company reputation and status. However it is not the job of researchers to interpret or evaluate the interviewees. Cornford and Smithson (2006) state that qualitative data help researchers' extract depth understanding on individual and / or group experiences and perceptions rather than abstract information. This process awards research results more value and coverage which produced richer and more informative research findings (Seaman, 1999). As such researchers must accept the truthfulness of what the respondents say during the study interview (Hansen and Kautz, 2005).

The second limitations are on the data collection issue. In conducted the data collection process, researchers had found some difficulties to get a full commitment and good response from the identified Irish Software VSEs. Limited number of staff, busy with current project, economy situation, project deadline, low level of interest and inappropriateness are among the reasons given by those companies that gave an impact to the progress and smoothness of present research study. This situation had been reflected to a low willingness in participate in the interview sessions and received a low return rate of questionnaire response. In particular issue especially in focus group interview approach, there were an occasion that researchers have need to convert the focus group session into an interview session due to the respondent commitment to the company project and dateline. This has been showed in table 4.2 in chapter 4. These situations and issues has been agreed and discussed in length by Fowler (2002). Moreover the low return rate of the questionnaire problems were also highlighted and experienced by some other researchers such as Syed Abdullah (2005), Newton et al., (2008) and Reed and Kelly (2002).

The third limitation is regarding the participation issue. Continuing from the above point, another potential limitation of research is that the fact that in stage one and two data collection process, which involved the interviews, focus group interviews and survey questionnaires, were dominated by the management group rather than a balance between the management and the development team. Therefore as results the finding could be more prone to the management views and less on the software development team views.

The final limitation is the issue of result generalisability which continue from the above issues. The issue of generalisability is an issue about the type and number of respondents participates in the research. Since the majority of the research data are more prone to the management viewed compared to development team, the issue of results generalisability is arise. This was added by the small research sample size which creates some limitation in the research results. However the appropriate type of companies and respondents were identified for this research have produced a valid indicator and the results could present the VSE environment as a whole. These could be understood from the consistent research results which produced in the analysis process.

8.5 Future works

From this research several directions for future research have emerged as follows:

The first area is on the holistic VSEs perspective. This research is concerned with how software process, KM and team issues in Irish Software VSEs. Since most of the Irish indigenous IT companies are felt under the definitions of VSEs as discussed in part 2.4, a wider perspectives should be consider in order to produce a holistic perspectives in this issues. Therefore number of research participants must be adequate enough which help to produce more reliable and generalise results. The results are expected could give VSEs an information and awareness in managing their software development process. Beside that, the balance involvement of two main research participants group as proposed in this research must take into a serious consideration. This is because the balance involvement from these two groups could help future research to gather both group perspective and opinions that could strengthen the results of the research. Beside that researches believe that the serious involvement especially the team development will produce more details, specific and valuable results.

The second potential area is a wider the current research spectrum. Continued from the first point above, in order to test current research findings and also to produce and provide more valid findings and results, a similar study could be deployed in other part of Ireland specifically and Europe countries generally. This activity is predicted could help to generalise the research results. This is because similar environment of software industries environment, which the majority of the software companies falls in VSEs category, as explained in chapter 2. Beside that as VSEs definition as explained and discussed in chapter 2, which does not cover and involve only specific to software development companies, could help to produce more input to the present research results. The involvement of non-IT company which having a small IT department could will help researchers to compare and produce a pattern of research results which will add to the current research result. By having all the information will generalise, extend and improve the research results in similar or different environment in future.

One of the issues that have a potential for further research is on the knowledge loss issues in SPI. The issue is related to the documentation strategy which the VSEs practices in maintain the software development knowledge in companies. The researchers found that one of the documentation strategies is via the electronic or online documentation which are more informal and not organize. Since this is the main documentation process method that VSEs are preferred to do, the future research could explore how this type of documentation process should be manage and organize. Researchers believed that the proper process and strategy could ease and encourage VSEs to be more serious in documenting all related process. This process also is believed could mitigate the knowledge loss problem from happened in VSEs. In addition the proper organization of software development knowledge also will help VSEs in lessen the process loss problem as explained in chapter 3.

As explained in part 8.2, that this research has listed some of the criteria that need to be aware or include in current or future process model or standard in order to encourage VSEs adopting process model or standard in their software process and process improvement. Therefore future research in the same issues and area could enhance or details the process model criteria to suit with the VSEs needs. In addition near future researchers could use the criteria that have been listed to evaluate the existing process models or standards.

References

Aaen, I., Arent, J., Mathiassen, L and Ngwenyama, O., 2001, 'A Conceptual MAP of Software Process Improvement', *Scandinavia Journal of Information System*, Vol. 13, Issue. 13, pp. 81-101.

Aaen, I., Börjesson, A. and Mathiassen, L., 2007, 'SPI agility: How to navigate improvement projects'. *Software Process: Improvement and Practice*, Vol. 12, Issue. 3, pp. 267 - 281.

Acs, Z.J, O'Gorman, C., Szerb, L. and Terjesen, S., 2007, 'Could the Irish Miracle be Repeated in Hungary', *Journal Small Business Economics*, Vol. 28, No. 2-3, pp.123-142.

Agile Alliance, (2001), The Agile Manifesto, *http://www.agilealliance.org/intro* [Accessed 12th April 2007].

Ahern, D.M., Clouse, A and Turner, R., 2004, *CMMI Distilled: A practical Introduction to Integrated Process Improvement*, 2nd Ed. Addison Wesley.

Agresti, W., 2000, 'Knowledge Management', *Advances in Computers*, Vol. 53, Issue. 5 , pp. 171-283

Alavi, M. and Leidner, D. E., 1999, 'Knowledge Management Systems: Issues, Challenges, and Benefits', *Communications of the AIS*, Vol.1, Issue. 2, pp. 4-37

Alavi, M. and Leidner, D. E., 2001, 'Knowledge Management and Knowledge Management Systems: Conceptual foundation and Research Issue', *MIS Quarterly*, Vol. 25, No.1, pp. 107-136.

Anquetil, N., Oliveira, K.M, Sousa, K.D. and Dias, MGB, 2007. 'Software Maintenance Seen as a Knowledge Management Issue', *Information and Software Technology*, Vol. 49, Issue 5, pp. 515 -529.

Albert, S.R and Fetzer, R.C., 2005, 'Smart Communities Network: self directed team effectiveness in action', *Journal Team Performance Management*, Vol.11, No. 5-6, pp. 144-156.

Allison, I. and Merali, Y., 2007, 'Software Process Improvement as Emergent Change: A structural analysis', *Information and Software Technology*, Vol. 49, Issue. 6, pp. 668-681

Arent, J. and Norbjerg, J., 2000, 'Software process improvement as organizational knowledge creation: a multiple case analysis'. *System Sciences, Proceedings of the 33rd Annual Hawaii International Conference***,** Maui, USA, pp. 105.

Argyris, C., 1977, 'Double loop learning in Organization', *Harvard Business Review*, Vol. 55, Issue. 5, pp. 115-125.

Armour, F.J. and Gupta, M., 1999, 'Mentoring for success', *IT Professional*, Vol. 1, Issue 3, pp. 64-66.

Arora, A., Gambardella, A. and Torissi, S., 2001, 'In the Footsteps of Silicon Valley/ Indian and Irish Software', *International Division of Labour, Stanford Institute For economy Policy Research (SIEPR) Discussion Paper*, No. 00-41, Stanford University, California USA.

Arora A. and Gambardella A. 2004, 'The Globalization of the Software Industry: Perspectives and Opportunities for Developed and Developing Countries. National Bureau of Economic Research', *Working Paper 10538,* June 2004.

Aurum, A., Daneshgar, F. and Ward, J., 2008, 'Investigating Knowledge Management practices in software development organisations – An Australian experience', *Information and Software Technology*, Vol. 50, Issue. 6, pp. 511-533.

Aurum, A., Jeffery, R., Wohlin, C. and Handzic, M., 2003, 'Managing Software Engineering Knowledge', *Springer, Germany*.

Awazu, Y., 2004, 'Knowledge management in distributed environments: roles of informal network players', *System Sciences, Proceedings of the 37th Annual Hawaii International Conference***,** pp. 1-6.

Ayman, R. 2000, 'Impact of team diversity on collaboration dynamics', *in Collaborating across Professional Boundaries*, available at http://www.stuart.iit.edu/ipro/papers/pdf/ayman.pdf. [Accessed on: 25 April 2007].

Babbington-Smith, B. 1979, *Training in Small Group* , Pergammon Press, Oxford.

Baddoo, N. and Hall, T. 2003, 'De-Motivator for Software Process Improvement: An Analysis of Practitioners' Views. *The Journal of System and Software*, Vol. 66, No.1, pp. 23-33.

Baker, R. 1996, 'The Corporate Politics of CMM Ratings', *in Communication of the ACM*, Vol. 39, No. 9, pp. 105-106.

Barnum, C.M., 2000, 'Building a Team for User Centered Design', *Proceeding of IEEE Professional Communication Society, International Professional Communication Conference and Proceedings of the 18th Annual ACM Internal Conference on Computer Documentation*: Technology and Teamwork.

Barry F. and Curran D., 2004, 'Enlargement and the European Geography of the Information Technology Sector', *The World Economy*. Vol. 27, No. 6, pp. 901–922.

Basham, R.E.; Jordan, C.; Hoefer, R.A., 2009, 'Reliability and Validity in Quantitative Measurement'. The *Handbook of Social Work Research Methods*, pp. 51-64, Sage *Publications,* Inc

Basili, V. R. and Weiss, D.M., 1984, 'A methodology for collecting Valid Software Engineering Data', *IEEE Transactions on Software Engineering*, Vol.10, Issue.6, pp.728-738.

Basili, V. R. and Turner, R., 1975. 'Iterative Enhancement: A Practical Technique for Software Development'. *IEEE Transactions on Software Engineering*. Vol. SE-1, No. 4, pp. 390-396.

Basili, V. R., Caldiera, G. and Rombach, D., 1994. 'The Goal Question Metric Approach', *Encyclopedia of Software Engineering*, Wiley.

Batisha, J. and de Figueiredo, A.D., 2000,' SPI in a Very Small Team: A Case with CMM', *Software Process Improvement and Practice*, Vol. 5, No. 4, pp. 243-255.

Benbasat, I., Goldstein, D.K. and Mead, M.,1987. 'The Case Research Strategy in Studies of Information Systems', *MIS Quarterly*, Vol. 11, Issue. 3, No. 369-386.

Beydeda, S. and Gruhn, V., 2004, 'Dynamic Evolution of Software Process to Evolve Software Systems during their Development', *Software Process Improvement Practice,* Vol. 9, Issue 4 , pp. 229 – 238.

Beaver, J.M. and Schiavone G. A., 2006. 'The effects of development team skill on software product quality'. *ACM SIGSOFT Software Engineering Notes*, Vol. 31, Issue 3, pp. 1 – 5.

Beck, K, 2000, *Extreme Programming Explained: Embrace Change*, Addison Wesley

Beck K, Beedle M, Van Bennekum A, Cockburn A, Cunningham W, Fowler M, Grenning Jhighsmith J, Hunt A, Jeffries R, Kern J, Marick B, Martin R, Mellor S, Schwaber K, Sutherland J and Thomas D (2001) *Manifesto for Agile Software Development*, *www.AgileManifesto.org.*

Belbin, R.M, 2000, *Beyond The Team*, Butterworth Heinemann, Oxford.

Bellinger, G., Castro, D. and Mills, A., 1997, 'Data, information, knowledge and wisdom' *available at http://www.systems-thinking.org/dikw/dikw.htm* [Accessed on: 27 April 2007].

Bergin, J., Caristi, J., Dubinsky, Y., Hazzan, O., and Williams, L. 2004, 'Teaching Software Development Methods: The Case of Extreme Programming'. *The Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education*, Norfolk, Virginia, USA, pp. 448-449.

Bergman, J., Jantunen, A. and Saksa J.M,. 2004, 'Managing knowledge creation and sharing – scenarios and dynamic capabilities in inter-industrial knowledge networks', *Journal of Knowledge Management*, Vol. 8, No. 6, pp. 63-76.

Beydeda, S. and Gruhn, V., 2004, ‘Dynamic Evolution of Software Process to Evolve Software Systems during their Development’, *Software Process Improvement Practice* Vol. 9, Issue. 4 , pp. 229 – 238.

Bharadwaj, A. and Tiwana, A. 2005, ‘Managing Knowledge in Software Development, System Sciences, HICSS '05’. *Proceedings of the 38th Annual Hawaii International Conference,* pp. 255- 264

Bjornson, F.O and Dingsoyr, T., 2005, ‘A Study of a Mentoring Program for Knowledge Transfer in a Small Software Consultancy Company’, *Lecture Notes in Computer Science*, Vol. 3547, Springer Berlin / Heidelberg.

Blaxter, L., Hughes, C. and Tight, M., 2003 ‘How to Research’ (2nd Ed),Open University Press Maiden Head Philadelphia.

Bratthall, L. and Jorgensen, M., 2002, ‘Can you Trust a single Data Source Exploratory Software Engineering Case Study?’, *Journal Empirical Software Engineering*, Vol. 7, No. 1, pp. 9-26.

Bloodgood, J. M. and Salisbury Wm. D. 2001. ‘Understanding the influence of organizational change strategies on information technology and knowledge management strategies’, *Journal Decision Support Systems,* Vol. 31, Issue 1, pp. 55-69

Boehm, B., 2002, ‘Get ready for agile methods, with care’, *Computer*, Vol. 35, Issue. 1, pp. 64 – 69.

Bollinger, A. S. and Smith R. D., 2001., ‘Managing organizational knowledge as a strategic asset’, *Journal of Knowledge Management*, Vol. 5, No. 1, pp. 8-18.

Borges L.M.S and Falbo, R.A , 2002, ‘Managing Software Process Knowledge’, *Proceedings of the International Conference on Computer Science, Software Engineering, Information Technology, e-Business, and Applications (CSITeA'2002*), pp. 227 – 232,.

Borjesson, A. and Mathiassen, L., 2005, ‘Improving Software Organization: agility Challenges and Implication’. *Journal Information Technology and People*, Vol. 18, No. 4, pp. 359-382.

Borjesson, A. and Mathiassen, L., 2004., ‘Successful Process Implementation’, *IEEE Software*, Vol. 21, Issue. 4, pp.36-44.

Brodbeck, F. C. 2001. ‘Communication and performance in software development projects’. *European Journal of Work & Organizational Psychology*, Vol. 10, No. 1, pp. 73-94.

Bryant, A. and Charmaz, K., 2008, Grounded Theory: The SAGE Handbook, *SAGE Publications*, Thousand Oak

Bryman, A., 1992, Quantity and Quality in Social Research, Routledge, London.

Bryman, A., 2001, Social Research Methods, Oxford: Oxford University Press

Bryman A., 2004. Social Research Methods, 2nd edn. Oxford: Oxford University Press

Bucci, G., Campanai, M. and Cignoni, G.A. 2000, 'RAPID Assessment to Solicit Process Improvement in SMEs', *Proc. 7th European Software Process Improvement Conf. (EuroSPI), 2000*; www.iscn.at/select_newspaper/assessments/italy.html. [Accessed on : 7 November 2006]

Burns, R. B., 2000. 'Introduction to Research Methods', 4th edition, Sage Publication.

Butler KL. 1997. 'Process lessons learned while reaching level 4'. *Crosstalk, The Journal of Defense Software Engineering*.

Canfora, G., García, F., Piattini, M., Ruiz, F. and Visaggio, C.A. 2005, 'A family of experiments to validate metrics for software process models'. *Journal of Systems and Software*, Vol. 77, Issue. 2, pp.113-129.

Carron, A. V. 1982. 'Cohesiveness in sport groups: Interpretations and considerations'. *Journal of Sport Psychology*, Vol. 4, No.6 , pp. 123-138.

Casey, V. and Richardson, I., 2004. 'A Practical Application of IDEAL Model', *Software Process Improvement and Practice,* Vol. 9, Issues. 3, pp. 123-132.

Cattanoe, F., A. Fuggetta and L. Lavazza. 1995. 'An experience in process assessment'. *Proceedings of the 17th International Conference on Software Engineering*, *IEEE Computer Society*, pp. 115-121.

Chan, I. and Chao, C-K., 2008. ' Knowledge management in small and medium-sized enterprises', *Communications of the ACM*, Vol. 51, Issue. 4, pp. 83-88.

Charmaz, K, 2006. Constructing Grounded Theory: A Practical Guide Through Qualitative Analysis, *Sage Publication*, Thousand Oak.

Chau, T., Maurer, F., and Melnik, G., 2003. 'Knowledge Sharing: Agile Method vs. Tayloristic methods'. *In Proceeding of the IEEE International Workshop on Enabling Technologies: Infrastructures for Collaborative Enterprises,*. IEEE Computer Society.

Chen, WS. and Hirschheim, R. 2004, 'A paradigmatic and methodological examination of information systems research from 1991-2001', *Journal of Information System*, Vol. 14, Issue. 3, pp. 197-235.

Chrissis, M.B., Konrad, M and Shrum, S. 2003, CMMI: Guidelines for Process Integration and Product Improvement, *Addison Wesley*, Boston, MA.

Chua, W. F., 1986, ‘Radical developments in accounting thought’, *The Accounting Review*, Vol. 61, No. 4, pp. 601–632.

Clark, V.L.P; Casas, C.A.H; Green,D.N. and Garrett, A.L., 2008., ‘Mix Method Approaches in Family Science Research’, *Journal of Family Issues*, Vol. 29, Issue. 11, pp. 1543- 1560

Clark, C.E., Cavanaugh, N.C., Brown,C.V. and Sambamurthy, V.. 1997. ‘Building change-readiness capabilities in the IS organization. Insights from the Bell Atlantic experience’. *MIS Quarterly*, Vol. 21, Issue. 4, pp. 425-455.

CMMI Product Team, 2002, CMMI for system engineering, software engineering, integrated product and process development, and supplier sourcing (staged representation)’, *Report no. CMU/SEI-2002-TR-012; ESC-TR-2002-012, CMMI, Pittsburg, PA,*

CMMI Product Team, 2006. CMMI for Development Version 1.2, Carnegie Mellon, Software Engineering Institute, Pittsburgh, PA at http://cc.ee.ntu.edu.tw/~farn/courses/SE/CMMI_DEV_V12.pdf [Accessed on : 16 February 2010]

Cockburn, A., 2002, Agile Software Development, *Addison Wesley*.

Cockburn, A. 2000, Writing Effective Use Case the Crystal Collection for Software Professional, *Addison Wesley*.

Cohen, S. G. and Bailey, D. E., 1997, ‘What Makes Teams Work: Group effective Research from The Shop Floor to the Executive Suite’, *Journal of Management*, Vol. 23 No. 3, pp 234-256.

Coleman, G. and O’Connor, R.V., 2008, ‘Investigating Software Process in Practice: A Grounded Theory Perspective’. *Journal of System and Software*, Vol. 81, Issues. 5, pp. 772-784.

Coleman, G. and O'Connor, R. V., 2008. ‘The influence of managerial experience and style on software development process’. *International Journal of Technology, Policy and Management*. Vol. 8, Issues. 1, pp. 91-109.

Coleman, G., 2006, ‘Investigating Software Process in Practice: A Grounded Theory Perspective’, *PhD Thesis DCU*, DCU.

Collins, K.M.T, Onwuegbuzie, A.J and Sutton, I.L., 2006, ‘A Model Incorporating the Rationale and purpose for conducting Mix methods research in Special Education and beyond. Learning Disabilities’: *A Contemporary Journal*, Vol. 4, Issue. 1, pp. 67-100.

Conradi, R., Dybå, T., Sjøberg, D.I.K. and Ulsund, T., 2003, ‘Lessons Learned and Recommendations from Two Large Norwegian SPI Programmes’, *Lecture Notes in Computer Science*, Springer, Vol. 2786/2003

Cornford, T and Smithson, S., 2006. ‘Project Research in Information Systems. A Student’s Guide’,(2nd edition). London *PalgraveMacmillan*. pp. 8-28

Cook, J. E. and Wolf ,A. L., 1998, ‘Discovering models of software processes from event-based data’, *ACM Transactions on Software Engineering and Methodology (TOSEM),* Vol.7, Issue. 3, pp: 215 – 249.

Creswell, J. W, 2003,’Research Design: Quantitative, Qualitative and Mix Methods Approaches’ (2nd Ed), *Sage Publications*: Thousand Oaks, Ca.

Creswell, J. W, 1994, ‘Research Design: Qualitative and Quantitative Approaches’. *Sage Publication*: Thousand Oaks, Ca.

Creswell, J. W. and Clark, V.L.P, 2007, ‘Designing and conduction Mixed Methods Research’, *Sage Publications*: Thousand Oaks, Ca.

Crone, M. 2002, A Profile of Software Irish Industry, *Northern Ireland Economic Research Center (NIERC)*, Belfast NI.

Curtis, B., 1989, ‘Three Problems Overcome with Behavioural Models of Software Development Process’, *International Conference on Software Engineering Pittsburg PA*, Pennsylvania, United States, pp. 398-399.

De Marco, T. and Lister, T., 1999, Peopleware: Productive Project and Team, 2nd Edition, *Dorset House* New York.

Debou, C and Kuntzmann-Combelles, A. 2000, ‘Linking Software Process Improvement to Business Strategies: Experience from Industries’ , in *Software Process Improvement and Practice*, Vol. 5, No.1, pp. 55-64.

Demirors, O. and Demirors, E., 2006, ‘Software Process Improvement in Small Organization: Difficulties and Suggestions Software Process Technoogy’. *Proceedings of the 6th European Workshop on Software Technology EWSPT.* Springer Heidelberg

Demirors, E. Sarmasik, G. Demirors, O. , 1997, ‘The Role of Teamwork in Software Development: Microsoft Case Study’, EUROMICRO 97. 'New Frontiers of Information Technology’, *Proceedings of the 23rd EUROMICRO Conference***.**

Denzin, N.K, 1970. ‘The Research Act in Sociology: A theoretical Introduction to Sociological Methods’. London: Butterworth.

Denzin, N. K. 1978. ‘The Research Act: a Theoretical Introduction to Mix Method Research’, New York: Praenger.

Denzin, N.K. 1989. ‘The research act (3rd Ed)’, *Eaglewood Cliffs*, NJ: Prentice Hall
Jick, T.D. 1972. ‘Mixing the Qualitative and Quantitative Methods: Triangulation in action’. *Administrative Science Quarterly*, Vol. 24, pp. 602-611

Denzin, N.K and Linconln, Y.S, 2000. 'The Discipline and Practice of Qualitative Research', *Handbook of Qualitative Research*, Sage Publication London.

Desanctis, G. and Jackson, B.M, 1994, 'Coordination of Information Technology Management: Team Based Structures and Computer Based Communication Systems', *Journal of Information Management Systems*, Vol. 10, No. 4, pp. 85-110.

Desouza, K.C., 2003., 'Facilitating Tacit Knowledge Exchange', *Communication of the ACM* ,Vol .46, Issue. 6, No. 6, pp 85-88.

Desouza, K.C., 2004, 'Knowledge Management: A New Commission for Industrial Engineers', *Journal Industrial Management* .Vol. 46, Issue. 1, p. 26-30.

Davenport, T. and Prusak, L. 1998. Working Knowledge, *Harvard Business Scholl Pres,* Boston, MA

Dick, R. 2001 "Grounded Theory: A Thumbnail Sketch at http://www.scu.edu.au/schools/gcm/ar/arp/grounded.html" [Accessed on : 7 November 2008]

Dictionary 2001, Merriam Webster 2001.

Dingsoyr, T. and Conradi, R., 2002. 'A Survey of Case Studies of the Use of Knowledge Management in Software Engineering', *International Journal of Software Engineering and Knowledge Engineering,*Vol. 12, No. 4, pp. 391-414.

Dingsøyr, T., Djarraya, H.K. and Royrvik, E. 2005. 'Practical knowledge management tool use in a software consulting company', *Communications of the ACM* , Vol. 48, No. 12, pp. 97–103.

Dion, R., 1993, 'Process Improvement and the Corporate Balance Sheet', *IEEE Software*, Vol.10, Issue. 4, pp 28-35.

Dove R. 2001. 'Response Ability - The Language, Structure, and Culture of the Agile Enterprise'. *Wiley*: New York.

Dyba, T., 2003, 'A dynamic Model of Software Engineering Knowledge Creation', book chapter in Aybuke Aurum, Ross Jeffrey, Claes Wohlin and Meliha Handzic (Eds.) : Managing Software Engineering Knowledge , *Springer Verlag*, ISBN 3-540-00370-3, pp. 95-117.

Dyba, T., 2005. 'An empirical investigation of the key factors for success in software process improvement'. *Software Engineering, IEEE Transactions* , Vol. 31, Issue. 5, pp. 410- 424.

Easterby-Smith, M., Thorpe, R., Lowe, A., 1991, 'Management Research: an Introduction', *Sage Publications*, London.

El-Emam, K., Goldenson, D., McCurley, J. and Herbsleb, J., 2001, 'Modelling the Likelihood of Software Process Improvement: An Exploratory Study'. *Journal Empirical Software Engineering*, Vol. 6, No. 3, pp. 207-229.

Elo, S and Kyngäs, H., 2008. ‘The qualitative content analysis process’, *Journal of Advanced Nursing*, Vol. 62, Issue 1, pp. 107 – 115.

Enterprise Ireland, 2005a, *Background to Software Irish Software Industry*, available at http://www.nsd.ie/htm/ssii/back.htm. [Accessed on: 16 January 2007].

Enterprise Ireland, 2005b, *Software Industry Statistics 1991- 2004*, available at http://www.nsd.ie/htm/ssii/stat.htm. [Accessed on: 16 January 2007].

Enterprise Ireland, 2009, *Software Strategy - A strategy for development of the indigenous software industry 2009-2013*, available at http://www.enterprise-ireland.com/NR/rdonlyres/117B1162-7B2D-41DC-81DB-A4E6B8E48F82/0/ SoftwarePublishedstrategy.pdf [Accessed on: 1 December 2009]

European Commission, 2005, The *New SME Definition: User Guide and Model Declaration* available at: http//europa.eu.int/comm./enterprise/enterprise policy/sme_definition/sme_user_guide.pdf. [Accessed on: 26 February 2007].

Faraj, S. and Sproull, L., 2000, ‘Coordinating Expertise in Development Teams’, *Management Science*, Vol. 46, No. 12, pp. 1554-1568.

Faraj, S. and Sambamurthy, V., 2006. ‘Leadership of Information Systems Development Project’, *IEEE Transactions on Engineering Management*, Vol. 53, No. 2, pp. 238-249.

Feagin, J., Orum, A., and Sjoberg, G. (Eds.). 1991. ‘A Case for Case Study’. Chapel Hill, NC: University of North Carolina Press.

Firestone, W.A, 1987, ‘Meaning in Method: The rhetoric of Quantitative and Qualitative Research’. *Education Researchers*, Vol. 16, Issue. 7, pp. 16-21.

Fiore, S. M.; Salas, E.; Cuevas, H.M.; Bowers, C.A., 2003. ‘Distributed coordination space: toward a theory of distributed team process and performance’, *Theoretical Issues in Ergonomics Science*, Vol. 4, No.3-4, pp. 340-364.

Fitzgerald, B., Russo, N. L. and O'Kane, Tom. 2003, ‘Software development method tailoring at Motorola’, *Communications of the ACM*, Vol. 46, No. 4, pp. 64-70.

Flick, U., 2005, ‘Introduction to the Qualitative Research’, *Thousand Oaks*, CA: Sage Publications Inc.

Fowler, M. and Highsmith, J. 2001. ‘The agile manifesto’. *Software Development*. Vol. 9, No. 8, pp. 28-35.

Bjørnson, F. O. and Dingsøyr, T., 2008. ‘Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used’, *Information and Software Technology,* Vol. 50, Issue. 11, pp. 1055-1068.

Flood, P., Heffernan, M., Farell, J., MacCurtain, S., O'Hara, T., O'Regan, P. and Carroll, C., Dramgoole, T. and Mangan, J., 2002, 'Managing Knowledge Based Organizations: Top management Teams and Innovation in the Indigenous Software Industry', *Blackhall Publishing.*

Fowler, F.J., 2002. 'Survey Research Method (3rd Edition)', *Thousand Oaks*, CA: Sage Publications Inc.

Fuggetta. A., 2000. 'Software Process: A Roadmap', *International Conference on Software Engineering*. Proceedings of the Conference on The Future of Software Engineering Limerick, Ireland, pp: 25 - 34.

Furumo, K. and Pearson, J.M., 2006. 'An Empirical Investigation of how Trust, Cohesion and Performance Vary in Virtual and Face to Face Teams'. *System Sciences, Proceedings of the 39th Annual Hawaii International Conference,* Vol. 1, pp. 26c- 26c

Galliers, R., 1991, 'Choosing Information Systems Research Approaches,' *Information systems Research: Issues, Methods and Practical Guidelines*, R. Galliers, Ed. Cambridge: NCC Blackwell

Garavelli, C., Gorgoglione, M. and Scozzi, B., 2004. 'Knowledge Management strategy and Organization: A Perspective of Analysis', *Knowledge and Process Management*, Vol. 11, No. 4, pp. 273-282.

Gerald I. S and Roger D. E., 1978. 'An Assessment of the Scientific Merits of Action Research', *Administrative Science Quarterly*, Vol. 23, No. 4., pp. 582-603.

Giartanna, M., Pagano, A and Torrisi, S., 2003. 'The Role of Multinational Firms in the Evolution of Software Industry in India, Ireland and Israel', *DRUID Summer Conference Elsinore*.

Grant, R. M. 1996. 'Prospering in dynamically-competitive environments: organisational capability as knowledge integration'. *Organisation Science*, Vol. 7, No. 4, pp. 375–387.

Green R, Cunningham J, Duggan I, Giblin M, Moroney M and Leo S., 2001, 'The Boundaryless Cluster: Information and Communications Technology in Ireland'. *Innovative Clusters: Drivers of National Innovation Systems*.

Gremba, J., and Myers, C. 1997. The IDEALSM model: A practical guide for improvement. *Pittsburgh: Software Engineering Institute, Carnegie Mellon University*. See URL http://www.sei.cmu.edu/ideal/ideal.bridge.html .

Gorla, N and Lam, Y.W., 2004, 'Who Should Work With Whom? Building Effective Software Project Teams', *Communications of the ACM*, Vol. 47, Issue. 6, pp. 79-82.

Golanfshani, N., 2003. 'Understanding Realibility and Validity in Qualitative Research', *The Qualitative Report*, Vol. 8, No. 4, pp. 597-607

Goulding, C. 2002, 'Grounded theory: a practical guide for management, business and market researchers', *Sage Publication*, London.

Guinan, P.J., Cooprider, J.G., Faraj, S., 1998. 'Enabling Software Development Team Performance during Requirement Definition: A Behavioural vs. Technical Approach' *Information Systems Research,* Vol. 9, Issue. 2, pp. 101 – 125.

Guzzo, R.A., and Shea, G. P. 1992. 'Group performance and intergroup relations in organizations', *Handbook of industrial and organizational psychology*, pp. 269-313. Chicago: Rand McNally.

Gist, M. E., Locke, E. A., and Taylor, M. S., 1987. 'Organizational behavior: Group, structure, process, and effectiveness'. *Journal of Management*, Vol.13, No. 2, pp. 237-257.

Gupta A.K. and Govindarajan, V., 2000. 'Knowledge flows within multinational corporations'. *Strategic Management Journal*, Vol. 21, Issue. 4 , pp. 473 – 496.

Gruber, T.R, 1995, 'Towards principles for the design of Ontologies used for knowledge sharing'. *International Journal of Human Computer Studies.* Vol. 43, Issue 5-6, pp. 907-928

Goldenson, D. and Gibson, D., 2003, Demonstrating the Impact and Benefit of CMMI: An Update and Preliminarily Result. *Technical Report CMU/SEI-2003-SR-009*, Software Engineering Institute, Pittsburgh, PA.

Haase, V., Messnarz, R., Koch, G., Kugler, H.J., and Decrinis, P., 'Bootstrap: Fine Tuning Process Assessment', *IEEE Software*, Vol. 11, No. 4, pp. 25-35.

Habra, N., Alexander, S., Desharnais, J-M., Laporte, C.Y. and Renault, A., 2008, 'Initiating software process improvement in very small enterprises. Experience with a light assessment tool', *Journal Information and Software Technology*, Vol. 50, No. 7-8, pp. 763-771.

Hackman JR, and Morris CG. 1975. 'Group tasks, group interaction process, and group performance effectiveness: A review and proposed integration'. *In Berkowitz L (), Advances in experimental social psychology*, New York : Academic Press. Vol. 8, pp. 45–99.

Hackman, J. R., 1987. 'The design of work teams'. *Handbook of Organizational Behavior*. Englewood Cliffs, NJ : Prentice-Hall, Inc., pp. 315–42.

Hackman, J.R., 2002, Leading Teams, *Harvard Business School Press*, Boston, MA.

Hall, T., Beecham, S., Verner, J. and Wilson, D., 2008 . 'The Impact of Staff turnover on Software Project: The Importance of Understanding What makes

Software Practitioners Tick', *Proceedings of ACM SIGMIS CPR*, ACM New York, pp. 30-39

Handzic M. and Hasan H. 2003, 'The Search for an Integrated KM Framework', Chapter 1, *Hasan H. and Handzic M. (eds) Australian Studies in Knowledge Management*, UOW Press, Wollongong, pp. 3-34,

Hansen, M. T., Nohria, N., and Tierney, T., 1999. 'What's Your Strategy for Managing Knowledge', *Harvard Business Reviews*, Vol. 77, Issues 2, pp. 106-118.

Hansen, B.H. and Kautz, K, 2004. 'Knowledge Mapping: A Technique for Identifying Knowledge Flows in Software Organisations',. *Lecture Notes in Computer Science*. *Springer*, Vol. 3281/2004

Hansen, B. and Kautz, K, 2005, 'Grounded Theory Applied – Studying Information Systems Development Methodologies in Practice', in *Proceedings of 38th Annual Hawaiian International Conference on Systems Sciences*, Big Island, HI.

Hanratty, P., 1997, Ireland as a Software Location, *Irish Development Agency*, Dublin.

Hall, T., Rainer, A. and Baddoo, N. 2002, 'Implementing Software Process Improvement: An empirical Study', *Software Process, Improvement and Practice*, Vol. 7, No 1, pp. 3-15.

Hall, T., Wilson, D., Rainer, A. and Jagielska, D., 2007, 'Communication: the neglected technical skill', *Proceedings of the 2007 ACM SIGMIS CPR conference on 2007 computer personnel doctoral consortium and research conference*: The global information technology workforce.

Hardgrave, B. C. and Armstrong, D. J., 2005, 'Technical opinion: Software process improvement: it's a journey, not a destination', *Communications of the ACM*, Vol. 48, Issue. 11, pp. 93.

Hauck, J. C. R., von Wangenheim, C.G., de Souza, R. H. and Thiry, M, 2008, " Process Reference Guides- Support for Improving Software Process in Alignment with reference Model and Standard", *Proceeding of the 15th European Conference, EuroSPI 2008,* CCIS 16,. Springer–Verlag Berlin Heildelberg, pp. 70-81

Heeks, R. and Nicholson, B., 2004, 'Software export success factors and strategies in follower' nations', *The Journal of Global Business and Political Economy,* Vol. 8, Issue. 3, pp. 267 – 303.

Hendricks, P.HJ. and Vriens, D.J. 1999, 'Knowledge-Based systems and Knowledge Management: Friends or Foe', *Information and Management Journal*, Vol. 35, Issue. 2, pp. 113-125.

Herbsleb, J. and Goldenson, D., 1996, 'A Systematic Survey of CMM Experience and Results', *Proceeding of the 18th International Conference on Software Engineering*, Berlin Germany, pp 323-330

Hirschheim, R. 1992, ‘Information Systems Epistemology: An Historical Perspective’, *Information Systems Research: Issues, Methods and Practical Guidelines*, R. Galliers (ed.), Blackwell Scientific Publications, Oxford, pp. 28-60.

Hoegl, M. and Proserpio, L., 2004, ‘Team Member Proximity and Teamwork in Innovative Projects’. *Research Policy*, Vol. 33, No. 8, pp. 1153–1165.

Horvat, R.V., Rozman, I. and Gyorkos, J., 2000, ‘Managing the Complexity of SPI in Small Companies’, *Software Process Improvement and Practice*, Vol. 5, No. 1, pp. 45-54.

Hosbond, JH. and Rtoft, R. 2003. ‘Knowledge Management: An Empirical Investigation Of Software Process Knowledge’. *MSc Dissertation*, Aalborg University.

Hughsmith, J. 2000, Adaptive Software Development: A Collaborative Approach to Managing Complex System, Dorset House.

Humphrey, W.S. 1989. Managing Software Process, *Addison Wesley*, Reading, MA.

Humphrey, W.S., 1995. A Discipline for Software Engineering, *Addison Wesley*, Boston, MA.

Holmberg, L and Mathiassen, L. 2002, Survival patterns in fast- moving software organizatitions, *IEEE Software*, Vol. 18, No.6, pp. 51-59.

HotOrigin, 2004, Ireland Software Cluster: Preparing for Consolidations, HotOrigin Ltd., Dublin, Ireland

Hsieh, H.F.; Shannon, S.E., 2005, Three approaches to qualitative content analysis, *Qualitative Health Research*, Vol. 18, No.6, pp. 51-59

ICT Ireland , 2009 , Ireland as Location , *http://www.ictireland.ie/Sectors/ ICT/ICTDoclib4.nsf /vLookupHTML/Key_Industry_ Statistics? OpenDocument [Accessed on 20 Frbruary 2010]*

Investopedia website, 2007 www.investopedia.com, [Accessed on: 25 April 2007]

Irish Software Association (ISA) , 2009 , ‘ Industry Profile’, *http://www.software.ie/Sectors/ISA/ISA.nsf/vPages/About_us~industry-profile? OpenDocument.* [Access on 20 February 2010]

ISO 2000, *Quality Management System – Requirements*, ISO 9000, ISO, Geneva.

ISO/IEC International Standard (ISO/IEC Standard 12207): Information Technology. Software Life Cycle Processes.

ISO 9001:2000 , 2001. ISO 9000 + ISO 14000 News (2001), *The First Group User of ISO 9001:2000. http://www.iso.org/iso/home.htm*[Access on 16 February 2010]

ISO 9001:2008, ISO 9000 essentials, International Standard for Business, Government and Society, *International Organizational for Standard, http://www.iso.org/iso/iso_catalogue/management_standards/iso_9000_iso_14000/ iso_9000_essentials.htm* [Access on 16 February 2010]

Janz,B.D., 1999, 'Self-directed Teams in IS : Correlates for Improved System Development Work Outcomes', *Information & Management,* Vol. 35, No. 3, , pp. 171-192.

Javed, T., Maqsood, M. and Durrani Q., 2004, 'A Survey to Examine the effect of Team Communication on Job satisfaction in Software Industry', *ACM SIGSOFT Software Engineering Notes***,** Vol. 29, Issue. 2, pp. 6-6.

Johanssen, R. T., 2004, 'An investigation into Software Process Improvement in Small and its application in Scandinavian picture agency group', *Dissertation Master of Science Information Technology*, University of Sunderland.

Johnson, M.D., Hollenbeck, J. R., Humphrey, S. E., Ilgen, D. R., Jundt. D, Meyer C. J., 2006, 'Cutthroat Cooperation: Asymmetrical Adaptation to Change in Team Reward Structures', *The Academy Management Journal*, Vol. 49, No. 1, pp. 103-119.

Joppe, M. 2000. 'The Research Process', *http://www.ryerson.ca/~mjoppe/rp.htm* [Accessed on: 12 January 2007]

Kaplan, B. and Maxwell, J.A. 1994. 'Qualitative Research Methods for Evaluating Computer Information Systems', *Evaluating Health Care Information Systems: Methods and Applications, J.G. Anderson, C.E. Aydin and S.J. Jay (eds.),* Sage, Thousand Oaks, CA, , pp. 45-68.

Kaplan, B. and Duchon, D. 1988. 'Combining Qualitative and Quantitative Methods in Information Systems Research: A Case Study', *MIS Quarterly*, Vol 12, No. 4, pp. 571-586.

Katzenbach, J.R., and Smith D.K., 1993 The Wisdom of Team. Creating the High Performance Organization. *Harvard Business Scholl Press*, Boston, MA.

Karlström D. and Runeson, P., 2006, 'Integrating agile software development into stage-gate managed product development', *Journal Empirical Software Engineering*, Vol. 11, Issue. 2, pp. 203-225.

Kaltio, T. and Kinlula, A. 2000, 'Deploying the Defined SW Process', *Software Process: Improvement and Practice*, Vol. 5, Issue. 1, pp. 65-83.

Kautz, K., 1998, 'Software Process Improvement in a Very Small Enterprise: Does it Pay Off?', *Software Process Improvement and Practice*, Vol. 4, No. 4, pp. 209-226.

Kayworth, T. R., and Leidner, D. E. 2001. 'Leadership effectiveness in global virtual teams'. *Journal of Management Information Systems*, Vol. 18, Issue. 3, pp. 7-40.

Keil, M., Cule, P. E., Lyytinen, K, and Smidth, C.R., 1998, A Framework for Identifying Software Project Risks. *Communications of the ACM*, Vol. 41, Issue 11.

Kess P.1; Haapasalo H. 2002. 'Knowledge creation through a project review process in software production'. *International Journal of Production Economics*, Vol. 80, No. 1, pp. 49-55.

Kettunen, P., 2003. 'Managing embedded Software project Team Knowledge', *Software, IEEE Proceedings,* Vol. 150, Issue: 6, pp. 359- 366.

Kiesler, S., 1978, 'Interpersonal Processes in Groups and Organization', *AHM Publishing Arlington Height.*

King, W.R. and Marks, P.V. 2008., 'Motivating knowledge sharing through a knowledge management system', *Omega*, Vol. 36, Issue. 1, pp. 131–146.

Kitzinger J. 1995 'Introducing focus groups', *British Medical Journal,* No. 311, pp.: 299-302.

Kirk, J. and Miller, M.L., 1986, 'Reliability and validity in quantitative research', *Beverly Hills, Sage Publications*.

Kirkman, B.L., Rosen. B, Tesluk, P. E. and Gibson, C.B., 2004, 'The impact of team empowerment on virtual team performance: The moderating role of face-to-face Interaction', *Academy of Management Journal*, Vol. 47, No 2, pp. 175-192.

Knauber, P., Muthig, D., Schmid, K. and Widen, T. 2000. 'Applying Product Line Concepts in Small and Medium-Sized Companies', *IEEE Software*, Vol. 17 No. 5, pp. 88-95.

Kraut, R.E. and Streeter, L. A. 1995. 'Coordination in Software Development', *Communication of the ACM*, Vol. 38, Issue. 3, pp. 69-81.

Krasner, H. and T. Ziehe. 1995. 'Lessons learned from the semiconductor industry initiative for improving software process quality, and reliability'. *Proceedings of the First World Congress for Software Quality, American Society for Quality Control (ASQC)*, San Francisco, CA, session D, pp.1-20.

Krasner, H. 1997. 'Accumulating the Body of Evidence for the Payoff of Software Process Improvement'. *In http://www.utexas.edu/coe/sqi/archive/krasner/spi.pdf* [Accessed 18th April 2007]

Kruchten. P, 2000, The Rational Unified Process, *Addison Wesley,* Reading, MA.

Krueger, R. and Casey, M., 2000, ‘Focus Groups: A Practical Guide for Applied Research’, *Thousand Oak*, CA: *Sage*.

Krueger, R. A., 1994. ‘Focus Groups: A Practical Guide for Applied Research’, *Thousand Oaks*, CA:Sage

Komiyama T, Sunazuka T, Koyama S. 2000. ‘Software process assessment and improvement in NEC - current status and future direction’. *Software Process Improvement and Practice,* Vol. 5, Issue. 1, pp. 31-43.

Kozlowski, S.W.J. and Ilgen, D. R., 2006, ‘Enhancing the Effectiveness of Work Groups and Teams’, *Psychological Science in the Public Interest*, Vol. 7, No. 3, pp. 77–124.

Kukko, M., Helander, N. and Virtanen, P., 2008. ‘Knowledge Management in Renewing Software Development Processes’, *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*, pp. 332-332.

Kohlbacher, F. 2006. The use of qualitative content analysis in case study research. *IN: Forum Qualitative Sozialforschung/Forum: Qualitative Social Research.*

Kvale, S., 2007. ‘Doing Interviews’, The Sage Qualitative Research Kit, *Thousand Oaks, CA: Sage*

Laporte, C., April, A and Renault, A. 2006,’Applying ISO/IEC Software Engineering Standard in Small Setting. Historical Perspective and Initial Achievement’, *SPICE Conference*, Luxemburg.

Laporte, C.Y., Alxender, S. and Renault, A., 2008a, ‘Developing International Standards for Very Small Enterprises’, *Journal of Computer*, Vol. 41, Issue. 3, pp.98-108.

Laporte, C. Y. and April, A., 2006, ‘Applying Software Engineering Standards In Small Setting : Recent Historical Perspectives and Initial Achievements’, *International Research Workshop in Small Setting*, Software Engineering Institue, Pittsburgh, Oct 19-20.

Laporte, C.Y., Alxender, S. and O’Connor, R.V., 2008b. ‘A Software Engineering Lifecycle Standard for Very Small Enterprise’, *Proceeding of the 15th European Conference, EuroSPI 2008 Industrial Proceeding,* Springer PUBLIZON, pp. 10.33-10.41.

Laporte, C.Y. and April, A., 2005. ‘Applying Software Engineering Standards in Small Setting: Recent historical perspectives and initial achievements’, *International Research Workshop for Process Improvement in Small Settings*, Software engineering Institute, Pittsburgh

Laporte, CY; Renault, A.; Desharnais, JM; Habra, N.; Abou El Fattah, M.; Bamba, JC. 2005. ‘Initiating Software Process Improvement In Small enterprise: experiments with CELTIC’s Micro-Evalustion Framework SWDC-REK’, *International Conference on Software Development*,University of Iceland, Reykjavik, Iceland May 27-June 1, pp. 153-163.

Larman, C., 2004. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. *Prentice Hall*, Third edition.

Larman, C. and Basili, V.R., 2003, 'Iterative and incremental developments. A brief history', *Computer*. Vol. 36, Issue. 6, pp. 47- 56.

Lee. B and Miller. J., 2004. 'Multi-Project Management in Software Engineering Using Simulation Modelling', *Software Quality Journal*, Vol.12, No. 1, pp. 59-82.

Levi, D. 2001, Group Dynamics for Teams, *Sage Publications*.

Lindval, M., Rus, I and Sinha, S. 2003, 'Software System Support for Knowledge Management', Journal *of Knowledge Management*, Vol. 7, Issue. 5, pp. 137-150.

Lethbridge, C.T., Sim, E.S. and Singer, J., 2005, 'Studying Software Engineering; Data Collection, Techniques for Software Field Studies', *Empirical Software Engineering,* Vol.10, Issue 3, pp.311-34.

Levin, D. M., 1988. 'The Opening of Vision: Nihilism and the Postmodern Situation' Routledge, London

Li, J. Y., 2006, 'Process Improvement and Risk Management in Off-the Shelf Componenet-Based Development', *PhD Thesis*, Norwegian University science and Technology.

Littlepage, G.E., Cowart, L. and Kerr, B., 1989. 'Relationships between Group Environment Scales and Group Performance and Cohesion', *Small Group Research*, Vol. 20, No. 1, pp. 50-61.

Litern, G., Diedrich F.J and Serfaty, D. 2002. 'Engineering the Community of Practice for Maintenance of Organizational Knowledge'. *Human Factors and Power Plants, Proceedings IEEE 7th Conference on 2002* pp. 6-7- 6-13.

Locke, F.L, Silverman, J.S and Spirduso, W.W, 1998, 'Reading and Understanding Research', *Sage Publication*, Thousand Oaks London.

Marks, A.M. and Panzer, F. 2004, 'The Influence of Team Monitoring on Team Process and Performance', *Human Performance*, Vol. 17, No. 1, pp. 25-41.

Maholtra, Y., 2000. 'Why Knowledge Management systems fail? Enablers and Constraints of Knowledge Management in Human Enterprise', *In Michael E.D. Koenig and T. Kanti Srikantaiah (Eds.), Knowledge Management Lessons Learned: What Works and What Doesn't, Information Today Inc. (American Society for Information Science and Technology Monograph Series)*, pp. 87-112, Mathiassen, L., Nielsen, P.A., and Pries-Heje, J. 2002. ‘Learning SPI in Practice’. *Chapter 1 from Improving Software Organization : From Principle to Practice* edited by Lars Mathiassen, Jan Priese Heje and Ojelenki Ngwenyama, Addision Wesley.

Mathiassen L.; Pourkomeylian P., 2003. ‘Managing knowledge in a software organization’. *Journal of Knowledge Management*, Vol. 7, No. 2, pp. 63-80.

Mathiassen, L.; Ngwenyama, O.K. and Aaen, I. 2005, ‘Managing change in software process improvement’, *Software IEEE*, Vol. 22, Issue. 6, pp. 84- 91.

Mathison, S.,1998.. ‘Why Triangulate?’ *Educational Research*, No. 17, Vol. 2, pp13-17.

McCaffery, F. and Coleman, G., 2008. ‘Analysing the cost of Lightweight SPI Assessments’, *Proceeding of the 15th European Conference, EuroSPI 2008, Industrial Proceedings,. Springer* Heildelberg, pp. 3.1-3.24.

McCarty, B., 2005. ‘Dynamics of a successful Team. What are the enablers and barriers to High Performing Successful Teams?’, *MSc Dissertation*, Dublin City University.

McFeeley, B. 1996. IDEAL: A User’s Guide for Software Process Improvement. Hand-book, *CMU/SEI-96-HB-001*. Pittsburgh: SEI.

McGrath, J. E., 1982, ‘Dilemmatic: The studies of the research choice and dilemma’ *Judgement Call in Research*, Kulka, R. A. Ed. Beverly Hills: Sage Publication, pp 69-102

McGibbon, T. 1999. ‘A business case for software process improvement revised—measuring return on investment from software engineering and management’, *Air Force Research Laboratory contract no. SP0700-98-4000,* http://dacs.dtic.mil/techs/roispi2. [Accessed on : 7 December 2006]

McGregor, D. 1985. *The Human Side of Enterprise: 25th Anniversary Printing*, McGraw-Hill/Irwin

McGuire, E.G. 1996. ‘Initial Effects of Software Process Improvement on an Experience Software Development Team’. *System Science, Proceeding on Twenty Nine Conference*. Vol.1, Issue. 10, pp. 713-721.

McIver Consulting, 1998, Manpower, Education and Training Study of Irish Software Sector, *A report submitted to Software Training Advisory Committee and FAS, Dublin,* Ireland.

Meehan, B., Richardson, I., 2002, ‘Identification of Software Process Knowledge Management’, *Software Process: Improvement and Practice*, Vol. 7, Issue. 2 , pp. 47 – 55.

Ming Huo, Verner, J., Liming, Zhu. and Babar, M.A. 2004. ‘Software quality and agile methods’, *Computer Software and Applications Conference* (COMPSAC) Proceedings of the 28th Annual International, Vol.1, pp.520- 525.

Miles, M.B. and Huberman, A.M., 1994 Qualitative Data Analysis: An Expanded Source Book (2nd Ed.) *Sage* Publication , p. 338

Mingers, J, 2001, Combining IS Research Methods: Towards a Pluralist Methodology, *Information System Research*, Vol 12, No. 3, pp. 240-259.

Mishra, D. and Mishra, A., 2008. Software Process Improvement for Small and Medium Enterprises: Techniques and Case Studies, *Information Science Reference*, p. 376.

Morgan, D.; Fellows, C.; Guevara, H., 2008. ‘Emergent Approaches to Focus Group Research’, *Handbook of Emergent Methods*, Guilford Press

Morse, J.M., 1991. ‘Approaches to Qualitative-Quantitative Methodologies Triangulation’, *Nursing Research*, Vol. 40, Issue. 2, pp. 120-123.

Mtigwe, B., 2005. ‘The entrepreneurial firm internationalization process in Southern African context: A comparative approach’. *International Journal of Entrepreneurial Behaviour and Research,* Vol.11, Issue. 5, pp 358-377

Mulligan, D., O’Leary, W. and O’Sullivan, D., 2003. ‘Innovation Management Practice in Irish SMEs’, *available at http: //fluffy.org /14_Innovation_Management_ Practice_SMEs.pdf.* [Accessed on 7 February 2007]

Mullin, R, 1996, ‘Knowledge Management A cultural evolution’, *Journal of Business Strategies,* Vol. 17, No. 5, pp. 56-59.

Myers, M.D., 1997, ‘Qualitative Research in Information Systems’, *Management Information System Quarterly*, Vol. 21, No. 2, pp. 241-242.

Newell, S., Tansley,C. and Huang, J., 2004, ‘Social Capital and Knowledge Integration in an ERP Project Team: The importance of Bridging and Bonding’, *British Journal Of Management*, Vol. 15, Issue. s1, pp. 43-57.

Newton, S.K., Blanton, J. E. and Will, R., 2008, ‘Innovative Work and Citizenship Behaviours from Information Technology Professionals: Effect of Their Psychologist Contract’, Information *Resource Management Journal*, Vol. 21, Issue. 4, pp.25-46.

Niazi. M, Wilson, D. and Zowghi D., 2006, ‘A framework for assisting the design of effective software process improvement implementation strategies’, *Journal of Systems and Software*, Vol. 78, Issue 2, pp. 204-222.

Nierstrasz, O. 2004. ‘Putting Change at the Center of the Software Process’, *Lecture Notes in Computer Science*, Springer, Vol. 3054/2004.

Nonaka, I., 1991. 'The knowledge-creating company', *Harvard Business Review*, Vol. 69, No. 6, pp. 96-104.

Nonaka, I., 1994. 'A Dynamic Theory of Organizational Knowledge Creation', *Organization Science*, Vol. 5, No. 1, pp. 14-37.

Nonaka, I., Toyama, R., Konno, N., 2000, 'SECI, Ba and leadership: a unified model of dynamic knowledge creation', *Long Range Planning*, Vol. 33, Issue. 1, pp. 5-34.

Nonaka, I. andToyama R., 2003. 'The knowledge-creating theory revisited: knowledge creation as a synthesizing process'. *Knowledge Management Research & Practice*, Vol 1, No.1, pp. 2-10

Nonaka, I. and Takeuchi, H., 1995. 'The Knowledge Creating Company'. *Oxford University Press*: New York.

O'Brien, R., 2001. 'An Overview of the Methodological Approach of Action Research'. In Roberto Richardson (Ed.), Theory and Practice of Action Research. João Pessoa, Brazil: http://www.web.ca/~robrien/papers/arfinal.html (Accessed 23 May 2008)

Oates, B.J., 2006, 'Researching Information Systems and Competing', Sage Publication, London.

Oestreich, P.C. and D.R. Webb. 1995. 'The race to level 3'. *Crosstalk - The Journal of Defense Software Engineering*, Vol. 8, No. 6, pp. 233-239.

Oktaba, H., 2005, '3.2 MoProSoft®: A Software Process Model for Small Enterprises', *Conference Proceeding*, *International Research Workshop for Process Improvement in Small Settings.*

Olsen, T, Humphrey, W S, and Kitson, D., 1989, Conducting SEI-assisted software process assessments. Pittsburgh, PA,Software Engineering Institute,. (*Technical Report CMU/SEI-89-TR-07.*)

Onwuegbuzie A. J. and Johnson R. B., 2004, 'Mixed Methods Research: A Research Paradigm Whose Time Has Come', *Educational Researcher*, Vol. 33, No. 7, pp. 14-26

Orci, T. and Laryd, A. 2000. 'Dynamic CMM for Small Organizations – Implementation Aspects', Journal, Stockholm University, April

Orlikowski, W.J., Markus, M.L. and Lee, A.S., 1991. 'A Workshop on Two Techniques for Qualitative Data Analysis: Analytic Induction and Hermeneutics' *Proceedings of the Twelfth International Conference on Information Systems*, pp. 390-1.

Pallant, J. 2005, 'SPSS Survival Manual: A Step by step guide to data analysis using SPSS for Windows (2nd Ed)', Maidenhead, *Open University Press*

Palmer, S. and Felsing, J. 2002, A Practical Guide to Feature-Driven Development, *Prentice Hall*, NJ.

Parent, M. Gallupe, R.B. Salisbury W.D. and Handelman, J.M., 2000, 'Knowledge creation in focus group: can group technologies help?' *Information and Management***,** Vol. 38**,** Issue. 1, pp. 47–58.

Patton, M.Q, 2002 'Qualitative Evaluation and Research Methods (3nd Ed.)'. Newbury Park, CA: *Sage Publications*, Inc.

Patton, M. Q. 1990, 'Qualitative Evaluation and Research Methods (2nd Ed.)'. Newbury Park, CA: *Sage Publications*, Inc.

Paulish, D.J. and Carleton, A.D. 1994. 'Case Studies of Software Process Improvement Measurement'. *IEEE Computer*, Vol. 27, No. 9, pp. 50-59

Paulk, M., Weber. C., Curtis , B. and Chrissis,M.B., 1994. 'The capability Maturity Model Guidelines for Improving the Software Process', *Addison Wesley* .

Paulk, M., 1994, 'A Comparison of ISO 9001 and Capability Maturity Model for Software', *CMU/SEI-94-TR-12, Software Engineering.*

Paulk, M.C., Weber, C.V., Curtis, B. and Chrissis, M.B., 1995 'The Capability Maturity Model: Guidelines for Improving the Software Process'. *Addison Wesley*, Reading MA.

Phongpaibul, M. and Boehm, B., 2005, 'Improving quality through software process improvement in Thailand: initial analysis', *ACM SIGSOFT Software Engineering Notes*, Vol. 30, Issue 4, pp. 1-6.

Pfleeger, S. and Kitchenham, B. 2001. 'Principles of survey research Part 1: Turning lemon into lemonade.' *Software Engineering Notes*, Vol. 26, No. 6, pp.16-18.

Pino, F.J.; García, F. and Piattini, M., 2008, 'Software process improvement in small and medium software enterprises: a systematic review', *Software Quality Journal* Vol. 16, Issue. 2, pp. 237-261.

Plessis, M., 2007, 'Knowledge management: what makes complex implementations successful?', *Journal of Knowledge Management,* Vol. 11, No. 2, , pp. 91-101.

Polanyi. 1966, The Tacit Dimension. London:Routledge

Politis, J.D., 2003. 'The connection between trust and knowledge management: what are its implications for team performance'. *Journal of Knowledge Management*, Vol. 7, No.5, pp. 55-66.

Probst, G., Raub, S and Romhardt, K. 1999. 'Knowledge Management building Blocks for Success'. *Wiley, Chicester.*

Punch, M. 1998, ‘Introduction to Social Research: Quantitative and Qualitative Approaches’, *Sage Publication*, London.

Ramesh, B. and Tiwana, A., 1999 ‘Supporting Collaborative Process Knowledge Management in New Product Development Teams Decision Support Systems', *Journal Decision Support System*, Vol. 27, No. 1-2, pp. 213-235.

Reed, T.F and Kelly, D 2002, ‘The Skill Gap in The Irish Software Industry’, *Irish Journal of Management*, Vol. 23, Issue. 2, pp. 95-110.

Rhodes, J.; Hung, R.; Lok, P.; Lien, BY; Wu, C.M., 2008, ‘Factors influencing organizational knowledge transfer: implication for corporate performance’ *Journal of Knowledge Management*, Vol. 12, Issue 3, pp. 84-100.

Richardson, I. and von Wangenbeim, C.G., 2007, ‘Guest editors’ Introduction: Why are Small Software Organizations Different?’, *IEEE Software*, Vol. 24, pp. 18-22.

Robillard, P.N. 1999., ’The role of knowledge in software development’, *Communications of the ACM*, Vol. 42, No.1, pp. 87–94

Rocha, A.R., Montoni, M., Santos,G., Mafra, S., Figueirdo, S., Albuqurque, A. and Mian, P., 2005. ‘Reference Model for Software Process Improvement: A Brazilian Experience’, *Proceeding of the 12th European Conference, EuroSPI 2005,* Springer–Verlag Berlin Heildelberg, pp. 130-141.

Ropponen, J. and Lyytinen, K. 2000. Components of software development risk: How to address them? A project manager survey. *IEEE Transactions on Software Engineering*. Vol. 26, Issue 2, pp 98-112.

Rosen, C.C.H., 2005, ‘The Influence of Intra Team relationships on the systems Development Process: A theoretical Framework of Intra-Group Dynamics.’ *17th Workshop of the Psychology off Programming Interest Group*, Sussex University

Rossman, G.B and Wilson, B.L, 1985, ‘Numbers and Words, Combining Quantitative and Qualitative Method in a Single Large Scale Evaluation Method’, *Evaluation Review*, Vol. 9, Issue. 5, pp. 627-643.

Royce, W.W., 1970, ’Managing The Development of Large Software Systems: Concepts and Techniques’, *Proceedings in Proceedings of the 9th international conference on Software Engineering*, pp. 328 - 338 .

Rout, T., 2004, ’ISO/IEC 15504 - Evolution to an International Standard’, *Software Process: Improvement and Practice*, Vol. 8, Issue. 1, pp. 27 – 40.

Rus, I. and Lindvall, M., 2002. ‘Knowledge management in Software Engineering’. *Software, IEEE*, Vol. 19, Issue. 3, pp. 26-38.

Salas, E., Burke, C.S. and Cannon-Bowers, J.A., 2000, ’Teamwork: Emerging Principles’, *International Journal of Management Reviews*, Vol. 2, Issue. 4, pp. 339-356.

Sanders, M., 1998, ‘The Spire Handbook: Better, Faster, Cheaper Software Development in Small Organization’, *Centre for Software Engineering*, Dublin Ireland.

Sandelowski, M., 2000, ‘Combining Qualitative and Quantitative Sampling, Data Collection and Analysis Techniques in Mixed Method Studies’, *Research in Nursing and Health*, Vol 23, No. 3, pp. 246-255

Sapsed, J., Bessant, J., Partington, D., Tranfield, D. and Young, M., 2002, ’Teamworking and Knowledge Management: A Review of Converging Themes’, *International Journal of Management Reviews*, Vol. 4, No. 1, pp. 71–85.

Sarma, A. and Van der Hoek, A., 2004, A Need Hierarchy for Teams, *www.ics.uci.edu/asarma/maslow.pdf* [Accessed on : 29 May 2007]

Sapovadia, Vrajlal K., 2006. ‘Micro Finance: The Pillars of a Tool to Socio-Economic Development. Development Gateway’. Available at SSRN: *http://ssrn.com/abstract=955062* [Accessed on: 9 October 2008]

Sawyer S. and Guinan P. J., 1998, ’Software development: processes and performance’, *IBM Systems Journal*, Vol. 37, Issue. 4, pp. 552 - 569

Scacchi, W. 2002 ‘Process Models in Software Engineering’, *in J. J. Marciniak (ed.), Encyclopedia of Software Engineering,* 2nd. Edition.

Scarborough, H., Swan, J., Preston, J., 1999, ‘Knowledge Management: A Literature Review: Issues in People Management’. *Institute of Personnel and Development*. London:

Scarnati, J. T., 2001, ’On becoming a team player’, *Team Performance Management*, Vol. 7, Issue.1/2, pp. 5 – 10.

Schwaber, K. and Beedle, M. 2002, ‘Agile Software Development with Scrum’, *Prentice Hall.*

Seaman, C.B, 1999. ‘Quantitative method in Empirical Studies in Software Engineering’, *IEEE Transaction for Software Engineering*, Vo. 25, No. 4, pp. 557-572.

Sfetsos, P., Angelis, L. and Stamelos, I. 2006. ‘Investigating the extreme programming system–An empirical study’, *Journal Empirical Software Engineering*, Vol.11, No. 2, pp. 269-301 .

Shaw, M. E., 1976. Groups Dynamics: the psychology of small group behavior. 2nd edition, *New York, Mac Grew Hill*

Shaw, M. E., 1981, Group Dynamics: The Psychology of Small Group Behavior (3rd Ed.). *McGraw-Hill, New York*

Shaw, D., Edward J.S, Baker, B. and Collier, PM. 2003, 'Achieving Closure Through Knowledge Management', *Electronic Journal on Knowledge Management*, Vol. 1, Issue. 2, pp. 197-204.

Sheard A.G. and Kakabadse A.P., 2004, 'A Process Perspectives on Leadership and Team Development', *Journal of Management Development*, Vol. 23, No. 1, pp. 7-106.

Sieber, S.D, 1973, 'The Integration of fieldwork and Survey Methods, American' *Journal Sociology*, Vol. 73, No. 6, pp.1335-1359.

Sirvio, K.S., Mantyniemi, A. and Seppanen, V., 2002. 'Toward a practical solution for capturing knowledge for software projects', *Software, IEEE*, Vol. 19, Issue. 3, pp. 60-62.

Singh, S. K., 2008., 'Role of leadership in knowledge management: a study', *Journal of Knowledge Management,* Vol.12, Issue. 4, pp.3 - 15

Simon, H. A., 1991, 'Bounded Rationality and Organizational Learning', *Journal Organizational Science*, Vol. 2, No. 1, pp. 125-134.

Sjoberg, D.I.K, Dyba, T. and Jorgensen, M., 2007, 'The Future of Empirical Methods in Software Engineering Research', *International Conference on Software Engineering*, pp: 358-378.

Sommerville, I., 2004, Software Engineering, *7th Edition, Addison Wesley, Reading MA.*

Sollingen, R.V. 2004, 'Measuring the ROI of Software Process Improvement', *IEEE Software*, Vol. 21, No. 3, pp. 32-38

Stahler, G. J. and Cohen, E, 2000.'Using Ethnographic Methodology in Substance abuse Treatment Outcome', *Research Journal of Substance Abuse Treatment*, Vol. 18, Issue. 1, pp. 1-8

Stalhane, T., Dingsoyr, T., Hassen. K.G., and Moe, N.B. 2003. 'Post Mortem- an assessment of two approaches'. *Proceedings of the European software Process Improvement 2001 (EuroSPI), LNCS*, Springer, Vol. 2765/2003

Staples D. S., Cameron A. F., 2005 'The Effect of Task Design, Team Characteristics, Organizational Context and Team Processes on the Performance and Attitudes of Virtual Team Members', *Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05), Track 1*, pp. 52a

Strauss, A., and Corbin, J., 1998. *Basics of qualitative research: Techniques and procedures for developing grounded theory* (2nd ed.). Thousand Oaks, CA: Sage.

Strauss, A., and Corbin, J., 1990. *Grounded theory research: Procedures, canons, and evaluative criteria, Journal of Qualitative Sociology,* Vol.3, No.1, pp. 3-21.

Steiner, I.D. 1966, 'Models for Inferring Relationships between Group Size and Potential Group Productivity', *Behaviourial Science*, Vol. 11, No. 4, pp. 273-283.

Steiner, I.D. 1972, 'Group Processes and Productivity', *NY Academic Press* , New York.

Stenmark, D., 2002. 'Information vs. Knowledge: The Role of Intranet in Knowledge Management', *35th Annual Hawaii International Conference on System Sciences (HICSS'02)*, Vol. 4, pp. 104b.

Stelzer, D. and Mellis, W., 1998. 'Success factors of organizational change in software process improvement', *Software Process: Improvement and Practice*, Vol. 4, Issue. 4, pp. 227 – 250.

Stelzer, D., Mellis, W. and Herzurm, G., 1996, 'Software Process Improvement via ISO9000. Result of two surveys among the European software houses', *Software Process Improvement and Practice*, Vol. 2, pp 197-210

Syed Abdullah, S., 2005, 'Empirical Study on Extreme Programming', *PhD Thesis Department of Computer Science, University Of Sheffield,* Sheffield University

Szulanski, G., 1996. 'Exploring internal stickiness: impediments to the transfer of best practice within firm', *Journal of Strategic Management*, Vol. 17, pp.27-43.

Tashakkori, A., and Teddies, C. 1998, 'Mix Methodology: Combining Quantitative and Qualitative Approaches'. *Applied Social Research Methods*, Vol. 46, Thousand Oaks, California: Sage Publication

Tallon, P.P. and Kraemer, K.L. 1999, 'The Impact of Technology on Ireland's Economic Growth and Development: Lessons for Developing Countries'. *Hawaiian International Conference on System Science Proceedings*, IEEE, Maui Hawaii.

Taylor, S. J. and Bordan, R, 1984, 'Introduction to Qualitative research Method', John Wiley & Sons

The Concise Oxford Dictionary , 1998.

Thomas, D., 1995, 'Component Based Software Construction; Making the Transition from Craft to Engineering', *Object Management Group*, New York.

Tracy, H., Beecham, S.,Verner, J. and Wilson, D. 2008. 'The Impact of Staff Turnover on Software Project: The importance of Understanding when Makes Software Practitioners Tick', *Special Interest Group on Computer Personnel Research Annual Conference Proceedings of the 2008 ACM SIGMIS CPR conference on Computer personnel doctoral consortium and research*, Association for Computing Machinery(ACM), pp. 30-39 .

Triplett. N., 1998. ‘The Dynamogenic Factors in Pace making and Competition’. *American Journal of Psychology*, Vol. 9, Issue. 4, pp. 507 -533.

Trienekens, Jos J. M., Kusters, R. J., van Genuchten, J. I. M. and Aerts, H. 2007. ‘Targets, drivers and metrics in software process improvement: Results of a survey in a multinational organization’, *Software*, Vol. 15, No. 2, pp. 135-153.

Turner, K., Makhija, M. 2006, ‘The role of organizational controls in managing knowledge’, *Academy of Management Review*, Vol. 31 No.1, pp.197-217.

Turner, R., Jain, A. 2002. ‘Agile Meets CMMI: Culture Clash or Common Cause?’ *Lecture Notes in Computer Science, Springer*, Vol. 2418/2002.

Urdanetta, F.C., 1999, ‘Teams or communities - Organizational structures for knowledge management’, *Special Interest Group on Computer Personnel Research Annual Conference Proceedings of the 1999 ACM SIGCPR conference on Computer personnel research, p*p. 128 – 134.

Umarji, M. and Seaman C., 2005. ’Predicting acceptance of Software Process Improvement’. *International Conference on Software Engineering Proceedings of the 2005 Workshop on Human and social factors of software Engineering*, pp. 1 - 6.

Valtanen, A. and Sihvonen, H.M., 2008. ‘Employees’ Motivation for SPI: Case Study in a Small Finnish Software Company’. *Proceeding of the 15th European Conference, EuroSPI 2008,* CCIS 16,. Springer–Verlag Berlin Heildelberg, pp. 152-163.

Verschuren, P. 2003. ‘Case Study as a Research Strategy: Some Ambiguities and Opportunities’, *International Journal of Social research Methodology*, Vol. 6, No. 2, pp.121-139

Visconti1, M. and Cook, C. R., 2004. ‘An Ideal Process Model for Agile Methods’ *Lecture Notes in Computer Science, Springer* , Vol. 3009

Von Wangenheim, C.G., Punter, T.D. and Anecleto, A., 2003, ’Software Measurement of Small and Medium Enterprises’. *Proceeding 7th International Conference on empirical Assessment in Software Engineering (EASE).*

Von Wangenheim, C.G., Anacleto, A. and Salviano, C.F., 2006. ‘Helping small companies assess software processes’, *IEEE Software*, Vol. 23, Issue. 1, pp. 91-98.

Wagner, J. A., 2000. ‘Organization Team’, *Encyclopedia of Psychology*, Vol.6, Issue. 5, pp. 14-20. New York .

Walczak, S., 2005, ’Organizational Knowledge Management Structure’, *The Learning Organization,* Vol. 12, No. 4, pp 330-339.

Ward, J. and Aurum, A., 2004, ‘Knowledge Management in Software Engineering - Describing the Process’, *Australian Software Engineering Conference (ASWEC'04)*, pp.137-146

Walsh, J.P. and Ungson, G. R., 1991.’Organizational Memory’. *Academy of Management Review*, Vol. 16, No.1, pp 57-61.

Ward, R.P., Fayad, M.E. and Laitinen, M. 2001. ‘Software Process Improvement in the Small’, *Communications of the ACM*, Vol. 43, Issue. 11, pp. 113-116.

Wiegers, K. E., 1998. ‘Software Process Improvement: Eight Traps to Avoid’, *Crosstalk, The Journal of Defense Software Engineering*.

Williams, L., 2004 ‘A Survey of Agile Development Methodologies’, *http://agile.csc.ncsu.edu/SEMaterials/AgileMethods.pdf.*

Weick. K.E ., 1995, ‘Sense Making in Organization’, *Sage Publication, Thousand Oak*: CA.

Wigg, K.M. 1999, ’Comprehensive Knowledge Management’, *Knowledge Research Institute Inc.*

Wittenbaum, G.M. 2003. ’Putting Communication into the Study of Group Memory’, *Human Communication Research*, Vol. 29, Issue. 4, pp. 616–623.

Wohlwend, H. and Rosenbaum, S., 1994, ‘Schlumberger’s Software Improvement Program’, *IEEE Transaction on Software Engineering*, Vol. 20, Issue 11, pp. 833-839.

Xu, J. and Quaddus, M., 2005, ‘A Reality-based guide to KMS diffusion’, *The Journal of Management Development*, Vol. 24, Issue. 4, pp. 374-389.

Yin, R.K. 1994. ‘Case Study Research: Design and Model’, *Sage Publications*, Newbury Park, CA.

Yin, R.K 2003, ‘Case Study Research’. *Design and Methods (3rd Ed.)* Thousand Oaks, CA:Sage

Zhang, D. and Zao, L., 2006.’ Knowledge Management in Organization’, *Journal of Database Management*, Vol.17, No.1, pp. 1-8.

Zahran, S., 1998, ’Software Process Improvement: Practical Guidelines for Business Success’, *Addison Wesley, Boston, MA.*

Zyngier, S., 2002. ‘Knowledge Management Obstacles in Australia’, *Proceeding of the 10th European Conference on Information Systems, Gdansk, June*

List of Appendices

APPENDIX A: List of Potential and Involved Companies

No	Company Name	Total Employee	VSE Category	Method Planned	Method Applied	Status
1.	Company A	1-9	Micro VSE	Interview/ Focus group	Not Available	Reject
2.	Company B	1-9	Micro VSE	Interview/ Focus group	Interview/ Focus group	Agree
3.	Company C	10-24	Larger VSE	Interview/ Focus group	Interview/ Focus group	Agree
4.	Company D	1-9	Micro VSE	Interview/ Focus group	Survey Questionnaire	Agree
5.	Company E	1-9	Micro VSE	Interview/ Focus group	No Replied	Reject
6.	Company F	1-9	Micro VSE	Interview/ Focus group	Not Available	Reject
7.	Company G	10-24	Larger VSE	Interview/ Focus group	No Replied	Reject
8.	Company H	10-24	Larger VSE	Interview/ Focus group	Not Available	Reject
9.	Company I	1-9	Micro VSE	Interview/ Focus group	No Replied	Reject
10.	Company J	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	No Replied	Reject
11.	Company K	10-24	Larger VSE	Interview/ Focus group/ Survey Questionnaire	Not Available	Reject
12.	Company L	10-24	Larger VSE	Survey Questionnaire	Survey Questionnaire	Agree
13.	Company M	10-24	Larger VSE	Interview/ Focus group/ Survey Questionnaire	No Replied	Reject
14.	Company N	10-24	Larger VSE	Interview/ Focus group/ Survey Questionnaire	No Replied	Reject
15.	Company O	10-24	Larger VSE	Survey Questionnaire	No Replied	Reject
16.	Company P	10-24	Larger VSE	Survey Questionnaire	Survey Questionnaire	Agree
17.	Company Q	1-9	Micro VSE	Survey Questionnaire	Survey Questionnaire	Agree
18.	Company R	10-24	Larger VSE	Survey Questionnaire	No Replied	Reject
19.	Company S	1-9	Micro VSE	Survey Questionnaire	No Replied	Reject
20.	Company T	10-24	Larger VSE	Survey Questionnaire	Survey Questionnaire	Agree
21.	Company U	1-9	Micro VSE	Survey Questionnaire	No Replied	Reject
22.	Company V	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	No Replied	Reject
23.	Company W	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	Not Available	Reject
24.	Company X	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	Not Available	Reject
25.	Company Y	10-24	Larger VSE	Interview/ Focus group/ Survey Questionnaire	Not Available	Reject
26.	Company Z	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	No Replied	Reject
27.	Company AA	10-24	Larger VSE	Interview/ Focus group/ Survey Questionnaire	No Replied	Reject
28.	Company AB	10-24	Larger VSE	Interview/ Focus group/ Survey Questionnaire	No Replied	Reject
29.	Company AC	10-24	Larger VSE	Survey Questionnaire	Survey Questionnaire	Agree
30.	Company AD	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	Survey Questionnaire	Agree
31.	Company AE	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	Interview/Interview	Agree
32.	Company AF	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	Interview/Interview	Agree
33.	Company AG	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	Survey Questionnaire	Agree
34.	Company AG	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	Survey Questionnaire	Agree
35	Company AH	1-9	Micro VSE	Interview/ Focus group/ Survey Questionnaire	Survey Questionnaire	Agree

APPENDIX B: Interview Guide

COMPANY NAME:	
NAME:	
CURRENT POSITION:	
CONTACT DETAILS:	
INTERVIEW DATE:	
LOCATION:	
START TIME:	
END TIME:	

Interview Guide

Introduction Script
Introduction:
First of all, I would like to express my appreciation for your willingness for this interview. For your information this interview will last approximately 60 minutes). The purpose of this interview is to discuss your and/or organizations experience and opinion in relation to the process of software development project especially in the issue of software development process (*software process maintenance and evolution*). Furthermore I would to make you aware that there is no right and wrong answer for all the questions.

FYI, there are six main parts in today's interview, which will cover:

a. Company background (approx. 5mins)
b. Employment Issues (approx.10mins)
c. Company software project and process issues (approx 30mins)
d. Software development knowledge issues (approx 20mins),
e. software development team issues (approx. 15mins)
f. Relation of all above questions (approx. 10mins)

FYAck, the context of this research is to understand how small software companies manage their previous and existing software project knowledge and software development team in software development project. In addition, this research also will study the awareness of software process standard among small companies. These issues have well stated that software development project is highly dependent to knowledge, people and procedure. Therefore these three main soft issues must study closely especially in small companies in order to remain relevant in nowadays software business.

Confidentiality
Any answers or explanations you make will be confidential. You name or any your identifying information will not be included in my report. I am interested in what you have said, not in who says what. So I want you feel like you are speak freely. Moreover, several small software companies will involved in this research, in order to help me to understand the trend/pattern how small software companies experience and plan in managing the software project in order to their relevancy in today dynamics business environment.

Recording
To assist me with analysing interview data, I would like to record the interview with your permission. Please understand that the main purpose this recording is to assist me in capturing all the interview data more accurately since I am not an English native speaker. Furthermore, this recording will only be listening by me and will be destroyed once data is analysed

I would like to stress here that your participant in this interview is very valuable and vital to assist me in understanding important issues surrounding the development of software systems and will ultimately benefit for the Irish Software industry. Finally, the results of the study will be shared with you when the research is completed.

Part 1: VSEs Company Background: Interviewee, Company and Staff

No	Question	
1.	Demographic (Person, name, designation , qualification, experience and role)	
2.	Company Background • History/Core business/Years of Foundation/ Client/Company developed since foundation	
3.	Hiring Issue – hiring process and criteria/ No of Staff/ Location	

Part 2: VSEs People Issue

No	Question	
4.	How do people enhance their expertise (skills)? – Formal training given- How/what	
5	Working Process • Explain how people work in here – in a team/ alone/ autonomous working • Formal Team -How team been created • Clear Role	
6.	Communication • Explain about typical employee communication and interaction each other – formal meeting/ online communication/ Autonomous communication • Formally documented all these activities • People relationship (Mgmt-Subordinate) (Staff-staff) Communication tools?	
7.	Management Style - Do you considered your staff turn overrate to be high? – Yes, why and how you control? No – why do you think is not a problem	

Part 3: VSEs Software Development Issue

No	Question	
8.	What you can tell me about your software development process - Specific model - Process clearly defines - How regular the development process been assess from one project to another? Why - Do you have standard documented procedure that in developing a system?	
9	Does your development process being change and evolve over time? Explain/ why **Yes** : Align with business process? Refer to any standard? Do people will use the same development process or the new one? How to encourage them. **No** : Is mean you use the same development process over time? Why? Do you think you should improve the development process overtime? Why	
10.	If there any changes in software project, how this issue been anticipated?	
11	Post Mortem - Did you do any post mortem for a deployed project? – how and why - Do documented this activities/ outcome?	
12	Does your software development project regularly receive feedback from the project team and client? ○ YES ○ NO IF ‘YES’ : i. Please explain how feedback has been delivered in your organization ?	

13	Quality standard - What is your opinion about software quality standard? – ISO - Do your company plan to adopt it? Why - What a barrier small companies like you encounter in adopting software standard? Any assistance needed	

Part 4: VSEs Software Development Knowledge Issue

No	Question	
14.	Sharing knowledge - Explain how knowledge been shared in this company? How and how frequent - Do people experience and idea been document? How - Do people need to document their activities? - Do the company have a standard documentation process? - Does people experience and lesson learn being documented?	
15.	How do you encourage people to share their knowledge?- any reward (tangible/ intangible)	
16.	How do you ensure that staff knowledge and experience in software development /project will be secure before they leave?	

Part 5: VSEs Software Development Team Issue

No	Question	
17.	Can you explain about development team in this company? Well define structure Appropriate team size Well balance team member Well define team process Good team communication People relationship How team been formed? Does any specific criteria involved in forming a team?	
18.	Based on your experience, are your software development teams involved directly in defining the software development process? ○ YES ○ NO IF 'YES' : i. Please explain how this process happens?	
19.	How do you see the relationship between staff (and also management)? – socialize, work place, team work, close	
20.	In your opinion, how to ensure companies competitiveness in current business environment?	

APPENDIX C: Focus group Interview Guide

COMPANY NAME:	
NAME:	
CURRENT POSITION:	
CONTACT DETAILS:	
INTERVIEW DATE:	
LOCATION:	
START TIME:	
END TIME:	

Focus Group Guide

Introduction

First of all, I would like to thank you for attending this session today. I know how busy all of you are and really appreciate your willingness to help me out with this focus group.

How many of you have been in a focus group before? Well the main reason why I bring all of you together is to hear all your different idea and experiences. Today I invited you because I want to hear about your experience and opinion about software development project and process.

For your information this discussion will last approximately 60minutes and there is no right and wrong answer for all the questions. There are six main areas in today's discussion, which will cover:

a. Company background
b. Employment Issues
c. Company software project and process issues
d. Software development knowledge issues
e. software development team issues
f. Relation of all above questions

(The context of this research is to understand how small software companies manage their previous and existing software project knowledge and software development team. In addition, this research also will study the acceptance and awareness of software process standard among small companies. Therefore these three main soft issues must study closely especially in small companies in order to remain relevant in nowadays software business.)

Moderator/Participant roles

The basic way this works is that you will be the talker and I will be the listener. Again remember you can freely express you taught and idea because there no right and wrong answer for this activities.

In fact, most of the talking you'll be doing be to each others. I'll have some question you need to ask, for you talk them among yourself. My basic job is to make sure that the topics get fully explore, and to make sure that we get to hear all different points of view.

Ground Rules

We do have a few basic ground rules, *The first*, thing is to participate. The reason that I've invited all of you today is so we can hear your different point of view. So we need everybody's help to have a good group.

The second thing is to take turn. I know that some people like to talk more than others, but sometimes you may have to hold on to some of things that you would like to say, so everyone in this group has a time to talk.

(*Finally*, it's all right to disagree with each others, but please be polite when do- no put downs. You want other people to listen to what you say and show some respect when they take to you, right? So you need going to do the same for everyone else in the group.)

Tape procedure

To assist me with analysing discussion data, I would like to record our discussion with your permission. Please understand that the main purpose this recording is to assist me in capturing all the interview data more accurately since I am not an English native speaker. Furthermore, this recording will only be listening by me and will be destroyed once data is analysed

Confidentiality

Any comments you make will be confidential. You name or any your identifying information will not be included in my report. I am interested in what you as a group have say, not in who says what. So I want you all feel like you are speak freely. Finally, I ask that you respect others privacy. What ever we say here today is just for this group. I know you don't want other people repeating anything that would violate your privacy, so all basically have to trust each other. The results of the study will be shared with you when the research is completed.

Opening

No	Questions	
	In brief (1 minute), describe your background, qualification, role and experience in this company. **Explain the company project that you have been involved?**	

Introductory/ Transition

No	Questions	
1	• **(10 mins) In your opinion, what is special about this company? (Turn over rate issue.- how the organization handle this / why is low)** • **In your opinion how can you explain about your organization environment? (communication, relationship and commitment)** o How people communicate each other?-typical employee communication o How frequent do you have a meeting? How it been conduct? o Relationship between team and management o Do you think the management are very supportive in all aspects? • **Do this company provide any formal training?** • **How do you enhance your skill and knowledge in performing your job?**	

Software development Issue

No	Questions	
2.	**(15 mins) In your opinion, what should organization do, in order to sustain relevance in today's software environment? How about your organization?** • What are challenges that your organization face in software business nowadays? • What are the important factors involved? • How management could play it role in this matter? • How development team could contribute? • How the management anticipate any changes in software project such as market change, standard change, requirement changes and others in software development project? • Does your software development project regularly receive feedback from the project management team? Do you aware and understand company business goals?	
No	Questions	
3.	• **(15 mins) How the development project been done? – in a team or individual (autonomous work)** • **any specific development model you follow** • **do you improve the development process/step over time** • Do you regularly assess your software process practice? Why and How? (periodic review) o How the management commitment on this matter? • **Do you think software process need to be changed and evolved overtime? Why and how?** • Do these changes aligned with the company business goal and process?	

	• In your opinion, should the development process be change or update over time? Why and how? • **In your opinion, how can we ensure the team follows the updated software process rather than inventing or creating a new process for every project?** **Based on your experience, are you involved directly in defining the software development process?**	
No	Questions	
4..	**(15 mins)** *(briefly explain about software process standard and outcome- Software process standard in the previous interview)* • **What is your opinion about software engineering standard?** • **Do you think about adopting a software engineering standard in software organization?** • **Why the acceptance of software engineering standard in small company is low?** • **How to encourage small company to adopt software engineering standard?** o Do you follow any standard in assessing your/project software process? How and Why? o Have the company look for external accreditation since your company's creation? o Barrier that exist from adapting any standard. o Awareness/Understand about certification- what should other VSE do. o Do you think your company or other small company should adapt any Software process standard? Why • Formal/ informal • Method/ standard apply and follow • Advantages/ Disadvantages • Knowledge issues • Company/management commitment • **In your opinion, what kind of software engineering standard that should be developed for VSEs? –** ***In your opinion, what are the desirable characteristics of standard for VSE?*** o In your point of view, what type of assistance (encouragement) do small companies like you needs in adopting/involving in software process standard (or new standard seriously)?	o Formal vs informal approach Software process maintenance and evolution Is it important to have periodic reviews of software project? Do employee get any training (formal/ informal) What are the factors that enforce/ encourage /discourage software process to be improved and evolved? Why Team issues o Management issues o Standard Issues o Knowledge Issues o Communication issues o *How the management commitment regarding this matter?*
No	Questions	
5...	(10 mins) "As stated in software development process literature, one of the biggest problems in software development projects is the process loss problem. Process loss is a situation where software team follow old and ineffective development processes instead of the recent process for developing software". What are your comments on the above issues? How this could be happened? How it should be handled? In your experience, has your organization had an experience with the process loss? ○ YES ○ NO IF 'YES' : i. Please explain why this problem happened? ii. Please explain how this problem is being handled in your organization? IF 'NO' : i. Please explain what action should your organization take to handle this process loss?	

Software Development Knowledge

No	Questions	
6...	**(15 mins) "***Software development projects by their nature is a teamwork effort and involve knowledge intensive exchange, the influence of software development teams and well organized software development knowledge could help in determined the success of software project.* • **In your opinion, does people knowledge are very important in all organization? why** • **How it should be managed? How knowledge been managed in this company** • **How does software development knowledge being managed in this organization?** o What are factors that could encourage people to share their knowledge in your organization? • **In your opinion, what are the factors could influence people share their knowledge and experience in organization?** • **Can you describe about sharing knowledge culture in this organization and how it has been done?** o *How does team development learn and understand previous project?* o In our opinion, how should team experience and knowledge be managed? Why? o It this knowledge been shared, transferred and document? How and why o How team are shared their knowledge? • **What is your opinion about documentation (is it important in software project?) why** • **Can you explain about the documentation process in this organization?** o Is the document been update/ compile over time? – formally/informal o Do people document their task /activity properly? o Management of software team experience and knowledge in developing software project o Formally or non formal o Documenting knowledge issues o Tools o Communication and interaction issues o Management encouragement • **Is turn over staff id critical in this company? Why?** • **How to secure staff knowledge and experience before they leave? Why and how it should be done?** **Do you agree that an organize people knowledge could give a good impact to software development project? why**	

Software Development Team

No	Questions	
7.	(10 mins) How you can explain about software development team in this company? Well define structure Appropriate team size Well balance team member Well define team process Good team communication What are typical types of project you work on? – In term of size, duration for each project, team in each different project Critical size How do you see about relationship among team member? --- And between management and non management? In your opinion, how to get full cooperation from team member in development project or organization? What organization should do in order to create an effective and efficient development team? Do you agree that a good team will enhance knowledge sharing and enhancing project success? Why	

Ending

No	Questions	
8.	**(2mins). As a summary, in your opinion how important is knowledge management and software team in software development process?** • **How should it be handled/managed -** Knowledge sharing and transferring/ knowledge management Team, standard	
No	Questions	
9.	**(1min)**Side Questions • Anything that you want to add more about these issues. – haven't discuss enough • Anything can improve(organization) in this issue (SPI, km and team) • Anything that you want to suggest for my future discussion/ interview?	

APPENDIX D: Research Survey Questionnaire

DCU

Dublin City University
Ollscoil Chathair Bhaile Átha Cliath

Dear Sir, Madam

We are requesting your assistance with a research project which is investigating the Software Development Process practices of Irish SMEs. A major part of this study is the gathering of industry data by Mr. Shuib Basri (a Ph.D. research student in DCU), who would requests your assistance in completing a questionnaire to ascertain your individual and organizational experiences in relation to the process of software development.

We realise that your time is valuable. The participation of organisations like yours is vital assist our collective understanding of important issues surrounding the development of software systems and will ultimately benefit the Irish Software industry as a whole. We will of course share our results and findings with you.

This questionnaire is divided into 5 main parts as follows:

- Background information
- Questions on people issues
- Questions focussed on the software development process
- Questions focussed on the software development knowledge
- Questions focussed on the software development team issues

We would like to assure you that the information gathered in this questionnaire will be treated as confidential and all completed questionnaires will be seen only by the Ph.D. student and his supervisor (named below). No personal / company identifying information will be disclosed to anyone whatsoever.

Mr. Shuib Basri
Ph.D. Research Student
sbasri@computing.dcu.ie
(01) 700 5828

Dr. Rory V O'Connor
Senior Lecturer
roconnor@computing.dcu.ie
(01) 700 5643

School of Computing, Dublin City University
Lero, The Irish Software Engineering Research Centre

Instructions

Please read all the instructions and complete this questionnaire. Our preference is that you type your replies into this document and email the completed questionnaire to us at email address below. Alternatively, you may print this questionnaire and hand writes the answers. In this case, please post or fax completed questionnaire to address or fax number below.

Email: *sbasri@computing.dcu.ie* *or* *roconnor@computing.dcu.ie*

Postal Address: Shuib Basri, PhD Student, School of Computing, Dublin City University, Glasnevin, D9, Dublin

Fax no: *(01)* 700 5442

PART A: RESPONDENT INFORMATION AND COMPANY BACKGROUND

COMPANY NAME:

NAME: ________

CURRENT POSITION:

EXPERIENCE (IN YEARS): **Current Company :**

Career to Date :

CONTACT DETAILS: **Email:**

Tel. No:

HOW LONG HAS THIS COMPANY BEEN IN EXISTENCE:

COMPANY TOTAL EMPLOYEES: **OVERALL:**
TECHNICAL:
SUPPORT/ADMIN :

COMPANY PROJECTS

- **NO. OF PROJECT CURRENTLY IN PROGRESS**
- **NUMBER OF PROJECTS COMPLETED WITHIN THE LAST 2 MONTHS:**
- **TYPICAL PERSON-MONTH EFFORT PER PROJECT:**

PART B: PEOPLE ISSUES

B1.	**When hiring new software development staff, what criteria does your organisation use to evaluate applicants?**
B2.	**Does your organization provide formal training to new software development employees?** ○ YES ○ NO **IF 'YES': Please explain what type of training and why given?** **IF 'NO': Please explain.**
B3.	**Do you consider the software development employee turnover rate in your organization to be high?** ○ YES ○ NO **IF 'YES': Please explain how your can organization control/prevent this situation?** **IF 'NO': Please explain, why do you think staff turnover is not a problem in this organization?**

PART C: SOFTWARE DEVELOPMENT PROCESS

C1.	**Do you consider the software development process in your organization changes / evolves over time?** ○ YES ○ NO **IF 'YES': i. What is the motivation for such changes?** **ii. Are these processes changes documented?** **IF 'NO': Do you consider the software development process should change / evolve over time. Please explain why?**
C2.	**Does management and / or software development team regularly review and update project documents?** ○ YES ○ NO **IF 'YES': How are these activities typically completed?** **IF 'NO': How would software project documented be reviewed and updated?**
C3.	***"As stated in software development process literature, a problem in software development projects is the process loss problem. Process loss is a situation where software developers follow older (possibly less effective) development processes instead of current processes".*** **Has your organization had an experience with 'process loss' as described above?** ○ YES ○ NO **IF 'YES': What do you think caused this to arise and how is it being handled?** **IF 'NO': How would you handle this situation if it arose?**

C4.	**Is your organization accredited by any external standards organization such as ISO?** o YES o NO **IF 'YES': i. What type of certification does your organisation hold?** **ii. What was the motivation to obtain such certification?** **(Please proceed to C6)** **IF 'NO': i. Will your organization consider such certification in future (next 1-3 years)?** o YES **(Why?)** (**Please proceed to C5)** o NO **(Why?)** **(Please proceed to C7)**
C5.	**i. What type of certification does your organization plan to obtain?** **ii. What type of assistance would your organization need to realise this certification goal?** **(Please proceed to C7)**

C	***Software Development Process***							
C6.	Only answer **C6** below if your organization is accredited by external standards organization such as ISO (**C4: Yes**). What are the reasons for implementation of the selected standards or certification? Please indicate your agreement or disagreement. Do so by circling **(O)** the appropriate number from the rating scale.			Strongly agree	Agree	Neutral	Disagree	Strongly disagree
	a.	Market regulation requirement		***5***	***4***	***3***	***2***	***1***
	b.	Customer demand		***5***	***4***	***3***	***2***	***1***
	c.	A general desire to improve product quality.		***5***	***4***	***3***	***2***	***1***
	d.	A general desire to increase productivity.		***5***	***4***	***3***	***2***	***1***
	e.	A general desire to improve software development process.		***5***	***4***	***3***	***2***	***1***

	f.	To increase customer satisfaction.		***5***	***4***	***3***	***2***	***1***
	g.	To reduce cost.		***5***	***4***	***3***	***2***	***1***
C7.	Only answer **C7** below if your organization is **NOT** accredited by external standards organization such as ISO (**C4: No**). Please indicate reasons your organization is **NOT** implementing international standards? For each statement below, please indicate your agreement or disagreement. Do so by circling **(O)** the appropriate number from the rating scale.			Strongly agree	Agree	Neutral	Disagree	Strongly disagree
	a.	Not required by customers.		***5***	***4***	***3***	***2***	***1***
	b.	Lack support from management.		***5***	***4***	***3***	***2***	***1***
	c.	Lack support from government/state agencies		***5***	***4***	***3***	***2***	***1***
	d.	Lack of internal resources.		***5***	***4***	***3***	***2***	***1***
	e.	Lengthy and difficult in implementation standard		***5***	***4***	***3***	***2***	***1***
	f.	Insufficient time to implement standard.		***5***	***4***	***3***	***2***	***1***
	g.	Insufficient guidance available.		***5***	***4***	***3***	***2***	***1***
	h.	No desire to change existing processes and procedures.		***5***	***4***	***3***	***2***	***1***
C8.	Consider the software development practices in your organisation. For each statement below, please indicate your agreement or disagreement. Do so by circling **(O)** the appropriate number for the following scale.			Strongly agree	Agree	Neutral	Disagree	Strongly disagree
	a.	Software development projects regularly receive feedback over stakeholder.		***5***	***4***	***3***	***2***	***1***
	b.	Software development staff are highly motivated.		***5***	***4***	***3***	***2***	***1***
	c.	Software development staff receive recognition for their work		***5***	***4***	***3***	***2***	***1***
	d.	Software development staff regularly receives guidance and support from management.		***5***	***4***	***3***	***2***	***1***
	e.	Software development staff knowledge is formally documented		***5***	***4***	***3***	***2***	***1***
	f.	We regularly document experiences/lessons learned from previous projects to use in future projects.		***5***	***4***	***3***	***2***	***1***
	g.	There is an effective communications channel between software development tem members and management		***5***	***4***	***3***	***2***	***1***
	h.	Software development staff are directly involved in planning and improving software development processes		***5***	***4***	***3***	***2***	***1***
	i.	There are regular formal meetings between software development staff and management.		***5***	***4***	***3***	***2***	***1***

	j.	There are regular informal (casual) communications between software development staff and management.		***5***	***4***	***3***	***2***	***1***
C9.	Consider the software development practices in your organisation. For each statement below, please indicate your agreement or disagreement. Do so by circling **(O)** the appropriate number for the following scale.			Strongly agree	Agree	Neutral	Disagree	Strongly disagree
	a.	We have established a Software Process Improvement (SPI) goals.		***5***	***4***	***3***	***2***	***1***
	b.	There is a broad understanding of SPI goals and policy within our organization.		***5***	***4***	***3***	***2***	***1***
	c.	Our SPI goals are closely aligned with organizational business goals.		***5***	***4***	***3***	***2***	***1***
	d.	We have a good balance between short term and long tem SPI goals.		***5***	***4***	***3***	***2***	***1***
	e.	Senior management actively supports SPI activities.		***5***	***4***	***3***	***2***	***1***
	f.	We are regularly consult software quality standards in performing our job		***5***	***4***	***3***	***2***	***1***
	g.	Software developers have freedom in planning and managing their work.		***5***	***4***	***3***	***2***	***1***
	h.	Software development staff are actively involved in setting goals for SPI activities.		***5***	***4***	***3***	***2***	***1***
	i.	The software process changes / evolves overtime		***5***	***4***	***3***	***2***	***1***
	j.	Software development staff are actively involved in creating process and procedure for software development		***5***	***4***	***3***	***2***	***1***
	k.	Management regularly assess software development process		***5***	***4***	***3***	***2***	***1***
	l.	When software processes are updated / changed, software developer always follow the new process.		***5***	***4***	***3***	***2***	***1***
	m.	Software development teams are regularly document and review their activities.		***5***	***4***	***3***	***2***	***1***
	n.	Management team are knowledgeable about software quality standards		***5***	***4***	***3***	***2***	***1***
	o.	Software development staff are knowledgeable about software quality standards		***5***	***4***	***3***	***2***	***1***
	p.	Management ensure that organisational standard processes are always followed by software developers.		***5***	***4***	***3***	***2***	***1***
Please provide any additional comments which you may have in relation to the software development issues in your organization.								

PART D: SOFTWARE DEVELOPMENT KNOWLEDGE

D1. **Does your organization formally manage software development team knowledge?**

o YES o NO

IF 'YES': i. Please explain how?

ii. Is such knowledge formally / informally documented?

IF 'NO': How is software development team knowledge in your organization managed?

D2. ***"As stated in software process improvement literature, one of the obstacles in improving the software development projects is the knowledge-loss problem. Knowledge loss is a situation where knowledge in software development becomes eroded and obsolete due to the departure of key staff in software organization".***

Has your organization experienced such 'knowledge loss'?

o YES o NO

IF 'YES': How is this problem addressed in your organization?

IF 'NO': How would you address this problem in your organization if it arose?

D	***Software Development Knowledge***							
D3.	Consider the software development practices in your organisation. For each statement below, please indicate your agreement or disagreement. Do so by circling **(O)** the appropriate number for the following scale.			Strongly agree	Agree	Neutral	Disagree	Strongly disagree
	a.	Good team communication is important in managing software development knowledge.		***5***	***4***	***3***	***2***	***1***
	b.	A knowledge management strategy is important in managing organisational knowledge.		***5***	***4***	***3***	***2***	***1***
	c.	Software communication tools (e.g. Skype, Blog, etc) support and enhance knowledge propagation.		***5***	***4***	***3***	***2***	***1***
	d.	It is necessary to have a reward / incentive mechanism to ensure knowledge sharing / transfer.		***5***	***4***	***3***	***2***	***1***
	e.	Management are very committed to sharing of knowledge and knowledge transfer activities.		***5***	***4***	***3***	***2***	***1***

	f.	Regular formal meetings are important for sharing and transfer of knowledge		***5***	***4***	***3***	***2***	***1***
	g.	Formal training is given in order to sustain and enhance software development knowledge.		***5***	***4***	***3***	***2***	***1***
	h.	Good working relationships between software development staff enhance knowledge sharing / transfer.		***5***	***4***	***3***	***2***	***1***
	i.	Good leadership is important in leveraging peoples knowledge and experience.		***5***	***4***	***3***	***2***	***1***
	j.	Formal project post-mortems are beneficial in capturing and transferring knowledge.		***5***	***4***	***3***	***2***	***1***
	k.	Clear documentation processes and guidelines are necessary to sustain knowledge in software development		***5***	***4***	***3***	***2***	***1***
	l.	We are follow an ‘agile’ type of software development methodology.		***5***	***4***	***3***	***2***	***1***
	m.	We always exploit existing organizational knowledge to the maximum.		***5***	***4***	***3***	***2***	***1***
	n.	We always learn from experiences of past projects.		***5***	***4***	***3***	***2***	***1***
	o.	We always collect experience data from past projects.		***5***	***4***	***3***	***2***	***1***
	p.	We regularly share our knowledge in software development projects.		***5***	***4***	***3***	***2***	***1***
	q.	We regularly document our knowledge and experience in software development projects.		***5***	***4***	***3***	***2***	***1***
	r.	We regularly documented our work / project progress.		***5***	***4***	***3***	***2***	***1***
	s.	We regularly share opinions and thoughts on our software development activities.		***5***	***4***	***3***	***2***	***1***

Please provide any additional comments which you may have in relation to software knowledge issue in your organization.

PART E: SOFTWARE DEVELOPMENT TEAM

E1. **Does your software development project regularly receive feedback from the project stakeholders (management, software development staff and customer)?**

o YES o NO

IF ‘YES’: How is such feedback delivered given?

IF ‘NO’: In your opinion, what are the potential benefits to the software development team, of such feedback?

E2. **Are the software development team involved directly in defining the software development process?**

o YES o NO

IF ‘YES’: Briefly explain how they are involved in this?

IF ‘NO’: Should the software development team be involved directly involved and why?

E3. **From your experience and observation, does your software development team work effectively and efficiently?**

o YES o NO

IF ‘YES’: Please explain why?

In your opinion, how can one create an effective and efficient team in a software organization?

E	***Software Development Team***							
E4.	Consider the software development practices in your organisation. For each statement below, please indicate your agreement or disagreement. Do so by circling **(O)** the appropriate number from the following rating scale.			Strongly agree	Agree	Neutral	Disagree	Strongly disagree
	a.	Software development staffs always understand projects goals.		***5***	***4***	***3***	***2***	***1***
	b.	There is clear communication between team members.		***5***	***4***	***3***	***2***	***1***
	c.	There is a good social relationship among software development staff		***5***	***4***	***3***	***2***	***1***
	d.	Software development staffs have good interpersonal skills.		***5***	***4***	***3***	***2***	***1***
	e.	Clear roles are defined within the software development team		***5***	***4***	***3***	***2***	***1***
	f.	The software development team is an appropriate size for projects		***5***	***4***	***3***	***2***	***1***
	g.	Software communication tools (e.g. Skype, email, etc) are regularly been used in software development projects.		***5***	***4***	***3***	***2***	***1***
	h	Software developers are physically located close to each other (office layout)		***5***	***4***	***3***	***2***	***1***
	i.	There is a diverse range of skills among the software developers.		***5***	***4***	***3***	***2***	***1***
Please provide any additional comments which you may have in relation to software development team issue in your organization.								

THANK YOU VERY MUCH FOR YOUR ASSISTANCE
Shuib Basri, PhD Student
School of Computing, Dublin City University, Glasnevin, D9, Dublin
Email: sbasri@computing.dcu.ie;

Appendix E : Open Coding (Illustrated)

(Notes: The following interview transcript is an extract interview from the original interview transcripts for interview open code process illustrations.)

Company : Company B -Interview : Interview
Date : 27/2/2009

Question	Open code
Q 1 and Q2 Company Background/ Interviewee Background The company was formed between me and my brother. We both are independent contractor providing software development service to no. of different companies. We always anticipate that at one stage to form business together. To do that we kind have a special client base and to do so we set up a company. Rather than trade as an independent contractor and trade as a single entities and to find obviously to find grows to the business The company was form in 2003 or 2004, cant remember exactly, the first year trading would be myself as an contractor and then in June , 2 years ago, we move to digital hub to start as software design as sole company. (1.12) Own background: Electronic engineering from UCD and my brother in Computer application from DCU.	• Owners having substantial IT experience before forming a company • Brothers Forming a family company • Merging both owner ambition in forming a company • Establishing company 5 years ago • Having own office space and employees • Both owner obtaining IT/ Computer relation degree/qualification
Q3. No of employees and new employee characteristics - 5 all of us together - we would generally look for kind of self starter, people above average abilities, good interest in computer software - interest above beyond just the academic interest , - not use of agencies for recruiting people apply job and base own our initiatives. Look for candidate with experience and work interest in computer outside their prime degree example there will be an advantage.	• Company having small number of employees • Hiring fresh graduate with average abilities and good interest in software • Not using employment agency in finding a suitable employee
Q5. Formal meeting? Formal meeting not very often because we always communicate each others so the formal meeting isn't there, we do might have a kick off our project and to the project and generally people discussing at different all of the time. Very open communication because we are very small everybody knows what every body doing or working on.	• Meeting been done informally • Frequently having casual meeting with others • Having free and open communication among staff • Perceiving everybody knows everybody in the company
Single office or share office - We have 2 offices , 2 rooms in 101 – one employee in Valencia and one employee in s/Africa	• Having offices in Dublin • Having staff working remotely in Valencia and South Africa
Q6. I mean usually either be face to face between 2 developers or over skype through 2 developer remotely communication In general the developer kind of work independently sometimes have a project, we have a sole responsibility for the project. Other time they assist each other in strategic and help each other for the output for a single project. It doesn't mean the notion of team isn't clearly set out in our company but just because of our size. In fact team will over kill someone of the project. It might be 2 months work for one person but the team such become heavy handed, you know there will be far more meeting for 3 months projects they extra kind of work. They cant be productive at the size of the project	• Talking with remote staff using communication tool e.g skype phone • Having a typical face to face communication for collocate staff • Staff having fully responsibility (autonomous) on their project • staffs are working independently • People working together only on strategic area. • No formal team notion exist currently • Being aware that people size not encourage team notion • Perceiving team will kill other in company project. • Perceiving that small project with two many people are very heavy handed and not productive
Do the development process structured and flexible? Yes , we follow the requirement specification and etc. and then we decided it didn't work. It wasn't the best model for the size at all we didn't like writing spec and customer didn't like reading it. So we just set what the point. It better to do in more interactive prototype based model where people can engage,	• Understanding old formal method required a lot of document • Understanding old formal development method is not working in small company • Doing more interactive and

assess usability seen system before to written and that's the model we use at the moment.	adopting prototype model
The evolvement processes … are you documented it? - A little bit. We set internal blog share information. We started to document as a basic knowledge of information should benefit everybody we encourage the developer to put in the blog.	• Starting share information using blog • Not all changes being documented • Using blog to share information
Do you assess your development process? We would do it a little step. What I haven't done is sat down and written how exactly the process works. And how we want the process work, to large extend I know authority what is the best it is/ for is and no doubt with the changes we will implement almost immediately. I think a lot of awful try and error in term to find out what work so. What we doing at this moment removing to like fully agile dev methodology, we doing it step by step because 2 things : i. continuity from everybody using the system. If we changes to may things at once, you can grind a halt and no one will do any development due (13.34) basically a new process. ii. we doing it little by little and that way we going to do we going to improve the process kind of and we open the discuss ask about what to do next, what is the basic/biggest problem of current process and what we could do to address that. It kind of organic at the moment and again we, with the limited resources as working on the project we do and what we can improve our process.	• Doing process assessment informally • Implementing changes immediately • Being aware of individually decision making • Adapting 'agile' method in development process • Improvement being done using try and error method • Keep using a similar working development method overtime • Being aware that changes/improvement in development process being done in small scale to prevent any major problem • Changes being done informally and small scale • Being aware that people are discussed each other on any changes • Lacking of resource lead to small changes and informal changes • Changes/improvement starting from inside
Is changes and evolvement referring to any standard? No we essentially looking at our own processes , our own work the job we are doing most are doing most are certainly automated and should be automated and area we lacking in and introducing automated user interface testing that in the pipe line. At the moment, the developer testing their own work and changes are made be to the slip on the track of the change somewhere else in user interface. The idea is to have down the road had a fully automated. Bring the idea the developer having the work on functionality you know and not as much on testing on stuff and more productive work they adding feature or new dev of any project.	• Not following any guideline or standard in improve/ change development process • Improving more focusing on find an area that could be automated • Being aware that the automated process will allow people more focusing in their work
Q13 Are these changes been documented? No and not often enough and in most cases like you know we are communicate quite well essentially what should be in and how to accomplish what to do so we looking at that respect. We also I mean this is the whole idea of formal documentation etc2. To a company of our size, you know, it really to a large extend u know it done to minimum it done at all Most of it done from the idea get the customer requirement, get their own words delivery then and making an awful a lot of assessment and decision of team behalf, u know, because they are not technical and we assess it use our knowledge and our experience, what the customer want in term of business POV, we know best deliver that.	• Update done by sharing and communicate informally • Any update process not being document often • Being aware having a small size of people , company need to focus on others rather than documentation • Team size influence documentation • Accessing requirement over time • Requirement coming from the customer • Technical solution providing by development team •
Q14 What is your opinion about ISO standard? I am only enquired about 2 days ago. Yes we do plan too, but one of things we done since started growth to quickly, when we haven't , u know' after pushing ourselves we spend time to learning how we want to do this and how we don't want to do this. We started to putting those processes in place so when we decide plan to growth we have a good platform, easy to growth , easy to sustain the growth, train people what your process and what your system is (23.20) I am going to have to do changes our process on system rather done before we started taking 10 peoples.	• Being interested in Software quality standard • Recently aware of the important of software quality standard • Looking for a lightweight software quality process • Planning to considering standard more important when having more people in company •
Which ISO standard you looking at?	• Being interest and prefer ISO9000

- ISO 9000	
Why not in standard?. Standard in one hand just seen as a sale tool. Is a good sale tool and for small company probably a lot of their client are more looking on ISO9000 b4 they looking at u. . Probably not interest and it a resources things and all time required to company with ISO9000. We can do it internally and ask somebody to do it.	- Perceiving Standard as a 'sale tool' - Perceiving standard is just an image to the company - Adopting standard required huge resource and time
Q17 Knowledge Management Our project will be use the same technology platform all the time , the domain of knowledge is not that large the size just need those little tricks and best practice you learn through experience and I don't particular think give a class one to one class to all developer will get all to get them all going and everybody has to learned from experience by looking to the others people code they implemented and all the project developer some one solves the problem, one ways can discuss among them and share knowledge that ways. As I said we want to put knowledge somewhere so we just set our blog to encourage people if they come across to took you a day or two to solve and post up there and available for everybody.	- Perceiving that not much knowledge been created - Company using same technology and process in every project - People learning through their experience and other experience - Sharing knowledge informally through casual communication and project code - Creating 'blog' to encourage people sharing opinion and knowledge
Why? Because their activities are working on the project we do have like an issue tracking system that we have in place, we start to use it more and more. So if the pieces of work come part of the project to be log in first and mark it off, I don't need they doing that , I don't need that level of micro management.	- Being confirm that all employee do not have to document their activities - PM very flexible and informal
Q19 We mean with the asset, I don't bother because to be honest you need to trust other people or you don't.	- The management are trusting people in working their work
Q25 can you explain about your software development team Team of one essential. Larger team of 2 or team of 3. No formal team structure and every body equal within the team PM are loose we good in estimate about our project, you know we good in understanding what is in requirement so the PM we don't necessarily have to dedicate the PM , just other people aware where we are in the project in one time and everybody know where we are at any one time. We again won't have dedicated test person, non dedicated UI person, so every body would be responsible for pitching in so we do have UI person (38.30) on the contract basis.	- Having flat development team structure - Having very small development team | No formal structure - Everybody having similar right and task - Having a loose project management - Alerting that all people are what happen in the company - Hiring external person in doing UI testing
Is there informal..Just like taking order that they see above? - Well I am sure there probably seen myself and brother as others or not, they do and Gavin have done a lot of dev work as well and then they will see them selves. Victor will probably seen himself due to experience in company as the head of Marcus but actually they is no real title	- Not having a formal division between management and staff

	Appendix E **Open Coding**	
Open Code (Company B/AF)	**Group Code (Category)**	**Open Code (Company C/AE)**
Owners having substantial IT experience before forming a company	People having Substantial Experience in IT/SW Dev (1)	Having IT qualification and background
Having experience in developing system		Having massive experience in IT work and project
Having substantial working experience		Knowledgeable and experience using different programming language and platform
:Having little software development experience		Company selling hardware and software
Having experience in various development environment		Having shifted company product from hardware to business (software) related to current needs
Having substantial working experience		Developing company internal software system
working mainly on dot net environment and mobile technology		Involved in most of organization software project
		Supporting company product
		Involving in several system platform before
		Working as developer and knowledge in many programming language
		Experiencing doing several IT work before
Role as Senior software development	Experience IT/SW People	Role as founder and director of the company
Having quiet substantial skill and working experience in software development		Having It background and qualification
Having IT related degree and qualification		Founder of the company Existed almost 10 years
Experience working with several big and small company in UK and Ireland		Having IT qualification , background and experience
Current role as CTO		
Both owner obtaining IT/ Computer relation degree/qualification	Having/hiring IT related background and qualification staff (1)	Having IT qualification and background
having degree in IT		Searching for innovation, hard work and able to work in stress environment
Obtaining electrical engineering degree		Having It background and qualification
Obtaining Degree in IT		Working with company for 8 years
Hiring fresh graduate with average abilities and good interest in software		Having IT background and qualification
Hiring people – having good degree qualification	Hiring Criteria	Hiring criteria more on experience, skills, passion and level of motivation
Having knowledge in specific platform		Least depends on academic qualification
Having creative and problem solving thinking		Looking for passion, skill and high motivated people
		Not very particular on qualification
Having IT related degree and qualification		Having IT qualification , background and experience
Experience and knowledgeable in many IT platform and projects		

Brothers Forming a family company	Having Family Based company (2)	Current company is small and family IT business
Merging both owner ambition in forming a company		
Gavin : Currently working with family company		
Establishing company 5 years ago	Being Establish and located at specific places (2)	2 people working in the office
Working space between people are very close		Working space between people are very close

Having own office space and employees		Having own office space and employees
Having offices in Dublin	Being small size company and having GSD operation (2)	2 people working in the office
Having staff working remotely in Spain and South Africa		2 people work remotely
Frequently coming over to Dublin office		Frequently coming over to Dublin office
It is non family business and 2 years joining his company	Company nature (2)	It is non family business and exist almost 10 years
All technical people located in WIT , Waterford	Office Location (2)	Located in new office in Cork
Mgmt in Dublin		Admitting people not understand company goal
Company having small number of employees	Having a flat structure and informal culture (3)	
Having flat development team structure		Having flat working structure/ Having small number of employee in IT department
Having very small development team | No formal structure		
Having very simple organization structure, informal work schedule and simple reporting system		Having same network set up with all people
Admitting that the team structure is flat and everybody have the same right		Frequently coming over to Dublin office
Having very simple organization structure, informal work schedule and simple reporting system		
Having good working environment	Team environment	Mgmt and staff having very close relationship
Having flexible and close relationship		Realizing age, interest, skills and experience create the environment
People expose to the new technology		No real gap between people- very flat structure
Having range of expertise		Being family and flexible environment
Having team work but each responsible their task and delivery time		Admitting we work more on team work environment
People having close relationship and easy to share idea		Having no big gap between staff and mgmt
Reason- good relation and excellent working environment		Admitting people work very close and sometime they socialize together
Working along with the developer.		This is due to the small team and easy to communicate
Having small number of staff in this department	Team Size and Structure (3)	Having small number of technical staff
Having flat structure		Having no formal team structure
People working across their area		Having very flat organization structure
Having small development team		No real gap between people- very flat structure
Having flat team and organization structure		Having small number of technical staff -Small company
Admitting no real team structure		Having a flat and informal organization structure
Having flat hierarchy		Having a flat and informal organization structure
Having close relationship and easy way		
Having small team/ technical team		
People being cut due to economy situation		
Having flat team structure		
Not giving any formal training to employee	Practicing informal culture in all development activities and in enhancing people expertise (4)	Putting all data in one place- centralize system
Allowing employee attend seminar that related to core technology that align with company process		Adopting 'train a trainer' concept
Currently under informal training purpose.		Sending staff training only when required
Easy getting help from others		Having more internal training

People could explore other technology in enhancing their skill		Exchange/sharing knowledge always happen informal
New employee need to do self learning in understanding current process and enhancing knowledge		Enhancing knowledge through self learning
No formal training been given to a new employee		Not attending any training
company not giving any for training to employee		Documentation process is depends on individual
Senior staff given frequent guide to new employee		
company not giving any for training to employee		
Team member helping others in understanding new tools and process		
New employee need to do self learning in understanding current process and enhancing knowledge		
No formal training been given to a new employee		
Company is continuing using a same technology and process in the entire project.	People Enhancing skills/expertise through self learning and sharing knowledge activities (5)	Not sending people for training/courses due to time and cost constraints
Being aware that company are busy with project		Giving opportunity people to attend short seminar and conference
People are trying to share knowledge and experience form time to time.		Adopting 'train a trainer' concept
Company is continuing using a same technology and process in the entire project.		Sending staff training only when required
Assuming most of the development technique being self understanding		Preferred conducting training internally
People could explore other technology in enhancing their skill		Having experience hiring expert given training
People could explore other technology in enhancing their skill		Being notice internal training could suit company objective and open to all
New employee need to do self learning in understanding current process and enhancing knowledge		Putting all data in one place-centralize system
No formal training been given to a new employee		Imposing contract to control trained people to leave company
company not giving any for training to employee		Perceiving giving training will encourage people to leave
Staff are reviewing their project/ code their own self		People enhancing knowledge toward exchange knowledge and experience
		Learning knowledge from new employee indirectly
		Not attending any training
Working according to their expertise and more to autonomous working	Enhancing Skill (4,5)	No formal training is given
Applying autonomous work and communication		People did self learning and sharing in mastering their job
Training is more on self learning and sharing		People are updating and sharing their progress over time and informal
Training more on hands-on or OJT		Admitting no formal training given to staff
Learning individually from internet/ sharing with people and expert		Giving in house. Internal training – more informal way
Enhanced through OJT, self learning and self explore.		People doing self learning and sharing among people
Having Frequent guidance from high/ expert people informally		
Meeting been done informally	Communication and Sharing activity being done informally, casual open , online and periodic (6)	Not sending people for training/courses due to time and cost constraints
Frequently having casual meeting with others		Giving opportunity people to attend short seminar and conference

Having free and open communication among staff		Not having team meeting frequently
People learning through their experience and other experience		Working with remote staff influence the people meeting
Sharing knowledge informally through casual communication and project code		Having difficulties in communication and observation with the remote staff
Creating 'blog' to encourage people sharing opinion and knowledge		Putting all data in one place- centralize system
Post mortem being done in informal way- through internal discussion		People having frequent communicate each others
People learning through their experience and other experience		Having open discussion prior changes
Being aware small company have open concept in communication		Changes/improve been done after series of open discussion with team development
Having open environment such as open to idea and open communication between people		Exchange knowledge happened informal
Being aware that people only response when they want it		Meeting been conducted informally in the company
Being perceived that small company having more informal and freedom in all aspect		meeting usually using skype conference
Most staff communicate with customer through email		Not having weekly meeting
Informing company basically always kept the updated database as in sourcesafe		Not having schedule meeting and informal meeting
Remote staff frequently visit main office for doing job		Having very informal meeting
Spending time in main office few times in a year		Learning knowledge from new employee indirectly
Sourcesafe generating automated updated database		important document not being shared
Using internal blog share idea		Not having a real team environment
Saving all programming code electronically and be share with other staff		Teams roles depending on project
People working across their area	Learning and Sharing Process (6)	People did self learning and sharing in mastering their job
Working according to their expertise and more to autonomous working		People are updating and sharing their progress over time and informal
People working support each other		Sharing code in order to get feedback and idea
Applying autonomous work and communication		Company having open and casual communication – open to discuss and express idea
Having open communication		Having informal sharing through casual communication and meeting/Not documented
Practicing informal environment and easy to access		People doing self learning and sharing among people
In general people are willing to share because of small team		Having an online sharing and through casual and informal discussion
Mgmt always supply related information		Encouraging people to talk to right person on specific issue all time
Sharing also been done through peer to peer programming		Sharing through on line
No documentation sharing or no formal sit down discussion		
Everybody can accept and express idea.		
More discussion and jot it down on white board		
Enhanced through OJT, self learning and self explore.		
Having Frequent guidance from high/ expert people informally		
People having close relationship and easy to share idea		
Admitting people are free to give idea and suggestion		
Sharing been done informally by shared books , direct ask question and discussion.		
Doing pair programming occasionally		
By giving frequent question and idea		

Ensure no one exclusively work in one project		
Hard to apply due to size		
People working across their area	Communication Process	People are updating and sharing their progress over time and informal
Working according to their expertise and more to autonomous working		Meeting more informal and casual
People working support each other		Company having open and casual communication – open to discuss and express idea
Practicing informal and casual communication		Having direct communication btw developer and mgmt
Having stand up meeting or short meeting normally		Not having proper meeting setup and more on standing/informal way
Following agile development practice		Discuss only the important and plan
Not having formal meeting		Practicing direct / face to face communication in all activities
Having open communication		
Having less structure communication and work process		
Customer - Feed back given by email or phone		
Customer - Feedback given directly to designer		
Enhanced through OJT, self learning and self explore.		
Having Frequent guidance from high/ expert people informally		
Practicing more face to face communication		
Having more informal, stand-up meting on daily basis		
Admitting people are free to give idea and suggestion		
Being aware that people only response when they want it	People having open and close relationship and working in informal and casual environment due to small team and low level of independent (7)	Having an open communication between management and staff
Perceiving people working happily and comfortable each other in casual and informal environment		People having frequent communicate each others
Being agree small company with small team are very close relationship and effective team work		Having family environment and flexible encourage people to stay
People/staff doing separate work/project in one time		Having open discussion prior changes
Considering all staff is in one team and having fully work autonomy		People having a good relationship
Company a flexible in doing works or assigning task		Having very informal meeting
Staff are reviewing their project/ code their own self		People working together in family environment
		Having direct contact and communication with team and management
		Company having a very informal communication and relationship
Perceiving everybody knows everybody in the company	People having an Autonomous work/No clear team notion definition exist/ People work independently (7)	Company having a flexible working structure
Staff having fully responsibility (autonomous) on their project		Being aware flexible flat working structure make people work effectively
staffs are working independently		Being flexible and freedom in working would prevents turn over
People working together only on strategic area.		Development model been use depends on the project
No formal team notion exist currently		Not following standard development process in all project
Being aware that people size not encourage team notion		Not following the same process model in all project
all staff are self dependent in their work		Being aware people are not following development standards
team culture exist very informally		Manager having freedom to make a decision

Staff are reviewing their project/ code their own self		Development team involve in decision making
Gavin : in small company people having more work autonomous and lot of autonomy		Staff doing all the related development task
Gavin : People in small company working independently in different project and own responsibility		Staff doing different project at one time
People could explore other technology in enhancing their skill		Not having standard team size for software project
Having freedom in work		Team size depend to the project
Being perceived that small company having more informal and freedom in all aspect		No formal team structure
Feeling comfortable with the small team because of flexibility and very open		Having frequent communication with other team member
Each developer working autonomously		No enforcing in following development procedure
Working autonomous encourage more interaction between team member		Following own development style in improve process
Not enforcing people to follow the guideline		Follows what is suit to company only
staffs having freedom choosing task in project		Doing development individually
Employees are able to work in a team		Documentation process is depends on individual
Staff are reviewing their project/ code their own self		new employee not convincing with the whole process
Having a lot of responsibility and freedom in doing jobs.		Team structure is depends on project
		Not having a real team environment
		Team structure is depends on project
		Not having a real team environment
		Team roles in project is define by project manager
		Team roles is based on the person expertise
Using more electronic documentation than paper documentation	Communication Tools	Most of the time using specific internal communication tools to interact – *teamwork-*
Using specific tools –xplanner- to document people activities		Sometime people using other tool to communicate personally
Using phone and other available tool communicate with management		Using company standard internal tool - teamwork
Casually using other communication tools when necessary		Using tools as needed but not frequent
Using development tool to communicate.		
Using specific tools- xplanner – to record people activity and progress		
Getting feedback through email, phoned or direct meting		
Talking with remote staff using communication tool e.g skype phone	Frequently having remote communication by using communication tools (8)	Realizing online communication is not effective in software development
Having a typical face to face communication for collocate staff		Using communication tools communicate with remote staff
Remote staff not feeling Isolated with support from communications tool		Team having a daily discussion and communication
		Frequently using communication tools to communicate
		Felling no different working with remote staff with support from communication tool
		meeting usually using skype conference
only one staff leaving since the company exist.	Not having serious staff turn over (9)	Being aware that economic situation give advantage
Perceiving current economy situation people will not leave their job.		Planning to get rid of in effective employee
“Not up to scratch” – not very good staff		Having not serious staffs turn over.
		Internal and external environment influencing people sustaining in

		company
		Being flexible and freedom in working would prevents turn over
		Having family environment and flexible encourage people to stay
		People having a good relationship
Having low turn over rate	Staff Turn Over Issue	Not a problem
No people left the company last 2 years		Confirming that no people so far left the company
		Being busy and having a lot of project in recession time compare to other company
Following similar development process from time to time	Using same technology and process in all project (10)	Technology not impact process and procedure change
Keep using a similar working development method overtime		Having same network set up with all people
Keep using a similar and workable development process		Following proof and own development style in doing project
Not able taking risk to impose a new process to the new project#		
Company using same technology and process in every project		Company not following a good coding process
Using a similar technology like others		Documentation process is depends on individual
remote staff having the same system setup like collocate staff		new employee not convincing with the whole process
People have the same operation in company system. environment		Using the same technology all the time
every staff have same set up.		Using the same technology and development process all the time
Using same method that being proof efficient in development		Following what process and technology is suit to company and been proof efficient development process only
Using the same process and procedure to avoid conflict in future		
Using any guideline/ standard that fit the project objective		
Being aware current process is workable and effective		
Perceiving current development process is better and no need to improve		
having interactive communication with client - especially when collecting requirement	Using interactive and prototype dev process/ Agile process development/ Not following similar development process (11)	
Using prototype method in develop project.		Having team discussion in evaluating changing and evolving process
Adopting prototype method encourage interaction and feedback from client		People having frequent communicate each others
Being aware prototype most suitable with the company project		Company following various development model in developing system
Follow customer needs and requirement		Development model been use depends on the project
Adapting 'agile' method in development process		Not following standard development process in all project
Adopting agile method type of methodology in doing software project		Not following the same process model in all project
Manager also working in the development		Having frequent communication with other team member
Implementing changes immediately		Helping each other in some task
Having open environment important to improve process		Perceiving agile method is more suitable for small company
Having less documentation process		Having direct contact and communication with team and management
Agile method could help improving development process		

Following agile development process and practice	SPIME	Using same development tools
Applying autonomous work and communication		Using similar development process in all product
Practicing informal environment and easy to access		Using standard development tools
Having every 2 week update and test		Being aware that development process always been improve
Having continuous integration on product		Improvement in small scale and informal way.
Having minor changes on major process.		Having rapid development process evolution
Assessing current process and product of problem arise		Involving all people in improve development process
Staring from inside rather than outside		Changes/improvement start from customer or internal.
Having a mock up/prototype to customer to get feedback		Discuss with customer and also the team
Designer improved based on the comment		Evolvement process start from inside and apply to client product
Not following other/outside standard guideline		Following coding standard depend on language/ Not following specific model
Having followed loosely extreme programming process		More following RAD development model
Having customer and internal input		Being aware customer feedback and internal view main reason for any process update
Developing and update road map to manage development process		Applying small scale and informal update/change but very rapid
Doing 2 weeks iteration		Admitting profitable drive the development process
Frequently review plan to tailored with job		Doing informal post mortem process
Having notice both process align but less people avoid meet company objective		Admitting post mortem is least priority activities
Using similar development process in all project		Realizing that process always being improve but indirect and informal ways
Having small changes overtime to suit objective and requirement.		
Plan well and study for changes		
No formal post mortem process		
Customer involve in start and end of process actively		
Explaining agile is more to out put then process		
Being aware that the development process been update continuously and indirectly		
Improve but not structure		
Being aware that detail technical specification and documentation is not useful	Having less documentation internally and externally/ Informal documentation due to high frequency of changes and similar process (12)	sharing problem experience through informal communication
understanding client not having technical background		Sharing process been done informally
Understanding old formal method required a lot of document		Sharing been done through informal communication within team and between department
Understanding old formal development method is not working in small company		Having very informal meeting
Understanding that the development process being evolve overtime and not suitable to formally document		Documentation process not following International standard process like ISO
Being aware that all process not clearly define and formally document		Company not have formal knowledge sharing culture
Following similar development process from time to time		Documentation is not priority in small company
Not all changes being documented		Documentation system is effective due to small no of people
Do not having complete development documentation		important document not being shared
Being aware that all improvement process not being documented		No clear documentation process been practice

Not all development activities being document – “not over document”		
Documenting specific document and issue	Having formal and controlling documenting procedure/ documenting specific issue only (12)	Controlling and documenting all people activities and work
Having proper documentation and feedback process		Being agreed that documentation is an important process to prevent knowledge loss
All process not being formally define and document		Having different version of documentation and product (evolvement process)
		Being aware that all problem being documented
		Having proper documentation and feedback process
□		Document information that very related to the main task - code and spec
□		Not all Knowledge/idea not being documented properly
		Having informal document process and action list
		Perceiving knowledge in term of codification knowledge
□		Having standardize documentation process
Working according to their expertise and more to autonomous working	Documentation Process	Not having formal documentation
Practicing informal environment and easy to access		Formal documentation more related to business process and product
Having less structure communication and work process		In development, documentation more on electronic code and other technical things
Admitting the documentation process not well organize		Having e personal documentation for their own used
Using more electronic documentation than paper documentation		Time and people are main reason on this issues
Not having standard documentation procedure		Minimal / simple documentation
Documentation is more informal process and personal documentation / Doing personal documentation for personal own benefit		Using company standard internal tool - teamwork
Using specific tools –xplanner- to document people activities		Not serious in documentation process
Having team work but each responsible their task and delivery time		Formal Documenting only task related to business process and product
No need to document in detail		Having internal documentation but simple and brief
Using specific tools- xplanner – to record people activity and progress		Having more on self documentation rather than standard documentation
Doing personal documentation		
Having light documentation that more toward specification		
Main documentation is programming code compare o others.		
Having formal documentation related to formal procedure and product		
People working across their area	Working Style	People more working on individual basis rather than team
Most of people having same level experience and skills		People work autonomously in small project
Working according to their expertise and more to autonomous working		People are updating and sharing their progress over time and informal
People working support each other		People working more individually most of the time
Applying autonomous work and communication		Explaining small project –people work very autonomously compare to big project
Having less structure communication and work process		Developer work directly with engineer in many areas
Enhanced through OJT, self learning and self explore.		Having direct communication btw developer and mgmt
Having Frequent guidance from high/ expert		Admitting we work more on team

people informally		work environment
General people woks according to their skill and expertise		
Having team work but each responsible their task and delivery time		
Chief also working as one of the developer / Chief work in all area		
Working along with the developer.		
Most of people having same level experience and skills	Mgmt Style	Applying loose PM in managing people
Working according to their expertise and more to autonomous working		Applying short and informal meeting early week before start work
Not having formal meeting		No real gap between people- very flat structure
Generally having no gap between mgmt and team		Applying loose supervision in managing people
Practicing informal environment and easy to access		Having very loose PM due to small team and flat structure.
Having unstructured management style		Having freedom to do the task.
Having loose and macro PM		Applying loose and flexible PM
Having team work but each responsible their task and delivery time		Having no big gap between staff and mgmt
Having less structure project management process		
Working along with the developer.		
Starting share information using blog	Most of sharing activities being done informally and using blog/online (13)	sharing problem experience through informal communication
Creating 'blog' to encourage people sharing opinion and knowledge		Sharing process been done informally
Setting blog to encourage people share information		Sharing been done through informal communication within team and between department
Encourage people to share their knowledge and experience in blog		No 'blog' being use
Being aware the important of sharing all information in blog		Having sharing and communication activities through email, online and phone
Using internal blog share idea		Having team discussion in evaluating changing and evolving process
		Having regular communicate with remote staff online ie skype, email and others
Knowledge being share in blog	Sharing knowledge informally in blog/online (13)	Team always update and communicate using communication tools such as skype, ym with remote staff
Finding interesting topic in blog		Felling comfortable working from home with internet support
Having open discussion in blog with others		Agree that communication tools help enhance the staff relationship and communication
Finding interesting topic in blog		Being aware the efficient of using internet in doing work
Finding blog able share knowledge/ opinion		Having frequent communicate with others through internet and phone
Aware spending amount time to share knowledge through blog		
Information in blog must be align with people interest and needs- could encourage people interest		
team culture exist very informally	Process improvement and assessment being done informal, naturally, personally, immediately, small scale and "try and error" (14)	Being aware small company not having in doing assessment frequently
Doing process assessment informally		Improving any process is aligns with client requirement
Implementing changes immediately		More following customer requirement with client
Being aware of individually decision making		Not doing process assessment frequently
Improvement being done using try and error method		Keep updating development process to avoid problem informally

Being aware that changes/improvement in development process being done in small scale to prevent any major problem		Direct implementation of process change.
Changes being done informally and small scale		Improving product via producing software version overtime
Being aware that all changes and improvement been done naturally		Changes /improvement happen depends on client needs
Changes/improvement being done aligning with the current project and process need but in small scale		Having continuous improvement on software product overtime
Update done by sharing and communicate informally		Continuous improvement happen informally
PM very flexible and informal		Being aware hard to set standard procedure in small company
Being aware that development process is been improve overtime but in a small scale		Believing small company hard to follow the ideal SPI
Using same method that being proof efficient in development		Being aware that SPI activities not suitable to small company
Company a flexible in doing works or assigning task		Constantly producing better module that affordable
team culture exist very informally	Process changes ,improvement and maintenance being done informally, organically and small scale/ Lacking in resource and time constraint (14)	Being aware that only experience people understand company business process
Being aware that people are discussed each other on any changes		Keep updating development process to avoid problem informally
Lacking of resource lead to small changes and informal changes		Direct implementation of process change.
Changes/improvement starting from inside		Improving product via producing software version overtime
Being aware that people are discussed each other on any changes		Changes /improvement happen depends on client needs
Being aware that development process not change/improve frequently		Having continuous improvement on software product overtime
• Suggesting that all team member follow company methodology , improve it in every company project		Continuous improvement happen informally
Using the same process and procedure to avoid conflict in future		Being aware small company not having luxury in doing assessment frequently
Using any guideline/ standard that fit the project objective		Being aware hard to set standard procedure in small company
Being aware current process is workable and effective		Believing small company hard to follow the ideal SPI
Adopting standard in small scale at one time		Being aware that SPI activities not suitable to small company
Time is main constraint in sharing knowledge		Constantly producing better module that affordable
		Process changes and evolution in software product happening so fast
		Having informal document process and action list
Being aware that people are discussed each other on any changes	knowledge sharing being done informally and internally/ Having sharing electronically and online (13)	Adopting 'train a trainer' concept
Having post mortem for every project		Sending staff training only when required
Having post mortem to improve development process based on customer requirement		Using communication tools communicate with remote staff
Doing post mortem internally and not with client		Putting all data in one place- centralize system
Any improvement and changes being discuss among team member		People having frequent communicate each others
People learning through their experience and other experience		Having details post mortem with client informally
PM very flexible and informal		Development team involve in decision making
Sharing is not on going basis activities		Knowledge sharing by having informal communication
Having open environment such as open to idea and open communication between people		Having frequent communicate with others through internet and phone
Using internal blog share idea		New staff and experience staff are communication effectively

Employee idea being accepted in small company		Helping each other in some task
Time is main constraint in sharing knowledge		Having informal knowledge sharing process
		Having casual and informal knowledge sharing with other staff
Not following any guideline or standard in improve/ change development process	Improvement or changes being done organically and not following standard (14)	Being aware hard to set standard procedure in small company
changes/improvement done base on situation		Development model been use depends on the project
Using internal blog share idea		Direct implementation of process change
		Having Least resource, time and man power made small company cant follow formal SPI
		Not having standard development process
		Following own development style in improve process
		Company not following a good coding process
Update done by sharing and communicate informally	Documentation Process being done informally and not frequently due to team size and less priority/ Self documentation and no standard documentation process (12)	Documentation is not priority in small company
Any update process not being document often		Documentation system is effective due to small no of people
Team size influence documentation		Changes or improvement been store in different version
Being aware having a small size of people , company need to focus on others rather than documentation		Current company not sustain/follow good documentation process
Documentation process is not a priority		Company having a very informal communication and relationship
Being confirm that all employee do not have to document their activities		Having informal document process and action list
Keeping all list of task in personal diary		Explaining document process is there but not been share with others
Having very simple organization structure, informal work schedule and simple reporting system		No clear documentation process been practice
Having very simple organization structure, informal work schedule and simple reporting system		Direct implementation of process change
Being aware that all improvement process not being documented		
Not all development activities being document		
Being interested in Software quality standard	Having plan to adopt standard/ (15)	ISO requiring /incurred high cost, heavy process, heavy documentation and time
Recently aware of the important of software quality standard		Preferring standard that align with current process
Looking for a lightweight software quality process		Standard that have template
Planning to considering standard more important when having more people in company		standard that could provide guideline to interact with customer
Planning to obtain the well recognize in order to enhance their company image and standard		standard that easy to follow
		Standard that not incurred a lot of overhead
		Standard that related to existing implemented standard
		Documentation process not following International standard process like ISO
		company have planned to adopt ISO before
Being aware that standard will lead to a quality product	Having standard will lead to development process consistency and success (15)	Being aware that having standard is important to making work more effective
Being aware of software standard benefit		Being aware standard giving consistency to development process
Being aware that standard will create consistency		being aware about the advantage having an engineering standard
Being aware that number of employee will	Critical employee size influence	Size giving impact in adopting

influencing the adapted software standard	company adopting standard (15)	standard
Requiring specific people to responsible and maintain the standard		Being aware ISO required time afford
Involving more cost and resources		
Being interest and prefer ISO9000	Plan to adopt ISO9000 (15)	• Being aware ISO not compulsory in software business
Starting looking more serious about standard		Being aware ISO not important to small company
Company having a more light and informal development process	Perceiving standard are heavyweight and complicated process that not suitable to small company (15)	ISO requiring /incurred high cost, heavy process, heavy documentation and time
Being aware that company are not interest in software standard		Documentation process not following International standard process like ISO
Company size and area influencing company in adopting software quality standard		Being ware ISO and CMMI not suitable for small company
being perceive that a lot of changes in current process need to done in adopting standard		Size giving impact in adopting standard
Being aware that small company is hard to meet and follow standard in detail.		ISO involved heavy weight process for small company
Company need to ready before implementing Quality standard		Complicated procedure in tendering system create problem to small company
Being aware that number of employee will influencing the adapted software standard		Standard having large documentation
Perceiving Standard as a 'sale tool'	Perceiving standard as "sales tool' / image/ required a lot of resources and just for public sector work. (15)	Being convinced that standard not important to company
Perceiving standard is just an image to the company		Not having problem without ISO in doing business
Adopting standard required huge resource and time		No problem without having standard so far
Aim to meet standard is important to satisfied customer and "image" builder		Perceiving that Irish company not interested adopting standard
Presenting what client need is very important not standard		Standard as a sale tool, and image
Having basic list that cover all company requirements.	Looking for new standard that related to the companies capabilities, lightweight process , easy to administrator, have a template and related to existing standard (15)	ISO requiring /incurred high cost, heavy process, heavy documentation and time
Company need to ready before implementing Quality standard		Preferring standard that align with current process
Must having light weight process easy to administrate		Standard that have template
Perceiving adopting maintaining standard will increase administration problem.		standard that could provide guideline to interact with customer
being perceive that a lot of changes in current process need to done in adopting standard		standard that easy to follow
Following company style in adopting standard		Standard that not incurred a lot of overhead
Looking for easy, flexible and not consume a lot of time		Standard that related to existing implemented standard
Looking for standard that align with current development process		Having light weight process
		Provide workshop
		standard align with current business process
		Standard having clear guideline
		Standard having clear template
		Need less documentation process and not time consuming
		Standard that easy to follow
Perceiving only one staff are aware most about software standard	Having aware people are less knowledgeable/ interested in software standard (15)	More interested in technical standard
Being unsure either the staff are aware and knowledgeable about software quality standard		Arguing that technical standard is more related to company business
Planning to obtain the well recognize in order to enhance their company image and standard		Manager less interested in software standard
Management not knowledgeable and less interested about software quality standard		Staff less interested in software quality standard
Staff less knowledgeable and interested in software quality standard		

More interested in technical standard rather than quality standard	Prefer Standard that related to technical standard (15)	more preferring technical standard
Marcus: Staff not knowledgeable about software quality standard more understand about technical standard		More interested in technical standard
More prefer technical software standard rather than quality standard		Arguing that technical standard is more related to company business
Not following other/outside standard guideline	Software Quality Standard	Having too detail documentation
No plan to apply any standard		Required extra man power
Focusing on delivery time then other process		Not important to developer
Having Less resources and time		No plan to adopt standard currently
Being aware that ISO involved repeatability process rather than quality		Admitting and realizing that quality standard make company more organize and improve work process.
Following agile type development		Aware that quality standard too much detail and not related with company current process
Being aware higher code quality is more important than process		Aware ISO good for business but current ISO not suitable to small company
Less resource		Doing some background study but no plan to adopt now
Not require by customer		
ISO more toward services rather than software product		
Perceiving that not much knowledge been created	Creating same knowledge all time (10)	Perceiving that similar knowledge been created
Company using same technology and process in every project		Using the same technology and development process all the time
Perceiving the main knowledge in company are similar		Following what process and technology is suit to company and been proof efficient development process only
• The management are trusting people in working their work	Trust element important in company (16)	
Being aware that people will leave the company with their knowledge	Knowledge Loss Problem not important issues (17)	Having not serious staffs' turns over.
Not worry about the loss of knowledge due to people leaving		Being aware of having documentation to present knowledge loss
Being aware loss of knowledge cannot be stop	Doing repetitive work, continuous documentation, continuous self learning and sharing prevent knowledge Loss / Starting organically (17)	Putting all data in one place-centralize system
Retaining knowledge through repetitive work and continuous learning		being aware having standard coding process help to retain knowledge
Believing that to retain knowledge must start with inside		Using the same technology all the time
Perceiving blog could secure people knowledge		Keep updating development process to avoid problem informally
People working across their area	Knowledge Preserve Issue	Relying on the programming code and other technical stuff in the database
People working support each other		
No having formal plan to preserver loss knowledge		
More rely on coding and any related document		
Having Frequent guidance from high/ expert people informally		
Ensure no one exclusively work in one project - Hard to apply due to size		
being aware that time is the most constrain in documenting people knowledge	Time is the main constrains in documentation process (12)	Being aware small company not having luxury in doing assessment frequently
time is a constraints in sharing tacit knowledge		Changes or improvement been store in different version
Being aware that time and effort is a main constraint to implement all the suggestion		Being aware ISO required time afford
		Time is the main constraint having all good process
Having flat development team structure	Having a flat and flexible/ no formal organization structure and	Being aware flexible flat working structure make people work

	team process/ Having direct, close working space, direct communication (3)	effectively
Having very small development team | No formal structure		Company having a flexible working structure
Not having a formal define job role		Having small number of employee in IT department
Treating people are in the same level		Counting manager as software development
Management involving directly with development		Having flat working structure
Being aware new staff not work individually		2 people working in the office
Being aware there no gap relation between team and management		People having frequent communicate each others
Admitting that the team structure is flat and everybody have the same right		Having family environment and flexible encourage people to stay
Having very simple organization structure, informal work schedule and simple reporting system		Development team involve in decision making
team culture exist very informally		Having a flat management and working structure
		Company having flat work structure
		Direct transmitting information from one to another
Not applying a micro project management process	Understanding everyone have similar right and informal PM (16)	Not forcing people to follow development procedure
Everybody having similar right and task		Having small number of employee in IT department
Having a loose project management		Having flat working structure
Not having a formal division between management and staff		Having family environment and flexible encourage people to stay
Management involving directly with development		Having a flat management and working structure
Being aware there no gap relation between team and management		Company having an open environment , everybody are free to suggest and talk
Admitting that the team structure is flat and everybody have the same right		
Not having formal project post mortem.		
Management reviewing people job over time		
Perceiving team work, team dynamic are important in knowledge sharing	Being aware that knowledge sharing need team dynamic and good team work (18)	Adopting 'train a trainer' concept
Assuming Knowledge sharing more effective if team doing the same project		Louie :People working and helping each other very close
		Louie : Felling no different working with remote staff with support from communication tool
		Sending staff training only when required
Being aware knowledge played big role in software development process	Admitting knowledge management and team important in enhancing current development process (18)	Believing the main asset of the company is experience and knowledge
Admitting sharing knowledge could carry more knowledge		Being agree that knowledge and experience giving benefit
Being aware good documentation help sustain knowledge		Being aware working in team could enhance knowledge and enhance team relationship
		Being aware new knowledge can improve existing software process
		Knowledge and team could having a standard process and grow the team
		Being aware that working alone cannot enhance knowledge and always follows old development style
		being aware having standard coding process help to retain knowledge
		Being aware having good documentation help in sustaining knowledge
		Having clear development process and procedure help to retaining knowledge
		Having a good repository place help in sustaining knowledge

		Team could enhance new idea in developing system
Adopting prototype method encourage interaction and feedback from client	Having good communication channel and process (19)	Being aware clear communication in team is important
Senior staff given frequent guide to new employee		Company having a clear communication ,goal and understanding
		Being aware direct communication related to fast feedback
		People working and helping each other very close
		Having team discussion in evaluating changing and evolving process
		Being aware that existing staff clear about business process
		Having effective feedback from other department too
Alerting that all people are aware happen in the company	Company having fast and effective communication/ team dynamics and cohesive in development team (19)	Having small number of employee in IT department
Not having a formal division between management and staff		Adopting 'train a trainer' concept
Management involving directly with development		Having different open communication level among employee.
Being aware people are casually communication each other openly and could tolerate in their work		People having frequent communicate each others
Perceiving staff relationship are very close and professional		Having team discussion in evaluating changing and evolving process
Senior staff given frequent guide to new employee		Development team involve in decision making
Staff having cross over interaction only for special purposes		Team are supporting each other within the organization
Working autonomous encourage more interaction between team member		All roles in team are clearly been define
Working space between people are very close		All team member working closely each other
		People having a good relationship
		Felling no different working with remote staff with support from communication tool
		Being aware team dynamic exist in the company due to small size and flat structure
		Having and excellent working environment
		People working and helping each other very close
		Direct transmitting information from one to another
		Management also working as a developer
		Management having direct involvement in development process and planning
		Company having an open environment , everybody are free to suggest and talk
Being ware that people are supporting and helping each other	Admitting collocate staff having good team work and dynamic (19)	Having different open communication level among employee.
Being aware people are actively communicate each other if they are working in the same project		Having difficulties in communication and observation with the remote staff
Having a great time together sometime		People having frequent communicate each others
Team member helping others in understanding new tools and process		Having family environment and flexible encourage people to stay
Socializing with other staff sometimes		Having team discussion in evaluating changing and evolving process
Having frequently conversation with others		Development team involve in decision making

Realizing that communication tools increase team relationship and team work		Team are fast learner and understand requirement
Company having active interaction among team and between team and customer		People are working each other in doing job/project
Having an open discussions in improving software process		All team member working closely each other
Staff having cross over interaction only for special purposes		Planning to focus on local suitable people
Having people opinion in improving current development process		Having a well balance development team
Helping each other in some task		Having direct contact and communication with team and management
Having flat development team structure		• Having frequent communication with other team member
Everybody having similar right and task		People working together in family environment
Having a loose project management		Company having an open environment , everybody are free to suggest and talk
PM very flexible and informal	Management having informal PM (16)	Following instruction from the superiors in any changes/improvement
		Having an open communication between management and staff
Not having a formal define job role		Having different open communication level among employee.
Management are very alert in finding error		Planning to focus on local suitable people
team culture exist very informally		Having direct contact and communication with team and management
Casually management share their opinion		Having very informal and close relationship
Management are very alert in finding error		Having frequent communication with other team member
Company having an open environment , everybody are free to suggest and talk		Management also working as a developer
Being aware people are casually communication each other openly and could tolerate in their work		Company having an open environment , everybody are free to suggest and talk
Perceiving staff relationship are very close and professional		Company having a very informal communication and relationship
		Manager having freedom to make a decision
		Being aware that most of employee idea being adopted indirectly
		Management having direct involvement in development process and planning
		Documentation process is depends on individual/ Not forcing people to follow development procedure
		Company not following a good coding process
Using communication tools daily e.g. skype, vpn	Frequently using communication tools in sharing and communicate each other (6)	Realizing online communication is not effective in software development
Using communication tools and software in synchronizing team work		Utilizing communication tools in remote communication
Remote staff not feeling Isolated with support from communications tool		Using communication tool to interact
Most staff communicate with customer through email		People having a good relationship
Being aware using communication tool increase teamwork effectiveness		Using email and skype exchange idea and opinion
Communication tool closing gap between remote staff and collocate staff		People working and helping each other very close
		Felling no different working with

		remote staff with support from communication tool
		Agree that communication tools help enhance the staff relationship and communication
		M: Being aware the efficient of using internet in doing work
		meeting usually using skype conference
		Having team discussion in evaluating changing and evolving process
working with remote staff not encourage team interaction and team work	Working with remote staff not encourage team dynamic or having a clear team notion (19)	Realizing online communication is not effective in software development
Perceiving team work and team dynamic is not exist if working with someone remotely		Having difficulties in communication and observation with the remote staff
Being perceive working remotely not impact team process and structured		Having difficulty with un control remote staff
Team are not really cohesive and dynamic due to location and team work		Planning to focus on local suitable people
Working space between people are very close		Admitting having challenge working with remote staff
		Louie : Being aware that working with remote staff change the team dynamic
		Team dynamic element not really exist due to location
		Documentation process is depends on individual
		Being aware working distributed impact on team dynamics
Having open environment such as open to idea and open communication between people	Development team having open and direct communication and relationship (6)	Having an open communication between management and staff
Communicate frequently with others.		Being aware flexible flat working structure make people work effectively
People helping each other		Having open discussion prior to change or improve
Having very open communication among team member		Having different open communication level among employee.
Senior staff given frequent guide to new employee		Utilizing communication tools in remote communication
Team member helping others in understanding new tools and process		Having family environment and flexible encourage people to stay
Team member are willing to accept other people opinion and suggestion		Having team discussion in evaluating changing and evolving process
Company having active interaction among team and between team and customer		Development team involve in decision making
Working autonomous encourage more interaction between team member		Planning to focus on local suitable people
Having an open discussions in improving software process		People having a good relationship
Having people opinion in improving current development process		Having very informal and close relationship
Employee idea being accepted in small company		Having a flat management and working structure
Staff having cross over interaction only for special purposes		Having direct contact and communication with team and management
		People working and helping each other very close
		Felling no different working with remote staff with support from communication tool
		Having team discussion in evaluating changing and evolving process
		Having frequent communicate with others through internet and

		phone
Admitting that formal meeting not usually happened	Having informal and casual meeting in development process (6)	Having different open communication level among employee.
meeting are more informally and casually		Utilizing communication tools in remote communication
Being aware that big company have very organize way of meeting compare to small company		People having frequent communicate each others
Being agree small company with small team are very close relationship and effective team work		Having a flat management and working structure
Being perceived that small company having more informal and freedom in all aspect		Having direct contact and communication with team and management
		People working and helping each other very close
		Meeting been conducted informally in the company
		Not having schedule meeting and informal meeting
		Having frequent communication with other team member
Being aware that adapting latest technology from time to time is important in order to sustain company competitiveness	Adapting new technology and updated and evolving dev process are important in sustain business competitiveness (20)	
Being are that company need to adapt latest technology overtime		
Being aware that the improving process giving a lot of benefit to the development project		
Being aware that having a reliable software is important		
Evolving the development process from time to time		
Being aware that the improving process giving a lot of benefit to the development project		
Being aware that company changed the development method in order to relate with the customer		
being aware that project success is depends on the right estimation	Having realistic estimation are important in sustain product/process Competitiveness (20)	
Having realistic in making decision		
Gavin : Being aware that realistic estimation and planning should be done especially in small company		
Understanding of company size in develop the system is important		
Having different process bound for different size of company		
Interaction between client and development is only happened in requirement gathering phase	Having good relationship with client / following client needs are important in sustain company competitiveness (20)	Having details post mortem with client
Prototype helping in get getting customer requirement quicker		Improving any process is aligns with client requirement
		Constantly producing better module that affordable
Using prototype method in order to enhance interaction with client		• More following customer requirement with client
Having a good relationship with customer		Being competitive by sustain customer relationship
Presenting or producing what customer really want only		Company having an open environment , everybody are free to suggest and talk
Perceiving interaction with customer is very important		People working and helping each other very close
Presenting what client need is very important not standard		
Being aware that SPI have indirectly and very informal	Software process being frequently improved/changed informally and indirectly in a small scale (14)	Having open discussion prior to change or improve
Being aware that improve happened indirectly and informally		Being perceive that improvement process important to save time and resources

Keeps continue using a basic (old) development process for every project every		Constantly producing better module that affordable
The improve development process not being implemented in all project		Being aware small company have least resource to follow SPI
Realizing the improvement process is happened but cannot determine the improvement		Having Least resource, time and man power made small company cant follow formal SPI
		Improve process in a small scale
		Having team discussion in evaluating changing and evolving process
		Being aware that most of employee idea being adopted indirectly
Informing company basically always kept the updated database as in sourcesafe	Being aware knowledge sharing either electronically or verbally important in maintaining software dev success. (18)	Believing the main asset of the company is experience and knowledge
Sourcesafe generating automated updated database		Company having a clear goal and proper planning
		Louie : Felling no different working with remote staff with support from communication tool
Claiming that consistency in programming code is important in saving knowledge		Being aware clear communication in team is important
		Company having a clear communication ,goal and understanding

Appendix F
Axial Coding (Abstraction Process)

Sub Category	Category	Main Category
IT Experience - Manager	People Skill and Experience (1)	
IT Experience - Team		
Expert		
IT related qualification/degree		
High Experience/ Skill full		
IT Background	Background of founder (1)	
Non IT Background		
Proactive	People Criteria and Qualification (1)	
High Interest and creative		
IT related qualification/degree		
Average Abilities		
Adaptable stress		
Family Business/ Company / Owner - Brothers	Company Nature (2)	
Non Family based - Partner		
Years Established	Establish and located at specific places (2)	
Small Companies/department		
Small office space		
Remote development team/ GSD operation	Development operation (2)	
Collocate development team		
Remote Office		
Dublin Main Office		
Frequent Visit		
Team Size - Small	Flat and Flexible Organizational Structure (3)	
Organizational and Team Flat Structure -		
Autonomous Work		
Team Involvement		
Work Structure - Flexible		
Team Role/ Job Role	Informal Team process and structure (3)	
Team Involvement		
Team Culture		
Autonomous work		
Close gap/ Work space		
Autonomous work		
Autonomous Communication		

Work independently	Autonomous work and communication (6)	
Strategic area		
Sole Responsibility		
Self Learning		
‘Team of One’		
Generic Documentation Process		
Generic Development Standard		
Isolated Project		
Training	Practicing Self Learning Culture (4)	
Self Learning		
Continuous Guidance		
Knowledge Sharing		
Centralize System		
Generic Documentation Process		
Internal Training	Practicing Team Sharing Culture (4)	
Meeting		
Online Communication		
Electronically		
Centralize Data		
Meeting	Practicing informally, casual open , online, direct and periodic communication and sharing (5)	
Communication		
Blog		
Post Mortem		
Open Environment		
Autonomous Communication		
Communication Tools		
Generic Documentation Process		
Family and Flexible Environment		
Open Exchange Information		
Communication tools	Main communication and sharing tool (5)	
Internet		
Electronically		
Economy Situation	Minimum case staff turn over (8)	
Same Level		
Family and Flexible Environment		
Macro Project Managemen/t Practice informal PM (16)		

Autonomous work and communication (6)		
Trust	Macro Project Management	
Family and Flexible Environment		
Loose PM		
Open Environment		
Autonomous work and communication	Autonomous PM	
Macro Project Management		
Informal documentation	Informal documentation tacit knowledge process (12/13)	
Individual Documentation		
Technical Business related document	Specific Information (12)	
Client related Document		
Repetitive work	Knowledge Loss Problem issues (17)	
Continuous learning		
Family relationship		
Electronic communication		
Standard Process and Technology		
Strategic area	Autonomous work and communication (6)	
Autonomous Communication		
Communication Tools		
Team Involvement		
Family and Flexible Environment		
Frequent Knowledge sharing	Collocate - good team work and dynamic (19)	
Close working Space		
Socializing		
Frequent Cross interaction		
Casual Communication		
Team Process Difficulties	GSD Less team dynamic/ team notion (19)	
Communication Difficulties		
Control Problem		
Local Focus		
Standard Technology	Applied standard technology and various development method (10)	
Non Standard development Process		
Agile Development Style	'Agile' development	

Interactive		
Informal/Indirect	Process Improvement, maintenance and evolution status. (14)	
Small Scale		
Process Loss and Focus		
Reason		
Not Enforcing Procedure	Process Loss Problem symptom (17)	
Less Project and Process Documentation		
Using same development process		
‘Try and Error’ concept’ (Not Following Standard guideline)		
Autonomous Work and communication		
Agile’ Documentation Process (Knowledge Management)		
Reactive and Proactive		
Company History	Interest Level - - standard (15)	
High awareness adopting Standard		
Critical Factors		
Standard Guideline	New Standard criteria (15)	
Light weight Process		
Flexibility		
Business and Process Align		
Technical Standard align		
Assistance/Guideline		
Less Knowledge	Level of Acceptance (15)	
Less Interest		
Less Priority		
Being aware knowledge played big role in software development process	KM and Team relationship (18)	
Admitting sharing knowledge could carry more knowledge		
Being aware good documentation help sustain knowledge		
Having team relation could relate new knowledge		
Knowledge management and good team - -improve standard process and Knowledge management and good team - -enhance teamwork		
experience and knowledge -- main asset of the company		
Knowledge and Experience - giving benefit development process		
Team work - enhance knowledge and work - enhance team		

relationship		
New knowledge - can improve existing software process		
Knowledge management and good team - -improve standard process and enhance teamwork		
working alone -cannot enhance knowledge and always follows old development style		
standard coding process -help to retain knowledge		
good documentation -help in sustaining knowledge		
clear development process and procedure help - retaining knowledge		
good repository place -help in sustaining knowledge		
Good development team - enhance new idea in developing system		
Team work/Team Dynamic	Enhanced knowledge sharing (18)	
Effective team		
Communication tools		
Internal training		
'train a trainer'		
Close relationship		
Latest Technology/ Reliable Software	Critical Factor - sustain business , product and business competitiveness (20)	
Improve development process		
Evolving development process		
Right estimation, decision, team size and interaction		
Following Customer Requirement		
Process Bound		
Right Development Method		
Open Environment		
Sustain Customer relationship		
Better Module and Version		
Frequent Knowledge sharing	Team dynamics/ Team work (19)	
Close working Space		
Socializing		
Frequent Cross interaction		
Casual Communication		
Team Process Difficulties		
Communication Difficulties		
Control Problem		
Local Focus		

Appendix F
Axial Coding

VSE Staff Background

Code (Company B/ Company AF)	Group Code (Category)	Code (Company C/Company AE)
IT experience – Manager/CEO	Experience i IT/SW Dev People (1)	IT qualification - Manager
Related IT qualification - Manager		IT experience - Manager
Substantial IT Experience -Team		Highly Knowledgeable in Development - Manager
IT qualification/degree - Team		Hardware
Related IT qualification/degree - Team		Development - Bespoke
Substantial IT experience - Team		Development- Customization
Different IT environment - Team		IT experience - Team
Development- Bespoke		Software Development- Platform
Development- Internally		IT qualification -Team
Highly Knowledgeable in Development		Related IT qualification - Manager
System Platform Development		Highly experience - team
[redacted]		[redacted]
Role as Senior software development	Experience IT/SW People	Role as founder and director of the company
Having quiet substantial skill and working experience in software development		Having It background and qualification
Having IT related degree and qualification		Founder of the company Existed almost 10 years
Experience working with several big and small company in UK and Ireland		Having IT qualification , background and experience
Current role as CTO		

IT Background	Background of founder (1)	Non IT background
IT Knowledgeable/Qualification		Selling Hardware
Having IT related degree and qualification		Having IT qualification , background and experience
Experience and knowledgeable in many IT platform and projects		

IT Related qualification - Hiring	Hiring IT related background and qualification staff (1)	Proactive
Average abilities- Hiring		Computer Related qualification- Hiring Substantial work experience
High Interest- Hiring		Hard work- Hiring
Substantial work experience		Adaptable Stress- Hiring
		Thinking Ahead

[redacted]	[redacted]	[redacted]
Hiring people – having good degree qualification	Hiring Criteria	Hiring criteria more on experience, skills, passion and level of motivation
Having knowledge in specific platform		Least depends on academic qualification
Having creative and problem solving thinking		Looking for passion, skill and high motivated people
		Not very particular on qualification

VSE Business Operation

Family Company	Company Nature (2)	Family business
Owner -Brothers It is non family business and 2 years joining his company		It is non family business and exist almost 10 years

Years Establish -5 year	Establish and located at specific places (2)	Years Establish -15 years
Small Office Space		Office Space/ Small Department
All technical people located in main office		Located in new office in Cork
Mgmt in Dublin		Admitting people not understand company goal

Main Office - Dublin	Development Activities (2)	Main Office - Dublin
Remote Development Team (spain/South Africa)		Remote Development Team – (Poland/Lithuania)
Collocate Development Team		Collocate Development Team –
Frequent Visit- Remote Having GSD operation		Frequent Visit- Remote Having GSD operation

Team Structure and Process

Team Size - Small		Team Structure – Flat

Team Structure – Flat	Organization and Team Structure (3)	Team Size - Small
Team Structure - Not Formal		Small team Involvement
Organization Structure _ simple and Flat		Small team set up - similar
Work schedule – Informal		Work Structure- Flexible
Reporting system – Simple no standard		Working environment- “family environment”

Team Role – Informal	Team process (3)	Team involvement - direct
Job Role - Informal		Team Role – Informal
Team Culture - informally		Team Culture - informally
Team involvement - direct		Software development - direct involvement
Team work - individually		Informal /direct communication
Team work – close gap management and team		High communication frequency
Team structure – same level. Same right		Work space close
Work space - close		Casual communication

Having small number of staff in this department	Team Size and Structure	Having small number of technical staff
Having flat structure		Having no formal team structure
People working across their area		Having very flat organization structure
Having small development team		No real gap between people- very flat structure
Having flat team and organization structure		Having small number of technical staff -Small company
Admitting no real team structure		Having a flat and informal organization structure
Having flat hierarchy		Having a flat and informal organization structure
Having close relationship and easy way		
Having small team/ technical team		
People being cut due to economy situation		
Having flat team structure		

Having good working environment	Team environment	Mgmt and staff having very close relationship
Having flexible and close relationship		Realizing age, interest, skills and experience create the environment
People expose to the new technology		No real gap between people- very flat structure
Having range of expertise		Being family and flexible environment
Having team work but each responsible their task and delivery time		Admitting we work more on team work environment
People having close relationship and easy to share idea		Having no big gap between staff and mgmt
Reason- good relation and excellent working environment		Admitting people work very close and sometime they socialize together
Working along with the developer.		This is due to the small team and easy to communicate

remote staff not encourage- team interaction and team work	GSD Less team dynamic/ team notion (19)	Online communication- not effective in software development
team work and team dynamic - not exist if working with someone remotely		Remote staff - difficulties communication and observation
Working remotely - not impact team process and structured		Control and observation problem
Team cohesive and dynamic - location and team work		Focus locally
Working space		Change team dynamic
		Location
		Being aware working distributed impact on team dynamics

Support	Collocate - good team work and dynamic (19)	open communication
active communicate		Clear communication and observation
Socializing family environment		frequent communicate
Team work		family environment flexible environment
Socializing with other staff sometimes		Team work
active interaction		Direct team involve
open discussions		Fast learner
cross over interaction		Small team
Small team		Socializing with other staff sometimes
Working space		Working space

High awareness- company		Small number of staff

No formal division- management team	Company having fast and effective communication/ team dynamics and cohesive in development team (19)	Open communication
Direct involvement		Frequent interaction
Open Communication Casual Communication Teamwork		Team discussion
Good staff relationship		Direct involvement Autonomous decision making
Frequent guidance		Close working space
Frequent Cross interaction		Good team support Teamwork
Woking autonomous Frequent sharing Communication tools		Team work Frequent sharing Communication tools
Working Area Working space		Socializing Family environment
Small No of people		Flat structure
Socializing Family environment		Open environment Direct transmit information

Management Style

Trust	Macro Project Management (16)	
Macro project management		Autonomous work
Same right and task		Flat working structure
Loose project management		Family and flexible PM
No formal division		Open PM
Direct management		Direct Involvement
No gap - relationship		Open PM - communication
Flat structure		Casual PM - communication
Post mortem- in formal and internal		Focus on local
Informal reviews		Direct contact and communication
Flexible and informal PM		Following instruction from the superiors in any changes/improvement
Job role – not explicit		Autonomous coding process
Management are very alert in finding error		Direct Involvement
Team culture- no explicit		Freedom
Casual PM - communication		Idea
Open PM - communication		No standard documentation
Close and professional relationship		Post mortem- in formal and internal

Most of people having same level experience and skills	Mgmt Style	Applying loose PM in managing people
Working according to their expertise and more to autonomous working		Applying short and informal meeting early week before start work
Not having formal meeting		No real gap between people- very flat structure
Generally having no gap between mgmt and team		Applying loose supervision in managing people
Practicing informal environment and easy to access		Having very loose PM due to small team and flat structure.
Having unstructured management style		Having freedom to do the task.
Having loose and macro PM		Applying loose and flexible PM
Having team work but each responsible their task and delivery time		Having no big gap between staff and mgmt
Having less structure project management process		
Working along with the developer.		

Standard Technology and similar development Process

Similar development process	Applied same development technology and standard development process (8)	Similar technology and process –(not impact process and procedure change)
Similar working development method		Similar network set up
Experience and workable development process		Low risk taking – adopted new process
Low risk taking – adopted new process		Standard technology and development process
Similar technology/ similar process		Similar development method – proof efficient/ avoid conflict
Similar technology set up		Standard - various
Similar development method – proof efficient/ avoid conflict		
Standard - various		

Following agile development process and practice	SPIME	Using same development tools
Using similar development process in all project		Using similar development process in all product
		Using standard development tools
		Realizing that process always being improve but indirect and informal ways

Working Style

Perceiving everybody knows everybody in the company	Autonomous work/and communication (6)	Work Structure- flexible
Autonomous work		Work structure- flat
Working independently		Development Process – no standard
Strategic area.		Work freedom
Team notion - informal		Direct involvement
Team notion – team size		Self control
Self independent		team structure- No formal
Team culture Team culture - informal		Development project - different
Own self review		Staff doing different project at one time
Autonomy		Team size- no standard
Different project Self responsibility		Team size -depend to the project
Self learning and explore		Team communication
Work freedom		Development process – suit company environment
Work flexible		Development process - individually
Work autonomous – Autonomous Communication		Team Notion - informal
No Forcing/ no work standard		Team environment - informal
"Team of one"		Team roles – not defined/ not clear/no standard
		Team structure – periodic basis
		Work remotely

People working across their area	Working Style	People more working on individual basis rather than team
Most of people having same level experience and skills		People work autonomously in small project
Working according to their expertise and more to autonomous working		People are updating and sharing their progress over time and informal
People working support each other		People working more individually most of the time
Applying autonomous work and communication		Explaining small project –people work very autonomously compare to big project
Having less structure communication and work process		Developer work directly with engineer in many areas
Enhanced through OJT, self learning and self explore.		Having direct communication btw developer and mgmt
Having Frequent guidance from high/ expert people informally		Admitting we work more on team work environment
General people woks according to their skill and expertise		
Having team work but each responsible their task and delivery time		
Chief also working as one of the developer / Chief work in all area		
Working along with the developer.		

KM Process

- **Communication Process**

Meeting Meeting- Informal Meeting- On line	Practicing informally, casual open , online, direct and periodic communication and sharing (5)	Meeting – casual Meeting – periodic Meeting- On line
Meeting – casual Meeting – periodic		Remote staff
Communication - free Communication – Open		Centralize data
Learning from experience		Having difficulties in communication and

		observation with the remote staff
Sharing – casual Sharing -communication Sharing -project code		Communication – online Communication – Frequent Communication Channel
Blog Blog – Sharing Blog – Knowledge Blog - internally		Open Discussion
Post mortem Post Mortem - informal Post Mortem - internal		Skype conference
Open to idea Open - discussion		Process Change - team Process Improve - team
Response – Periodic Cross over interaction – periodic Autonomous Communication – assistance/direction/ambiguity		Exchange knowledge - informal
Informal environment Freedom environment		Documentation Documentation – no standard Documentation- Not shared
Communicate - email Communicate – online Communication Channel		Family environment Flexible environment
Sourcesafe Database Automated Electronically		Direct Involvement Direct Contact Sourcesafe Database Electronically
Visit – casual Autonomous work		Visit – casual Autonomous work
Development process Accept and exchange idea		Development process Accept and exchange idea
New tools New Process		

People working across their area	Communication Process	People are updating and sharing their progress over time and informal
Working according to their expertise and more to autonomous working		Meeting more informal and casual
People working support each other		Company having open and casual communication – open to discuss and express idea
Practicing informal and casual communication		Having direct communication btw developer and mgmt
Having stand up meeting or short meeting normally		Not having proper meeting setup and more on standing/informal way
Following agile development practice		Discuss only the important and plan
Not having formal meeting		Practicing direct / face to face communication in all activities
Having open communication		
Having less structure communication and work process		
Customer - Feed back given by email or phone		
Customer - Feedback given directly to designer		
Enhanced through OJT, self learning and self explore.		
Having Frequent guidance from high/ expert people informally		
Practicing more face to face communication		
Having more informal, stand-up meting on daily basis		
Admitting people are free to give idea and suggestion		

Communication Tools Skype, VPN, email	Main communication and sharing tool (5)	Online Communication – Not effective
Synchronize work		Remote Communication – online tool
Team effectiveness		email , skype exchange
Closing Gap/Remote Staff/ Collocate staff		People having a good relationship
		Using idea and opinion
		Communication Tool
		Online communication – enhance relationship

		Internet Internet efficient work
		Process change Process evolving Process change and Process evolving – small scale and team discussion

Using more electronic documentation than paper documentation	Communication Tools	Most of the time using specific internal communication tools to interact – *teamwork-*
Using specific tools –xplanner- to document people activities		Sometime people using other tool to communicate personally
Using phone and other available tool communicate with management		Using company standard internal tool - teamwork
Casually using other communication tools when necessary		Using tools as needed but not frequent
Using development tool to communicate.		
Using specific tools- xplanner – to record people activity and progress		
Getting feedback through email, phoned or direct meting		

- **Learning and sharing**

Training - informal	Self learning and self enhancing (4)	Centralize System
Seminar/ Core technology		“Train a Trainer” concept
Informal training purpose.		Sending staff training only when required
Knowledge Sharing – frequent/informal		seminar / core technology / Company development process
Self Enhancing – others skill		Training - internally
Self learning- development process		Knowledge exchange/ Knowledge sharing - informally
Guidance - informal and continuous		Self learning
Team work – sharing periodic (autonomous communication)		Self exploring and self enhancing
Self Exploring- New tools/new process		Guidance - frequent
		Knowledge sharing -informal

People working across their area	Learning and Sharing Process	People did self learning and sharing in mastering their job
Working according to their expertise and more to autonomous working		People are updating and sharing their progress over time and informal
People working support each other		Sharing code in order to get feedback and idea
Applying autonomous work and communication		Company having open and casual communication – open to discuss and express idea
Having open communication		Having informal sharing through casual communication and meeting/Not documented
Practicing informal environment and easy to access		People doing self learning and sharing among people
In general people are willing to share because of small team		Having an online sharing and through casual and informal discussion
Mgmt always supply related information		Encouraging people to talk to right person on specific issue all time
Sharing also been done through peer to peer programming		Sharing through on line
No documentation sharing or no formal sit down discussion		
Everybody can accept and express idea.		
More discussion and jot it down on white board		
Enhanced through OJT, self learning and self explore.		
Having Frequent guidance from high/ expert people informally		
People having close relationship and easy to share idea		
Admitting people are free to give idea and suggestion		
Sharing been done informally by shared books , direct ask question and discussion.		
Doing pair programming occasionally		
By giving frequent question and idea		
Ensure no one exclusively work in one project		

Hard to apply due to size		

Working according to their expertise and more to autonomous working	Enhancing Skill	No formal training is given
Applying autonomous work and communication		People did self learning and sharing in mastering their job
Training is more on self learning and sharing		People are updating and sharing their progress over time and informal
Training more on hands-on or OJT		Admitting no formal training given to staff
Learning individually from internet/ sharing with people and expert		Giving in house. Internal training – more informal way
Enhanced through OJT, self learning and self explore.		People doing self learning and sharing among people
Having Frequent guidance from high/ expert people informally		

Same Technology/same development process	Enhancing knowledge and skills through self learning and sharing knowledge activities (4)	Time constraint/ Cost constraint
Busy		Training - internal
Sharing Knowledge - casually		Training – when require
Self understanding		Hiring expert
Self explore		suit company objective/ open to all
Self learning/ enhancing		Contract -training
Self renewing		Enhancing Knowledge- Sharing
		Sharing experience

Blog	Shared Tacit Knowledge - 'blog' or online (4)	Online sharing – remote staff
Blog – sharing knowledge		No 'Blog'
Blog - Share information		Email, Skype and phone
Blog- Internal Share		Internet support
Blog- Open Discussion		
Blog – time constraints		
Blog – must align with people interest		

- **Documentation Process**

Documentation – technical/ non technical Technical Documentation – not useful	Informal documentation tacit knowledge process (11)	Tacit knowledge
Technical documentation – not meet client requirement		Having very informal meeting
old formal method Old formal method- heavy process and not suitable for small company		Documentation process – not systematic and standardize Banana Product -
" Not over documentation"		Documentation – less priority
Documentation -Frequent change and evolve		Documentation – small team size
Process not clearly define and document		Documentation – no clear process and standard
Documentation – no standard		Main documentation – not shared
Documentation – time consuming		Main development process – self explain
Documentation -Small team Size		Documentation – individual
Documentation – not complete		Documentation – not follow existing standard

Working according to their expertise and more to autonomous working	Documentation Process	Not having formal documentation
Practicing informal environment and easy to access		Formal documentation more related to business process and product
Having less structure communication and work process		In development, documentation more on electronic code and other technical things
Admitting the documentation process not well organize		Having e personal documentation for their own used
Using more electronic documentation than paper documentation		Time and people are main reason on this issues
Not having standard documentation procedure		Minimal / simple documentation
Documentation is more informal process and personal documentation / Doing personal documentation for personal own benefit		Using company standard internal tool - teamwork
Using specific tools –xplanner- to document people activities		Not serious in documentation process
Having team work but each responsible their task and delivery time		Formal Documenting only task related to business process and product
No need to document in detail		Having internal documentation but simple and

		brief
Using specific tools- xplanner – to record people activity and progress		Having more on self documentation rather than standard documentation
Doing personal documentation		
Having light documentation that more toward specification		
Main documentation is programming code compare o others.		
Having formal documentation related to formal procedure and product		

Documentation - specific document and issue	Documentation Specific Information 11	Version of documentation
Documentation - feedback process		Loose documentation - unofficial
Documentation – no standard		Proper Documentation - client
□		Document information - main task - code and spec
		‘Banana Product’- mature with client
□		Standardize documentation process

Perceiving that not much knowledge been created	Created same knowledge all time (9)	Perceiving that similar knowledge been created
Company using same technology and process in every project		Using the same technology and development process all the time
Perceiving the main knowledge in company are similar		Following what process and technology is suit to company and been proof efficient development process only

Knowledge loss Not serious Staff turn over-low	Knowledge Loss Problem issues (17)	Knowledge loss Staff turn over-low
Repetitive work Continuous learning Blog Economy		Repetitive work Documentation Sharing Centralize system Standard coding process Economy Self learning
[redacted]	[redacted]	[redacted]
People working across their area	Knowledge Preserve Issue	Relying on the programming code and other technical stuff in the database
People working support each other		
No having formal plan to preserver loss knowledge		
More rely on coding and any related document		
Having Frequent guidance from high/ expert people informally		
Ensure no one exclusively work in one project - Hard to apply due to size		

Less knowledge documentation	Knowledge Loss Problem symptom (17)	Less knowledge documentation

SPIME

Interactive communication	‘Agile’ development (10)	Open communication-n evaluating changing and evolving process
Prototype		Communication Frequent
Prototype -Interaction/feedback– Suit Company project		Adapting 'agile' method philosophy
Prototype handle customer needs and requirement		Development model and process –standard
Adapting 'agile' method		Interactive communication
Implementing changes immediately		Work autonomous
Less documentation process		
Agile method - improving development process		
Open environment important to improve process		
Hiring expert		

Organic	Process Improvement, maintenance and evolution (14)	Reactive vs proactive
Interactive		Process Focus – technology change/ change requirement
Not documented		Product version
Frequent change		Product Orientation
Try and Error		Less resources/ Time constraints
Small scale Autonomous work No specific standard/guideline		Client orientation
Informal No Standard Current Standard - heavy process		Interaction/engagement
Natural Change		Module orientation
Agile type		luxury
Process Loss		Informal/ direct changes/ autonomous work/ rapid change
Process Focus		Indirect changes
New Voice/idea		Market Change/ Customer requirement
Time and resources constraints		Technology change
Market Change/ Customer requirement Product oriented		

Following agile development process and practice	SPIME	Using same development tools
Applying autonomous work and communication		Using similar development process in all product
Practicing informal environment and easy to access		Using standard development tools
Having every 2 week update and test		Being aware that development process always been improve
Having continuous integration on product		Improvement in small scale and informal way.
Having minor changes on major process.		Having rapid development process evolution
Assessing current process and product of problem arise		Involving all people in improve development process
Staring from inside rather than outside		Changes/improvement start from customer or internal.
Having a mock up/prototype to customer to get feedback		Discuss with customer and also the team
Designer improved based on the comment		Evolvement process start from inside and apply to client product
Not following other/outside standard guideline		Following coding standard depend on language/ Not following specific model
Having followed loosely extreme programming process		More following RAD development model
Having customer and internal input		Being aware customer feedback and internal view main reason for any process update
Developing and update road map to manage development process		Applying small scale and informal update/change but very rapid
Doing 2 weeks iteration		Admitting profitable drive the development process
Frequently review plan to tailored with job		Doing informal post mortem process
Having notice both process align but less people avoid meet company objective		Admitting post mortem is least priority activities
Using similar development process in all project		Realizing that process always being improve but indirect and informal ways
Having small changes overtime to suit objective and requirement.		
Plan well and study for changes		
No formal post mortem process		
Customer involve in start and end of process actively		
Explaining agile is more to out put then process		
Being aware that the development process been update continuously and indirectly		
Improve but not structure		

Not enforcing procedure Less project and process documentation Using same development process Try and error Not following standard and guideline	Process Loss Problem symptom (17)	Not enforcing procedure Autonomous and freedom No development standard and guideline Reactive vs proactive Less dev process documentation

Autonomous work		Not following good practice – set up before Not update documentation

Latest Technology	Critical Factor - sustain business , product and business competitiveness (20)	Requirement Align
Improve Development Process		Better Module and version
Reliable Software		Customer Relationship
Evolving Development Process		Follow Customer Requirement
Follow Customer Requirement		Sustain customer relationship
Right Estimation		Post Mortem
Realistic Decision		Open environment
Realistic Planning		
Right Team size		
Process Bound		
Right interaction		
Good relationship		
Right development method		

Software Quality Standard

Software quality Software quality standard Software quality standard – shows interest ISO9000 Small Company	Interested adopted ISO9000 (15)	Software quality standard ISO9000 – company plan history Small Company
Software quality standard - high awareness		Preferring standard that align with current process

Lead to a quality product	High Awareness having benefit (15)	Effective work
Creating consistency		Creating consistency
Company image		Company Advantage

People Size	Critical factors - adopting standard (15)	Team size
Specific people – to look after		Time constraint
Team size		High Cost
High cost Huge resources		Heavy Documentation Heavyweight process
Company image		Complicated procedure
Heavy weight process Heavy documentation Huge different from current process Details process		Not compulsory/low demand
Less interest		No problem
Not compulsory		No request -client
Public Company/ Government agency		Level of readiness
No problem		Sales tools
No request -client		Not suitable – small company
Level of awareness		Level of awareness
Complicated procedure		Company image

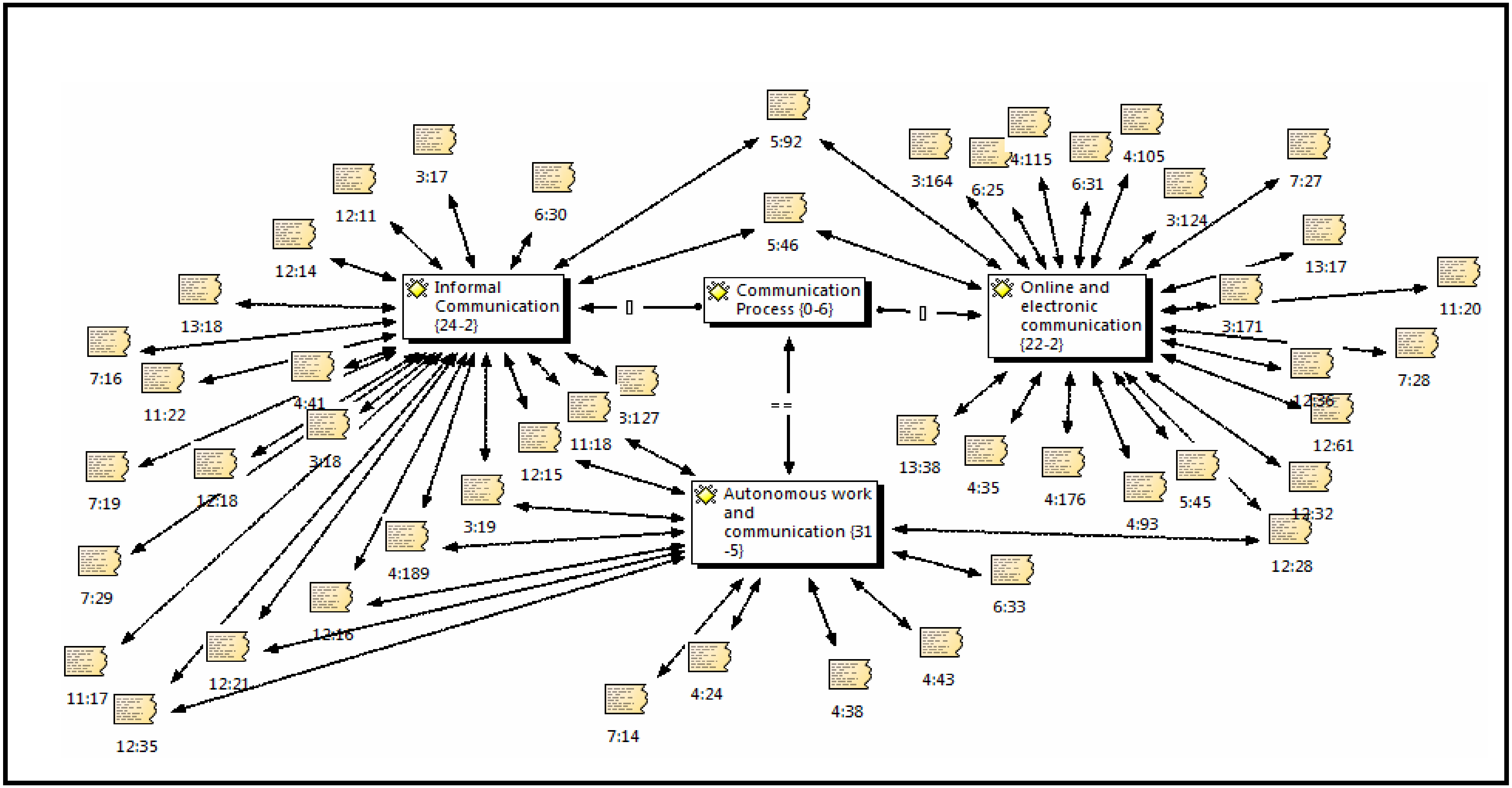
Not following other/outside standard guideline	Software Quality Standard	Having too detail documentation
No plan to apply any standard		Required extra man power
Focusing on delivery time then other process		Not important to developer
Having Less resources and time		No plan to adopt standard currently
Being aware that ISO involved repeatability process rather than quality		Admitting and realizing that quality standard make company more organize and improve work process.
Following agile type development		Aware that quality standard too much detail and not related with company current process
Being aware higher code quality is more important than process		Aware ISO good for business but current ISO not suitable to small company
Less resource		Doing some background study but no plan to adopt now
Not require by customer		
ISO more toward services rather than software product		

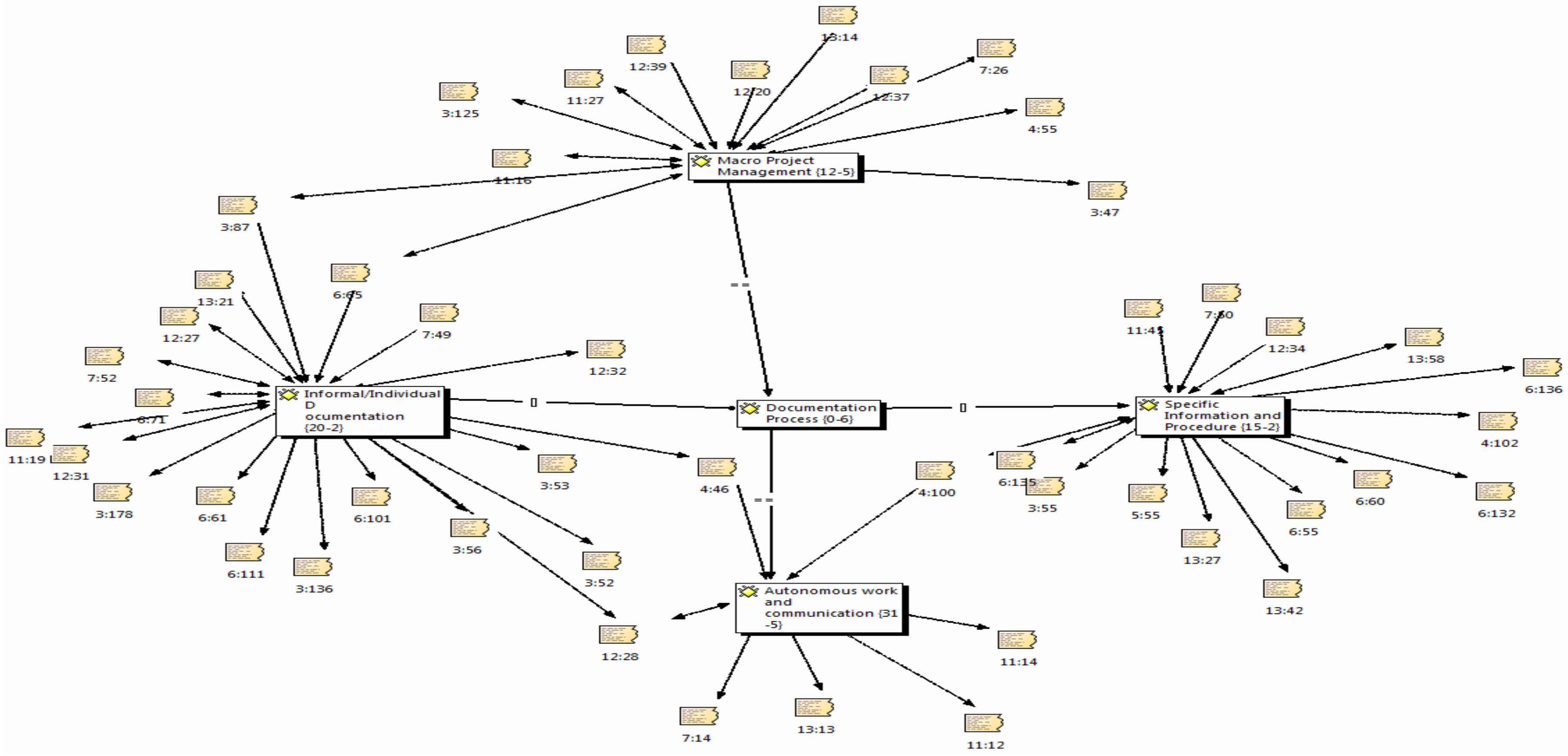
Basic list company operation	New Standard criteria	Align with current process

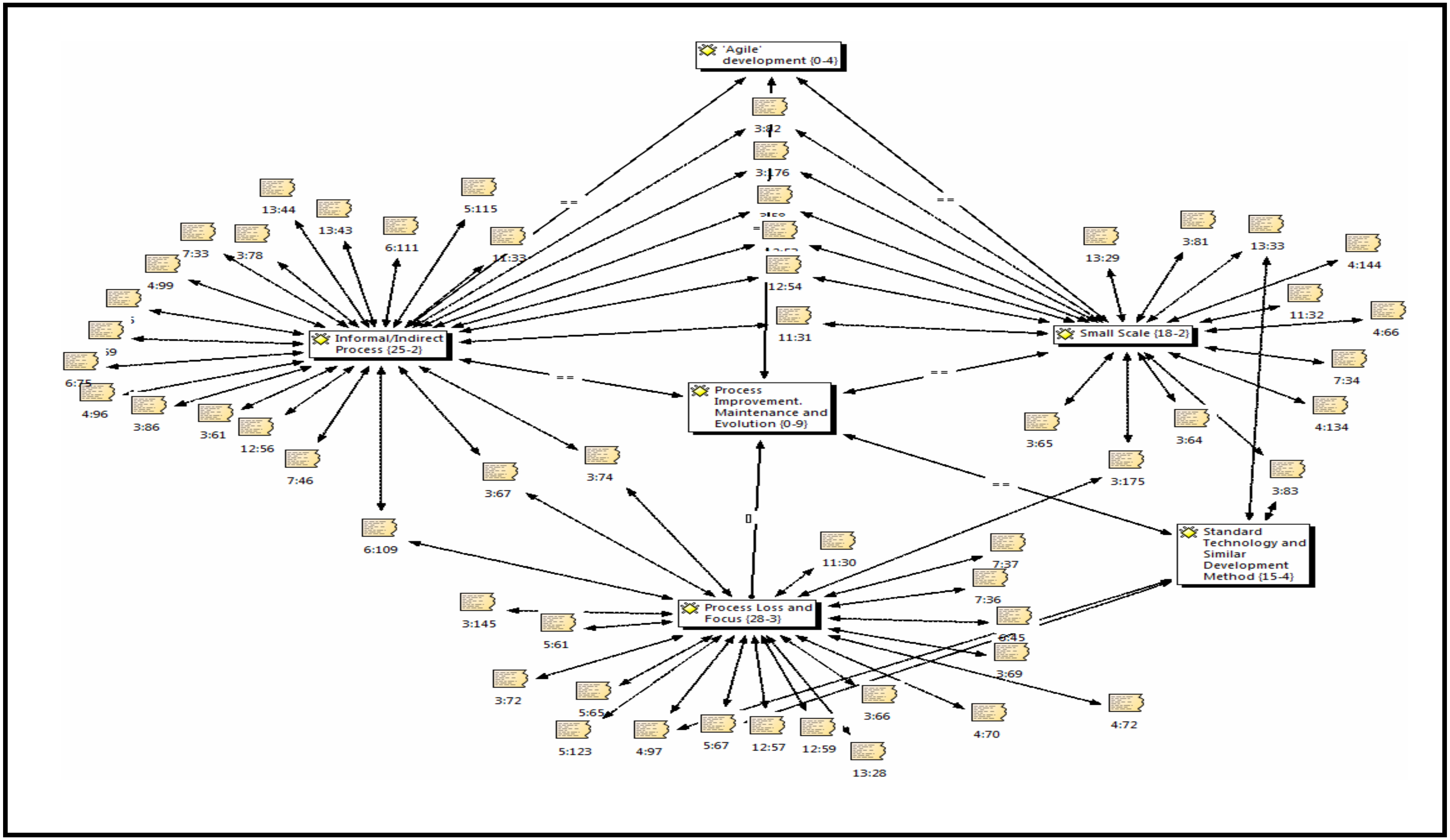
Level of readiness	(15)	Light weight process Not time consume
Easy and light administration		Light weight process
Less changes in current process		Align with technical standard
Align current business operation		Less complicated
Standard Style		Less cost and resources
Light weight process Not time consume		Workshop
Flexible		Detail guideline
Less cost and resources		Clear Template
Clear Template		Related to existing standard
Align with technical standard		

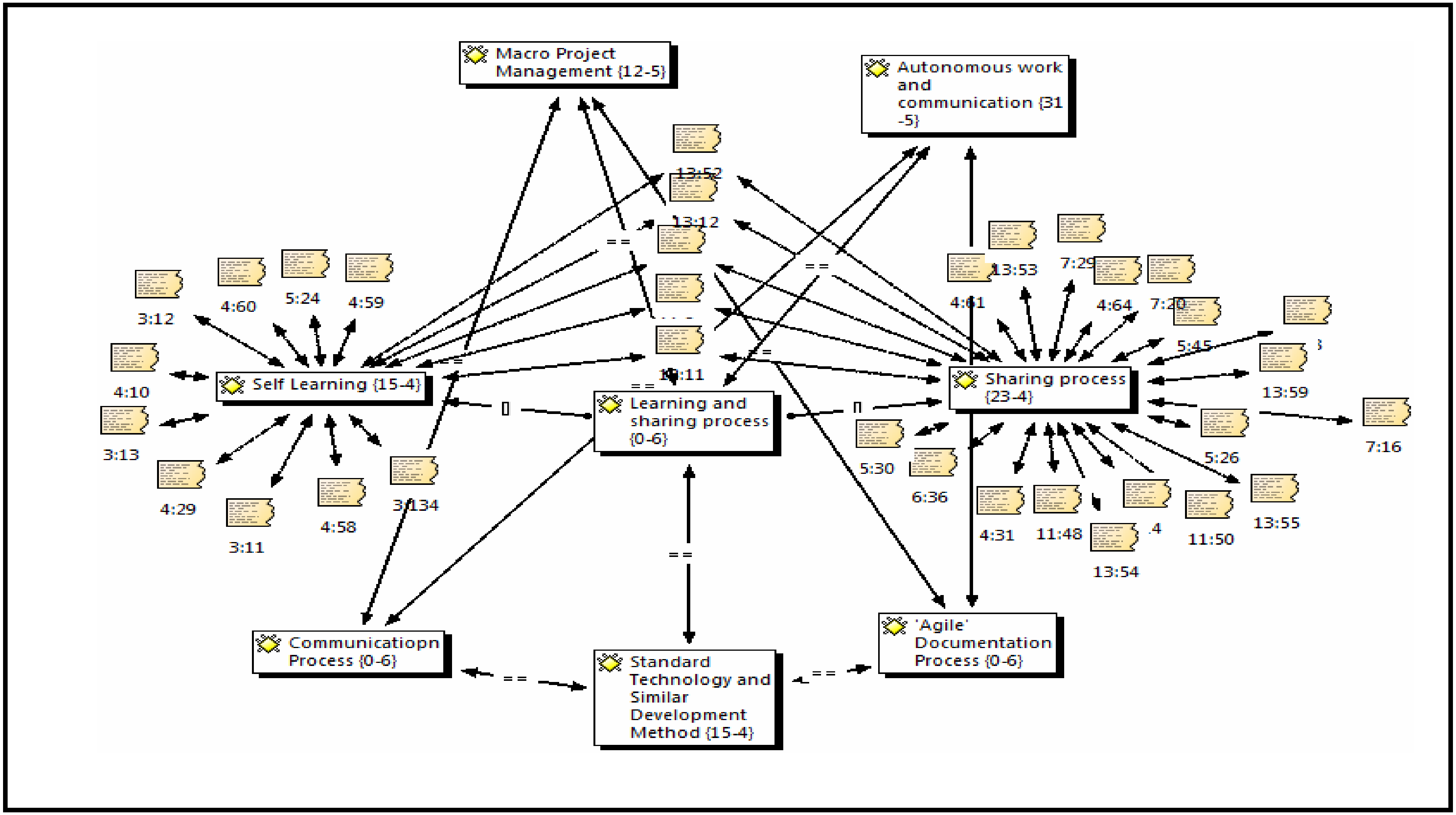
‘Only one staff aware of standard’	Less interest/knowledgeable/ Motivation on software standard (15)	More interested in technical standard
Being unsure either the staff are aware and knowledgeable about software quality standard		Arguing that technical standard is more related to company business
Planning to obtain the well recognize in order to enhance their company image and standard		Manager less interested in software standard
Management not knowledgeable and less interested about software quality standard		Staff less interested in software quality standard
Staff less knowledgeable and interested in software quality standard		Being aware ISO not compulsory in software business
Starting looking more serious about technical standard		Being aware ISO not important to small company

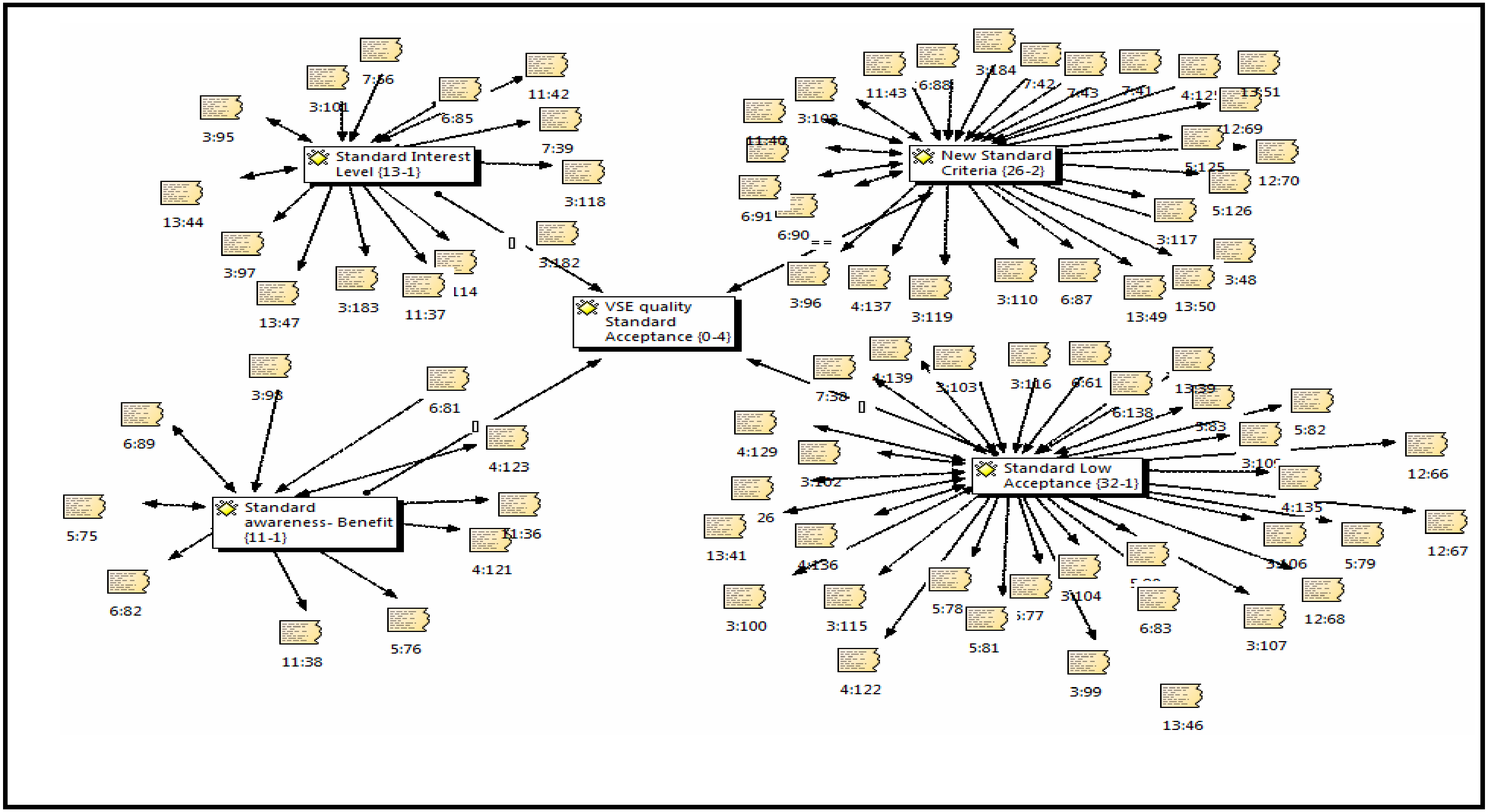
APPENDIX G
Individual Diagram

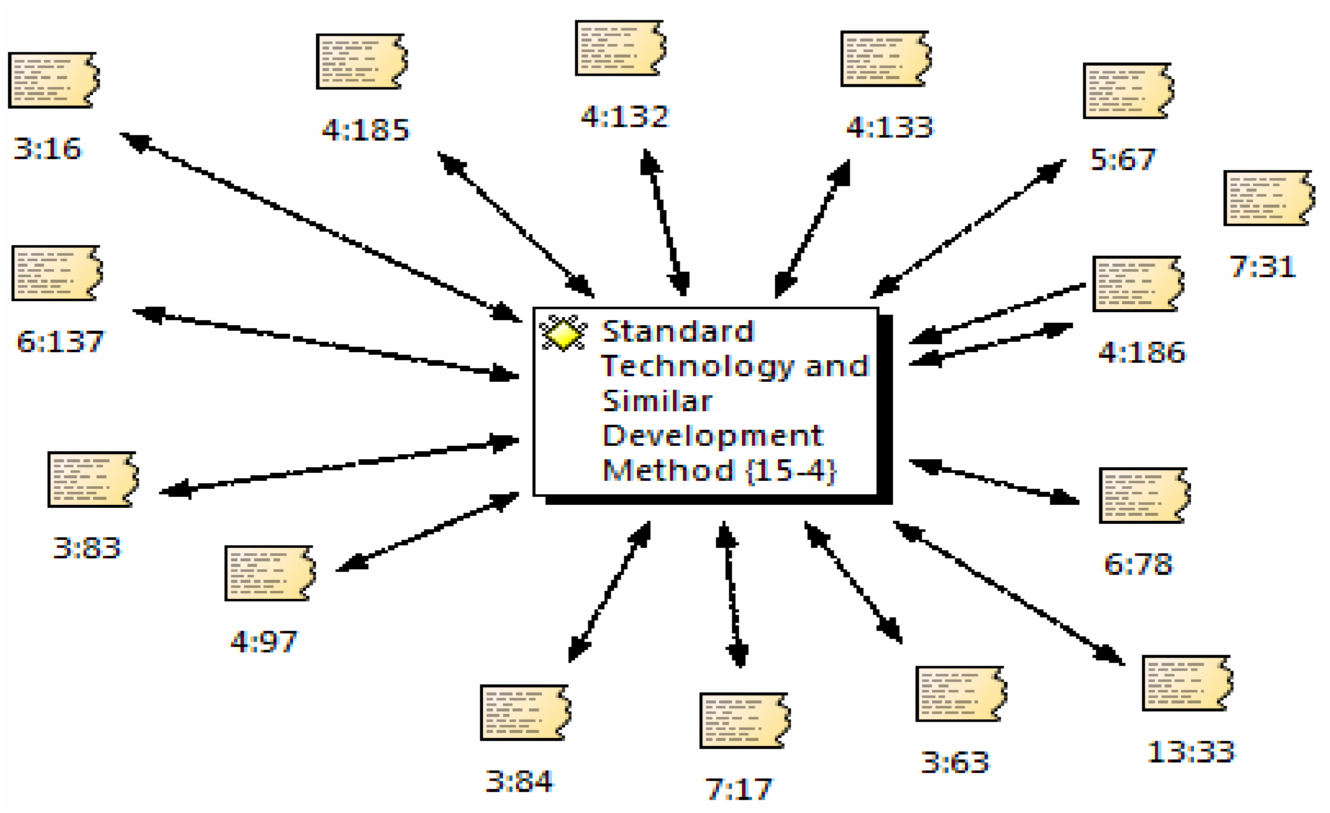


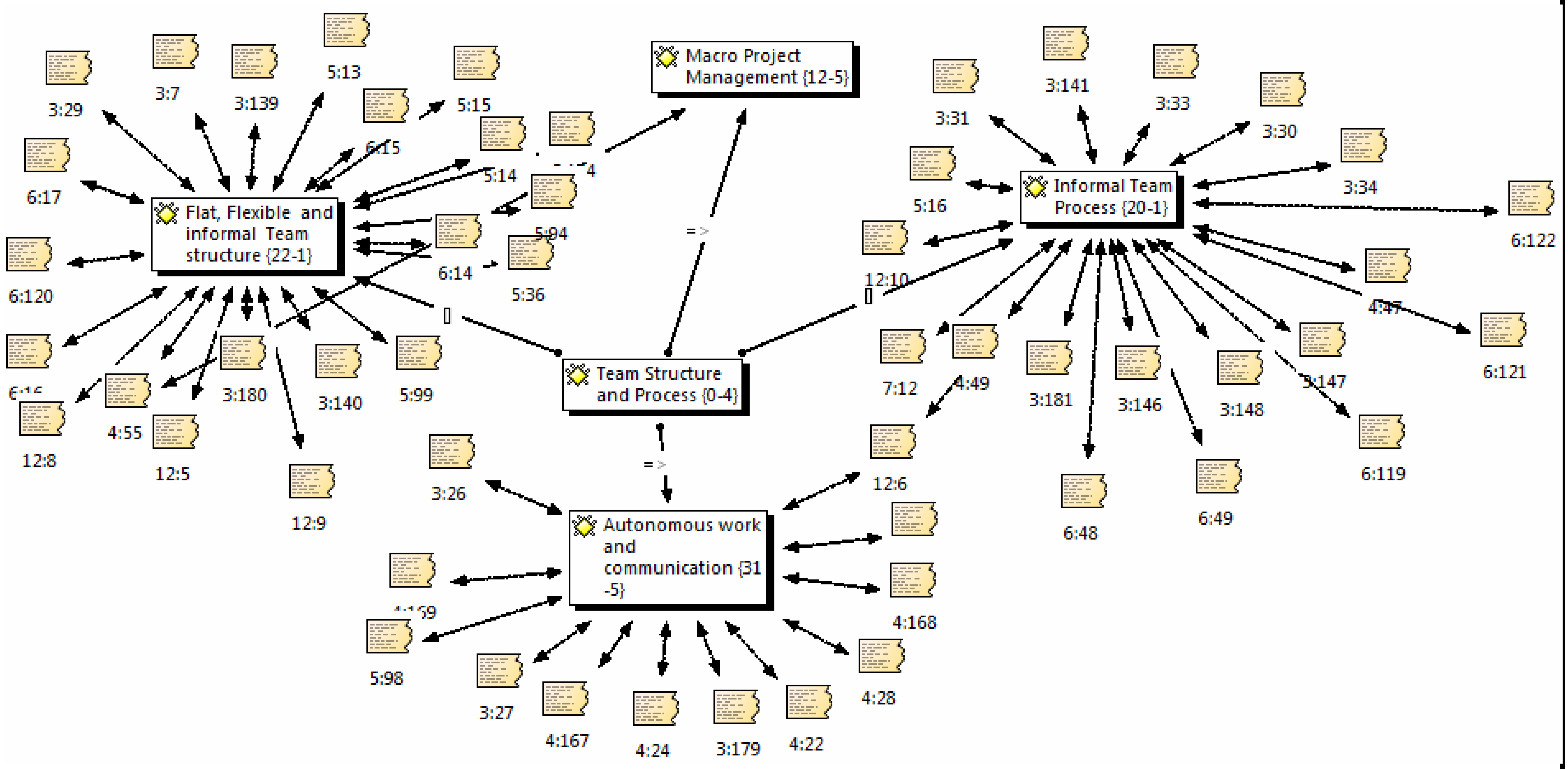

13:14
12:39
7:26
11:27
12:20
12:37
3:125
4:55
11:16
Macro Project Management {12-5}
3:47
3:87
==
13:21
6:65
12:27
7:49
7:52
12:32
Informal/Individual D ocumentation {20-2}
Documentation Process {0-6}
Specific Information and Procedure {15-2}
6:71
11:19
12:31
3:178
6:61
6:101
3:53
4:46
4:100
3:56
6:111
3:136
3:52
Autonomous work and communication {31 -5}
12:28
7:14
13:13
11:12
11:14
11:45
7:50
12:34
13:58
6:136
4:102
6:135
3:55
5:55
13:27
6:55
6:60
6:132
13:42


'Agile' development {0-4}
3:82
3:176
12:54
11:31
Informal/Indirect Process {25-2}
Small Scale {18-2}
Process Improvement. Maintenance and Evolution {0-9}
Process Loss and Focus {28-3}
Standard Technology and Similar Development Method {15-4}
13:44
13:43
5:115
6:111
11:33
7:33
3:78
4:99
6:75
4:96
3:86
3:61
12:56
7:46
3:67
3:74
6:109
13:29
3:81
13:33
4:144
11:32
4:66
7:34
4:134
3:65
3:64
3:175
3:83
11:30
7:37
7:36
6:45
3:69
3:145
5:61
3:72
5:65
5:123
4:97
5:67
12:57
12:59
13:28
3:66
4:70
4:72
= =
= =
= =
= =
= =


Macro Project Management {12-5}
Autonomous work and communication {31 -5}
13:52
13:12
10:11
Self Learning {15-4}
Learning and sharing process {0-6}
Sharing process {23-4}
Communicatiopn Process {0-6}
Standard Technology and Similar Development Method {15-4}
'Agile' Documentation Process {0-6}
3:12
4:60
5:24
4:59
4:10
3:13
4:29
3:11
4:58
3:134
4:61
13:53
7:29
4:64
7:20
5:45
13:59
7:16
5:30
6:36
5:26
4:31
11:48
13:54
.4
11:50
13:55
= =
[]


Standard Interest Level {13-1}
7:56
3:101
6:85
11:42
3:95
7:39
3:118
13:44
3:97
13:47
3:183
11:37
L14
3:182
New Standard Criteria {26-2}
11:43
6:88
3:184
7:42
7:43
7:41
4:12'
13:51
3:108
11:40
12:69
5:125
12:70
6:91
6:90
5:126
3:117
3:48
3:96
4:137
3:119
3:110
6:87
13:49
13:50
VSE quality Standard Acceptance {0-4}
Standard awareness- Benefit {11-1}
3:98
6:81
6:89
4:123
5:75
11:36
4:121
6:82
11:38
5:76
Standard Low Acceptance {32-1}
7:38
4:139
3:103
3:116
6:61
13:39
6:138
5:83
5:82
4:129
3:102
3:10
12:66
26
4:135
12:67
13:41
4:136
5:79
3:106
3:100
3:115
5:78
3:104
5:77
6:83
12:68
3:107
5:81
4:122
3:99
13:46


3:16
4:185
4:132
4:133
5:67
7:31
6:137
4:186
Standard Technology and Similar Development Method {15-4}
3:83
6:78
4:97
3:84
7:17
3:63
13:33


Macro Project Management {12-5}
Flat, Flexible and informal Team structure {22-1}
Informal Team Process {20-1}
Team Structure and Process {0-4}
Autonomous work and communication {31-5}
=>
=>
3:29
3:7
3:139
5:13
5:15
6:15
5:14
6:17
6:120
6:14
5:94
5:36
6:16
12:8
4:55
12:5
3:180
3:140
5:99
12:9
3:26
4:169
5:98
3:27
4:167
4:24
3:179
4:22
4:28
4:168
12:6
3:31
3:141
3:33
3:30
3:34
5:16
6:122
12:10
4:47
6:121
7:12
4:49
3:181
3:146
3:148
3:147
6:119
6:48
6:49

APPENDIX H:
OVERALL DIAGRAM

QUALITATIVE CONTENT ANALYSIS

Sub Category	Category	Main Category
IT Experience - Manager	People Skill and Experience (1)	VSE Staff Background
IT Experience - Team		
Expert		
IT related qualification/degree		
High Experience/ Skill full		
IT Background	Background of founder (1)	
Non IT Background		
Proactive	People Criteria and Qualification (1)	
High Interest and creative		
IT related qualification/degree		
Average Abilities		
Adaptable stress		
Family Business/ Company / Owner - Brothers	Company Nature (2)	VSE Buisness Operation
Non Family based - Partner		
Years Established	Establish and located at specific places (2)	
Small Companies/department		
Small office space		
Remote development team/ GSD operation	Development operation (2)	
Collocate development team		
Remote Office		
Specific office		
Frequent Visit		
Autonomous work	Autonomous work and communication (6)	
Autonomous Communication		
Work independently		
Strategic area		
Sole Responsibility		
Self Learning		
‘Team of One’		
Trust	Macro Project Management	

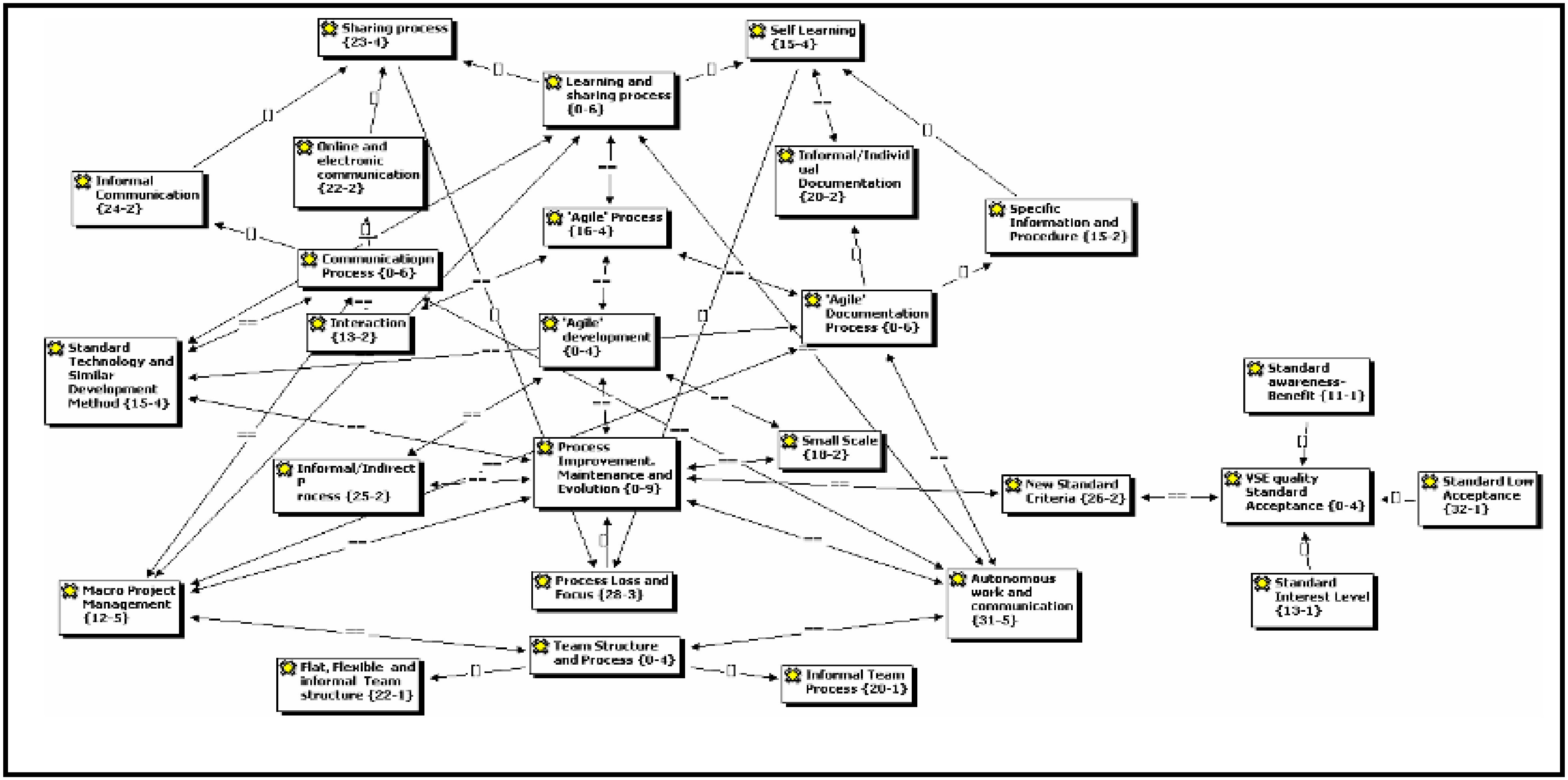
Family and Flexible Environment	(16)	
Loose PM		
Open Environment		
Team Size - Small	Organizational and Team Structure (3)	VSE Team Structure and Process
Organizational and Team Flat Structure -		
Team Role	Team process (3)	
Team Involvement		
Team Culture		
Autonomous work and communication (6)		
Applied standard technology and standard development method Communication Process		VSE Team Learning and Sharing Process (Knowledge Management)
Training	Self Learning Culture (4)	
Self Learning		
Continuous Guidance		
Internal Training	Sharing Culture (4)	
Meeting		
Knowledge Sharing		
Open Communication	Open Informal Communication (5)	VSE Team Communication Process
Informal Communication		
Autonomous work and communication (6)		
Communication tools	Online communication (5)	
Internet/ Electronically		
Economy Situation	Minimum case staff turn over(7)	
VSE Team Structure and Process		
Autonomous work and communication(6)		
Macro Project Management (16)		
Autonomous work and communication **Macro Project Management** (Autonomous PM)		'Agile' Documentation Process (Knowledge Management)
Informal documentation	Informal documentation tacit knowledge process (11/12)	
Individual Documentation		

Technical Business related document	Specific Information (11)	
Client related Document		
Applied standard technology and standard development method (8)		Improvement , Maintenance and Evolvement Development Process
Informal/Indirect	Process Improvement, maintenance and evolution status.) (14)	
Small Scale		
Process Loss and Focus		
‘Agile’ Development Style	‘Agile’ development	
‘Agile’ Documentation Process		
Interactive		
Standard Technology	Applied standard technology and standard development method(8)	
Standard development Process		
High awareness adopting Standard	Interest Level - - standard (15)	Quality Standard Acceptance Level
Critical Factors		
Standard Guideline	New Standard criteria (15)	
Light weight Process		
Flexibility		
Business and Process Align		
Technical Standard align		
Assistance/Guideline		
Standard Benefit Awareness	Level of Awareness	
Low acceptable	Level of Acceptance(15)	
Less Interest and Priority		
Autonomous work and communication(6)	Knowledge Loss Problem Thread (17)	Knowledge Loss Tread and Treat
Macro Project Management(16)		
Agile’ Documentation Process (Knowledge Management)		
Macro Project Management(16)	Knowledge Loss Problem -Treat (17)	
Autonomous work and communication(6)		
Applied standard technology and standard development method(8)		
VSE Team Communication Process		
VSE Team Learning and Sharing Process (Knowledge Management)		

Not Enforcing Procedure		Process Loss Problem symptoms(17)
Less Project and Process Documentation		
Using same development process		
‘Try and Error’ concept’ (Not Following Standard guideline)		
Autonomous Work and communication		
Agile’ Documentation Process (Knowledge Management)		
Reactive and Proactive		
Autonomous work and communication(6)		Team work Vs Team dynamics
Macro Project Management(16)		
VSE Team Communication Process		
Frequent Knowledge sharing	Collocate - good team work and dynamic (19)	
Close working Space		
Socializing		
Frequent Cross interaction		
Casual Communication		
Team Process Difficulties	GSD Less team dynamic/ team notion (19)	
Communication Difficulties		
Control Problem		
Local Focus		
Being aware knowledge played big role in software development process		KM and Team and SPI relationship awareness (18)
Admitting sharing knowledge could carry more knowledge		
Being aware good documentation help sustain knowledge		
Having team relation could relate new knowledge		
Knowledge management and good team - -improve standard process and Knowledge management and good team - -enhance teamwork		
experience and knowledge -- main asset of the company		
Knowledge and Experience - giving benefit development process		
Team work - enhance knowledge and work - enhance team relationship		
New knowledge - can improve existing software process		
Knowledge management and good team - -improve standard process and enhance teamwork		
working alone -cannot enhance knowledge and always follows old development style		
standard coding process -help to retain knowledge		

good documentation -help in sustaining knowledge	
clear development process and procedure help - retaining knowledge	
good repository place -help in sustaining knowledge	
Good development team - enhance new idea in developing system	
Team work/Team Dynamic	Critical Factor - Enhanced knowledge sharing (18)
Macro Project Management	
Communication tools	
' Organizational and Team Structure	
Autonomous work and communication(6)	
Latest Technology/ Reliable Software	Critical Factor - sustain business , product and business competitiveness (20)
Improve development process	
Evolving development process	
Right estimation, decision, team size and interaction	
Following Customer Requirement	
Process Bound	
Right Development Method	
Open Environment	
Sustain Customer relationship	
Better Module and Version	

Appendix H
Overall Network Diagram


Sharing process {23-4}
Self Learning {15-4}
Learning and sharing process {0-6}
Online and electronic communication {22-2}
Informal Communication {24-2}
Informal/Individ ual Documentation {20-2}
'Agile' Process {16-4}
Specific Information and Procedure {15-2}
Communicatiopn Process {0-6}
Interaction {13-2}
'Agile' Documentation Process {0-6}
'Agile' development {0-4}
Standard Technology and Similar Development Method {15-4}
Standard awareness-Benefit {11-1}
Small Scale {10-2}
Process Improvement. Maintenance and Evolution {0-9}
Informal/Indirect P rocess {25-2}
New Standard Criteria {26-2}
VSE quality Standard Acceptance {0-4}
Standard Low Acceptance {32-1}
Macro Project Management {12-5}
Process Loss and Focus {28-3}
Autonomous work and communication {31-5}
Standard Interest Level {13-1}
Team Structure and Process {0-4}
Flat, Flexible and informal Team structure {22-1}
Informal Team Process {20-1}

APPENDIX I:
Survey Questionnaire: Quantitative Analysis

Communication

E4	b.	There is clear communication between team members.	***Communication***	***5***	***4***	***3***	***2***	***1***
D3	a.	Good team communication is important in managing software development knowledge.		***5***	***4***	***3***	***2***	***1***
D3	f.	Regular formal meetings are important for sharing and transfer of knowledge		***5***	***4***	***3***	***2***	***1***
C8	a.	Software development projects regularly receive feedback over stakeholder.		***5***	***4***	***3***	***2***	***1***
C8	g.	There is an effective communications channel between software development tem members and management		***5***	***4***	***3***	***2***	***1***
C8	i.	There are regular formal meetings between software development staff and management.		***5***	***4***	***3***	***2***	***1***
C8	j.	There are regular informal (casual) communications between software development staff and management.		***5***	***4***	***3***	***2***	***1***

SWDevPracticeInCmmpany - regularFdBk

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	1	10.0	10.0	10.0
	AGREE	4	40.0	40.0	50.0
	STRONGLY AGREE	5	50.0	50.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany - EffCommTeamMgmt

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	4	40.0	40.0	40.0
	STRONGLY AGREE	6	60.0	60.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany - RegularFrmalMtg

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	5	50.0	50.0	50.0
	NEUTRAL	4	40.0	40.0	90.0
	AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany -InformalComm

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	2	20.0	20.0	20.0
	STRONGLY AGREE	8	80.0	80.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg- GoodTeamComm

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	1	10.0	10.0	10.0
	STRONGLY AGREE	9	90.0	90.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-RegularMeeting

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	7	70.0	70.0	70.0
	NEUTRAL	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

SDevTeam-ClearComm

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	4	40.0	40.0	40.0
	STRONGLY AGREE	6	60.0	60.0	100.0
	Total	10	100.0	100.0	

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SWDevPracticeInCmmpany – regularFdBk	10	3	5	4.40	.699
SWDevPracticeInCmmpany - EffCommTeamMgmt	10	4	5	4.60	.516
SWDevPracticeInCmmpany - RegularFrmalMtg	10	2	4	2.60	.699
SWDevPracticeInCmmpany -InformalComm	10	4	5	4.80	.422
SDevKnowldg-GoodTeamComm	10	4	5	4.90	.316
SDevKnowldg-RegularMeeting	10	2	3	2.30	.483
SDevTeam-ClearComm	10	4	5	4.60	.516
Valid N (listwise)	10				

Report

EmpSize Group		SDevTeam-ClearComm	SDevKnowldg-GoodTeamComm	SDevKnowldg-RegularMeeting	SWDevPracticeInCmmpany - regularFdBk	SWDevPracticeInCmmpany - EffCommTeamMgmt	SWDevPracticeInCmmpany - RegularFrmalMtg	SWDevPracticeInCmmpany - InformalComm
1-9	Mean	4.80	5.00	2.20	4.40	4.80	2.20	5.00
	N	5	5	5	5	5	5	5
	Std. Deviation	.447	.000	.447	.548	.447	.447	.000
10-24	Mean	4.40	4.80	2.40	4.40	4.40	3.00	4.60
	N	5	5	5	5	5	5	5
	Std. Deviation	.548	.447	.548	.894	.548	.707	.548
Total	Mean	4.60	4.90	2.30	4.40	4.60	2.60	4.80
	N	10	10	10	10	10	10	10
	Std. Deviation	.516	.316	.483	.699	.516	.699	.422

Learning and Sharing

D3	m.	We always exploit existing organizational knowledge to the maximum.	***Learning and Sharing***	***5***	***4***	***3***	***2***	***1***
D3	n.	We always learn from experiences of past projects.		***5***	***4***	***3***	***2***	***1***
D3	o.	We always collect experience data from past projects.		***5***	***4***	***3***	***2***	***1***

SDevKnowldg-ExploitKnowledge

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	2	20.0	20.0	20.0
	AGREE	4	40.0	40.0	60.0
	STRONGLY AGREE	4	40.0	40.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-LearnPastExperience

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	3	30.0	30.0	30.0
	AGREE	4	40.0	40.0	70.0
	STRONGLY AGREE	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-CollectPastExperience

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	4	40.0	40.0	40.0
	AGREE	5	50.0	50.0	90.0
	STRONGLY AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SDevKnowldg-ExploitKnowledge	10	3	5	4.20	.789
SDevKnowldg-LearnPastExperience	10	3	5	4.00	.816
SDevKnowldg-CollectPastExperience	10	3	5	3.70	.675
Valid N (listwise)	10				

Report

EmpSizeGroup		SDevKnowldg-ExploitKnowledge	SDevKnowldg-LearnPastExperience	SDevKnowldg-CollectPastExperience
1-9	Mean	4.00	4.20	4.00
	N	5	5	5
	Std. Deviation	.707	.837	.707
10-24	Mean	4.40	3.80	3.40
	N	5	5	5
	Std. Deviation	.894	.837	.548
Total	Mean	4.20	4.00	3.70
	N	10	10	10
	Std. Deviation	.789	.816	.675

Tools

D3	a.	Good team communication is important in managing software development knowledge.	***Tool***	***5***	***4***	***3***	***2***	***1***
D3	c.	Software communication tools (e.g. Skype, Blog, etc) support and enhance knowledge propagation.		***5***	***4***	***3***	***2***	***1***
E4	g.	Software communication tools (e.g. Skype, email, etc) are regularly been used in software development projects.		***5***	***4***	***3***	***2***	***1***

SDevKnowldg- GoodTeamComm

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	1	10.0	10.0	10.0
	STRONGLY AGREE	9	90.0	90.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg- KnowldgCommTools

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	2	20.0	20.0	20.0
	AGREE	5	50.0	50.0	70.0
	STRONGLY AGREE	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

SDevTeam-SWCommTools

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	3	30.0	30.0	30.0
	AGREE	1	10.0	10.0	40.0
	STRONGLY AGREE	6	60.0	60.0	100.0
	Total	10	100.0	100.0	

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SDevKnowldg-GoodTeamComm	10	4	5	4.90	.316
SDevKnowldg-KnowldgCommTools	10	3	5	4.10	.738

SDevTeam-SWCommTools	10	3	5	4.30	.949
Valid N (listwise)	10				

Report

EmpSizeGroup		SDevKnowldg-GoodTeamComm	SDevKnowldg-KnowldgCommTools	SDevTeam-SWCommTools
1-9	Mean	5.00	4.60	4.80
	N	5	5	5
	Std. Deviation	.000	.548	.447
10-24	Mean	4.80	3.60	3.80
	N	5	5	5
	Std. Deviation	.447	.548	1.095
Total	Mean	4.90	4.10	4.30
	N	10	10	10
	Std. Deviation	.316	.738	.949

Documentation Process

D3	k.	Clear documentation processes and guidelines are necessary to sustain knowledge in software development		***5***	***4***	***3***	***2***	***1***
C8	e.	Software development staff knowledge is formally documented	***Documentation***	***5***	***4***	***3***	***2***	***1***
C8	f.	We regularly document experiences/lessons learned from previous projects to use in future projects.		***5***	***4***	***3***	***2***	***1***
C9	m.	Software development teams are regularly document and review their activities.		***5***	***4***	***3***	***2***	***1***
D3	q.	We regularly document our knowledge and experience in software development projects.		***5***	***4***	***3***	***2***	***1***
D3	r.	We regularly documented our work / project progress.		***5***	***4***	***3***	***2***	***1***

Frequencies

Statistics

		SDev Knowldg-ClearDoc Process NGuideline	SWDev PracticeIn Cmmpany - Formal Document	SWDev PracticeIn Cmmpany - Doc Experience	SWDev PracticeIn Cmmpany-DocActvtand Rev	SDev Knowldg-Regular Documente dxperience	SDev Knowldg-RegularDoc Work&proj Prog
N	Valid	10	10	10	10	10	10
	Missing	0	0	0	0	0	0

Frequency Table

SDevKnowldg-ClearDocProcessNGuideline

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	3	30.0	30.0	30.0
	AGREE	7	70.0	70.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany - FormalDocument

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	STRONGLY DISAGREE	1	10.0	10.0	10.0
	DISAGREE	5	50.0	50.0	60.0
	NEUTRAL	2	20.0	20.0	80.0
	AGREE	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany - DocExperience

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	STRONGLY DISAGREE	1	10.0	10.0	10.0
	DISAGREE	4	40.0	40.0	50.0
	NEUTRAL	2	20.0	20.0	70.0
	AGREE	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- DocActvtandRev

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	2	20.0	20.0	20.0
	NEUTRAL	6	60.0	60.0	80.0
	AGREE	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-RegularDocumentedxperience

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	6	60.0	60.0	60.0
	NEUTRAL	3	30.0	30.0	90.0
	AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-RegularDocWork&projProg

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	6	60.0	60.0	60.0
	NEUTRAL	4	40.0	40.0	100.0
	Total	10	100.0	100.0	

Descriptives

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SDevKnowldg-Clear DocProcessNGuideline	10	3	4	3.70	.483
SWDevPracticeIn Cmmpany - FormalDocument	10	1	4	2.50	.972
SWDevPracticeIn Cmmpany - DocExperience	10	1	4	2.70	1.059
SWDevPracticeIn Cmmpany- DocActvtandRev	10	2	4	3.00	.667
SDevKnowldg-Regular Documentedxperience	10	2	4	2.50	.707
SDevKnowldg-Regular DocWork&projProg	10	2	3	2.40	.516
Valid N (listwise)	10				

```
MEANS
  TABLES=D3k C8e C8f C9m D3q D3r  BY EmplyeeSIzeGRoup
  /CELLS MEAN COUNT STDDEV  .
```

Means

```
[DataSet1] C:\shuib\QuestionaireData.sav
```

Case Processing Summary

	Cases					
	Included		Excluded		Total	
	N	Percent	N	Percent	N	Percent
SDevKnowldg-Clear DocProcessNGuideline * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SWDevPracticeIn Cmmpany - FormalDocument * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SWDevPracticeIn Cmmpany - DocExperience * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SWDevPracticeIn Cmmpany- DocActvtandRev * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SDevKnowldg-Regular Documentedxperience * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SDevKnowldg-Regular DocWork&projProg * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%

Report

EmpSizeGroup		SDevKnowldg-ClearDocProcessNGuideline	SWDevPracticeInCmmpany - FormalDocument	SWDevPracticeInCmmpany - DocExperience	SWDevPracticeInCmmpany-DocActvtandRev	SDevKnowldg-RegularDocumentedxperience	SDevKnowldg-RegularDocWork&projProg
1-9	Mean	3.40	2.20	2.20	2.60	2.20	2.20
	N	5	5	5	5	5	5
	Std. Deviation	.548	1.095	.837	.548	.447	.447
10-24	Mean	4.00	2.80	3.20	3.40	2.80	2.60
	N	5	5	5	5	5	5
	Std. Deviation	.000	.837	1.095	.548	.837	.548
Total	Mean	3.70	2.50	2.70	3.00	2.50	2.40
	N	10	10	10	10	10	10
	Std. Deviation	.483	.972	1.059	.667	.707	.516

KM Strategy

D3	b.	A knowledge management strategy is important in managing organisational knowledge.	***Strategy***	***5***	***4***	***3***	***2***	***1***
D3	i.	Good leadership is important in leveraging peoples knowledge and experience.		***5***	***4***	***3***	***2***	***1***
D3	g.	Formal training is given in order to sustain and enhance software development knowledge.		***5***	***4***	***3***	***2***	***1***
D3	j.	Formal project post-mortems are beneficial in capturing and transferring knowledge.		***5***	***4***	***3***	***2***	***1***
D3	d.	It is necessary to have a reward / incentive mechanism to ensure knowledge sharing / transfer.		***5***	***4***	***3***	***2***	***1***

Frequencies

Statistics

		SDev Knowldg-KMStrategy	SDev Knowldg-Good Leadership	SDev Knowldg-Postmortem	SDev Knowldg-Formal Training	SDev Knowldg-Reward IncentiveKn Sharing
N	Valid	10	10	10	10	10
	Missing	0	0	0	0	0

Frequency Table

SDevKnowldg- KMStrategy

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	3	30.0	30.0	30.0
	AGREE	7	70.0	70.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-Good Leadership

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	5	50.0	50.0	50.0
	STRONGLY AGREE	5	50.0	50.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-Postmortem

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	8	80.0	80.0	80.0
	NEUTRAL	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-FormalTraining

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	STRONGLY DISAGREE	3	30.0	30.0	30.0
	DISAGREE	5	50.0	50.0	80.0
	NEUTRAL	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg- RewardIncentiveKnSharing

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	2	20.0	20.0	20.0
	NEUTRAL	8	80.0	80.0	100.0
	Total	10	100.0	100.0	

Descriptives

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SDevKnowldg-KMStrategy	10	3	4	3.70	.483
SDevKnowldg-Good Leadership	10	4	5	4.50	.527
SDevKnowldg-Postmortem	10	2	3	2.20	.422
SDevKnowldg-FormalTraining	10	1	3	1.90	.738
SDevKnowldg-RewardIncentive KnSharing	10	2	3	2.80	.422
Valid N (listwise)	10				

Means

Case Processing Summary

	Cases					
	Included		Excluded		Total	
	N	Percent	N	Percent	N	Percent
SDevKnowldg-KMStrategy * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SDevKnowldg-Good Leadership * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SDevKnowldg-Postmortem * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SDevKnowldg-Formal Training * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SDevKnowldg-RewardIncentiveKn Sharing * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%

Report

EmpSizeGroup		SDev Knowldg-KMStrategy	SDev Knowldg-Good Leadership	SDev Knowldg-Postmortem	SDev Knowldg-Formal Training	SDev Knowldg-Reward IncentiveKn Sharing
1-9	Mean	3.40	4.60	2.40	1.40	2.60
	N	5	5	5	5	5
	Std. Deviation	.548	.548	.548	.548	.548
10-24	Mean	4.00	4.40	2.00	2.40	3.00
	N	5	5	5	5	5
	Std. Deviation	.000	.548	.000	.548	.000
Total	Mean	3.70	4.50	2.20	1.90	2.80
	N	10	10	10	10	10
	Std. Deviation	.483	.527	.422	.738	.422

People and Management Commitment in KM

D3	e.	Management are very committed to sharing of knowledge and knowledge transfer activities.	***Mgmt and People Commitment***	***5***	***4***	***3***	***2***	***1***
D3	h.	Good working relationships between software development staff enhance knowledge sharing / transfer.		***5***	***4***	***3***	***2***	***1***
D3	s.	We regularly share opinions and thoughts on our software development activities.		***5***	***4***	***3***	***2***	***1***
D3	p.	We regularly share our knowledge in software development projects.		***5***	***4***	***3***	***2***	***1***

Frequencies

Statistics

		SDev Knowldg-Management Commitment	SDev Knowldg-Goodworking Relationship	SDev Knowldg-Regular Share Opinion& Thought	SDev Knowldg-Regular Share Sxperience
N	Valid	10	10	10	10
	Missing	0	0	0	0

Frequency Table

SDevKnowldg-ManagementCommitment

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	4	40.0	40.0	40.0
	AGREE	3	30.0	30.0	70.0
	STRONGLY AGREE	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-GoodworkingRelationship

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	4	40.0	40.0	40.0
	STRONGLY AGREE	6	60.0	60.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-RegularShareOpinion&Thought

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	6	60.0	60.0	60.0
	STRONGLY AGREE	4	40.0	40.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-RegularShareSxperience

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	1	10.0	10.0	10.0
	AGREE	7	70.0	70.0	80.0
	STRONGLY AGREE	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

Descriptives

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SDevKnowldg-Management Commitment	10	3	5	3.90	.876
SDevKnowldg-GoodworkingRelationship	10	4	5	4.60	.516
SDevKnowldg-Regular ShareOpinion&Thought	10	4	5	4.40	.516
SDevKnowldg-Regular ShareSxperience	10	3	5	4.10	.568
Valid N (listwise)	10				

Report

EmpSizeGroup		SDev Knowldg-Management Commitment	SDev Knowldg-Goodworking Relationship	SDev Knowldg-Regular Share Opinion& Thought	SDev Knowldg-Regular Share Sxperience
1-9	Mean	4.40	4.80	4.40	4.20
	N	5	5	5	5
	Std. Deviation	.894	.447	.548	.837
10-24	Mean	3.40	4.40	4.40	4.00
	N	5	5	5	5
	Std. Deviation	.548	.548	.548	.000
Total	Mean	3.90	4.60	4.40	4.10
	N	10	10	10	10
	Std. Deviation	.876	.516	.516	.568

Process Improvement and Assessment

C9	i.	The software process changes / evolves overtime	***Process Improve***	***5***	***4***	***3***	***2***	***1***
C9	k.	Management regularly assess software development process		***5***	***4***	***3***	***2***	***1***
C9	l.	When software processes are updated / changed, software developer always follow the new process.		***5***	***4***	***3***	***2***	***1***
D3	l.	We are follow an 'agile' type of software development methodology.		***5***	***4***	***3***	***2***	***1***

Frequencies

Statistics

		SWDev PracticeIn Cmmpany-SPIevolveovertime	SWDev PracticeIn Cmmpany-MgmtReg AccessSPI	SWDev PracticeIn Cmmpany-FllwNw Process	SDev Knowldg-AgileType development
N	Valid	10	10	10	10
	Missing	0	0	0	0

Frequency Table

SWDevPracticeInCmmpany- SPIevolveovertime

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	2	20.0	20.0	20.0
	AGREE	7	70.0	70.0	90.0
	STRONGLY AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- MgmtRegAccessSPI

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	1	10.0	10.0	10.0
	NEUTRAL	3	30.0	30.0	40.0
	AGREE	6	60.0	60.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- FllwNwProcess

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	2	20.0	20.0	20.0
	NEUTRAL	4	40.0	40.0	60.0
	AGREE	4	40.0	40.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-AgileType development

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	1	10.0	10.0	10.0
	AGREE	5	50.0	50.0	60.0
	STRONGLY AGREE	4	40.0	40.0	100.0
	Total	10	100.0	100.0	

Descriptives

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SWDevPracticeIn Cmmpany- SPIevolveovertime	10	3	5	3.90	.568
SWDevPracticeIn Cmmpany- MgmtRegAccessSPI	10	2	4	3.50	.707
SWDevPracticeIn Cmmpany- FllwNwProcess	10	2	4	3.20	.789
SDevKnowldg-Agile Type development	10	2	5	4.20	.919
Valid N (listwise)	10				

Means

Case Processing Summary

	Cases					
	Included		Excluded		Total	
	N	Percent	N	Percent	N	Percent
SWDevPracticeIn Cmmpany- SPIevolveovertime * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SWDevPracticeIn Cmmpany- MgmtRegAccessSPI * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SWDevPracticeIn Cmmpany- FllwNwProcess * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%
SDevKnowldg-Agile Type development * EmpSizeGroup	10	100.0%	0	.0%	10	100.0%

Report

EmpSizeGroup		SWDev PracticeIn Cmmpany-SPIevolveovertime	SWDev PracticeIn Cmmpany-MgmtReg AccessSPI	SWDev PracticeIn Cmmpany-FllwNw Process	SDev Knowldg-AgileType development
1-9	Mean	3.80	3.40	3.20	4.20
	N	5	5	5	5
	Std. Deviation	.447	.894	.837	.447
10-24	Mean	4.00	3.60	3.20	4.20
	N	5	5	5	5
	Std. Deviation	.707	.548	.837	1.304
Total	Mean	3.90	3.50	3.20	4.20
	N	10	10	10	10
	Std. Deviation	.568	.707	.789	.919

People Involvement and Management Involvemet

C8	h.	Software development staff are directly involved in planning and improving software development processes	***Involvement***	***5***	***4***	***3***	***2***	***1***
C9	g.	Software developers have freedom in planning and managing their work.		***5***	***4***	***3***	***2***	***1***
C9	h.	Software development staffs are actively involved in setting goals for SPI activities.		***5***	***4***	***3***	***2***	***1***
C9	j.	Software development staff are actively involved in creating process and procedure for software development		***5***	***4***	***3***	***2***	***1***

C8	d.	Software development staff regularly receives guidance and support from management.	***Mgmt Involvement***	***5***	***4***	***3***	***2***	***1***
C8	b.	Software development staff are highly motivated.		***5***	***4***	***3***	***2***	***1***
C8	c.	Software development staff receive recognition for their work		***5***	***4***	***3***	***2***	***1***
C9	e.	Senior management actively supports SPI activities.		***5***	***4***	***3***	***2***	***1***

SWDevPracticeInCmmpany - TeamDrctInvolvemt

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	1	10.0	10.0	10.0
	AGREE	5	50.0	50.0	60.0
	STRONGLY AGREE	4	40.0	40.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- StaffAutonomouswork

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	7	70.0	70.0	70.0
	STRONGLY AGREE	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- StaffactivesetSPIgl

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	4	40.0	40.0	40.0
	AGREE	6	60.0	60.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- StaffDirectlyInvSPI

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	1	10.0	10.0	10.0
	AGREE	8	80.0	80.0	90.0
	STRONGLY AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SWDevPracticeInCmmpany - TeamDrctInvolvemt	10	3	5	4.30	.675
SWDevPracticeInCmmpany- StaffAutonomouswork	10	4	5	4.30	.483
SWDevPracticeInCmmpany- SPIevolveovertime	10	3	5	3.90	.568
SWDevPracticeInCmmpany- StaffDirectlyInvSPI	10	3	5	4.00	.471
Valid N (listwise)	10				

Report

EmpSizeGroup		SWDevPracticeInCmmpany - TeamDrctInvolvemt	SWDevPracticeInCmmpany- StaffAutonomouswork	SWDevPracticeInCmmpany- StaffactivesetSPIgl	SWDevPracticeInCmmpany- StaffDirectlyInvSPI
1-9	Mean	4.20	4.20	3.80	4.00
	N	5	5	5	5
	Std. Deviation	.837	.447	.447	.000
10-24	Mean	4.40	4.40	3.40	4.00
	N	5	5	5	5
	Std. Deviation	.548	.548	.548	.707
Total	Mean	4.30	4.30	3.60	4.00
	N	10	10	10	10
	Std. Deviation	.675	.483	.516	.471

SWDevPracticeInCmmpany - GuideSupport

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	8	80.0	80.0	80.0
	STRONGLY AGREE	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- HighSuppHgMgmt

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	2	20.0	20.0	20.0
	NEUTRAL	6	60.0	60.0	80.0
	AGREE	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany - HighMotivated

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	1	10.0	10.0	10.0
	AGREE	2	20.0	20.0	30.0
	STRONGLY AGREE	7	70.0	70.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany - RecvdRecog

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	1	10.0	10.0	10.0
	AGREE	5	50.0	50.0	60.0
	STRONGLY AGREE	4	40.0	40.0	100.0
	Total	10	100.0	100.0	

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SWDevPracticeInCmmpany - GuideSupport	10	4	5	4.20	.422
SWDevPracticeInCmmpany - HighMotivated	10	2	5	4.50	.972
SWDevPracticeInCmmpany - RecvdRecog	10	2	5	4.20	.919
SWDevPracticeInCmmpany- HighSuppHgMgmt	10	2	4	3.00	.667
Valid N (listwise)	10				

Report

EmpSizeGroup		SWDevPracticeInCmmpany - GuideSupport	SWDevPracticeInCmmpany - HighMotivated	SWDevPracticeInCmmpany - RecvdRecog	SWDevPracticeInCmmpany- HighSuppHgMgmt
1-9	Mean	4.20	4.80	4.40	3.00
	N	5	5	5	5
	Std. Deviation	.447	.447	.548	.707
10-24	Mean	4.20	4.20	4.00	3.00
	N	5	5	5	5
	Std. Deviation	.447	1.304	1.225	.707
Total	Mean	4.20	4.50	4.20	3.00
	N	10	10	10	10
	Std. Deviation	.422	.972	.919	.667

Goal and Planning

C9	a.	We have established a Software Process Improvement (SPI) goals.	***Goal and Planning***	***5***	***4***	***3***	***2***	***1***
C9	b.	There is a broad understanding of SPI goals and policy within our organization.		***5***	***4***	***3***	***2***	***1***
C9	c.	Our SPI goals are closely aligned with organizational business goals.		***5***	***4***	***3***	***2***	***1***
C9	d.	We have a good balance between short term and long tem SPI goals.		***5***	***4***	***3***	***2***	***1***
E4	a.	Software development staffs always understand projects goals.		***5***	***4***	***3***	***2***	***1***

SWDevPracticeInCmmpany -EstablishSPIGoal

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	STRONGLY DISAGREE	2	20.0	20.0	20.0
	DISAGREE	1	10.0	10.0	30.0
	NEUTRAL	4	40.0	40.0	70.0
	AGREE	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany - BrdUndstdSPIGoal

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	3	30.0	30.0	30.0
	NEUTRAL	7	70.0	70.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- SPIgoalAllignBussG

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	5	50.0	50.0	50.0
	NEUTRAL	2	20.0	20.0	70.0
	AGREE	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- BalShrtLongTermSPI

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	4	40.0	40.0	40.0
	NEUTRAL	5	50.0	50.0	90.0
	AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

SDevTeam-UnderstandGoal

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	6	60.0	60.0	60.0
	STRONGLY AGREE	4	40.0	40.0	100.0
	Total	10	100.0	100.0	

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SWDevPracticeInCmmpany - EstablishSPIGoal	10	1	4	2.80	1.135
SWDevPracticeInCmmpany - BrdUndstdSPIGoal	10	2	3	2.70	.483
SWDevPracticeInCmmpany- SPIgoalAllignBussG	10	2	4	2.80	.919
SWDevPracticeInCmmpany- BalShrtLongTermSPI	10	2	4	2.70	.675
SDevTeam- UnderstandGoal	10	4	5	4.40	.516
Valid N (listwise)	10				

Report

EmpSizeGroup		SWDevPracticeInCmmpany - EstablishSPIGoal	SWDevPracticeInCmmpany - BrdUndstdSPIGoal	SWDevPracticeInCmmpany- SPIgoalAllignBussG	SWDevPracticeInCmmpany- BalShrtLongTermSPI	SDevTeam- UnderstandGoal
1-9	Mean	2.00	2.60	2.60	2.60	4.60
	N	5	5	5	5	5
	Std. Deviation	1.000	.548	.894	.548	.548
10-24	Mean	3.60	2.80	3.00	2.80	4.20
	N	5	5	5	5	5
	Std. Deviation	.548	.447	1.000	.837	.447
Total	Mean	2.80	2.70	2.80	2.70	4.40
	N	10	10	10	10	10
	Std. Deviation	1.135	.483	.919	.675	.516

Development Standard Process and Quality Standard

	a.	Not required by customers.		***5***	***4***	***3***	***2***	***1***
	b.	Lack support from management.		***5***	***4***	***3***	***2***	***1***
	c.	Lack support from government/state agencies		***5***	***4***	***3***	***2***	***1***
	d.	Lack of internal resources.		***5***	***4***	***3***	***2***	***1***
	e.	Lengthy and difficult in implementation standard		***5***	***4***	***3***	***2***	***1***
	f.	Insufficient time to implement standard.		***5***	***4***	***3***	***2***	***1***
	g.	Insufficient guidance available.		***5***	***4***	***3***	***2***	***1***
	h.	No desire to change existing processes and procedures.		***5***	***4***	***3***	***2***	***1***

C9	f.	We are regularly consult software quality standards in performing our job	***Standard***	***5***	***4***	***3***	***2***	***1***
C9	n.	Management team are knowledgeable about software quality standards		***5***	***4***	***3***	***2***	***1***
C9	o.	Software development staff are knowledgeable about software quality standards		***5***	***4***	***3***	***2***	***1***
C9	p.	Management ensure that organisational standard processes are always followed by software developers.		***5***	***4***	***3***	***2***	***1***

ReasonNotAdaptStandard - NotReqCustomer

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	3	30.0	30.0	30.0
	STRONGLY AGREE	7	70.0	70.0	100.0
	Total	10	100.0	100.0	

ReasonNotAdaptStandard - LackSupportMgmt

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	STRONGLY DISAGREE	3	30.0	30.0	30.0
	DISAGREE	1	10.0	10.0	40.0
	NEUTRAL	3	30.0	30.0	70.0
	AGREE	2	20.0	20.0	90.0
	STRONGLY AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

ReasonNotAdaptStandard - LackSpptGoven

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	10	100.0	100.0	100.0

ReasonNotAdaptStandard - LackinternalRcsr

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	3	30.0	30.0	30.0
	AGREE	5	50.0	50.0	80.0
	STRONGLY AGREE	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

ReasonNotAdaptStandard - LengthnDiffStd

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	3	30.0	30.0	30.0
	AGREE	4	40.0	40.0	70.0
	STRONGLY AGREE	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

ReasonNotAdaptStandard - InsuffTime

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	4	40.0	40.0	40.0
	AGREE	3	30.0	30.0	70.0
	STRONGLY AGREE	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

ReasonNotAdaptStandard - InsuffGuidance

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	5	50.0	50.0	50.0
	AGREE	3	30.0	30.0	80.0
	STRONGLY AGREE	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

ReasonNotAdaptStandard - NoDesire

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	4	40.0	40.0	40.0
	AGREE	4	40.0	40.0	80.0
	STRONGLY AGREE	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
ReasonNotAdaptStandard - NotReqCustomer	10	4	5	4.70	.483
ReasonNotAdaptStandard - LackSupportMgmt	10	1	5	2.70	1.418
ReasonNotAdaptStandard - LackSpptGoven	10	3	3	3.00	.000
ReasonNotAdaptStandard - LackinternalRcsr	10	3	5	3.90	.738
ReasonNotAdaptStandard - LengthnDiffStd	10	3	5	4.00	.816
ReasonNotAdaptStandard - InsuffTime	10	3	5	3.90	.876
ReasonNotAdaptStandard - InsuffGuidance	10	3	5	3.70	.823
ReasonNotAdaptStandard - NoDesire	10	3	5	3.80	.789
Valid N (listwise)	10				

Report

EmpSize Group		ReasonNotAdaptStandard - NotReqCustomer	ReasonNotAdaptStandard - LackSupportMgmt	ReasonNotAdaptStandard - LackinternalRcsr	ReasonNotAdaptStandard - LackSpptGoven	ReasonNotAdaptStandard - LengthnDiffStd	ReasonNotAdaptStandard - InsuffTime	ReasonNotAdaptStandard - InsuffGuidance	ReasonNotAdaptStandard - NoDesire
1-9	Mean	5.00	2.40	3.80	3.00	4.00	4.00	3.40	3.80
	N	5	5	5	5	5	5	5	5
	Std. Deviation	.000	1.949	.837	.000	.707	1.000	.548	1.095
10-24	Mean	4.40	3.00	4.00	3.00	4.00	3.80	4.00	3.80
	N	5	5	5	5	5	5	5	5
	Std. Deviation	.548	.707	.707	.000	1.000	.837	1.000	.447
Total	Mean	4.70	2.70	3.90	3.00	4.00	3.90	3.70	3.80
	N	10	10	10	10	10	10	10	10
	Std. Deviation	.483	1.418	.738	.000	.816	.876	.823	.789

SWDevPracticeInCmmpany- ConsultStdinSPI

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	STRONGLY DISAGREE	5	50.0	50.0	50.0
	DISAGREE	1	10.0	10.0	60.0
	NEUTRAL	3	30.0	30.0	90.0
	AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- MgmtKnQualtyStad

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	2	20.0	20.0	20.0
	NEUTRAL	4	40.0	40.0	60.0
	AGREE	3	30.0	30.0	90.0
	STRONGLY AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- StafftKnQualtyStad

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	2	20.0	20.0	20.0
	NEUTRAL	5	50.0	50.0	70.0
	AGREE	2	20.0	20.0	90.0
	STRONGLY AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

SWDevPracticeInCmmpany- StaffFlwNwProcess

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	1	10.0	10.0	10.0
	NEUTRAL	2	20.0	20.0	30.0
	AGREE	5	50.0	50.0	80.0
	STRONGLY AGREE	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SWDevPracticeInCmmpany-ConsultStdinSPI	10	1	4	2.00	1.155
SWDevPracticeInCmmpany-MgmtKnQualtyStad	10	2	5	3.30	.949
SWDevPracticeInCmmpany-StafftKnQualtyStad	10	2	5	3.20	.919
SWDevPracticeInCmmpany-StaffFlwNwProcess	10	2	5	3.80	.919
Valid N (listwise)	10				

Report

EmpSizeGroup		SWDevPracticeInCmmpany-ConsultStdinSPI	SWDevPracticeInCmmpany-MgmtKnQualtyStad	SWDevPracticeInCmmpany-StafftKnQualtyStad	SWDevPracticeInCmmpany-StaffFlwNwProcess
1-9	Mean	1.60	3.40	3.00	3.80
	N	5	5	5	5
	Std. Deviation	.894	1.140	.707	1.095
10-24	Mean	2.40	3.20	3.40	3.80
	N	5	5	5	5
	Std. Deviation	1.342	.837	1.140	.837
Total	Mean	2.00	3.30	3.20	3.80
	N	10	10	10	10
	Std. Deviation	1.155	.949	.919	.919

Team Dynamic

D3	h.	Good working relationships between software development staff enhance knowledge sharing / transfer.	***Dynamic***	***5***	***4***	***3***	***2***	***1***
D3	s.	We regularly share opinions and thoughts on our software development activities.		***5***	***4***	***3***	***2***	***1***
E4	c.	There is a good social relationship among software development staff		***5***	***4***	***3***	***2***	***1***
E4	d.	Software development staffs have good interpersonal skills.		***5***	***4***	***3***	***2***	***1***
E4	h	Software developers are physically located close to each other (office layout)		***5***	***4***	***3***	***2***	***1***

SDevKnowldg-GoodworkingRelationship

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	4	40.0	40.0	40.0
	STRONGLY AGREE	6	60.0	60.0	100.0
	Total	10	100.0	100.0	

SDevKnowldg-RegularShareOpinion&Thought

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	6	60.0	60.0	60.0
	STRONGLY AGREE	4	40.0	40.0	100.0
	Total	10	100.0	100.0	

SDevTeam-GoodSocialRelationship

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	6	60.0	60.0	60.0
	STRONGLY AGREE	4	40.0	40.0	100.0
	Total	10	100.0	100.0	

SDevTeam-GoodInterpersonalSkill

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	AGREE	7	70.0	70.0	70.0
	STRONGLY AGREE	3	30.0	30.0	100.0
	Total	10	100.0	100.0	

SDevTeam-CloselyLocated

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	1	10.0	10.0	10.0
	STRONGLY AGREE	9	90.0	90.0	100.0
	Total	10	100.0	100.0	

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SDevKnowldg-GoodworkingRelationship	10	4	5	4.60	.516
SDevKnowldg-RegularShareOpinion&Thought	10	4	5	4.40	.516
SDevTeam-GoodSocialRelationship	10	4	5	4.40	.516
SDevTeam-GoodInterpersonalSkill	10	4	5	4.30	.483
SDevTeam-CloselyLocated	10	2	5	4.70	.949
Valid N (listwise)	10				

Report

EmpSize Group		SDevKnowldg-Goodworking Relationship	SDevKnowldg-RegularShare Opinion&Thought	SDevTeam-GoodSocialRelationship	SDevTeam-GoodInterpersonalSkill	SDevTeam-CloselyLocated
1-9	Mean	4.80	4.40	4.80	4.40	4.40
	N	5	5	5	5	5
	Std. Deviation	.447	.548	.447	.548	1.342
10-24	Mean	4.40	4.40	4.00	4.20	5.00
	N	5	5	5	5	5
	Std. Deviation	.548	.548	.000	.447	.000
Total	Mean	4.60	4.40	4.40	4.30	4.70
	N	10	10	10	10	10
	Std. Deviation	.516	.516	.516	.483	.949

Team Structure

E4	e.	Clear roles are defined within the software development team	***Structure***	***5***	***4***	***3***	***2***	***1***
E4	f.	The software development team is an appropriate size for projects		***5***	***4***	***3***	***2***	***1***
E4	i.	There is a diverse range of skills among the software developers.		***5***	***4***	***3***	***2***	***1***

SDevTeam-ClearRoles

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	4	40.0	40.0	40.0
	AGREE	6	60.0	60.0	100.0
	Total	10	100.0	100.0	

SDevTeam-ApropriateSize

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	DISAGREE	1	10.0	10.0	10.0
	NEUTRAL	7	70.0	70.0	80.0
	STRONGLY AGREE	2	20.0	20.0	100.0
	Total	10	100.0	100.0	

SDevTeam-DiverseSkillRange

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	NEUTRAL	3	30.0	30.0	30.0
	AGREE	6	60.0	60.0	90.0
	STRONGLY AGREE	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
SDevTeam-ClearRoles	10	3	4	3.60	.516
SDevTeam-ApropriateSize	10	2	5	3.30	.949
SDevTeam-DiverseSkillRange	10	3	5	3.80	.632
Valid N (listwise)	10				

Report

EmpSizeGroup		SDevTeam-ClearRoles	SDevTeam-ApropriateSize	SDevTeam-DiverseSkillRange
1-9	Mean	3.60	3.20	3.60
	N	5	5	5
	Std. Deviation	.548	1.095	.548
10-24	Mean	3.60	3.40	4.00
	N	5	5	5
	Std. Deviation	.548	.894	.707
Total	Mean	3.60	3.30	3.80
	N	10	10	10
	Std. Deviation	.516	.949	.632

APPENDIX J:
Survey Questionnaire: Qualitative Analysis

Company	Company Size	B1 : HiringNewStaffCriteria	
Company AD	0	Problem solving skills, communication skills, experience, reference and knowledge of specific technologies that are required	Good Technical Skill, Hardwork and adaptable stress, Good Communication Skill (Written and Verbal), Reasonable Experience and related Qualification,
Company D	0	High Skill in IT/Having reasonable business knowledge	
Company AG	0	technical Ability, Communication skill, ability to interact with customers	
Company AG	0	Technical Knowledge, how well they ' fit in'	
Company AH	0	Problem Solving, Technical Skills, Attitude/approach	
Company Q	1	Technical Skill, ability to work alone, good English, good communication (verbal and written)	
Company L	1	Basic stuff, smart, quick on the uptake, willing to work hard and late if required. Also knowledge of the dev tools that KBS use. VB + SQL.	
Company P	1	High Skill, Reasonable qualification, Communication Skill, reasonable experience	
Company T	1	Degree, interview Assessment communication, technical skill and work in team	
Company AC	1	Good Degree, Communication Skill, Quick Learning, Communication skills, adaptable stress, hard work	

Company	Company Size	(B2 FormalTraingStatus)	B2aYes/TypeofTrainingandWhy	B2b No/NoFormalTraining
Company AD	0	0		No requirement for same has been identified to date. We have some simple docs defining coding practices and the tools we use but from there we manage/train through feedback as progress made
Company D	0	0		We are doing more on self learning and sharing among us/Using same technology and process for our project/Money Constraint
Company AG	0	0		On job training, mentoring
Company AG	0	0		We are too small to need to do that
Company AH	0	0		Myself working with a single employee, so training was on the job as and when required
Company Q	1	1	our framework, Microsoft exam	
Company L	1	0		They are expected to "Pick it up as they go". Typically new
Company P	1	0		Individual training given. Very small company therefore small number of developers.
Company T	1	0		On job training
Company AC	1	0		Internal, self learning and OJT
		90% - VSE not provide formal training	Concentrate to Technical/ Specific requirement	Self Learning and Sharing/ Self Review Standard Documentation and tool/ Informal Feedback/ Internal Training/ ' OJT' --- Reason Cost, small team, standard technology and process

Company	Company Size	B3 EmpTurnOverate	B3b No/NotProblem
Company AD	0	0	We handle many varying projects of different sizes and complexities and have a very loose/informal and friendly atmosphere. This means the work is challenging and rarely gets boring while it also being enjoyable here
Company D	0	0	All of us are basically the owner of the company and this is a family business
Company AG	0	0	We are too small for this to be an issue, only 4 of us, no body had ever left
Company AG	0	0	We are a young company, so nobody has left us to date. One person went part time to look after her children in afternoons, so i guess being flexible helps
Company AH	0	0	Small no of people
Company Q	1	0	Fun Culture, Sense of Ownership
Company L	1	0	14 employees. Last one in was 3 yrs ago. We operate a relax and informal environment. We also pay top percentage
Company P	1	0	Direct involvement, family and flexible environment, good team environment
Company T	1	0	Good place to work
Company AC	1	0	Family and Flexible Environment, Good Place to work
		100 % - no problem with staff turn over	Informal culture/ environment, Open and casual environment, Macro PM, Family and flexible environment, Direct Involvement, Family business,, Autonomous work and communication

Company	Company Size	(C1 SoftwareDevProcessStatus)	C1ai Yes/ MotivationChanges	C1aii ProcessChangeDocStatus	C1bi no/ MotivationChanges
Company AD	0	1	Changing requirements due to growth of the organization. We started out as 2 people 4 years ago and now have 11, so things had to change along the way	No. We aren't too good on process documentation.	
Company D	0	1	Basically yes , not evolve/changes due to the same technology and process we applied. Not aware on development process	Document only the important work	
Company AG	0	1	need to meet customer needs, technology change and we need to keep up		
Company AG	0	0			Not really. we still do the same basic things, we change some aspects of how we works. Its a little bit adhoc.. agile i suppose
Company AH	0	1	it will evolve as we grow in size and get more applications in production environment	No - simple documentation	
Company Q	1	1	Best Practice	yes - specific work	
Company L	1	1	typically our larger client impose some new method/ approach upon our organization. If it has merit and provides internal improvement then we will adopt it in the long term	No - we implement with minimal documentation	
Company P	1	1	to improve current development process and meet customer requirement	yes - Documented	

Company T	1	1	staff suggestion, ISO9000, constant learning	Yes- document specific work	
Company AC	1	1	Client related, improving current process	No - we implement informal document	
		90% feel that their development process are improve and evolve (small scale)	Follow the best Practice'/ New method and approach/ Client requirement/Team Size growth/ New idea/ Follow ISO Standard -- most small scale and informal	Formal Documented - specific issue, important work, specific work/ (Others -Informal Document or Minimal Document - Others)	Follow 'Agile' Methodology

Company	Company Size	C2 ReviewUpdateProjDocStatus	C2a Yes/ActivitiesDone	C2b No/HowActivitiesDone
Company AD	0	0		If we get feedback from a customer after they have seen a milestone that requires changes to requirements that further affects the timeline, then we'll review the doc. Otherwise they stay pretty static and are used as the touchstone for the project definitions
Company D	0	1	We documented it electronically, and having an equal decision on it	
Company AG	0	1	Informal manner, Just keep an eye on these things every now and again. We looks at things if something goes wrong or there is a major problem	
Company AG	0	0		We are too small to need to document
Company AH	0	0		
Company Q	1	1	Daily update of time spends on coding/testing. Weekly Project reporting	
Company L	1	1	If by project documents you mean project and work specifications then yes we would. Constant updates until such time as the represent the change /work to be complete	
Company P	1	1	Following Release	
Company T	1	0		Ad hoc work
Company AC	1	0		Ad Hoc basis
		50% - felt they update their document regularly	Regular update on specific work and procedure only	Periodic basic/ Minimal Update/ Adhoc job/ Informal process

Company	Company Size	C3 ProcessLossStatus	C3a Yes/CauseHowofProcessLoss	C3b No/HowtoHandleProcessLoss
Company AD	0	0		As a manager, I don't believe in using the latest and greatest techniques for the sake of it. We'll use something that fits our team dynamics and we'll spurn something that doesn't. Whether that count
Company D	0	0		document mostly electronically /Always sharing knowledge informally - since this is family business/Having informal regular meeting
Company AG	0	0		I trust people to do the thing right. If i don't agree with them then i tell them what to do and they will do it. I am a manager after all
Company AG	0	0		Not really. we still do the same basic things, we change some aspects of how we works. Its a little bit adhoc.. agile i suppose
Company AH	0	0		Single source of document for current process
Company Q	1	1	Laziness, Proactive coaching and evaluation	

Company L	1	1	Our development process is informal and we tend to follow RAD development	
Company P	1	0		knowledge sharing and teamwork
Company T	1	0		Guidance, mentoring
Company AC	1	0		Regular sharing process, internal sharing and team work
		80% - felt they don't have a process loss problem	Cause- Laziness, Informal or rapid development process change / informal changes	Using standard technology and process - team dynamic/, Guidance, mentoring, sharing and team work/ proactive coaching

Company	Company Size	C4 AccreditationExtStandStatus	C4bi PlanforCertificationinFutureStatus	C4bia Yes/Reason
Company AD	0	0	0	
Company D	0	0	0	
Company AG	0	0	0	
Company AG	0	0	0	
Company AH	0	0	1	improve managing IT infrastructure development and operation
Company Q	1	0	1	Best Practice required for large corporate and good tender
Company L	1	0	0	
Company P	1	0	0	
Company T	1	0	1	Improve process, quality
Company AC	1	0	1	Improve process and Public tender
		100% - Not accreditate by any quality standard currently	40% - Plan to obtain Quality Standard/ 60% don't acquire it	Best Practice/ Public tender/ Improve process and quality- (VSE is aware of standard benefit)

Company	Company Size	C4bib NO/Reason	C5i TypeofCertification	C5ii Assistance Needed
Company AD	0	Because there has been market demand for it. We trade on our reputation in the small to medium sized project space. These clients do not look for any standards accreditation, more a proven track record	No plan	No idea
Company D	0	No interest on ISO because not required by customer. More interested on technical. Being aware ISO is for public sector. No knowledgeable about ISO, heavy weight	ISO9000	Lightweight Process, related to technical, not high overhead, follow current guideline
Company AG	0	Probably Not	No plan in place currently	
Company AG	0	i don't think we could	No plan in place currently	
Company AH	0		'ITIL' - Based on ISO standard	Any assistance useful (- financial to cover learning cost)
Company Q	1		ISO9000	Training
Company L	1	No cost benefit to us	No plan	No idea
Company P	1	No plan and interest	No plan and interest	No idea
Company T	1		ISO9000	External expert

Company AC	1		ISO9000	Less Overhead, Template and Training
		No important(benefit)/ No demand from market/Not required by client/ More interest -technical standard/ Less knowledgeable/ No interest	50% - look forward related ISO9000	Training / lightweight process/technical related/Less Overhead/ align current process/ template/ External expert guide

Company	Company Size	D1 SWDevTeamKnowledgeStatus	D1b Explain ManageKnowledge
Company AD	0	0	Informally through ad-hoc conversations and some code review
Company D	0	0	informal sharing knowledge/Changing/documenting electronically/Informal transferring and knowledge sharing
Company AG	0	0	We all talk all the time about the work we do, the problem we have and what solutions we can use. People must talk all the times to exchange information
Company AG	0	0	We talk a lot and we use email as well. We did look at using blog, but that didn't really work out well for us
Company AH	0	0	Central shared area for all development -related document and procedures and it done informally
Company Q	1	0	depends to Team and the required doc
Company L	1	0	we operate informally
Company P	1	0	Informal documented, no standard applied, staff initiative
Company T	1	0	Adhoc
Company AC	1	0	Informally through ad-hoc conversations, sharing and some code review
		100%- not formal manage sw team knowledge	Informal and personal documentation process/Code review(code documentation)/Informal transfer/ impromptu (adhoc)

Company	Company Size	D2 KnowledgeLoss HappenStatus	D2a Yes/KL address	D2b No/ Plan if Arose
Company AD	0	0		Ensuring that no single member of staff has any exclusive knowledge by using a mentoring/buddy system.
Company D	0	0		Not a problem since we using same technology and process in all our project. /Occasionally sharing and transferring knowledge among brothers
Company AG	0	0		Ensuring everyone talks and exchanges information about projects on an ongoing basis we can mitigate against leaving the company or forgetting knowledge
Company AG	0	0		i think we keep fresh with what happening in the market all the time, so we gain and not lose knowledge
Company AH	0	0		Formal documentation of procedure via wiki/confluence page
Company Q	1	0		Roll up the sleeves and get stuck into the problem
Company L	1	1	Another developer will take over the code base previously assigned. Some business support for management	
Company P	1	0		Documentation/training prior to staff leave
Company T	1	0		knowledge sharing, document
Company AC	1	0		Informally through ad-hoc conversations, sharing and some code review

only 10 % - face knowledge loss problem	due to less/informal documentation and people leaving. Team and management support	team work/ knowledge sharing activities/ mentoring and pair system/ documentation/ regular communication

Company	Company Size	E1 : SDevTeam- RegReceivedFdback	E1a : FdbackDelivered	E1b : Fdbackbenefit
Company AD	0	0		Less risk of delivering the wrong product that doesn't satisfy the business requirements.
Company D	0	1	online communication, informal feedback, internal discussion, informal communication	
Company AG	0	1	We are sit in one office so i talk to them all the time	
Company AG	0	1	Customer will always say if there is a problem and we find out quick. It is because we are small and we listen to our customers.	
Company AH	0	1	Informal conversation	
Company Q	1	1	Show and Tell, We show to the customer and tell us what they want	
Company L	1	1	The odd " well done mail"	
Company P	1	1	Meeting and documentation	
Company T	1	1	Status report, comment	
Company AC	1	1	Informal meeting, casual communication, report	
		90% - claimed that development project get feedback regularly	Informal discussion, online communication, internal feedback - internally / specific documentation and status report - related to client	Improve product delivery

Company	Company Size	E2 ;SDevTEaminvolvedDefingSdevprocess	E2a ; Yes/Explained	E2b ; No/Explained
Company AD	0	1	Explicit requests for input and feedback on any thoughts/ideas for changing the way things are done	
Company D	0	1	Having equal right in companies share and decision/Having active sharing knowledge in development process/Teams are companies main stakeholder	
Company AG	0	1	i welcome input from developers on what we are doing and how to make it better	
Company AG	0	1	We do things the way we always did them in the main, but when things don't workout we change and do what best.	
Company AH	0	1	Direct on the best way to develop, easiest processes	

Company Q	1	1	100% Owner Ship with Mgmt	
Company L	1	0		This is mainly down how the client want it developed
Company P	1	1	Meeting following release to determine if the process can be improved in any way for future release	
Company T	1	1	Review/Create	
Company AC	1	1	Specific input and idea in managing problem or planning	
		100% - admit that team involved directly in defining software development process	High sense of ownership, equal right, actively sharing , frequent review and create	Depend on client requirement

Company	Company Size	E3 :TeamStatus	E3a Yes /Explain	E3b Howto create effectiveand efficint team
Company AD	0	1	They gel well as a social group and communicate regularly/openly. Also the projects we manage are normally 1 to 2 man projects and hence easily manage in an adhoc manner by two people that get on and communicate well	Pay attention to culture and ensure you foster an open/friendly environment where mistakes are declared and discussed with an eye to avoiding them in the future as opposed to blaming those responsible. Bugs happen, it's just a case of discussing why and trying to minimize the risk of re-occurrence.
Company D	0	1	Clear communication, active informal knowledge sharing , family culture, following specific strategic planning, actively using communication tools	active and open communication- directly and indirectly/having good interaction between team
Company AG	0	1	They all work well together. we have a friendly atmosphere and a good work ethic and culture	Its all about people you have to have the right people in the first place and then trust them to get on with the job
Company AG	0	1	We are all up to date with the tool and technology. Knowledge and ability is key	
Company AH	0	1	Yes but always could improve	Regular face to face discussion, out of office activity to bong team members
Company Q	1	1	They are highly skilled and motivated, great team atmosphere	Great Team atmosphere and highly skill and motivated
Company L	1	1	Motivated by being the best at we do and money.	
Company P	1	1	Good teamwork, clear communication, dynamics and good relationship	flexible, freedom and direct involvement
Company T	1	1	Project On time, On budget	good People and processes
Company AC	1	1	High Motivated and Clear Communication	Great Team, Team work and clear Communication
		100% - claim their development teams are efficient and effective	high skill and motivated/ dynamics, socialize, good teamwork and family culture/ Regular and open communicate/ Meet deadline and budget/ Have strategic planning and active knowledge sharing	good team work/ highly team motivated/open and friendly environment/ open interaction / direct involvement in planning and decision