ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

EXPERIENCE BASED FACTORY MODEL FOR SOFTWARE DEVELOPMENT PROCESS: ITEM CONSTRUCT VALIDATION ON QUESTIONNAIRE DESIGN

[1]MASTURA HANAFIAH, [2]RUSLI ABDULLAH, [3]MASRAH AZRIFAH AZMI MURAD, [4]JAMILAH DIN, [5]MOHD ZALI MOHD NOR

Faculty of Computer Science and Information Technology, University Putra Malaysia, Serdang, 43400 Selangor, MALAYSIA.

E-mail: [1]mastura.hanafiah@daimler.com, [2]rusli@upm.edu.my, [3]masrah@upm.edu.my, [4]jamilahd@upm.edu.my, [5]mohd.zali@my-newstar.com

ABSTRACT

Software development is a highly intensive knowledge process. Information, data, knowledge and experiences are accumulated daily and it is crucial that they are managed appropriately for the purpose of sharing and future reuse. Today, software development has spread across geographical boundaries; therefore, the need for knowledge retention has risen, and the need for collaboration among the community of practice has been further in demand. Based on this motivation, we posit a model of managing the experiences of software development process by using experience based factory approach. An initial conceptual model has been constructed based on relevant theoretical frameworks which include knowledge management, experience factory, software development process, community of practice, technology and infrastructure, and influences from managerial and organizational levels. Based on the literature review, questionnaire items have been designed to form the identified latent constructs. A pilot survey has been conducted to verify the questionnaire items and the results are tested against Rasch measurement analysis. By using Rash logit measures, the items' quality is ensured. Findings indicate that the item fitness is good (0.73), outfit and infit mean square values are very much close to 1, and Z-standardized value is within the expected range. Unidimensionality shows that there is no visible secondary dimension even though the scale category structure is rather high. Nonetheless, there are several misfitting items which are further calibrated and revised for future study. The initial model will be the basis of the future model development.

Keywords: *Experience Factory, Knowledge Management, Software Development Process, Questionnaire Design, Rasch Analysis*

1. INTRODUCTION

The advancement of digital world has triggered more competition among software organizations. From traditional desktop to web applications, from normal web to mobile applications, and from mobile to augmented realities, software development has become a prominent industry. Software development involves a lot processes, activities, artefacts and multiple roles. Instinctively, software development accumulates data, information and knowledge each day especially in a fast and productive working environment. They eventually become valuable experiences for both the organizations and the employee themselves. Experiences and knowledge are transferred either from one form to another, or from one person to another. By some means, the people involved in the activities or events gain experiences either through observations, or emotions with respect to the observed event, or making conclusions or hypotheses derived from that observed event and emotion [1]. The accumulation of experiences and knowledge however creates issues for organizations: knowledge are difficult to transfer, errors occur due to lack of knowledge, critical knowledge are only in the hands of few people, unable to perform measurement related to knowledge use, relevant knowledge are lost at critical time, and lack of knowledge sharing process [2].

A recent literature review analysis reveals that there are issues in managing experience and knowledge in software organizations in terms of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

knowledge transfer and information flow, explicit and tacit knowledge management, and misinterpretation and inconsistencies, mainly in global software development [3]. There are also challenges in leveraging experiences as they are often localized to individuals or teams [4]; they are not shared due to inefficiency of communication, diverging cultures, high complexity and inefficiency of project management [5]; and due to complex context specific knowledge resulted from different technologies, needs and expectation of users [6]. Moreover, it is frequent that the contents from sources such as emails, phones and personal meetings are not documented or shared to other team members [7]. Tacit knowledge, additionally, is more difficult to articulate due to cultural and experiential differences that may affect team relationship and successful communication. This could result to unsuccessful knowledge transfer and eventually lead to difficulties, delays or failures [8]. Furthermore, the differences in background, culture, education, terminology, practices, and standards being used in multi-side development always lead to inconsistencies and misinterpretation [9][10].

This paper aims to contribute the initial development of the model formulation for managing experiences and knowledge in software development process based on experience factory (EF) approach. The strength on EF is on the reuse of life cycle experiences, processes and products particularly for software development. Motivation on EF comes from the need to build successful software products by initially fulfilling the basic requirements: first, by understanding the process and product; second, by defining business needs; third, by evaluating success and failures; fourth, by having closed-loop process with feedback process; fifth, by learning from experiences; and finally, by packaging and reusing experiences for future benefits [11].

In this study, several concepts have been taken into the consideration as the main components of the model development. The model, known as Experience Based Factory for Software Development Process (EBF-SD), has undergone a survey research to seek experts' agreement on the proposed components. A set of questionnaire is used as the instrument. The questionnaire items are analyzed by using Rasch measurement model [13] in terms of logit measures whether they corroborate to the underlying theories or possibly agreeable by the experts.

The paper is structured as follows: Section 2 discusses on some related works in knowledge management and experience based solutions in software development; Section 3 explains the research methodology that has been carried out; Section 4 discusses the result and findings; and Section 5 summarizes the findings and concludes the study.

2. RELATED WORKS

There are numerous solutions that have been proposed previously for knowledge management (KM) in software development. In an earlier analysis [3], it has been discovered that ontology and semantic based solution have been gaining popularity while other frameworks have also been well-thought-out like pattern-based, experience factory, social networking, wikis, Software and Systems Process Engineering Meta-Model (SPEM), agent-based and taxonomy. In a study by [14], Web Ontology Language (OWL) for the 2-layer ontology modelling has been introduced for an effective KM system to ease the process of knowledge, search and learning. In another study by [5], a domain specific ontology based system for distributed teams has been proposed as a tool to enable the handling and searching information from the knowledge base. An approach to combine multi-agent and ontology solutions for processing context information has been introduced with a tool called Distributed Software Engineering Environment (DiSEN) that is supposed to support communication, persistence and collaboration among teams geographically distributed [15]. More collaborative solutions are presented such as FLOW Maps [7], Enterprise Software Engineering Model (ESEM) [9], and collaborative KMS framework [16].

In terms of experience based solutions, Knowledge Experience Package (KEP) has been introduced to facilitate knowledge comprehension and acquisition which contains knowledge contents and training units for the purpose of e-learning with the features for knowledge searching and navigation [6]. An Experience Based Model (EBM) has been proposed in [17] to overcome the issues in two software process improvement models namely the model based approach (e.g. CMM) and pragmatic approach (TQM) by managing up-to-date experience about software engineering items or objects which may include any technique, method or tool used for software engineering. In one study, an approach named ReBEC (Reflection-Based Experience Capture) has been introduced that enables organizations to integrate the experience capture activities into daily software project tasks including lessons learned and best practices [18].

Journal of Theoretical and Applied Information Technology
15th January 2017. Vol.95. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Another experience based solution has been proposed namely Practice Selection Framework (PSF) which is used as a tool support for utilizing postmortems based on EF approach that includes method, technique, procedure, tool, or model used in software development [4]. Postmortem reviews are used to evaluate the development practices used, and the results from the evaluation are stored as experiences in the tool, which are later available for the organization.

Although there are a number of available experience based solutions, they are not specifically tailored for structured process for software development and lack of user interactivity and collaboration. The solutions and proposals however provide useful insights for the development of the model proposed in this study. It has been acknowledged that ontology and multi-agent systems can be incorporated together with the experience based approach to provide such a valuable masterpiece for software organizations. The idea can be extended further by empowering the model with collaborative technologies such as internet, Web 2.0 and cloud based environment for maximum possible efficiency and effectiveness.

3. METHODOLOGY

The research methodology is shown in Figure 1.

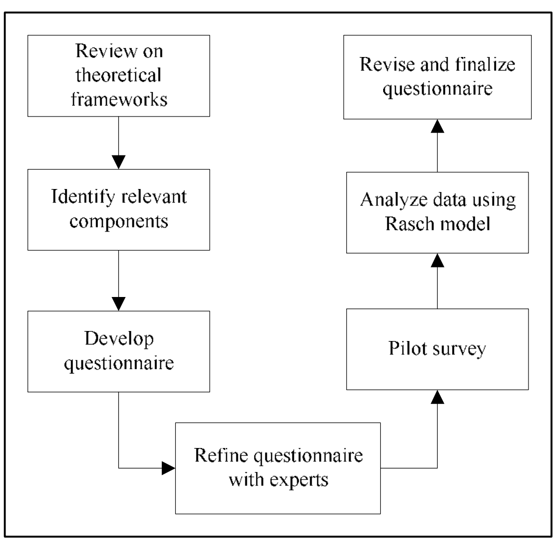

Figure 1: Research Methodology

This research begins with the analysis of the relevant theoretical concepts. This includes the main subjects that will be incorporated in the model: knowledge management, experience factory, software development process, community of practice, appropriate technology and infrastructure and some influences from organizational and managerial perspectives. The initial conceptual model is further formulated based on these components which are then designed as the latent constructs with identified questionnaire items. Some reviews from the KM and software development experts have been carried out. Moving forward, a pilot study has been conducted with the purpose to verify the items as a means of item quality control. The results are analyzed using Rasch measurement model [13] that can help to identify misfitting items and persons. Misfitting items and persons are removed or amended, and the revised questionnaire will be used in further study.

3.1 Construct Validity

Constructs are the abstractions that are deliberately created based on theoretical foundations for the purpose to conceptualize latent variables. Cronbach and Meehl (1955) advocated the concept of construct validity with four divided types: predictive, concurrent, content and construct validity [19]. While all types are important, construct validity appears to be the most crucial. Construct validity refers to the degree to which a test assesses the underlying theoretical construct it is supposed to measure. Two main threats to construct validity are construct under-representation and construct-irrelevant variance, as highlighted by [20] in [21]. Construct under-representation means imperfectness of the test in which the constructs that we gather do include the features according to our definition, but we might miss some important features that should have been included. Construct-irrelevant variance is the existence of unrelated sub-dimensions that are irrelevant to the focal constructs and they produce reliable variances in test scores but irrelevant to the constructs. This could result invalidly low scores for difficult items as well as invalidly high scores for easy items. These invalid measures could contaminate the data and it is highly recommended that these items go through some sort of item quality control.

3.2 Item Quality Control with Rasch Analysis

In survey research design, there is always an emphasis on the quality of the questions or items used. It could exist whether there is any abnormality of the measured latent traits, or whether there are any items that could trigger erratic responses. Especially for self-developed questionnaire, it is important to perform such quality assessment before the questionnaire or the results can be used in further analysis. One most common approach on item reduction is by using

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

classical test theory (CTT). In CTT, however, there exist two major conceptual limitations: the lack of an explicit ordered continuum of items that represent a unidimensional construct, and the lack of additivity of rating scale data; this could be achieved in Rasch scaling methodology by examination of hierarchical structure, unidimensionality and additivity of questionnaire items [22]. An example of the usage of Rasch usage in preliminary work is the item construct verification study for a proposed collaborative software maintenance framework where misfitting items were identified and the survey questions were revised [23].

Rasch analysis is a unique approach of mathematical modeling based upon a latent trait and accomplishes measurement of persons and items on the same scale; one can ask whether there is a substantial number of persons who actually do respond as anticipated by the Rasch model [24][22]. The Rasch model, named after Georg Rasch, is a psychometric model for analyzing categorical data, as a function of the trade-off between person ability versus item difficulty.

Rasch principle states that "a person having a greater ability than another person should have the greater probability of solving any item of the type in question; and similarly, one item being more difficult than another means that for any person the probability of solving the second item is the greater one" [13]. In other words, a person with higher ability should be able to obtain the most correct answers on lower difficulty items, and able to answer some of higher difficulty items. In reverse, a person with lower ability should be able to answer only a few higher difficulty items and some of lower difficulty items. The respondents' agreements towards the proposed components are measured using Rasch Rating Scale Model as follows [25]: the probability of a person n scoring x on item i is given by β_n and δ_i at different threshold level τ_k which is defined as:

$$\Pr\{X_{ni} = x\} = \frac{\exp \sum_{k=0}^{x}(\beta_n - (\delta_i - \tau_k))}{\sum_{j=0}^{m} \exp \sum_{k=0}^{j}(\beta_n - (\delta_i - \tau_k))} \quad (1)$$

where β_n is the person ability and δ_i is the item difficulty.

The reliability in Rasch model is determined by using three measures: Cronbach Alpha, Person Reliability and Item Reliability. Reliability measures show whether similar results can be obtained when the same study is repeated using same instruments. The model fitness is measured on the person fitness and item fitness based on acceptable point measure correlation, outfit and infit mean square, and infit and outfit Z-standardized value. The acceptable ranges of Rasch model are as shown in Table 1.

Table 1: Acceptable range of Rasch measurement model [26]

Person/Item	Acceptable range
Point measure correlation (PTMEA CORR)	0.4 < x < 0.8
Infit/Outfit means square (MNSQ)	0.5 < x < 1.5
Infit/Outfit Z-Standardized value	-2 < x < 2

The items that are out of these ranges or do not fit the model are candidates for modification, discard or indications that the construct theory needs amending [21].

3.3 Theoretical Frameworks Review

3.3.1 Knowledge management (KM)

KM has caught researchers' attention since about two decades ago. In many domains, knowledge is considered as the key element of an organization for future references and improvements. Knowledge is created through the interaction between tacit and explicit, with four ways of knowledge conversation that may exists: from tacit to tacit knowledge (socialization), from tacit to explicit knowledge (externalization), from explicit to explicit knowledge (combination) and from explicit to tacit knowledge (internalization) [27]. While explicit knowledge is the knowledge that can be written down and relatively easy to transfer from one person to the next, tacit knowledge on the other hand is more difficult to articulate because it often arises out of experiences. Experts have difficulties in organizing and formalizing knowledge at the tacit level especially during conceptual design; therefore, it is essential that knowledge should be organized and formalized for each step and activity [28].

In collaborative environment, KM involves four main processes: knowledge acquisition, knowledge storage or mapping, knowledge dissemination and knowledge application [29]. Knowledge acquisition is the process of identifying, collecting, adapting, organizing, and storing the knowledge. Knowledge storage is the organizational knowledge in repositories, either in the form of documentation or in a special format to enable future browsing and quicker access. Knowledge dissemination is where knowledge is published and shared among the users of KM system. Four techniques of disseminating knowledge are available: synchronous technique

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(same time, same place), asynchronous techniques (different time, same place), distributed synchronous collaboration (same time, different place) and distributed asynchronous collaboration (different time, different place) [30]. The process of disseminating knowledge can be effectively achieved with appropriate collaboration and communication means, while the automation can be achieved with the incorporation of software agents.

3.3.2 Experience Factory (EF)

Figure 2 shows the EF framework which consists of two main organizations: Project organization and Experience Factory organization. Introduced by Basili in the 1990s, EF is as a way to improve software development processes by reusing products, processes and other forms of knowledge [11] [12].

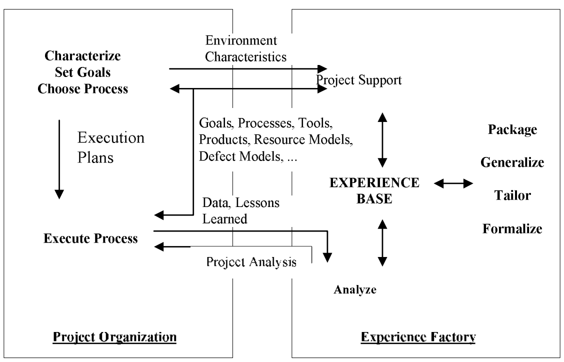

Figure 2: Experience Factory [12]

Project or development organization is responsible to develop and deliver systems, and further provides the EF organization the product development environment characteristics, data and models it currently uses. The EF organization processes the information and returns direct feedback to the project activity with goals and models tailored from previous increment.

EF is based on Quality Improvement Paradigm (QIP) which consists of six fundamental steps: (i) characterize the project environment in terms of its models and metrics, (ii) set quantifiable goals for successful project performance and increment, (iii) choose the appropriate processes, methods and tools, (iv) execute process, constructs the products, collect and analyze data for real-time feedback, (v) analyze and evaluate data, and make recommendations for project improvements, and (vi) package experience in the form of updated and structured knowledge and save it as experience base for future reuse.

Managing experiences are crucial due to several reasons: first, it needs to be less dependent on its employees to prevent loss of knowledge; second, it needs to unload the experts' experiences by eliciting, storing and make it available; third, it needs to create productive employees sooner by speeding up the learning curve; and finally, it needs to improve the business process by analyzing and synthesizing experiences, and further make it captured, structured and available [12]. EF brings benefit to organizations by establishing software improvement process, producing a repository for everyday practices, developing organizational internal support for substantial cost and quality performance benefits, providing a mechanism for incorporating new technologies, and supporting reuse in software development process [11].

3.3.3 Software development process (SD)

Software development process goes through a lifecycle model where many phases, activities, techniques, methods, tools and artifacts are involved. Nowadays, there are several known different software process models, however, the most fundamental required phases are: specification, design and implementation, validation and evolution [31]. Software Engineering Body of Knowledge (SWEBOK) specifies the following phases in software development process lifecycle: (i) Software Requirement (RE): the elicitation, analysis, specification, and validation of software requirements as well as the management of requirements during the whole life cycle of the software product; (ii) Software Design (DE): the analysis of requirements that describe the software internal structures resulting into software architecture with subcomponents and interfaces defined; (iii) Software Construction (CO): the detailed creation of working software through a combination of coding, verification, unit testing, integration testing, and debugging; (iv) Software Testing (TE): the dynamic verification of a program by executing a predefined set of test cases against the expected behaviors; and (v) Software Maintenance (MA): continuous tasks and activities required to support the software [32]. These activities are tailored into different types of lifecycle models that we currently have such as waterfall, spiral, iterative and incremental, prototyping and agile methodologies.

In each model, there are specific methods, techniques and tools used to achieve certain objectives and tasks. It is also important that these processes are made known and available for every software practitioners in an organization to promote organizational learning. In learning organizations, the collection of best practices and lessons learned

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

enhances and harnesses individual and team learning that already occurs in the organization; however, many organizations miss the opportunity to take its valuable advantage because the information is often lost and not captured in a timely manner as it is being gained [33].

2.2.2 Community of practices (CoP)

In a learning organization, CoP plays an important role in successful implementation of KM. CoP is defined as "groups of people who share a concern, a set of problems, or a passion about a topic, and who deepen their knowledge and expertise in this area by interaction on an ongoing basis" [34]. It is assumed that CoP is a good mechanism for capturing, transferring and sharing of tacit knowledge to those that are from different units to discuss topics in interest [35]. As communities now spread across the boundaries, there exists the necessity to provide the means for effective collaboration.

Collaborative activity between the participants depends on the shared understanding of an organization's internal and external context, which is likely to occur with a combination of face-to-face meetings and machine-mediated communications. With effective tool support, the operations among distributed teams or virtual organizations can be done in a more unified way [36]. The emergence of Internet and Web 2.0 as well as older but still relevant technologies like email, instant messaging, and bulletin board has actually allowed CoP to work more efficiently in which they can eventually form the 'virtual community' [37]. It is further analyzed that more sophisticated technologies can further enhance the growth of CoP including Wiki, Social Networking, Forums, Weblogs, Learning Management System, and Content Management System [37].

2.2.3 Technology and infrastructure (TI)

In implementing knowledge based solution, proper technology and infrastructure are the fundamental requirements. With today's distributed environment, such structure should be able to support the needs of the community to communicate and collaborate in an efficient manner. Functionalities of a KM system such as knowledge portal, document management system, information retrieval, data warehousing or data mining can be supported by relevant technologies such as intranets (workgroup), groupware and agent technology [16].

Technologies that are relevant for the knowledge management process includes content generation tool, discovery tools (e.g. data mining, web data capture), technologies for storage (e.g. databases or warehouses), for codification (e.g. case based or rule based approach), organization technologies (e.g. taxonomies, indexes and directories, ontologies), transformation tool (e.g. validation, compilation, reconstruction), use tools (e.g. expert systems, ERP, CRM), technologies for access and transfer (e.g. portals, groupware), and knowledge sharing tools (e.g. interface, internet/intranet tools, intelligent agents) [38].

KM portal has been considered as a virtual workspace that can promote knowledge sharing among CoP. It has the core features such as taxonomy (or classification schemes), publishing content in different formats, integrated search capability, personalization for users according to preferences, integration with different organizational repositories, and collaboration [39]. Additionally, a portal allows access to knowledge repository or structured knowledge base. Ontology, as a standard representation of the structured knowledge with a set of concepts and relationship, can be used to represent knowledge classification within a particular domain. Likewise, ontology based organizational memory can represent the scattering knowledge based in the software development process [40].

Some artificial intelligence techniques have also been applied in KM based solution for automation purposes such as software agents, expert system, and case or rule based system. Software agents have been widely used in KM systems as it has the capability to communicate and negotiate, to learn how to improve performance over time and to react proactively [30].

The high volume of knowledge and experiences also raises the need to have such a reliable and secure platform. It, therefore, makes sense to take the advantage of cloud computing environment as it offers adequate infrastructure and storage by providing on-demand services with high availability, reliability, elasticity and scalability [2].

2.2.4 Organizational and managerial influences (OM)

The organization plays a vital role in the successful implementation of KM solution. A previous review shows that organizational culture is considered as a critical factor in developing and reinforcing knowledge creation and knowledge management in organization as it impacts how members learn, acquire, and share knowledge [41]. KM is also a value-added to the organization in such a way that it encourages innovation, maximize

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

profits, and improve decision making by means of knowledge and information sharing among the people working within the organization [42].

Personal values within organization may also affect knowledge retention. These values include the belief on the importance of knowledge sharing, trusting element at individual levels, organizational support and encouragement, and the lack of these elements is considered as the greatest factor obstructing knowledge behavior [43]. A study suggests that organizations need to provide constant monitoring, reward, training, support, policy implementation and awareness on the benefits and importance of KM to the employees [44]. Apart from these, budgeting especially on labor, equipment and materials must also be considered rationally [45].

In addition to organizational influences, leadership is also a critical success factor in KM strategy. Leadership has been shown to be positively related to KM success and its adoption will increase the effectiveness of KM projects [46]. The best form of leadership style for managing knowledge in organization is when employees are given adequate power, authority, and responsibility to experiment and innovate in their tasks [47]. It is also important to have a performance indicator for KM strategy. Such indicators could include the following: (i) process (knowledge quality, efficiency due to new routines, incentives, knowledge contributor); (ii) human (knowledge sharing attitude and activities, participation in activities, awareness); and (iii) information technology (active involvement, knowledge structure, usability) [48].

3.4 The initial conceptual framework

Based on the review of the theoretical frameworks, the following components have been identified. Table 2 shows the components' relevant items and their source of references. Figure 3 illustrates the corresponding components. For experience factory component (EF), we have two distinguished organizations: EF1 (project organization) and EF2 (experience factory organization). EF1, which consists of the CoP and SD, provides EF2 with the project data, its processes, and environment. Based on QIP, EF1 is responsible for the planning (characterize, set goals, choose process) and executing the model, while EF2 is accountable for analyzing and packaging the end product.

CoP is the people who are directly involved in developing software products such as software engineers, developer, software architects, system and business analysts, database administrator, software testers, consultants and project managers. SD is the process of software development itself which includes the software process models or lifecycles, their phases, activities, best practices, methods, techniques and tools. As EF2 is responsible to provide the platform for managing the experiences of the project organization, it equips itself with the appropriate KM processes for the purpose of knowledge acquisition, mapping and storage, dissemination and reuse. This platform allows massaging of the information it receives and producing more customized outputs based on requests. It is then that the new output becomes a new knowledge in the repository.

In order to achieve the KM objectives, the platform should be supported with the appropriate technology and infrastructure (TI). Four main areas are identified as significant; (i) knowledge portal and repository, (ii) automation and discovery, (iii) collaboration and communication, and (iv) cloud based environment. Knowledge portal and repository provides the interface for knowledge capture, classification, storage, search and retrieval. Automation is supported by agent technologies, social networks, mobile technology and RSS feeds. Collaboration and communication are important to serve asynchronized and synchronized communications, as well as for knowledge sharing and dissemination. The employment of cloud computing is to ensure that the efficiency of the collaborative model can be maximized in terms cost effectiveness, high availability and scalability.

The success of the model is also largely influenced by the organizational and managerial role (OM). In organizational level, these aspects are important: culture and values, awareness, reward and motivation, value-added of KM towards organization as well as financial needs and training. Additionally, the management is expected to have leadership values, provide coordination and control of the KM implementation, ensure the readiness of the organization and provide adequate measurement from time to time. The measurement of the model (shown at the most right of Figure 3) will be based on the perceived benefits of experience factory and identified success quality factors; however, this is not the scope of this research.

The six identified components become the latent constructs of the initial framework. 80 questions were considered with the total of 141 items. The questionnaires were repeatedly reviewed by two KM experts in software development until an acceptable set of questionnaire is realized.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. RESULTS AND DISCUSSION

The sample for this pilot study consists of 28 software practitioners from 2 software companies based on convenient sampling. 19 are software engineers and the others are project managers, software testers and database administrators. Data were collected using a structured questionnaire with 4-point Likert scale for agreeableness measure: 1-strongly disagree, 2-disagree, 3-agree and 4-strongly agree. They were given one week duration to answer the questionnaire.

Analysis tool WINSTEPS Version 3.68.2 was used to examine the data. Rasch analysis can be carried out several times until a satisfactory result is achieved. The summary statistics as shown in Figure 4 reveals excellent person reliability measures and Cronbach alpha with 0.96 values. For reliability, a value of greater than 0.7 shows high reliability [26]. The spread of the responses is 0.36 logit; this is close to 1. Outfit and infit mean square is 1.01 and 0.98 respectively, and this is very close to the expected value of 1. Z-standardized value is -0.9, and this is also close to value 0 and within the normality range: $-2 < Z < +2$.

The summary of measured items, as also shown in Figure 4, shows the overall fitness of the instrument whether it fits with Rasch model. Item reliability value with 0.67 is quite fair. Item mean square values are also very close to 1 and within the expected range of $0.5 < x < 1.5$. Z-standardized value is .1; it is expected to be at norm and within the normality range $-2 < Z < +2$. This indicates overall fit to the Rasch model.

A close examination to the person misfit order shows that person ST1 has negative point measure correlation (-0.19) (Figure 5). In Rasch analysis, negative correlation gives the perception there could be something wrong with the item or person. This is a clear candidate for removal. With this finding, we remove this person from the data, and run Rasch analysis again. Summary statistics for the second run in shown in Figure 6. Person reliability stays at 0.96 while item reliability has a slight improvement with value at 0.73. This indicates higher reliability as compared to the previous result before this elimination. Item mean square values are 1 and Z-standardize values are 0.1; these are still within range. This again confirms the overall fitness of the instrument with Rasch model.

Rasch item maps shows the distribution of item difficulties aligned with the distribution of person abilities measured in terms of logit scale, a common measurement unit used for both [49]. As shown in Figure 7, the person-item map illustrates that the items at the top are more difficult to endorse while items at the bottom are easier to endorse on agreeableness. Respondents at the higher scale are more agreeable with the items, while those at the bottoms are less agreeable with the items. Only one person that is item free, which means he or she could easily endorse the items; and there is one person that is item free at the bottom, which means that he or she hardly agrees with all items. The logit scale for person is 3.69 with 2.14 as the maximum and -1.55 as the minimum logit values. The item logit scale is 3.65 with 2.19 and -1.46 as the maximum and minimum values respectively. Person spread (3.69) is just a little higher than item spread (3.65); this indicates that item difficulty is within respondents' ability.

We further examine the unidimensionality. Unidimensionality in Rasch is the key component of content validity. It refers to how well the items fit the constructs. It is suggested that that data dimensionality can be done in three stages: (i) by using point biserial correlation (this is more applicable for dichotomous variable, therefore, it does not apply here); (ii) by analyzing misfitting items and persons using Rasch fit indicators; and (iii) by examining unidimensionality using Rasch factor analysis [50]. Infit and outfit statistics for item and measure are too much influenced by accidents in the data such as guessing, and generally do not detect the more indirect but pervasively impact a second dimension [51].

We proceed with (ii) by analyzing misfitting items. Figure 8 shows that several items have negative point measure correlation with Z-standardized values of more than 2. A more comprehensive item measures can be seen in Table 4. Even though A4_C has negative measure value, the Z-standardized value is still within range; therefore this item will be retained. The items A4_D, A3_E, F11_H and A3_F are generally sub-items, and will not have major impact; thus, removal of these sub-items is insignificant. It is more important to pay higher attention to item A1 because the point measure value is the lowest (-0.21). On further thought, the meaning of Item A1 (*I am familiar with any of software engineering or software development standard (e.g. IEEE, CMMI, ISO*) is more towards checking persons' profiles or demographic data, and the answer should be either 'yes' or 'no'. Therefore, it is more suitable to move this item as one of the demographic items.

With the elimination of item A1, we run the Rasch analysis again. We proceed to step (iii) by examining unidimensionality using Rasch factor

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

analysis. In dimensionality analysis, the variance explained by the first contrast in the residuals indicates whether there could be another dimension exists. Figure 9 shows the empirical values which are very close with the predicted modelled values.

For unexplained variance for first to fifth contrast, value of more than 15% is poor, 10-15% is fair, 5-10% is good, 3-5% is very good and less than 3% is excellent [26]. The unexplained variance for first is considered good (6.6%), but the eigenvalue unit shows the strength of around 13 items and is considered very high. According to Rasch model simulations, it is unlikely that the 1st contrast in the unexplained variance will have a size larger than 2.0. Higher value indicates that there could potentially exist secondary dimensions. However, on further investigation on the contrast loading, it is observed that the plot looks random (Figure 10) and they are generally within the acceptable range of -0.6 and 0.6. Therefore, we can conclude that there is no visible secondary dimension.

5. CONCLUSION

KM is a critical area in many domains including software development. With the advancement of technologies and the demand from software organizations, it is crucial to employ the right components that can help practitioners to perform their work more effectively and efficiently. This research begins with the initial model development by first identifying the relevant components based on the identified theoretical concepts, and further, we proceed by developing the questionnaire items for the constructs. Relevant components including knowledge management, experience factory, software development process, community of practices, technology and infrastructure, and influences from managerial and leadership are incorporated. Agreements from the community of practice on the constructs are collected through a survey research using questionnaire as the instruments.

The quality of the items is verified using Rasch analysis to ensure the fitness of the item constructs. Analysis with Rasch model gives preliminary insights on the model development whether the model being built constitutes the right components, and whether the items measure or fit a constructs. The study reveals that the reliability and validity of the measurement instruments can be enhanced through the removal or adjustment of misfitting items and persons. Successive analyses are performed with some calibrations until a satisfactory data is achieved. There is no clear secondary dimension and no excessive amount of misfitting items or persons, henceforth, it can be concluded that the items are reliable to be used in further study. The study also shows that Rasch measurement model is a powerful tool for analyzing item constructs validity by calibrating person ability and item difficulty, and distinguishes misfit responses to finally achieve the data be fitted in the model.

REFRENCES:

[1] K. Schneider, Experience and Knowledge Management in Software Engineering, Berlin: Springer-Verlag, 2009.

[2] E. H. M. Huzita, G. C. L. Leal, R. Balancieri, T. F. C. Tait, E. Cardoza, E., R. R. de Moura Penteado, and R.L.Vivian, "Knowledge and Contextual Information Management in Global Software Development Perspectives", IEEE Seventh International Conference on Challenges and Global Software Engineering Workshops, 2012, pp. 43-48.

[3] M. Hanafiah, R. Abdullah, J. Din and M. A. A. Murad, "Towards Developing Collaborative Experience Based Factory Model for Software Development Process in Cloud Computing Environment", International Review on Computers and Software (I.RE.CO.S.), Vol. 10 No. 3, 2015, pp. 340-350.

[4] M. Ivarsson, and T. Gorschek, "Tool Support for Disseminating and Improving Development Practices", Software Quality Journal, Vol. 20 No. 1, 2011, pp. 173-199.

[5] R. G. C. Rocha, R. Azevedo, and S. Meira, "A Proposal of an Ontology-Based System for Distributed Teams", 40th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA), 2014, pp. 398.

[6] P. Ardimento, M. Cimitile, and G. Visaggio, "Distributed Software Development with Knowledge Experience Packages", Lecture Notes in Computer Science, Vol. 8186, 2013, pp. 263-273.

[7] K. Stapel, and K. Schneider, Managing Knowledge on Communication and Information Flow in Global Software Projects, Expert Systems, Vol. 31, No. 3, 2012, pp. 234-252.

[8] E. Wende, T. Philip, G. Schwabe, and G. King, "Kaiwa: Towards a method for knowledge transfer in the transition phase of offshore

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

outsourced projects", Lecture Notes in Business Information Processing, Vol. 163, 2013, pp. 180-191.

[9] F. Salger, F. Sauer, G. Engels, and A. Bauman, "Knowledge Transfer in Global Software Development - Leveraging Ontologies, Tools and Assessments", 5th IEEE International Conference on Global Software Engineering (ICGSE), pp. 336.

[10] P. Wongthongtham, and N. Kasisopha, "An Ontology-Based Method for Measurement of Transferability and Complexity of Knowledge in Multi-site Software Development Environment", Lecture Notes in Computer Science, Vol. 6746, 2011, pp. 238-252.

[11] V. R. Basili, and G. Caldiera, "Improve Software Quality by Reusing Knowledge and Experience", Sloan Management Review, 1995, pp. 55-64.

[12] V. R. Basili, M. Lindvall, and P. Costa, "Implementing the Experience Factory concepts as a set of Experience Bases", Proceedings of 13th International Conference on Software Engineering and Knowledge Engineering, 2001.

[13] G. Rasch, Probabilitics models for some intelligence and attainment tests, Copenhagen: DanmarksPpaedagogoske Institut, 1960.

[14] M. M. Salam Khan, M. A. Salam Khan, T. Goto, T. Nishino, and N. Debnath, "Software ontology design to support organized open source software development", 15th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/ Distributed Computing (SNPD 2014), 2014.

[15] H. H. L. C. Monte-Alto, A.B. Biasão, L.O. Teixeira, E.H.M. Huzita, "Multi-agent Applications in a Context-Aware Global Software Development Environment", Advances in Intelligent and Soft Computing, Vol. 151, 2012, pp. 265-272.

[16] R. Abdullah, S. Sahibudin, R. A. Alias, and M. H. Selamat, "Collaborative Knowledge Management Systems for Learning Organisations", Journal of Information and Knowledge Management, Vol. 4, No.4, 2005, pp. 237-245.

[17] N. Sharma, K. Singh, and D. P. Goyal, "Experience Base Approach to Software Process Improvement: Comparative Analysis and Design of Improved Model", Advanced Computing & Communication Technologies (ACCT), 2012, pp. 30 – 35.

[18] G. Maturro, and A. Silva, "ReBEC: A Method for Capturing Experience during Software Development Projects", Lecture Notes in Computer Science, Vol. 6317, 2010, pp. 524-533.

[19] L. J. Cronbach, and P. E. Meehl, "Construct validity in psychological tests", Psychological Bulletin, Vol. 52 No. 4, 1955, pp. 281–302.

[20] S. Messick, Validity. In R. Linn (Ed.), Educational measurement (3rd ed.), Washington, D.C.: American Council on Education, 1989.

[21] P. Baghaei, "The Rasch Model as a Construct Validation Tool", Rasch Measurement Transactions, Vol. 22 No.1, 2008, pp. 1145-1146.

[22] L. Prieto, J. Alonso, and R. Lamarca, "Classical test theory versus Rasch analysis for quality of life questionnaire reduction", Health Quality Life Outcomes, Vol. 1 No. 27, 2003, pp.1–13.

[23] M. Z. M. Nor, R. Abdullah, M. A. A. Murad, M. H. Selamat, and A. A. Aziz, "KMS Components for Collaborative Software Maintenance – A Pilot Study", International Conference on Information Retrieval and Knowledge Management, 2010, pp. 332-337.

[24] C. Granger, "Rasch Analysis is Important to Understand and Use for Measurement", Rasch Measurement Transactions, Vol. 21, No.3, 2008, pp. 1122-1123.

[25] D. Andrich, "A rating formulation for ordered response categories", Psychometrika, Vol. 43, 1978, pp. 561-73.

[26] W. P. Fisher, "Rating Scale Instrument Quality Criteria", Rasch Measurement Transactions, Vol. 21, No. 1, 2007, pp.1095.

[27] I. Nonaka, and H. Takeuchi, The Knowledge-Creating Company, Oxford University Press, New York, NY, 1995.

[28] Z. Pourzolfaghar, R. Ibrahim, R. Abdullah, and N.M. Adam, "A Technique to Capture Multi-Disciplinary Tacit Knowledge During the Conceptual Design Phase of a Building Project, Journal of Information and Knowledge Management, Vol. 13, No. 2, 2014,

[29] R. Abdullah, M. H. and Selamat, "Facilitating knowledge sharing with groupware among faculty communities in higher learning institution", IJCSNS International Journal of Computer Science and Network Security, Vol. 5, 2007, pp. 220-229.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[30] R. Abdullah, Knowledge Management System in a Collaborative Environment, Universiti Putra Malaysia Press, 2008.
[31] I. Sommerville, Software Engineering. 9th ed. Boston: Addison-Wesley, 2011.
[32] P. Bourque, and R. E. Fairley, "SWEBOK v3.0: Guide to Software Engineering Body of Knowledge", IEEE Computer Society, retrieved 14 Dec 2014 from https://www. computer.org/ web/swebok/v3.
[33] J. V. Vandeville, "Organizational Learning Through the Collection of Lessons Learned", Special Series on Organizational Learning, Vol. 3, No. 3, 2000, pp. 127-133.
[34] E. C. Wenger, R. McDermott, and W. C. Snyder, Cultivating Communities of Practice: A Guide to Managing Knowledge, Harvard Business School Press, Cambridge, USA, 2002.
[35] A. Mestad, M. Myrdal, T. Dingsøyr, and T. Dybå, "Building a Learning Organization - Three Phases of Communities of Practice in a Software Consulting Company," Proceedings of the 40th Hawaii International Conference on System Sciences (HICSS'07), 2007.
[36] P. Clarke, and M. Cooper, "Knowledge management and collaboration", Proceedings of the Third International Conference On Practical Aspects of Knowledge Management, Vol. 6, 2000.
[37] P. Penfold, "Virtual Communities of Practice: Collaborative Learning and Knowledge Management", Third International Conference on Knowledge Discovery and Data Mining (WKDD '10), 2010, pp. 482-485.
[38] A. Antonova, E. Gourova, and R. Nikolov, "Review of technology solutions for knowledge management", 2nd IEE International Conference on Intelligent Environments (IET), Athens, Greece, 5-6 July 2006.
[39] H. Benbya, G. Passiante, and N. A. Belbaly, (2004), "Corporate Portal: A tool for knowledge management synchronization", International Journal of Information Management, Vol. 24, 2004, pp. 201-220.
[40] M. A. Jabar, M. S. Khalefa, R. Abdullah, and S. Abdullah, "Meta-Analysis of Ontology Software Development Process", International Review on Computers and Software (IRECOS), Vol. 9 No. 1, 2014, pp. 29-37.
[41] R. K. Rai, "Knowledge management and organizational culture: a theoretical integrative framework", Journal of Knowledge Management, Vol. 15, No. 5, 2011, pp. 779 – 801.
[42] Z. A. Mohsen, M. Ali, A. Jalal, "The Significance of Knowledge Management Systems at Financial Decision Making Process", International Journal of Business and Management, Vol. 6, No. 8, 2011, pp. 130-142.
[43] E. C. Martins, and H. W. J. Meyer, "Organizational and behavioral factors that influence knowledge retention", Journal of Knowledge Management, Vol. 16, No.1, 2012, pp. 77-96.
[44] D. Chandran, and K. Raman, "Awareness and Problems in Implementing Knowledge Management Systems in Medium Sized Business Organizations in Malaysia", Journal of Social Science, Vol. 19, No. 2, 2009, pp. 155-161.
[45] I. Goodluck, "Budgeting for knowledge management in organisations", Chinese librarianship: An International Electronic Journal, Vol. 32, 2011, retrieved 21 Feb 2016 from http://www.whiteclouds.com/iclc/cliej/ cl32goodluck.pdf.
[46] M. Mas-Machuca, "The Role of Leadership: The Challenge of Knowledge Management and Learning in Knowledge-Intensive Organizations", Journal of Educational Leadership and Management, Vol. 2, No. 1, 2014, pp. 97-116.
[47] S. K. Singh, "Role of leadership in knowledge management: a study", Journal of Knowledge Management, Vol. 12, No. 4, 2012.
[48] S. Rifat, "Measuring Knowledge Management Performance", European Journal of Scientific Research, Vol. 35, No. 2, 2009, pp. 242-253.
[49] T. G. Bond, and C. M. Fox, Applying the Rasch Model: Fundamental Measurement in the Human Sciences, 3rd. edition. New York, NY: Routledge, 2015.
[50] J. M. Linacre, "Prioritizing misfit indicators", Rasch Measurement Transactions, Vol. 9, No. 2, 1992, pp. 422-423.
[51] J. M. Linacre, "Dimensionality: when is a test multidimensional?", retrieved 21 Jan 2016 from http://www.winsteps.com/winman/ dimensionality.htm.
[52] M. L. Badger, T. Grance, R. Patt-Corner, and J. M. Vaos, "Cloud Computing Synopsis and Recommendations", NIST Special Publication, 2012.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[53] S. E. Bartczak, W.R. Boulton, R.K. Rainer, and S.L. Oswald, "Investigating Barriers to Knowledge Management Implementation In the U. S. Military: A Focus on Managerial Influences", Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA, 25 - 26 March 2010.

[54] I. Becerra-Fernandez, and R. Sabherwal, Knowledge Management: Systems and Processes, Armonk (N.Y.); London: M.E. Sharpe, 2010.

[55] M. Montoni, R. Miranda, A. R. Rocha, and G. H. Travassos, "Knowledge Acquisition and Communities of Practice: An Approach to Convert Individual Knowledge into Multi-organizational Knowledge", Advances in Learning Software Organizations, Lecture Notes in Computer Science, Vol. 3096, 2004, pp. 110-121.

Journal of Theoretical and Applied Information Technology
15th January 2017. Vol.95. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2: EBF-SD components and items

Components	Items	Sources
Experience Factory (EF)	EF framework; Quality Improvement Paradigm (QIP); Perceived benefits;	Basili and Caldiera (1995) [11]; Basili et al. (2001) [12]
Community of Practice (CoP)	Knowledge capture; transfer and sharing; facilitation on knowledge sharing; Skills and experts; Communication & Collaboration; Technological support (Web 2.0, internet);	Mestad et al. (2007) [35]; [54]Montoni et al. (2004); [53]Becerra-Fernandez and Sabherwal (2010);
Software Development (SD) Process	Development methodology; best practices; methods; tools; techniques; lessons learned	Bourque and Fairley (2014) [32]; Sommerville (2011) [31]; Vandeville (2000) [33];
Knowledge Management Process (KM)	Acquisition; storage and mapping; dissemination; application	Nonaka and Takeuchi (1995) [27]; Abdullah and Selamat (2007) [29]; Becerra-Fernandez and Sabherwal (2010); Pourzolfaghar et al. (2014);
Technology and Infrastructure (TI)	Portal and repository; automation and discovery; communication and collaboration; cloud computing	Becerra-Fernandez and Sabherwal (2010) [53]; Antonova et al. (2006) [38]; Benbya et al. (2004) [39]; Jabar et al. (2014) [40]; [51] Badger et al. (2012); Huzita et al. (2012) [2];
Organizational and Managerial Influences (OM)	Culture and values; awareness; Reward and motivation; value-added; financial, training; leadership; coordination; control; readiness; measurement	Martins and Meyer (2012) [43]; Chandran and Raman (2009) [44]; Mohsen et al. (2011) [42]; Goodluck (2011) [45]; Bartczak et al. (2011) [52]; Rifat (2010) [48]; (Mas-Machuca, 2014) [46]; (Singh, 2012) [47];

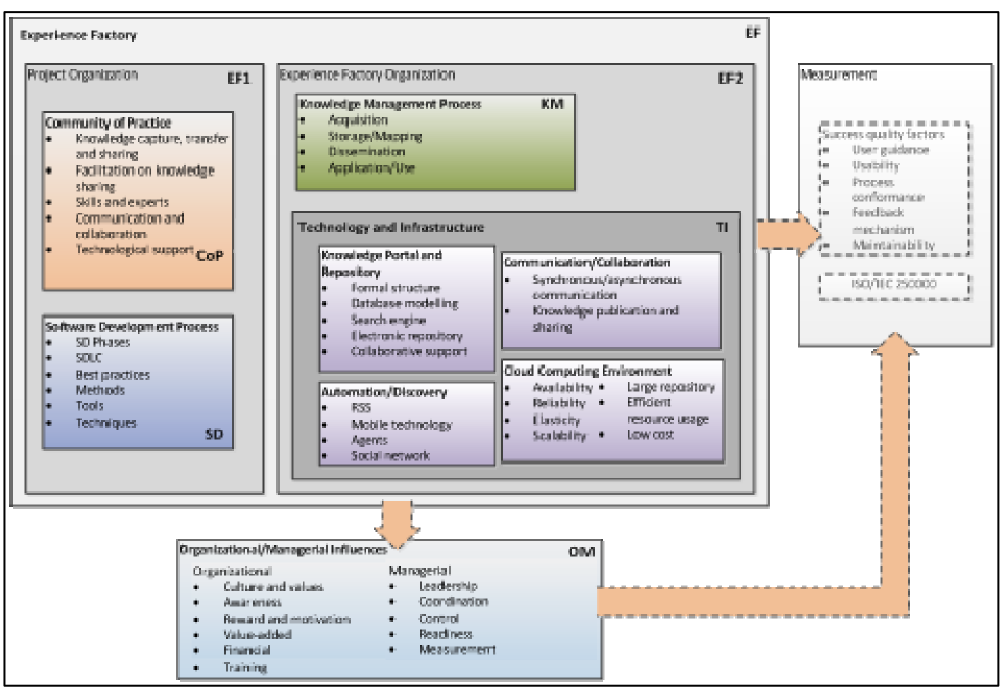

Figure 3. Initial conceptual model

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

```
     SUMMARY OF 28 MEASURED PERSONS
-------------------------------------------------------------------------------
|          RAW                          MODEL         INFIT        OUTFIT     |
|         SCORE      COUNT    MEASURE   ERROR      MNSQ  ZSTD   MNSQ  ZSTD   |
|-----------------------------------------------------------------------------|
| MEAN    438.0      140.7        .36     .16       .98  -1.1   1.01   -.9   |
| S.D.     31.2        1.1        .77     .01       .74   3.8    .76   3.7   |
| MAX.    506.0      141.0       2.17     .18      4.30   9.9   4.49   9.9   |
| MIN.    381.0      136.0       -.96     .14       .20  -8.2    .22  -8.1   |
|-----------------------------------------------------------------------------|
| REAL RMSE    .17  ADJ.SD     .75  SEPARATION  4.36  PERSON RELIABILITY  .95 |
|MODEL RMSE    .16  ADJ.SD     .76  SEPARATION  4.80  PERSON RELIABILITY  .96 |
| S.E. OF PERSON MEAN = .15                                                   |
-------------------------------------------------------------------------------
        VALID RESPONSES:  99.8%
PERSON RAW SCORE-TO-MEASURE CORRELATION = .99 (approximate due to missing data)
CRONBACH ALPHA (KR-20) PERSON RAW SCORE RELIABILITY = .96 (approximate due to missing data)

     SUMMARY OF 141 MEASURED ITEMS
-------------------------------------------------------------------------------
|          RAW                          MODEL         INFIT        OUTFIT     |
|         SCORE      COUNT    MEASURE   ERROR      MNSQ  ZSTD   MNSQ  ZSTD   |
|-----------------------------------------------------------------------------|
| MEAN     87.0       27.9        .00     .37      1.00    .1   1.01    .1   |
| S.D.      6.6         .3        .66     .06       .17    .7    .20    .8   |
| MAX.    103.0       28.0       2.07     .53      1.66   2.5   1.68   2.7   |
| MIN.     70.0       26.0      -1.24     .24       .68  -1.4    .63  -1.4   |
|-----------------------------------------------------------------------------|
| REAL RMSE    .39  ADJ.SD     .53  SEPARATION  1.37  ITEM   RELIABILITY  .65 |
|MODEL RMSE    .37  ADJ.SD     .54  SEPARATION  1.44  ITEM   RELIABILITY  .67 |
| S.E. OF ITEM MEAN = .06                                                     |
-------------------------------------------------------------------------------
UMEAN=.000 USCALE=1.000
ITEM RAW SCORE-TO-MEASURE CORRELATION = -.11 (approximate due to missing data)
3940 DATA POINTS. LOG-LIKELIHOOD CHI-SQUARE: 5746.41 with 3609 d.f. p=.0000
```

Figure 4: Summary statistics (first run)

```
-------------------------------------------------------------------------------------
|         MODEL|   INFIT    |  OUTFIT    |PT-MEASURE  |EXACT MATCH|          |
|MEASURE  S.E. |MNSQ  ZSTD|MNSQ  ZSTD|CORR.  EXP.| OBS%  EXP%| PERSON   |
|--------------+-----------+-----------+-----------+-----------+----------|
|  -.52     .15|4.30   9.9|4.49   9.9|A-.16   .38| 12.1  68.3| ST1      |
|   .79     .16|1.50   3.4|1.60   4.0|B .26   .34| 51.1  67.2| SE13     |
|  2.17     .18|1.58   5.2|1.55   4.3|C .13   .25| 63.1  66.3| SE10     |
|   .45     .16|1.51   3.1|1.50   3.1|D .21   .36| 54.6  69.3| OT3      |
|  -.02     .16|1.25   1.5|1.15   1.0|E .21   .38| 70.9  70.6| SE4      |
|   .91     .17|1.18   1.4|1.24   1.8|F .40   .33| 56.5  66.4| SE14     |
|   .45     .16|1.13    .9|1.19   1.3|G .32   .36| 59.6  69.3| SE2      |
|   .50     .16|1.18   1.2|1.13    .9|H .40   .36| 61.0  69.0| OT9      |
|   .56     .16|1.12    .9|1.12    .9|I .45   .35| 61.7  68.7| PM1      |
|  -.63     .14|1.06    .5|1.02    .2|J .43   .38| 63.8  66.4| OT5      |
|  1.50     .17|1.04    .4|1.03    .3|K .43   .29| 66.7  63.1| SE17     |
|  -.23     .15| .91   -.5|1.00    .0|L .43   .38| 70.9  70.3| OT4      |
|  1.17     .16| .96   -.4| .97   -.3|M .31   .32| 62.4  64.8| PM3      |
|  -.96     .14| .68  -3.1| .95   -.3|N .34   .37| 66.7  63.2| SE19     |
|   .40     .16| .86  -1.0| .94   -.3|n .37   .36| 70.9  69.5| SE9      |
|  1.06     .16| .85  -1.3| .92   -.6|m .49   .32| 66.0  65.6| SE20     |
|  1.52     .17| .91   -.9| .83  -1.7|l .43   .29| 69.5  63.0| PM4      |
|  1.50     .17| .75  -2.6| .74  -2.6|k .27   .29| 71.6  63.1| SE16     |
```

Figure 5: Person misfit order

```
     SUMMARY OF 27 MEASURED PERSONS
-------------------------------------------------------------------------------
|          RAW                          MODEL         INFIT        OUTFIT     |
|         SCORE      COUNT    MEASURE   ERROR      MNSQ  ZSTD   MNSQ  ZSTD   |
|-----------------------------------------------------------------------------|
| MEAN    439.3      140.7        .18     .17      1.00   -.4   1.00   -.4   |
| S.D.     31.0        1.1        .91     .00       .41   3.5    .41   3.3   |
| MAX.    506.0      141.0       2.14     .18      1.73   5.8   1.76   5.0   |
| MIN.    381.0      136.0      -1.55     .17       .23  -8.3    .22  -8.1   |
|-----------------------------------------------------------------------------|
| REAL RMSE    .19  ADJ.SD     .89  SEPARATION  4.76  PERSON RELIABILITY  .96 |
|MODEL RMSE    .17  ADJ.SD     .89  SEPARATION  5.16  PERSON RELIABILITY  .96 |
| S.E. OF PERSON MEAN = .18                                                   |
-------------------------------------------------------------------------------
        VALID RESPONSES:  99.8%
PERSON RAW SCORE-TO-MEASURE CORRELATION = 1.00 (approximate due to missing data)
CRONBACH ALPHA (KR-20) PERSON RAW SCORE RELIABILITY = .96 (approximate due to missing data)

     SUMMARY OF 141 MEASURED ITEMS
-------------------------------------------------------------------------------
|          RAW                          MODEL         INFIT        OUTFIT     |
|         SCORE      COUNT    MEASURE   ERROR      MNSQ  ZSTD   MNSQ  ZSTD   |
|-----------------------------------------------------------------------------|
| MEAN     84.1       26.9        .00     .41      1.00    .1   1.00    .1   |
| S.D.      5.9         .3        .79     .06       .20    .8    .26    .9   |
| MAX.     99.0       27.0       2.19     .57      1.67   3.0   1.83   3.2   |
| MIN.     66.0       25.0      -1.46     .27       .62  -1.7    .50  -1.7   |
|-----------------------------------------------------------------------------|
| REAL RMSE    .43  ADJ.SD     .67  SEPARATION  1.56  ITEM   RELIABILITY  .71 |
|MODEL RMSE    .41  ADJ.SD     .68  SEPARATION  1.63  ITEM   RELIABILITY  .73 |
| S.E. OF ITEM MEAN = .07                                                     |
-------------------------------------------------------------------------------
UMEAN=.000 USCALE=1.000
ITEM RAW SCORE-TO-MEASURE CORRELATION = -.22 (approximate due to missing data)
3799 DATA POINTS. LOG-LIKELIHOOD CHI-SQUARE: 5074.30 with 3508 d.f. p=.0000
```

Figure 6: Summary statistics (second run)

Journal of Theoretical and Applied Information Technology
15th January 2017. Vol.95. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

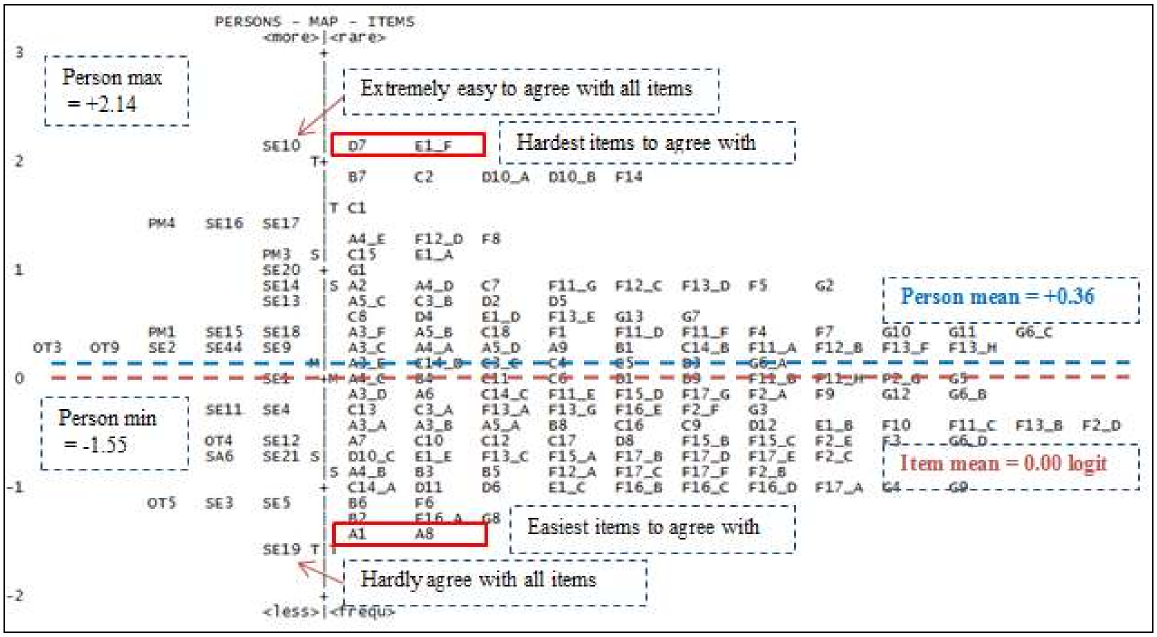

Figure 7: The variable map

MEASURE	MODEL S.E.	INFIT MNSQ	INFIT ZSTD	OUTFIT MNSQ	OUTFIT ZSTD	PT-MEASURE CORR.	PT-MEASURE EXP.	EXACT MATCH OBS%	EXACT MATCH EXP%	ITEM	G
.02	.38	1.49	1.3	1.83	1.8	A-.02	.42	64.0	71.3	A4_C	0
-1.40	.38	1.67	2.4	1.80	2.7	B-.21	.42	37.0	63.3	A1	0
.90	.45	1.33	1.7	1.78	2.5	C-.03	.39	65.4	71.1	A4_D	0
.09	.42	1.50	3.0	1.72	3.2	D-.15	.39	40.7	67.0	A3_E	0
.06	.34	1.62	2.2	1.71	2.5	E-.08	.46	55.6	60.8	F11_H	0
.44	.42	1.44	2.6	1.64	2.8	F-.09	.39	48.1	66.6	A3_F	0
-.90	.31	1.21	.7	1.54	1.6	G .29	.47	59.3	62.0	F17_F	0
.73	.36	1.45	1.6	1.44	1.6	H .03	.44	61.5	64.2	A5_C	0
.35	.41	1.43	1.3	1.35	1.1	I-.08	.39	74.1	74.1	A4_A	0
.99	.44	1.07	.4	1.42	1.5	J .24	.38	77.8	71.2	G1	0
-.09	.42	1.23	1.4	1.40	1.8	K .13	.39	55.6	67.8	A3_D	0
.80	.43	1.16	.9	1.39	1.6	L .18	.38	59.3	69.4	A2	0
-.76	.29	1.13	.6	1.39	1.4	M .35	.51	51.9	55.0	F15_A	0
-.42	.54	1.22	.6	1.35	.8	N-.01	.30	85.2	85.2	D12	0
-.50	.32	1.22	1.0	1.33	1.3	O .33	.49	59.3	55.0	F11_C	0
.41	.48	1.21	.7	1.32	.9	P .04	.33	81.5	81.5	A5_B	0
.26	.42	1.22	1.4	1.30	1.5	Q .16	.39	51.9	66.2	A5_D	0
-.24	.37	1.27	1.0	1.29	1.1	R .16	.42	66.7	67.7	F13_G	0
.29	.34	1.24	1.0	1.29	1.2	S .25	.46	55.6	60.8	F11_A	0
-.84	.44	1.27	1.0	1.28	.9	T .04	.37	70.4	75.0	B3	0
-.45	.43	1.25	1.4	1.22	.9	U .14	.38	63.0	69.7	A3_B	0
.48	.39	1.23	.9	1.25	.9	V .17	.41	66.7	70.5	F11_F	0
-.05	.48	1.16	.5	1.25	.7	W .12	.33	81.5	81.5	D9	0
-.57	.39	1.24	.9	1.23	.9	X .17	.41	51.9	68.6	C10	0
-.04	.33	1.23	1.0	1.23	1.0	Y .28	.47	51.9	58.8	F11_B	0
-.45	.43	1.23	1.3	1.16	.7	Z .17	.38	63.0	69.7	A3_A	0

Figure 8: Item misfit order

Journal of Theoretical and Applied Information Technology
15th January 2017. Vol.95. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 4: Item measure order

Entry No.	Total score	Count	Mea-sure	Model S.E	INFIT		OUTFIT		PT MEA. CORR.	Item
					MNSQ	ZSTD	MNSQ	ZSTD		
A. Software Development Process										
1	66	27	-1.4	0.38	1.67	2.41	0.8	2.7	-0.21	A1 Familiarity
2	91	27	0.8	0.43	1.16	0.91	0.39	1.6	0.18	A2 Understanding
3	98	27	-0.45	0.43	1.23	1.31	0.16	0.7	0.17	A3_A Requirement
4	98	27	-0.45	0.43	1.25	1.41	0.22	0.9	0.14	A3_B Analysis/Design
5	94	27	0.26	0.42	1.09	0.61	0.05	0.3	0.32	A3_C Construction
6	96	27	-0.09	0.42	1.23	1.41	0.4	1.8	0.13	A3_D Testing
7	95	27	0.09	0.42	1.5	3.01	0.72	3.2	-0.15	A3_E Implementation
8	93	27	0.44	0.42	1.44	2.61	0.64	2.8	-0.09	A3_F Maintenance
9	80	27	0.35	0.41	1.43	1.31	0.35	1.1	-0.08	A4_A Waterfall
10	83	26	-0.81	0.44	1.13	0.51	0.08	0.4	0.19	A4_B Iterative
11	69	25	0.02	0.38	1.49	1.31	0.83	1.8	-0.02	A4_C Spiral
12	87	26	0.9	0.45	1.33	1.71	0.78	2.5	-0.03	A4_D Agile
13	69	25	1.33	0.43	1.14	0.61	0.17	0.6	0.23	A4_E Prototype
14	80	26	-0.39	0.54	0.92	0	0.94	0	0.41	A5_A Heuristics
15	80	27	0.41	0.48	1.21	0.71	0.32	0.9	0.04	A5_B Formal
16	74	26	0.73	0.36	1.45	1.61	0.44	1.6	0.03	A5_C Prototyping
17	94	27	0.26	0.42	1.22	1.41	0.3	1.5	0.16	A5_D Agile
18	96	27	-0.09	0.42	1.06	0.51	0.08	0.4	0.32	A6 Best practices
19	99	27	-0.64	0.44	0.84	-0.8	0.8	-0.7	0.53	A7 Domain knowledge
20	94	27	-1.46	0.37	0.79	-0.8	0.96	-0.1	0.56	A8 Technologies
21	94	27	0.26	0.42	0.96	-0.2	0.91	-0.4	0.44	A9 Tool usage
B. Community of Practice										
22	94	27	0.26	0.42	0.98	-0.1	0.94	-0.2	0.43	B1 Transfer/Share
23	91	27	-1.26	0.38	0.94	-0.2	0.92	-0.3	0.48	B2 Relationship/Trust
24	86	27	-0.84	0.44	1.27	1.01	0.28	0.9	0.04	B3 Engaged
25	82	27	0.01	0.41	1.15	0.61	0.16	0.6	0.22	B4 Common interest
26	89	27	-0.82	0.36	0.79	-0.8	0.79	-0.9	0.63	B5 Common skill/expertise
27	90	27	-1.19	0.39	0.83	-0.7	0.81	-0.7	0.58	B6 Communication
28	86	27	1.9	0.52	0.84	-0.5	0.68	-0.6	0.49	B7 Collaboration
29	85	27	-0.47	0.4	1.18	0.71	0.22	0.8	0.22	B8 Technology support
C. Knowledge Management Process										
30	87	27	1.64	0.49	0.88	-0.4	0.8	-0.4	0.46	C1 Knowledge capture
31	86	27	1.9	0.52	0.96	0	0.89	-0.1	0.36	C2 Convert information
32	83	27	-0.22	0.44	1.12	0.51	0.17	0.6	0.22	C3_A Formal document
33	76	27	0.7	0.33	1.08	0.41	0.09	0.4	0.42	C3_B Informal document
34	81	27	0.18	0.35	0.95	-0.1	0.95	-0.1	0.49	C3_C Internal knowledge
35	81	27	0.18	0.39	0.88	-0.3	0.89	-0.3	0.52	C4 Corporate memory
36	81	27	0.18	0.39	0.96	0	0.98	0	0.45	C5 Software phases
37	82	27	0.06	0.34	0.74	-1.1	0.74	-1.1	0.69	C6 Repositories
38	91	27	0.8	0.43	0.88	-0.7	0.8	-0.8	0.52	C7 Keywords
39	92	27	0.62	0.43	0.75	-1.7	0.66	-1.7	0.66	C8 Mapping
40	85	27	-0.47	0.4	0.91	-0.2	0.88	-0.3	0.49	C9 Visual representation

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

41	86	27	-0.57	0.39	1.24	0.91	0.23	0.9	0.17	C10 Distributed
42	82	27	0.01	0.41	1.19	0.71	0.21	0.7	0.18	C11 Knowledge push
43	86	27	-0.57	0.39	1.06	0.31	0.07	0.3	0.36	C12 Profiling
44	84	27	-0.35	0.42	1.11	0.51	0.11	0.4	0.23	C13 Awareness
45	88	27	-1.03	0.41	1.06	0.3	0.96	-0.1	0.35	C14_A Best practices
46	80	27	0.31	0.37	1	0.11	0.02	0.2	0.43	C14_B Document search
47	83	27	-0.08	0.35	0.97	0	0.96	0	0.47	C14_C Expert locator
48	81	27	0.18	0.39	1	0.11	0.01	0.1	0.4	C14_D Advanced search
49	76	27	1.2	0.44	0.93	-0.2	0.96	0	0.4	C15 Personalization
50	83	27	-0.42	0.54	0.89	-0.1	0.87	-0.1	0.45	C16 Knowledge reuse
51	84	27	-0.59	0.49	1.06	0.31	0.01	0.2	0.33	C17 Improve performance
52	80	27	0.41	0.48	0.96	0	0.99	0.1	0.37	C18 Innovation
D. Experience Factory										
53	82	27	-0.05	0.48	1	0.1	0.96	0	0.34	D1 Reusing experience
54	78	27	0.71	0.42	1.08	0.41	0.14	0.5	0.29	D2 Package experience
55	81	27	0.18	0.35	1.2	0.81	0.2	0.8	0.26	D3 Evaluate
56	78	27	0.6	0.37	1	0.11	0.02	0.2	0.42	D4 Support
57	79	27	0.78	0.53	1	0.21	0.03	0.2	0.32	D5 Integration
58	87	27	-0.94	0.42	1.01	0.11	0.06	0.3	0.37	D6 Characterize
59	85	27	2.19	0.57	0.78	-0.5	0.5	-0.8	0.55	D7 Set goals
60	84	27	-0.59	0.49	1.14	0.51	0.15	0.5	0.15	D8 Execute
61	82	27	-0.05	0.48	1.16	0.51	0.25	0.7	0.12	D9 Choose process
62	86	27	1.9	0.52	0.79	-0.6	0.61	-0.8	0.54	D10_A automation
63	86	27	1.9	0.52	0.8	-0.6	0.56	-0.9	0.55	D10_B collaboration
64	85	27	-0.72	0.46	0.88	-0.3	0.77	-0.5	0.46	D10_C reliability
65	88	27	-1.03	0.41	0.97	0	0.92	-0.2	0.42	D11 Analyze
66	83	27	-0.42	0.54	1.22	0.61	0.35	0.8	-0.01	D12 Package
E. Perceived Benefits										
67	89	27	1.19	0.45	1.03	0.2	0.87	-0.3	0.38	E1_A Create values
68	83	27	-0.42	0.54	0.96	0.1	0.83	-0.2	0.31	E1_B Improvement
69	87	27	-0.94	0.42	1.08	0.4	0.94	-0.1	0.31	E1_C Quality
70	79	27	0.58	0.44	0.98	0.01	0.03	0.2	0.38	E1_D Cost benefit
71	85	27	-0.72	0.46	0.87	-0.3	0.86	-0.3	0.51	E1_E Reuse best practices
72	85	27	2.19	0.57	0.95	0	0.8	-0.2	0.35	E1_F Learning
F. Technology and Infrastructure										
73	80	27	0.41	0.48	1.09	0.41	0.15	0.5	0.19	F1 KM enabler
74	83	27	-0.13	0.39	0.86	-0.4	0.86	-0.4	0.54	F2_A Secure
75	86	27	-0.84	0.44	0.91	-0.2	0.89	-0.2	0.47	F2_B Organize
76	85	27	-0.72	0.46	0.82	-0.5	0.78	-0.5	0.57	F2_C Search knowledge
77	83	27	-0.42	0.54	1.08	0.3	0.87	-0.1	0.17	F2_D Search experts
78	84	27	-0.59	0.49	0.92	-0.1	0.85	-0.2	0.39	F2_E Collaboration
79	84	27	-0.35	0.42	1.01	0.11	0	0.1	0.35	F2_F Automatic push
80	82	27	0.01	0.41	1.1	0.41	0.16	0.6	0.26	F2_G Personalization
81	84	27	-0.59	0.49	0.94	0	0.88	-0.2	0.44	F3 Formal structure
82	79	27	0.43	0.36	1.03	0.21	0.03	0.2	0.41	F4 Database modeling
83	91	27	0.8	0.43	0.9	-0.5	0.79	-0.9	0.51	F5 Search engine
84	89	27	-1.12	0.4	0.82	-0.7	0.8	-0.7	0.59	F6 Electronic repositories
85	80	27	0.41	0.48	0.77	-0.5	0.7	-0.7	0.6	F7 Automation support

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

86	75	27	1.3	0.42	0.91	-0.2	0.97	0	0.46	F8 RSS
87	70	27	-0.11	0.32	0.88	-0.4	0.88	-0.4	0.58	F9 Mobile
88	75	27	-0.45	0.3	1.06	0.31	0.08	0.4	0.45	F10 Social network
89	80	27	0.29	0.34	1.24	1.01	0.29	1.2	0.25	F11_A Audio conf.
90	83	27	-0.04	0.33	1.23	1.01	0.23	1	0.28	F11_B Web conf.
91	88	27	-0.5	0.32	1.22	1.01	0.33	1.3	0.33	F11_C Video conf.
92	79	27	0.43	0.36	0.87	-0.4	0.87	-0.4	0.56	F11_D Chat
93	83	27	-0.13	0.39	0.96	-0.1	0.96	0	0.45	F11_E Instant message
94	79	27	0.48	0.39	1.23	0.91	0.25	0.9	0.17	F11_F White boarding
95	91	27	0.8	0.43	0.95	-0.3	0.88	-0.4	0.45	F11_G Screen share
96	82	27	0.06	0.34	1.62	2.21	0.71	2.5	-0.08	F11_H Online meeting
97	90	27	-0.89	0.35	0.84	-0.6	0.81	-0.8	0.59	F12_A Messaging
98	80	27	0.28	0.32	0.85	-0.6	0.85	-0.6	0.61	F12_B Audio/video stream
99	75	27	0.87	0.34	0.8	-0.8	0.81	-0.7	0.62	F12_C Calendar
100	75	27	1.3	0.42	0.84	-0.5	0.82	-0.5	0.56	F12_D Files replication
101	85	27	-0.27	0.33	0.79	-0.9	0.79	-0.9	0.64	F13_A Forums
102	85	27	-0.47	0.4	0.91	-0.2	0.91	-0.2	0.5	F13_B Blog
103	87	27	-0.66	0.37	1.04	0.21	0.04	0.3	0.38	F13_C Wiki
104	76	27	0.79	0.35	1.03	0.21	0.03	0.2	0.42	F13_D RSS
105	76	27	0.64	0.31	1.21	0.91	0.15	0.7	0.37	F13_E Social network
106	80	27	0.29	0.34	1.21	0.91	0.22	0.9	0.27	F13_F File sharing
107	84	27	-0.24	0.37	1.27	1.01	0.29	1.1	0.16	F13_G Document management
108	80	27	0.31	0.37	1.21	0.81	0.18	0.7	0.23	F13_H Task management
109	86	27	1.9	0.52	0.83	-0.5	0.62	-0.7	0.52	F14 Collaboration
110	81	27	-0.76	0.29	1.13	0.61	0.39	1.4	0.35	F15_A Large repository
111	82	27	-0.56	0.28	0.75	-0.8	0.76	-0.7	0.67	F15_B 24x7
112	82	27	-0.51	0.3	0.69	-0.9	0.64	-1.1	0.69	F15_C Low setup and maintenance cost
113	83	27	-0.13	0.39	0.71	-1	0.7	-1.1	0.7	F15_D Adapt to workload
114	91	27	-1.26	0.38	0.69	-1.4	0.67	-1.5	0.72	F16_A Availability
115	87	27	-0.94	0.42	0.74	-0.9	0.72	-0.9	0.66	F16_B Reliability
116	87	27	-0.94	0.42	0.62	-1.5	0.58	-1.5	0.79	F16_C Elasticity
117	88	27	-1.03	0.41	0.63	-1.6	0.61	-1.5	0.78	F16_D Scalability
118	85	27	-0.34	0.36	0.71	-1.2	0.7	-1.2	0.71	F16_E Pay-per-use
119	88	27	-1.03	0.41	0.77	-0.9	0.75	-0.9	0.63	F17_A Increase collaboration
120	84	27	-0.64	0.28	0.91	-0.2	0.96	0	0.57	F17_B Price flexibility
121	83	27	-0.81	0.31	0.73	-0.9	0.7	-1.1	0.69	F17_C No upfront investment
122	86	27	-0.73	0.29	1.01	0.11	0.08	0.4	0.53	F17_D Hardware cost saving
123	85	27	-0.7	0.28	0.84	-0.4	0.87	-0.3	0.64	F17_E Software cost saving
124	85	27	-0.9	0.31	1.21	0.71	0.54	1.6	0.29	F17_F Operational cost saving
125	96	27	-0.09	0.42	0.82	-1.2	0.76	-1.3	0.59	F17_G Ability to grow/shrink

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

G. Organizational and Managerial Influences										
126	90	27	0.99	0.44	1.07	0.41	0.42	1.5	0.24	G1 Organization values
127	91	27	0.8	0.43	0.86	-0.8	0.88	-0.5	0.51	G2 Culture
128	84	27	-0.24	0.37	0.73	-1	0.72	-1.1	0.69	G3 Funding
129	87	27	-0.94	0.42	0.98	0.01	0.06	0.3	0.38	G4 Awareness
130	74	27	-0.06	0.27	0.8	-0.7	0.83	-0.5	0.69	G5 Reward
131	81	27	0.18	0.44	0.96	0	0.97	0	0.41	G6_A Promote innovation
132	83	27	-0.13	0.39	0.98	0	0.97	0	0.44	G6_B Decision making
133	79	27	0.48	0.39	0.86	-0.4	0.85	-0.4	0.55	G6_C Profit
134	84	27	-0.59	0.49	1.08	0.41	0.07	0.3	0.26	G6_D Efficiency
135	79	27	0.58	0.44	0.97	0	0.96	0	0.42	G7 Trainings
136	91	27	-1.26	0.38	0.79	-0.9	0.77	-0.9	0.62	G8 Key players
137	88	27	-1.03	0.41	0.78	-0.8	0.77	-0.8	0.62	G9 Leadership
138	79	27	0.39	0.33	0.83	-0.7	0.84	-0.6	0.61	G10 Coordination
139	79	27	0.43	0.36	0.91	-0.3	0.91	-0.3	0.52	G11 Control
140	83	27	-0.08	0.35	0.88	-0.4	0.88	-0.4	0.55	G12 Readiness
141	79	27	0.58	0.44	0.74	-0.7	0.67	-0.9	0.66	G13 Measurement
MEAN	84.1	26.9	0	0.41	1	0.1	1	0.1		
S.D.	5.9	0.3	0.79	0.06	0.2	0.8	0.26	0.9		

```
     Table of STANDARDIZED RESIDUAL variance (in Eigenvalue units)
                                               -- Empirical--          Modeled
Total raw variance in observations     =   196.2  100.0%                100.0%
  Raw variance explained by measures   =    56.2   28.7%                 27.5%
    Raw variance explained by persons  =    20.1   10.2%                  9.8%
    Raw Variance explained by items    =    36.1   18.4%                 17.6%
  Raw unexplained variance (total)     =   140.0   71.3%   100.0%        72.5%
    Unexplned variance in 1st contrast =    13.0    6.6%     9.3%
    Unexplned variance in 2nd contrast =    10.9    5.5%     7.8%
```

Figure 9: Table of Standardized Residual

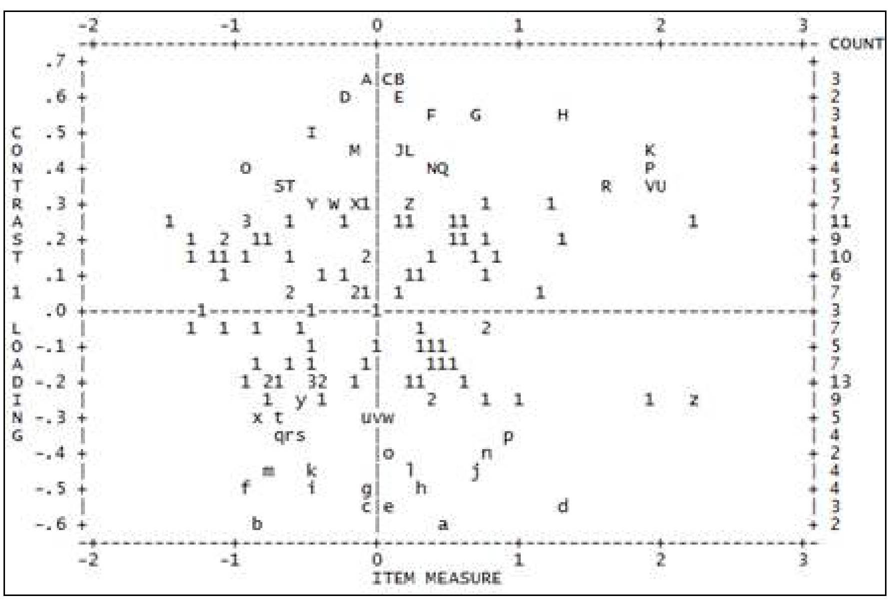

Figure 10: Standardized residual contrast 1 plot