Analyzing Data for Grounding a Problem Theory in Software Engineering

Sara Shahzad[1], Shaukat Hayat[1], Shah Nazir[1], Huma Javed[1], Yousaf Hayat[2], Muhammad Aamir[1], Ali M.Serat[1]

[1]Department of Computer Science, University of Peshawar, Pakistan
[2]Department of Mathematics, Statistics and Computer Science, University of Agriculture, Peshawar
{sara, shahnzr, humajaved15}@upesh.edu.pk, s.hayat.ibms@gmail.com
{serat_af, aamirmuhammad_26}@yahoo.com

Abstract: Empirical research aims to understand data trends and draw conclusions regarding the phenomena under study. Aim of this paper is to present and understand various steps and stages of data analysis which are performed at the initial stage of Grounded Theory (GT) process. The ultimate objective of this research is to understand effect of teaching and instruction on the learning outcome of software engineering students. To understand this phenomenon a number of variables have been identified. Associations among different variables of interest are explained by applying Chi-square test on primary data collected using questionnaires. The results have been interpreted at 5% level of probability. The data analysis shows a significant association between teachings of theoretical concepts along with practical implementation level exercises covering all different phases of software engineering process; from analysis to implementation. Results are also drawn to show learning outcome of students at different phases of the said course. Furthermore, it reflects areas of software engineering course which are weaker from the point of view of learning outcome of students or lacking attention in teaching. In addition this research can be extended to understand problems relating to different aspects of any field of study.

[Sara Shahzad, Shaukat Hayat, Shah Nazir, Huma Javed, Yousaf Hayat, Muhammad Aamir, Ali M.Serat. **Analyzing Data for Grounding a Problem Theory in Software Engineering.** *Life Sci J* 2014;11(10s):61-69]. (ISSN:1097-8135). http://www.lifesciencesite.com. 11

Keywords: Qualitative Research, Grounded Theory, Software Engineering Education, Chi-Square

1. Introduction

Qualitative research methods are applicable in numerous diverse academic disciplines. Its main aim is to provide insight into phenomena relating to areas of study where human behavior has to be judged and understood. Grounded Theory (GT) in itself is a qualitative research method in which research theory is generated and acquired from the data through a systematic procedure (Glaser, 1978, Glaser 1998). This theory (GT) focuses on generation of a new theory based on collection and analysis of data during the research process. For validity of the newly emerged theory it is important to avoid pre-conceived ideas and mindset which results in a thorough literature review before starting the study (Goulding, 2002). Grounded theory process is generally being used successfully to study and observe relationships among people and different phenomena with in a society. It is successfully used in diverse fields of study; in social sciences (Parry,1998; Weed 2009 and Holt,2010), in medical sciences (Thomson et al., 2013) and in engineering sciences (Adolph et al.,2012; Kroeger et al., 2014). In the field of information and education sciences grounded theory approach has been used to investigate student behavior in graduate and undergraduate study programs. Various studies have been conducted by instructors as well as by the researchers to analyze the output of teaching computer science related subjects for understanding students' behavior, learning satisfaction, and educational gain from graduate and undergraduate courses. Wellons and Johnson (Wellons & Johnson, 2011) used grounded theory to analyze student learning problems in order to plan how the course should be structured and organized. The objective of the overall research is aimed at analyzing a fundamental software engineering course and to make sure that the students are learning the necessary theory and skills about software development process; which they need as a basis of their training as professional software engineers.

A number of empirical studies have been presented by researchers which are conducted on the platform of software development industry to investigate the various aspects of software engineering. De la Vara et al. (De la Vara, 2011) presented an empirical study on the significance of software quality requirements in an industry. The study emphasized important and diverse managing and constructive roles in a development team and their respective duties. The study is based on data collected through questionnaire based survey feedback. The study also helped in measuring the importance of the quality requirements for each concerned role, nature of project, and the intensity for application realm is also defined. Similar studies have been conducted by many other researchers including Sadraei et al. (2007) investigated the effort sharing phenomena within projects. Curtis et al. (1988) conducted a study in different organizations to take an insight into the design process which is used in software projects. Damian et al. (2004) presented various changes in the practices of requirement engineering and scrutinized its effects on overall software development activities. But studies which are conducted on academic grounds like the one presented by (Wellons & Johnson, 2011) are very less. Even in this study only programming skill and problems have been identified, it is not covering the overall spectrum of software engineering including the theory behind software engineering practices.

This research study looks at software engineering education from a broader perspective, giving importance to the theory of software engineering process and its practical use, which is necessary to give an organized and structured view to perspective software engineers. The research starts with adopting Glaser's grounded theory (GT) approach (Glaser and Strauss, 2009) to find out the deficiencies and weakness in the organization and curriculum design of an undergraduate software engineering foundation course. The presence of such deficiencies and weakness result in failing to meet the objectives to be achieved from teaching of these courses to perspective software engineers

2. Materials and Methods

The study uses primary data that were collected through a well-structured questionnaire from undergraduate students majoring in computer science at a public sector university. The goal of this research was to evaluate the students' learning outcome and the effectiveness of the teaching approach (Kyriakides et al., 2013) for a SE foundation course. The research was prompted due to an informal understanding, through final year capstone project reports, that the students were having problem in understanding the basics of SE processes (Schmidt,2013), and hence were unable to correlate theory with practice. The questionnaires were distributed among a group of undergraduate students who recently attended a foundation course of SE. The questionnaire contained eleven main questions, designed as multiple choice questions and scaled on rank order. The major emphasis of the course was on teaching general SE skills, software development. The data were collected in order to check the effectiveness of the teaching and the learning outcome of the course and were analyzed using Statistical Package for Social Sciences (SPSS v.20).

The current study uses qualitative variables so all the results are presented in terms of counts and percentages (Siegel, 2011). To test the hypotheses in empirical studies in SE research the worth of statistical testing plays an important role (Dybå et al., 2006). In this connection, Chi-square test was employed to test the associations (Curtis and Youngquist,2013) of SE concepts with other related attributes.

For the convenience of the reader, a Chi-square test is defined as:

$$\chi^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{(O_{ij}-e_{ij})^2}{e_{ij}}$$

which, under the null hypothesis, follows a χ^2-distribution with $(r-1)(c-1)$ degrees of freedom. In the given equation the terms O_{ij} and e_{ij} indicates the observed and expected frequencies respectively (Nachmias and Nachmias,1992).

2.1 Research Hypothesis

Following are the hypotheses selected for the research under study. In statistical terms, the null (H_0) and alternative (H_1) hypotheses can be expressed as:

H_0: there is no significant association between theoretical and practical concepts of SE course,

H_1: there is significant association between theoretical and practical concepts of SE course.

The following points of interest (variables) were considered while testing the hypotheses:

- Depth of teaching in theoretical concepts of software development process
- Depth of teaching in practical implementation of the theoretical concepts
- Learning outcome from the teaching of theoretical concepts
- Learning outcome and depth of understanding regarding practical exercises
- Knowledge grading about SE course
- Various process models knowledge and implementation ability
- Extent of practical exercises supporting the theoretical class assignment.
- Level of confidence of students about knowledge gained relating to software development process
- Level of confidence of students about knowledge gained relating to team work.
- Rate of knowledge to conduct good tests against developed software/any module.
- Respondent perception about any changes in current SE course.

2.2 Research Problem

There is a continuous need to learn and enhance SE techniques to deal with the advancing computer technology and the dynamic software business market needs. This understanding makes it more crucial to have a proper college and university level training of perspective software engineers. The young graduates of SE should have core knowledge and concepts of different dimensions of the SE. For this purpose it is important to find out if the required SE related knowledge is being dispersed properly by the instructors and the concepts are being grasped by the students.

To investigate the problem under study, this research was conducted to study a number of variables of interest addressing both the practical and theoretical concepts of SE, in addition to establish an association among these variables.

3. Results and Discussion

The study aimed to find out statistical associations among different variables that are selected for achieving the required objectives of the study. The variables are tested using Chi square test (Nachmias and Nachmias,1992) and the associations are calculated among the defined variables. The following section provides the details of the whole process of data analysis and results interpretation. The results are presented in two separate sections: one category showing the significant results while the other one representing non-significant results in terms of association. All the results were presented in contingency tables in terms of counts and percentages, and are presented with the help of simple bar diagrams. All the results are interpreted at 5% level of significance.

3.1 Significant Associations

3.1.1 Knowledge grading, work ability, confidence level and teaching depths in different dimensions of the software engineering course

Students' perceptions were recorded about the SE course that was taught to them during regular academic session. Teaching sessions included theoretical concepts, practical work class assignments, and project base hand-on practices. Additionally, the students' feedback was also recorded about the knowledge level, their practical work ability and confidence they have in different sub-area of SE. Association test is applied and the associative results are shown in Table 1 and Figure 1 (A, B). The results suggest that any change in knowledge about SE course will significantly affect the participant's practices in its different dimensions. It was recorded that maximum percentage of the respondents (22.2%) in bi-variate analysis about the knowledge grading and practices in different dimensions of SE course falls in "average" category. It further suggests (Figure1A) that the theory teaching and practical teaching in requirement engineering (RE) taught to graduate students during semester are significantly ($P < 0.01$) related to each other and in the long run by increasing the sample size, a significant association could be established between those two parameters. Furthermore, it was observed (from the bivariate analysis) that maximum feedback (27.8%) of the total students lies in "more detail" category of the teaching depth. Similarly, Figure1B indicates the perception of students regarding Teaching depth in both theory and practical analysis and design methods, and there is highly significant ($P < 0.01$) relationship between the two attributes. These results suggests that any change in teaching depth in theory of analysis and design methods will bring significant change in teaching depth in practical of analysis and design methods. Maximum Associative response (33.3%) of the total students regarding teaching depth in both variables falls in the category of "enough to have general concept". A highly significant ($P < 0.01$) association was recorded between SE course knowledge grading, and Software development ability & confidence which suggests that knowledge and ability to work on real projects developments are directly related to each other. Therefore, it is concluded that if knowledge about SE course is increased then the student's ability in software development will also be increased. Maximum associative responses at average level of knowledge (38.9%) of the total students are confident about their practical work ability on software projects but look forward for more guidance in software development activities.

Table 1. Associations in different dimensions of the software engineering course

S.No	Variable-A	Variable-B	χ^2	P-value	Association Status
1.	Knowledge grading about SE course	Practices in different dimensions of SE course	22.5	0.032	Significant
2.	Teaching depth in theory of RE	Teaching depth in practical of RE	29.184	0.001	Highly significant
3.	Teaching depth in theory of analysis and design methods	Teaching depth in practical of analysis and design methods	22.767	0.007	Highly significant
4.	SE course knowledge grading	Software development ability and confidence	25.104	0.003	Highly significant

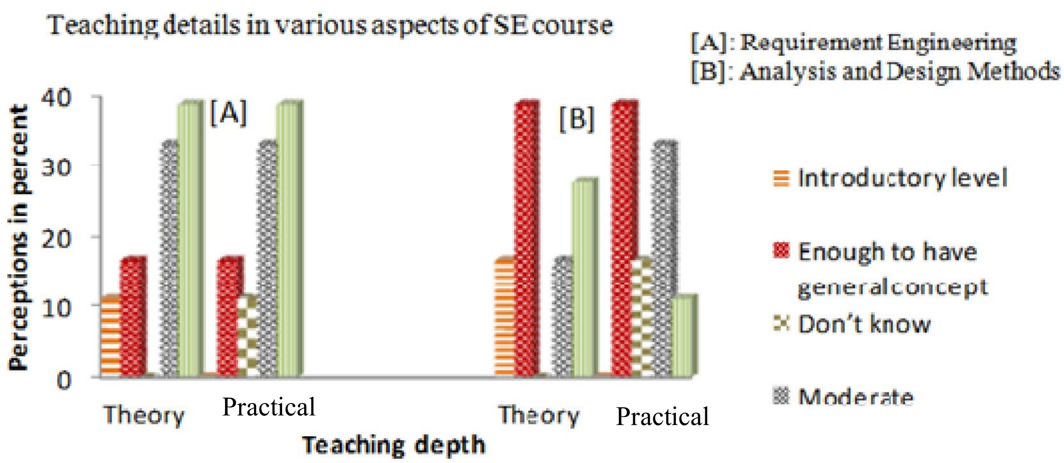

Figure 1. Associative responses of the graduates about teaching depth of SE course

3.1.2 Learning capability about SE Course

Learning capability of the students about various sub-area of the SE course was computed. This capability is concerned about theoretical concepts leads to its respective practical area. Chi-square test is applied between theory and practical learning about the various sub-parts of the SE course and the results are displayed in Table 2 and Figure 2 (A – D).

It is evident that a significant ($P < 0.05$) association was recorded between the theory and practical learning in requirement engineering (RE). The graduate's students as individual learning in theoretical concepts are same as they learned during practical work. It was recorded that maximum associative response (16.7%) of the total students falls in "learn a lot" category (Figure 2A) suggesting that any increase in theory learning of requirement engineering will significantly enhance the practical capability of students in RE. Similarly, it depicts that theory and practical learning in process models are dependent ($P < 0.05$) indicating that the learning performance of SE students will be significantly affected when there is any change in the practical learning of process model and vice versa. It further reveals that maximum associative response (22.2%) of the total students about their learning as well as practical abilities of the process models falls in "moderate working knowledge" (Figure 2B). In addition, Student's perceptions regarding learning in both theory and practical in object oriented concepts and technology are same (Figure 2C) while the association between theory and practical is highly significant ($P < 0.01$) suggesting that both the attributes of SE course are dependent on each other. Furthermore, a highly significant ($P < 0.01$) relationship between theory and practical learning in software project management implying that any change in the two factors of the management activities (software project management) will significantly affect its value in terms of learning capabilities. Maximum associate response about the software project management was recorded in the "expert" category of learning (Figure 2D).

Table 2. Learning Outcomes (theory and practical) of the Software Engineering course in its different dimensions

S.No	Variable-A	Variable-B	χ^2	P-value	Association Status
1.	Theory Learning in RE	Practical Learning in RE	18.360	0.031	Significant
2.	Theory Learning in process Models	Practical learning in Process Model	18.360	0.031	Significant
3.	Theory Learning in object oriented concept and technology	Practical Learning in object oriented concept and technology	20.143	0.003	Highly significant
4.	Theory Learning in software project management	Practical Learning in software project management	22.185	0.008	Highly significant

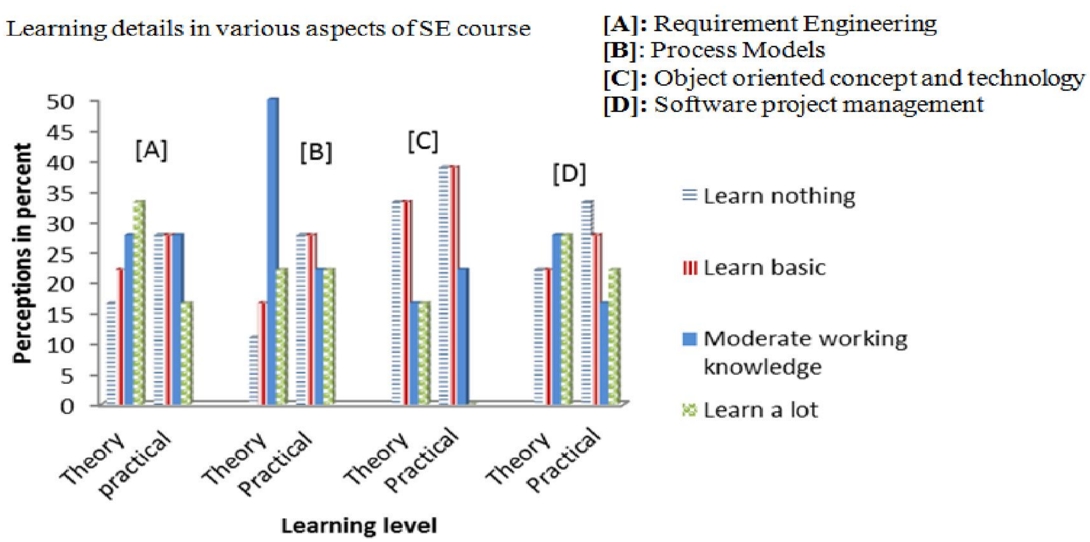

Figure 2. Associative responses of the graduates about learning of the SE course

3.1.3 Process models implementation ability and confidence on the basis of knowledge gained during study SE course

The ability of the graduates to work on software development based on knowledge they evoke while study the SE course and use of different traditional process models implementation confidence in practical environment. The association test is applied and the perceptions of the participated graduates are recorded. Table 3 indicate the results of the ability of the graduates to work on software development based on knowledge they evoke while studying the SE course and use of different traditional process models implementation confidence in practical environment. The results show that a highly significant ($P < 0.01$) association exists between waterfall model and the gained knowledge and software development confidence and ability, suggesting that if the knowledge increase the confidence about model implementation will also be increased. In total, associative perception (27.8%) of the total graduates having a lot of knowledge was observed. This increase their confidence and have the ability to implement the under study model (Waterfall model) in practical environment. Similarly, according to the results of the total graduates (27.8%) having moderate knowledge and have the ability to endorse the spiral model in real software development projects but they were also demanding more control to empower the implementation skills. Improvement in any factor will positively affect each other as evident from the P-value (0.016) of a Chi-square test suggesting highly significant association between the two attributes under study. In addition, the same and maximum percentage (27.8%) of the total graduates showing that they have no clear knowledge about the current processes but have the ability to implement agile concept in real software development, additionally they added that if more guidance are provided then it will increase their confidence as evident from the significant ($P < 0.05$) Chi-square test. Furthermore, a highly significant ($P < 0.01$) association was observed for other related concepts and the gained knowledge and software development confidence and ability, which suggest that these are dependent on each other and its further suggested that if the knowledge increases about model implementation then it will also increase the ability and confidence of respondents to endorse the models implementation in real environment.

Table 3. Different process models implementation ability in real world environment on the basis of knowledge gained during study SE course.

S.No	Variable-A	Variable-B	χ^2	P-value	Association Status
1.	Gained knowledge and software development confidence and ability	Waterfall Model	26.786	0.008	Highly significant
2.	-do-	Spiral Model	20.241	0.016	Significant
3.	-do-	Agile Software process	19.623	0.020	Significant
4.	-do-	Other related concepts	19.607	0.003	Highly significant

3.2 Non-Significant Associations

This section presents non-significant results in graphical form. Insignificance was calculated using Chi square test. Calculated Chi-square along with p-values are also shown in the graphs.

Teaching details in different sub-area of Software Engineering course

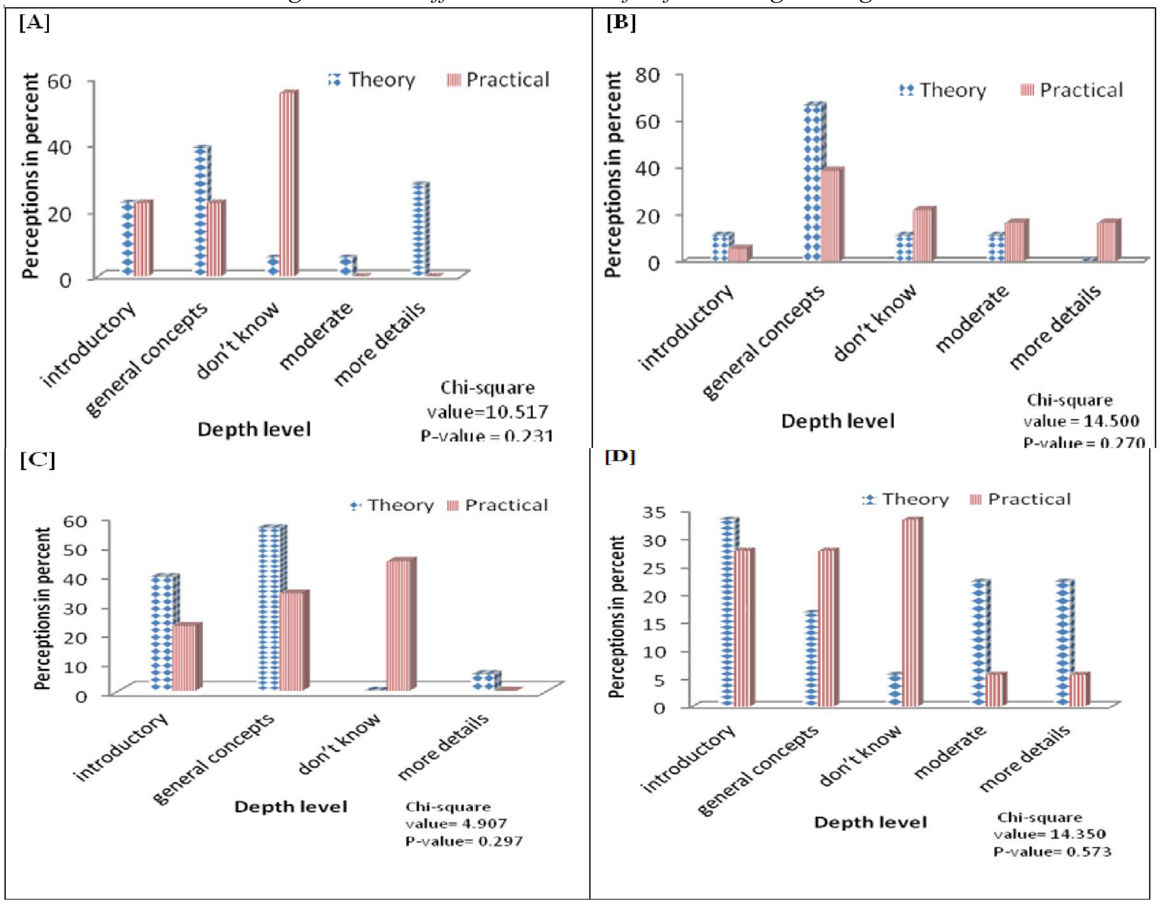

Figure 3. Perceptions regarding detail of teaching and practical of SE course: [A] Software testing, [B] software development process models, [C] object oriented concept and technology, [D] software project management.

Learning outcomes in different sub-areas of Software Engineering course

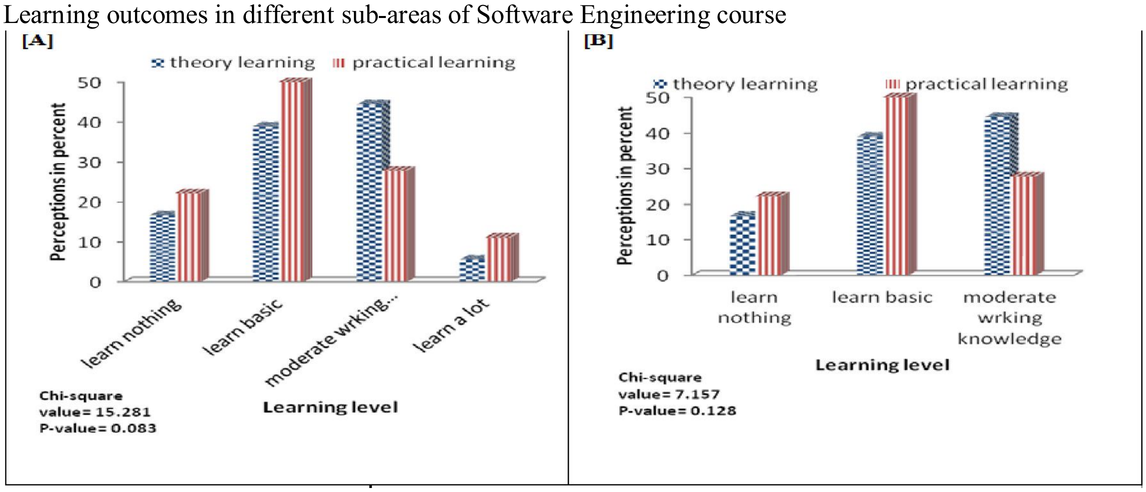

Figure 4. Perceptions regarding learning detail in theory and practical of SE course: [A] Analysis and Design methods [B] Software Testing.

3.2.1 Class assignments support to practical learning and change suggested about SE course for effective teaching

The rest of the part of the SE course about class assignments support to practical learning in association with team work activities having insignificant results. It suggests that both variables are not affecting each other's and may not supporting learning performance of the graduates. While Figure 5 shows the perception of graduate students about a better change for effective learning about the SE course. It is evident that maximum percentage (61.1%) of the respondents suggests that SE contents should be more practical to educate the academia graduates when delivering SE contents.

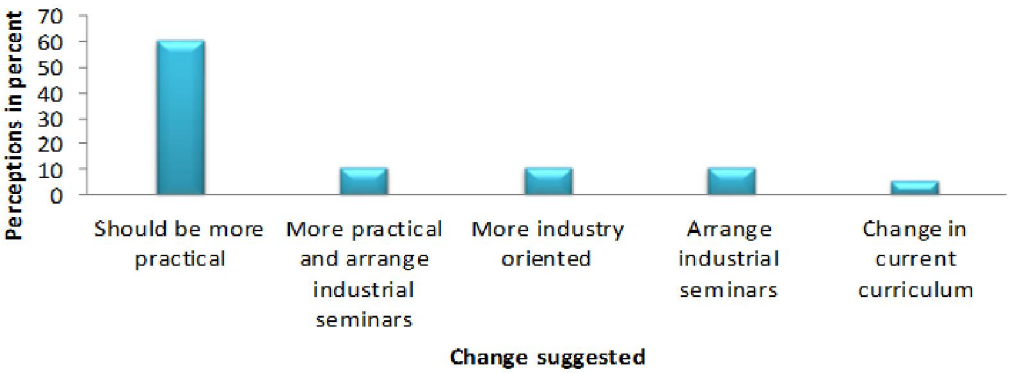

Figure 5. Change suggested by gradates to teach SE course

3.2.2 Ability to conduct test after learning testing techniques

The results showing (Figure 6) the respondent's perception about to perform a test on the basis of gained knowledge during study the testing techniques. It is evident that for each and every statement describing the testing knowledge of the respondents, maximum percentage (38.9%) of the total responses falling in conducting good testing ability against any developed software. Similarly (16.7%) of the total respondents having an average and (11.1%) having the ability to conduct very good test on developed software. These results suggest that the respondents having knowledge can handle the testing procedures easily.

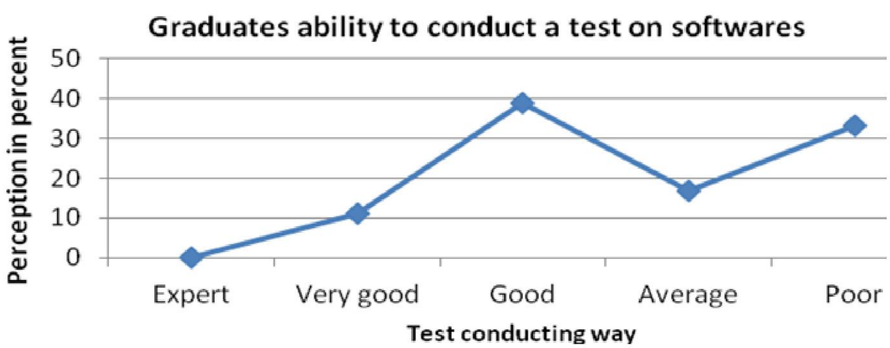

Figure 6. Testing knowledge to conduct test against developed software

4. Conclusion and recommendation

This study describes the relationships among various aspects of a graduate level SE course. The students' feedbacks were gathered based on their personal experience of learning and understanding SE through the said SE course. Based on the obtained results, it is concluded that there is a significant association among the various SE theoretical aspects and related practical exercises. Students show more learning outcome, understandability and implementation confidence in real world environment for all those concepts which are taught along with practical exercises. There are also some concepts which are not well understood by the students, or the students are not sure about the practical effectiveness in learning and its implementations. These are the areas of the part of the SE course which need more attention, including redesigning the course contents and structure and should be backed with effective pedagogy techniques and with more hands-on practices. These changes may provide significant and more pragmatic results in the study area.

This data analysis and interpretation is performed as part of a GT process. These interpretations and results will provide a major input for generating the required theory. This effort is only the part of the first cycle of the overall GT process. Reporting the details of this process is aimed to provide a general understanding on how one can analyze data collected using questionnaires and how it is to be interpreted.

References

1. Adolph, S., P. Kruchten and W. Hall. (2012). Reconciling perspectives: A grounded theory of how people manage the process of software development. Journal of Systems and Software, 85(6):1269-1286.
2. Bohnet, I. C., B. Roberts, E. Harding and K. J. Haug. (2011). A typology of graziers to inform a more targeted approach for developing natural resource management policies and agricultural extension programs. Land Use Policy, 28(3): 629-637.
3. Curtis, B., H. Krasner and N. Iscoe. (1988). A field study of the software design process for large systems. Communications of the ACM, 31(11): 1268-1287.
4. Curtis, K. and S.T. Youngquist, Part 21: Categoric Analysis: Pearson Chi-Square Test. Air Medical Journal, 32(4): 179-180.
5. Damian, D., D. Zowghi, L. Vaidyanathasamy and Y. Pal. (2004). An industrial case study of immediate benefits of requirements engineering process improvement at the Australian Center for Unisys Software. Empirical Software Engineering, 9(1-2): 45-75.
6. De la Vara, J. L., K. Wnuk, R. Berntsson-Svensson, J. Sánchez and B. Regnell. (2011). An empirical study on the importance of quality requirements in industry. In SEKE (pp. 438-443).
7. Dybå, T., V. B. Kampenes and D.I. Sjøberg (2006). A systematic review of statistical power in software engineering experiments. Information and Software Technology, 48(8): 745-755.
8. Glaser, B. G. and A. L. Strauss, (2009). The discovery of grounded theory: Strategies for qualitative research. Transaction Books.
9. Glaser, B. (1978). Theoretical sensitivity: Advances in the methodology of grounded theory. Mill Valley, CA: Sociology Press.
10. Glaser, B. (1998). Doing grounded theory: Issues and discussions. Mill Valley, CA: Sociology Press.
11. Goulding, C. (2002). Grounded Theory: A Practical Guide for Management. In Business and Market Researchers: Sage Publications.
12. Holt, N. L. and Tamminen, K. A. (2010). Moving forward with grounded theory in sport and exercise psychology. Psychology of sport and exercise, 11(6): 419-422.
13. Hunziker, M. (1995). The spontaneous reafforestation in abandoned agricultural lands: perception and aesthetic assessment by locals and tourists, Landscape and Urban Planning, 31 (1-3): 399-410.
14. Kroeger, T. A., N. J. Davidson and S. C. Cook. (2014). Understanding the characteristics of quality for software engineering processes: A Grounded Theory investigation. Information and Software Technology, 56(2):252-271.
15. Kyriakides, L., C. Christoforou and C. Y. Charalambous. (2013). What matters for student learning outcomes: A meta-analysis of studies exploring factors of effective teaching. Teaching and Teacher Education, 36:143-152.
16. Morgan, A. C., Rudd, R. D. and Kaufman, E. K. (2013). Elements of an undergraduate agricultural leadership program: A Delphi study. Volume 12, Number 1–Winter 2013, 140
17. Nachmias, F. and D. Nachmias. (1992). Research methods in the Social Sciences. 4th ed. 1992: St. Martin Press, Inc. London, Melbourne Auckland
18. Oreszczyn, S. (2000). A Lane, The meaning of hedgerows in the English landscape: Different stakeholder perspectives and the implications for future hedge management. Journal of Environmental Management, 60(1):101-118.
19. Outhwaite, O., R. Black and A. Laycock. (2007). The pursuit of grounded theory in agricultural and environmental regulation: a suggested approach to empirical legal study in biosecurity. Law and Policy, 29(4): 493-528.
20. Parry, K. W. (1998). Grounded theory and social process: A new direction for leadership research. The Leadership Quarterly, 9(1), 85-105.
21. Sadraei, E., A. Aurum, G. Beydoun and B. Paech. (2007). A field study of the requirements engineering practice in Australian software industry. Requirements Engineering, 12(3): 145-162.
22. Schmidt, R.F. (2013). Software Engineering: Architecture-driven Software Development, 1st Ed. Morgan Kaufmann.ISBN: 0124078788, 9780124078789
23. Siegel, A.F.(2011). Chapter 17 - Chi-Squared Analysis: Testing for Patterns in Qualitative

Data, in Practical Business Statistics (Sixth Edition), Academic Press: Boston. p. 507-522.

24. Thomson, O. P., N. J. Petty and A. P. Moore. (2013). Osteopaths' professional views, identities and conception- A qualitative grounded theory study. International Journal of Osteopathic Medicine. PII: S1746-0689(13) 00171-5, DOI: 10.1016/j.ijosm.2013.12.002.
25. Thomson, O. P., N. J. Petty and J. Scholes. (2013). Grounding osteopathic research– Introducing grounded theory. International Journal of Osteopathic Medicine. http://dx.doi.org/10.1016/j.ijosm.2013.07.010.
26. Van Der Ploeg, J. D., H. Renting, G. Brunori, K. Knickel, J. Mannion, T. Marsden, K. De Roest, E. Sevilla-Guzmán and F.Ventura. (2000). Rural Development: From Practices and Policies towards Theory. Sociologia Rural-is, 40: 391–408.
27. Weed, M. (2009). Research quality considerations for grounded theory research in sport and exercise psychology. Psychology of Sport and Exercise, 10(5): 502-510.
28. Wellons, J., & Johnson, J. (2011). A Grounded Theory Analysis of Introductory Computer Science Pedagogy. Journal of Systemics, Cybernetics & Informatics, 9(6).

5/29/2014