2011 5th Malaysian Conference in Software Engineering (MySEC)

Software Engineering Curriculum:
A Systematic Mapping Study

Muhammad Manan Qadir, Muhammad Usman
Department of Computer Science and Software Engineering
International Islamic University Islamabad, Pakistan
{manan.msse202,m.usman}@iiu.edu.pk

Abstract - **Software Engineering (SE) discipline has come a long way since the 1968 NATO conference when the term SE was first used. Lot of work has been done for developing and revising SE curriculum and body of knowledge (e.g. SE 2004, GSwE2009, SWEBOK efforts). Different universities are developing and revising SE program and curricula at graduate and undergraduate levels all over the world. Large number of SE curriculum related efforts are being published in conferences such as CSEET, FIE and REET etc. There is a need to see the state of the art of this abundant literature. In this paper, we report a systematic mapping study conducted to synthesize and aggregate the SE curriculum related reported efforts and to provide a broad overview of the area. Systematic mapping studies are performed to evaluate quantity and types of primary studies in an area of interest in an unbiased and systematic manner.**

Keywords – Software Engineering Curriculum; Systematic Mapping Study;

I. INTRODUCTION

Software engineering (SE) is growing since 1968 and is now being considered as important computing field. In the beginning, Software engineering was conceived as a sub-area of computer science (CS) and usually one or two SE courses used to be the part of undergraduate CS curriculum. Software engineering evolved with the passage of time and is maturing into a discipline now. Different universities now offer SE programs at both undergraduate and graduate level. Lots of efforts have been reported about software engineering curriculum design, revision, implementation and assessment. Research in software engineering education depicts the efforts in courses level and as well as overall software engineering curriculum. Curriculum at a program level is reported in a study that puts forward three phase curriculum system [4] and focused on systematic knowledge and its organization. Curriculum at a course level is reported in [6] which Curriculum development for Aspect oriented software development (AOSD) which is a model curriculum in postgraduate curriculum from author's experiences. Lot of work has been done to mature software engineering education. Software engineering body of knowledge (SWEBOK) defines the field and describes the knowledge. IEEE-CS and ACM SE 2004 developed curriculum for software engineering which was a great effort. A graduate level curriculum GSwE2009 is indicated which provides the curriculum guidelines for graduate degree programs in software engineering.

Software engineering education research is promoting by different professional bodies such as IEEE, ACM and conferences such as CSEET (Conference on Software Engineering Education and Training), FIE (Frontiers in Education), EDUCON (Engineering Education Conference) and IEEE transaction on education etc. Curriculum of software engineering education evolved in terms of new design, revised, minor and major changes. Software engineering curriculum implementation and assessment in academia took place in different regions in all over the world.

Studies relevant to software engineering curriculum are abundant and dispersed. There is a need to see the state of the art in software engineering curriculum. In order to cope with this goal, we have conducted a systematic mapping study and reported in this paper. Systematic mapping study is also called scoping study stated in [1] and it is a type of secondary research. Systematic mapping studies are performed to evaluate quantity and types of primary studies in an area of interest in an unbiased and systematic manner. Proper procedure of executing systematic mapping study required sound planning.

The paper is organized as follows; in section 2, we presents background of systematic mapping, section 3 presents the systematic mapping process, section 4 presents the systematic mapping plan, section 5 presents the execution of systematic mapping plan, section 6 presents discussion on this study, section 7 presents conclusion, section 8 presents future research and at the end references are listed. There are 517 references of this mapping study, so that all references are not listed here.

II. BACKGROUND

Evidence is important and much practiced in medical science [1,2,3]. Evidence in Software engineering is getting

978-1-4577-1531-0/11/$26.00 ©2011 IEEE

much attention. Systematic mapping study is a technique in Evidence Based Software Engineering and has already been used in different sub-domains of software engineering i.e. software testing and requirement specification [3,5].

Systematic mapping requires sound planning, execution and analysis of systematic map. Data extraction conducts with the identification of categorization by screening (Title, Keywords and Abstracts).

In particular, Software engineering education has a rich literature in which lot of contributions towards curriculum is presented. Towards, overall software engineering curriculum, such type of work is not still presented in literature.

III. SYSTEMATIC MAPPING PROCESS

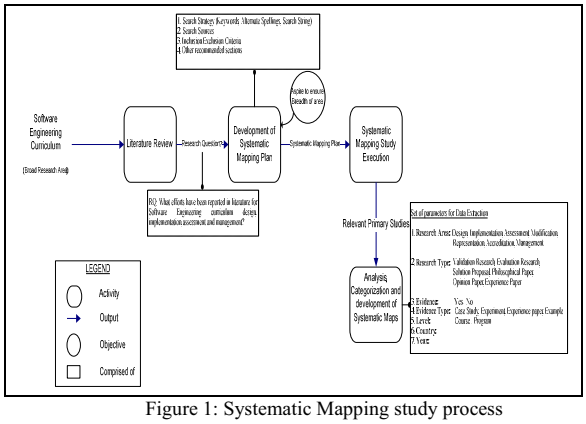

Figure 1: Systematic Mapping study process

To conduct a Systematic Mapping study, figure 1 summarizes the systematic mapping process which we have performed. Literature review leads to formulate research question which directs to develop a systematic mapping plan and executed to get the results.

In subsequent sections systematic mapping study plan, execution and analysis will represent the coverage of software engineering education literature.

IV. SYSTEMATIC MAPPING PLAN

Systematic mapping plan has several sections as stated in [1]. Systematic mapping plan which we have developed for this systematic mapping study is as follows;

A. Research Question

In order to meet our goal, we have formulated following research question;

What efforts have been reported in literature for Software Engineering curriculum design, implementation, assessment and management?

B. Search Strategy

The title, abstract and keywords of the articles in the included electronic databases and conference proceedings will be searched according to the following search strategy:

1) *Keywords:* Software engineering, Curriculum
2) *Alternate Spellings and Acronyms for Major Terms:* SE, Software Engineering; Curriculum: Curricula, syllabus, syllabi
3) *Search String:* Following general search string is formed;
((“SE” OR “Software Engineering”) AND (“Curriculum” OR “Curricula” OR “syllabus” OR “syllabi”))

The reason for formulating a generic string is that in mapping study we want to cover all literature in which software engineering curriculum related work is reported; work can be about curriculum design, implementation, assessment or management.

These search string will be modified according to the search criteria provided by different sources like IEEE Explore, EI Compendex and ACM digital Library etc. All searches and their responses will be saved. We will use Endnote to manage the citations and abstracts and to download/save the results.

C. Search Sources

Following databases will be searched for finding candidate primary studies:

- IEEE Explore
- ACM digital library
- Compendex
- ScienceDirect
- Inspec (IET)

Search results from all the databases will be saved separately. A master file comprising all searches will be created to remove duplicates. This master file will be used for applying inclusion/exclusion criteria and for data extraction.

D. Inclusion and Exclusion Criteria

1) Study inclusion Criteria

We will include studies:

a. That are directly related to software engineering curriculum
   i. This will include SE curriculum design, representation, management, assessment, analysis and accreditation efforts **AND**
b. That are published in a peer reviewed conference or journal.

2) Study exclusion Criteria

We will exclude studies

a. That are not related directly with software engineering curriculum **OR**
b. That are not published in a peer reviewed conference or journal

For the mapping study, the inclusion/exclusion criteria will be applied on abstracts. In case of any ambiguity conclusion and introduction of the papers will be read for resolution. In extreme case, we will go for full paper reading.

V. EXECUTION OF SYSTEMATIC MAPPING PLAN

We have conducted systematic mapping study according to plan which we have stated in previous section.

A. Conduction of Search

These search string was modified for different syntax of databases according to the search criteria. Search was conducted on 14-April-2011. Table 1 represents the result of searching from all databases. Search was conducted from year 1970-2011. We have found 1565 papers as a candidate studies from all selected sources.

TABLE 1: SEARCH RESULTS

Search Source	Year	Candidate Studies	Included Primary Studies
IEEE Xplore	1977 - Present	341	340
ACM	1970 - Present	488	205
Compendex	1970 - Present	608	435
Science Direct	1970 - Present	120	93
Inspec (IET)	1970 - Present	8	8
Total		1565	1081

B. Application of Inclusion and Exclusion Criteria

We have used Endnote reference manager to save the search results. Duplicate papers were removed at the time of importing. So, at the end of this activity we had 1081 research papers as mentioned in last column of Table 1.

We have selected abstracts of 1081 papers which fulfilled following keywords “SE”, “Software Engineering”, “Curriculum”, “Curricula”, “syllabus” and “syllabi”.

We have performed data extraction to extract the relevant studies. Research papers excluded which were not specific to software engineering discipline, courses. E.g. If studies come to different domain like information science, service engineering etc. to become part of software engineering curriculum, course or approach, so, that studies was included. If software engineering courses, approaches were used for another discipline was excluded. Software engineering courses, approaches design, implementation, and assessment took place in the domain of computer science was included. So, 447 papers were excluded after applying inclusion and exclusion criteria as mentioned in plan.

Therefore we have 517 research papers at the end which fulfilled our criteria. Results of our study are presented in next section.

C. Results

Research papers were found mainly categorized as program level and course level. Figure 2 shows the research papers as separately categorized.

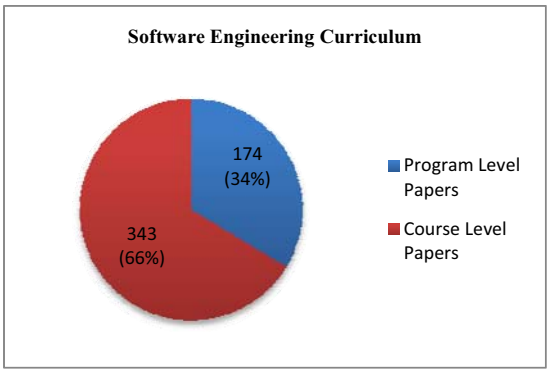

Figure 2: Software Engineering Curriculum

174 research papers found which were focusing on overall software engineering program while 343 research papers focused on course level.

Evidence also has been collected in this systematic mapping study. Evidence was categorized as case studies, experiments, experience reports and examples.

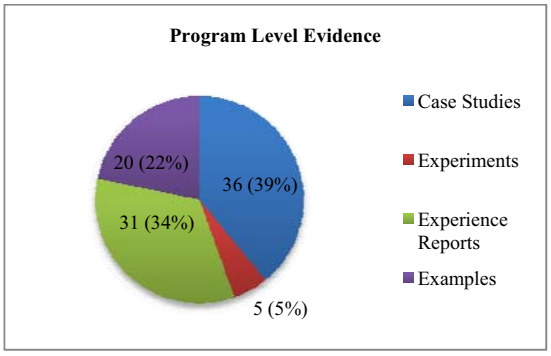

Figure 3: Evidence in Program level studies

Figure 3 depicts that Case studies were found at the large extent. In this way, there were 5 experiments found at program level studies.

Figure 4 shows that experience reports in course level studies has contributed at big level. Case studies also have in a large extent. There were 19 experiments found in course level studies which were more as in program level studies.

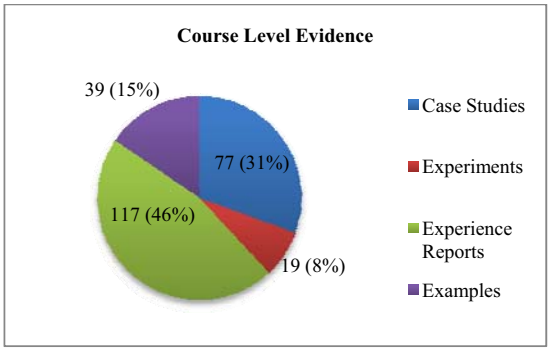

Figure 4: Evidence in Course level studies

To view the curriculum focus of research papers as mentioned in our research question, we have collected research papers focused on curriculum design, implementation, assessment, modification, representation, accreditation and management.

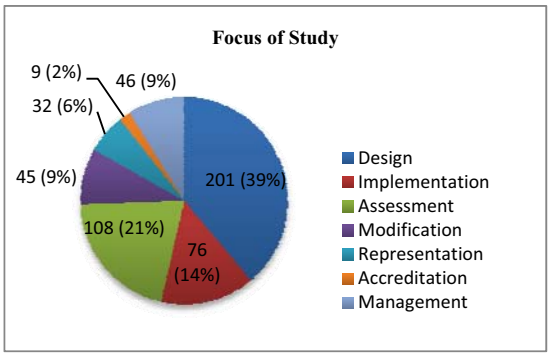

Figure 5: Focus of Studies at Program and Course Level

Figure 5 shows that huge efforts were reported in design of software engineering programs as well as courses. Assessment also reported in a great quantity and Accreditation efforts reported in 9 papers.

TABLE 2: YEAR VS. FOCUS OF STUDY

Year	DESIGN	IMPLEMENTATION	ASSESSMENT	MODIFICATION	REPRESENTATION	ACCREDITATION	MANAGEMENT
1976 - 1980	6	0	3	0	2	0	1
1981- 1985	2	1	0	1	1	0	0
1986 - 1990	10	6	6	0	2	0	0
1991 - 1995	16	3	10	4	1	0	5
1996 - 2000	50	11	12	7	6	3	13
2001 - 2005	47	25	35	16	5	5	16
2006 - 2011	70	30	42	17	15	1	11
Total	**201**	**76**	**108**	**45**	**32**	**9**	**46**

As we can see that in 1976-1986 software engineering education literature were in less quantity and focused less on design, implementation and so on. 1987-1997 contribution towards focus of study increased and in 1998-2011 there is a huge contribution towards software engineering education.

TABLE 3: LEVEL VS. FOCUS OF STUDY

Level	DESIGN	IMPLEMENTATION	ASSESSMENT	MODIFICATION	REPRESENTATION	ACCREDITATION	MANAGEMENT
Program Level	68	10	43	8	20	8	17
Course Level	133	66	65	37	12	1	29
Total	**201**	**76**	**108**	**45**	**32**	**9**	**46**

In table 3, Focus of study in research papers at course level was greater with respect to program level. Curriculum design is much focused at course level and program level and accreditation is less focused.

VI. DISCUSSION

By analyzing the results of this study, we can see that Software Engineering curriculum related efforts have been extensively reported in literature. Courses and approaches level papers are more than the program level papers which inform that much work have been done for the improvement of software engineering courses, topics and approaches.

In order to see the overall Software engineering education at program level as well as at course level since 1976 software engineering field were less matured as we can see in Table 2; from 1970's to 1995 fewer studies are reported pertaining several focus of study which is mentioned in figure 5. With the passage of time software engineering education improved with the help of research contributions.

However, lessons learned from this papers is keeping in view the less contribution towards software engineering education at program level, there is a need of improvement and more contribution from researchers, academia and other stakeholders to grow up the software engineering education at program level.

These findings open the direction for academia and researchers to see what has been done? And where is the need of improvement in software engineering education.

VII. CONCLUSION

Software engineering education literature efforts are in great quantity. To collect software engineering education at one place, we have conducted systematic mapping study. We have mapped the results of systematic mapping study in form of charts and tables that indicated the focus of software engineering curriculum and showed the number of contributions year wise vs. program and course level.
Contributions towards curriculum at course level are greater than at program level. In order to improve software engineering education, there is a need to put more efforts at overall software engineering discipline.

VIII. FUTURE RESEARCH

In this research we have synthesized the studies related to software engineering curriculum. In order to proceed, at next stage, we will conduct systematic literature review and extract data by implementing systematic literature review protocol which we have prepared and we will analyze data in terms of tendencies. The resulted analyzed data will be presented to inform the state of the art in software engineering curriculum design/revision at program level, evolution of software engineering curriculum of program level and industry/academia relationship and gaps.

Systematic mapping study output will provide ground for systematic literature review. Systematic literature review also known as systematic review stated in [1] and it is a type of secondary research. It is a deeper view of specific research area in mode of depth to identify the evidence available related to research question.

This work will be enlightening for software engineering researchers, software engineering educators to instruct software engineering students and train software engineering professionals by designing efficient and effective software engineering curriculum with focusing market trends.

REFERENCES

[1] Kitchenham, B.A. and Charters, S. Guidelines for performing Systematic Literature Reviews in Software Engineering Technical Report EBSE-2007-01, 2007.

[2]Elisa Yumi Nakagawa, Daniel Feitosa, Katia Romero Felizardo, Using Systematic Mapping to Explore Software Architecture Knowledge. *SHARK'10* May 2, 2010, Cape Town, South Africa, pages 29-36, 2010.

[3] R. S. M. Petersen, K.; Feldt and M. Mattsson. Systematic mapping studies in software engineering. In *EASE'08*, pages 1–10, Bari, Italy, June 2008.

[4] Duanyang Zhao; Qingxiang Xu; Wuheng Zuo; , "Research and Practice in the Curriculum System for Majors in Software Engineering," *Education Technology and Computer Science (ETCS), 2010 Second International Workshop on* , vol.1, no., pp.417-420, 6-7 March 2010 doi: 10.1109/ETCS.2010.203.

[5] Condori-Fernandez, N.; Daneva, M.; Sikkel, K.; Wieringa, R.; Dieste, O.; Pastor, O. (2009). "A systematic mapping study on empirical evaluation of software requirements specifications techniques", ESEM 3rd International Symposium, 502-505.

[6] Brichau, J.; Chitchyan, R.; Garcia, A.; Rashid, A.; Clarke, S.; D'Hondt, E.; Haupt, M.; Joosen, W.; Katz, S.; Noye, J.; Sudholt, M.; , "A Model Curriculum for Aspect-Oriented Software Development," *Software, IEEE* , vol.23, no.6, pp.53-61, Nov.-Dec. 2006 doi: 10.1109/MS.2006.146.