Online Repository for Software Engineering Education

Emanuel S. Grant, *Member, IAENG*

***Abstract*— The profession of software engineering is one of the faster growing in the world; in the United States, it is projected that there will be over two million unfilled computer technology positions by the year 2020. Educational institutions are working to address this increasing demand for graduates to fill the personnel vacuum. An approach being pursued by a group of researchers is to develop an open-source Cloud-based repository of teaching artifacts. The repository will be a relational database of searchable case-based material that are tied to related program/course outcomes/topics and structurally attached to assessment material. Other projects have offered similar products and services, but this project focuses on sustainability by designing a dynamic content repository with broad participation.**

***Index Terms*— Curriculum, education, repository, software engineering**

I. Introduction

THE The pervasiveness of technology in society demands a workforce that is capable of addressing the increasing demand for products and services that are worldwide applicable. The Internet World Statistics website (http://www.internetworldstats.com/stats.htm) shows there has been a 832.5% increase in worldwide Internet usage between 2000 and 2015, with a corresponding increase of 1,319% increase in Asia, over the same period. A description for some Asia countries is presented in Table 1.

TABLE I: Asian countries internet usage increase

Country	Population	Users 2000	Users 2015	% Change
China	1,361,512,535	22,500,000	674,000,000	2,895.55
Hong Kong	7,141,106	2,283,000	5,751,370	151.92
India	1,251,695,584	5,000,000	375,000,000	7,400.00
Japan	126,919,659	47,080,000	114,963,827	144.19
South Korea	49,115,196	19,040,000	45,314,248	138.00
Malaysia	30,513,848	3,700,000	20,596,847	456.67
Philippines	109,615,913	2,000,000	47,134,843	2,256.74
Singapore	5,674,472	1,200,000	4,653,067	287.76
Taiwan	23,415,126	6,260,000	19,666,364	214.16
Thailand	67,,976,405	2,300,000	38,000,000	1,552.17

Table 1 illustrates the growing population of Internet usage, which is an indication of the growing application of computing technology. This implies a growing need for educated software developers to design, implement, and maintain these systems. The Swiss nonprofit foundation, World Economic Forum (http://www.weforum.org/) listed 14 technology predictions for the year 2020 and six of them were directly computer technology related (job learning, Internet of everywhere, data-driven healthcare, Internet of things, tele-communication, humanized Internet). A Computerworld article by Patrick Thibodeau [1] presents a breakdown of jobs figure in the computer technology industry for the year 2020 that is derived from official United States government figures. These figures predict an average growth of 22% in the computer technology fields, with the highest percentage growth of 32 in the software developers/systems software area.

A report from July 2013 by Allie Bidwell in the online US New, projects that by 2020 there will be approximately 170 million jobs in the USA, with 65% of those jobs requiring some post-secondary education or training. These figures project a 5 million shortfall in the number of needed graduates to fill these positions. While these figures are from the USA Government Department of Labor, it can be assumed that similar figures are represented in other developed countries.

These statistics and their implications are justification for the work presented in this paper. The work seeks to address the need for educated personnel in the information technology, specifically in the sub-discipline of software engineering (SE). The research effort seeks to develop a repository of teaching artifacts for SE education. The repository will implement an ontological relationship across a heterogeneous collection of teaching material that facilitates searching and selection of a subset of the repository content. This selection will be driven on a set of specified program/course outcomes for a four-year undergraduate-level of education in SE. The key artifact of the repository will be a set of case examples for SE that is ontologically related to the desired outcomes, topics, lectures, and assessment instruments.

The next section of this paper is the background and related works of the research project, followed by the methodology of the research. A section on current state of the research precedes the conclusion and future work.

II. Background

A. SWEET Research Group

A team of SE educators, led by the author, is trying to advance the teaching of SE has started a project to develop and support the creation and adaptation of a core set of learning materials and teaching strategies for undergraduate education. The outcome of this project is a framework that is a repository of essential SE teaching and assessment artifacts, course projects, and assignments that will be available to all SE educators. This will advance the learning

Manuscript received January 7, 2017; revised January 15, 2017.
E. S. Grant is an Associate Professor with the Department of Computer Science, University of North Dakota, Grand Forks, ND 58202-9015 USA Work: 701-777-4133; fax: 701-777-3330; e-mail: grante@aero.und.edu.

goals, and promote the adoption of teaching strategies that are in the best interest of the students'.

The goal of the research is the identification of commonalities in teaching SE across multiple territories and institutions and the identification of a fundamental set of topics, and course material for teaching SE at the undergraduate level. The research will result in an integrated environment that leverages a set of best practices, with respect to SE teaching techniques, processes, and material - towards the definition of an international collaborative teaching paradigm. The group created a project name, SoftWare Engineering Education and Training (SWEET).

B. Sweet Workshops

SWEET has hosted a series of workshop that were aimed at capturing input data for the research project. The first seminar was hosted at the Holy Angel University, Philippines July 2010 [2]. This seminar had in attendance ninety-five students and faculty members from twenty educational institutions in the central Philippines region, along with representatives from government agencies and business professionals. A second workshop that was held at the 27th IEEE Conference on Software Engineering Education and Training (CSEE&T) in Alpen-Adria Universität Klagenfurt, Austria in April 2014 [3] and a third at the 1st International Workshop on Case Method for Computing Education, co-located with APSEC 2015 [4]. This workshop focused on the presentations of case studies, as used in SE courses. Four presentations were of the educators' experience of applying case-based pedagogy in their respective SE courses, with the fifth presentation focusing on the requirements and features of case-based pedagogy in SE education. The workshop proceeding and agenda are attached as appendices.

C. Related Works

The Software Engineering Disciplinary Commons (SEDC) (http://sec.cs.siue.edu/) is a project that is funded by the National Science Foundation (NSF) (http://www.nsf.gov) of the United States, and the program has two objectives. The first is the documentation and sharing of knowledge about student learning on courses in SE, in four-year degree-granting institutions within a single geographic region. The second objective is to improve the quality of teaching in SE (and computer science in general) by establishing practices for the scholarship of teaching by making it public, peer-reviewed, and amenable for future use and development by other SE educators.

This work encompasses that of the SEDC, and adds three layers; (1) international participation, (2) collaborative distributed teaching, and (3) retrieval and update of teaching material. Two of the researchers involved in this project are participants in the SEDC project, and their experience from the SEDC project will be incorporated into this project.

The IEEE Computer Society hosted a series of workshops in the early 1980s that initiated work on establishing standards for SE products and processes. That effort motivated the establishment of a Software Engineering Body of Knowledge (SWEBOK) [5], which came into focus ten years later. The mission statement of SWEBOK is "To establish the appropriate sets(s) of criteria and norms for professional practice of SE upon which industrial decisions, professional certification, and educational curricula can be based." This work is currently embodied in the 2004 SWEBOK Guide that is comprised of twelve chapters and four appendices. The guide sets out a baseline for body of knowledge of SE. The IEEE Computer Society seeks to keep the Guide relevant by updating it based on comments it receives from educators and practitioners in SE.

In addition to SWEBOK the IEEE Computer Society, in conjunction with the Association of Computing Machinery (ACM) have compiled the Computer Science Curricula 2013, Curriculum Guidelines for Undergraduate Degree Programs in Computer Science [6]. The document serves as a guideline for academic and accreditation institutions on the content of undergraduate degree program in SE. Similar to the SWEBOK guide [3] is updated by a management committee with input from educators and industry practitioners.

D. Case-based Pedagogy

The case method uses two elements namely the case, and a set of activities related to that case [7]. The case is a rich narrative that provides detailed information about a situation in which an individual or group must make a decision or solve a problem. Cases take many forms, and there are varieties of ways to write them. Usually it comprises the following:

- A detailed description of the problem's context, which at the least include the current situation and background information,
- A character that plays the central role and that character has to be solve the problem,
- Supporting data, this can include a range; data tables, quoted statements from the various actors in the case, supporting documents, images, video, or audio
- The case is not required to provide any analysis or conclusions.

Students working on the case during classroom activities focus on analyzing the case to understand and explain the events, evaluate and propose solution options to solve the problem, predict the effects of taking actions, etc. The activities can be classified in the following four steps [5]:

1. Understanding the case-identifying the important facts of the case,
2. Analyzing the case by understanding the issues and challenges from multiple perspectives, evaluating solutions proposed in the case,
3. Taking action by proposing alternate solutions, and evaluating the pros and cons of the solutions and their short- and long-term impact, and
4. Finally, being able to "take away" the generalizable concepts and principles from the case.

The activities can differ depending on the course that is being taught. For example, in a management course, an activity can be answering discussion questions of the the case, "Should the marketing manager launch the product?". In a technology oriented course, an activity can include the students having to design and evaluate solutions, "Propose

ISBN: 978-988-14047-3-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

and evaluate two alternative IT solution architectures for implementing the automated fulfilment process".

E. The Rationale for Case-based Teaching

Case teaching method provides a number of benefits to enhance student learning through "interactive pedagogy" by stimulating critical thinking and problem solving skills and creating replicas of actual situations--which include incomplete information, time constraints, and conflicting goals. This leads to enhances student motivation and well-aligned learning environment, where practice and theory come together. This is especially true when teaching technical topics through traditional lecture that is often very dry and boring. Following are some benefits of using cases in computing education:

- Introduce real world scenarios into the classroom,
- Convey knowledge of what computing professionals do and how they work,
- Develop effective problem-solving skills, which are situated in a real world context,
- Helps students to better connect theory and practice,
- Enhance cooperative learning skills in the class.

F. The Way Forward

In order to drive the case method in computing education, as a community involved in teaching computing courses, there is the need to develop a set of sample cases that can be used in teaching specific computing courses, for example, object oriented design, enterprise integration, software testing, etc., and share them with other faculty. These cases should be associated with desired program/course outcomes and the linked to a set of necessary teaching and assessment artifacts. In order to achieve these goals, there need to be an online repository where faculty can share the case.

III. THE RESEARCH PROJECT

This section, the infrastructure for an online open-source repository of ontologically related heterogeneous teaching components is presented. The repository will be implemented on a Cloud platform that is accessible over the Internet. This will require the use of established middleware and its associated security components. This ensures the integrity of the repository content to its worldwide users.

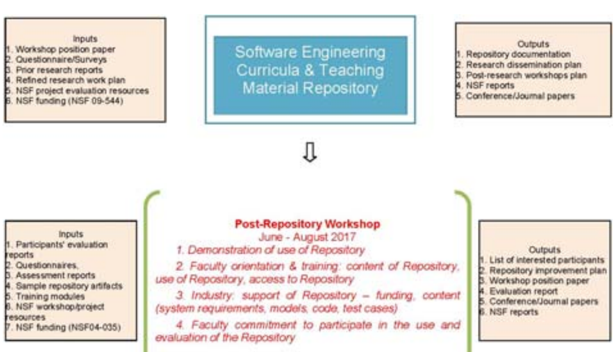

Figure 1. Research Project Development Plan

Figure 1 illustrates the Repository development phase of the research. The Repository is intended to advance the teaching and learning of SE topics at the undergraduate level, by providing a collection of artifacts that captures the best practices in the field. SE educators will be able to access and use the Repository to develop course curriculum and syllabus, or enhance existing courses with high quality teaching and learning components. Educators will be able to select items for teaching and learning across the years of an undergraduate program. The Repository will store items for curricula development, syllabi development, and course assessment material.

Access to the Repository will be over the Internet, which will also provide the gateway for updates to the Repository (addition/deletion of items, and modification of existing items). The architectural platform for the Repository will be Cloud, and associated computing technologies [9]. Cloud Computing, software as a service (SaaS) [9], and service-oriented architecture (SOA) [9] are growing phenomena in the business world. In contrast to traditional software systems, where organizations maintain technical staff and enough computer hardware to run their business applications effectively, SaaS and Cloud Computing are beginning to change this traditional way of managing information technology (IT). SaaS has an incentive to release new features as soon as they are completed to the users.

Service-Oriented Architecture (SOA) [10] evolved in stages over the last few decades, since industrial automation increased. The services in use today, process requests as input and produce output for customers of other systems or services. These systems or services orchestrate the data when generating messages among the users. The operational users of these services can monitor or manage many requests simultaneously. These operations can also be performed by a mediator service designed to follow the agreed policies and procedures among each of these services. Each service is owned and governed by a business entity and works within a certain body of rules defined by the policymakers.

The Cloud [11] is a union of computers, over the world, that provide data and software as a service (SaaS) [9]. The goal is to process, display, and communicate data with each other without ownership, management, nor operation of the intervening hardware and software. The Cloud offers millions of users a way to find and share their relative information, and conduct research, and accelerated learning.

Figure 2: A Cloud with several Data Centers

Devices that compose the Cloud and provide service to

ISBN: 978-988-14047-3-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

users are servers situated across the globe as depicted in Figure 2. A user's need to process data over the Cloud is served by any or many of these servers using some specific pieces of software applications designed and developed for Cloud services. Several of these services are freely available to use, such as Google™ search facility or Amazon's™ book search. The servers providing such and many more services are stationed at data centers. It is a fact that the cloud is a combination of all of the data centers.

The following section will set out the methodology behind the research, and provide a description of the conceptual framework of the repository.

IV. METHODOLOGY

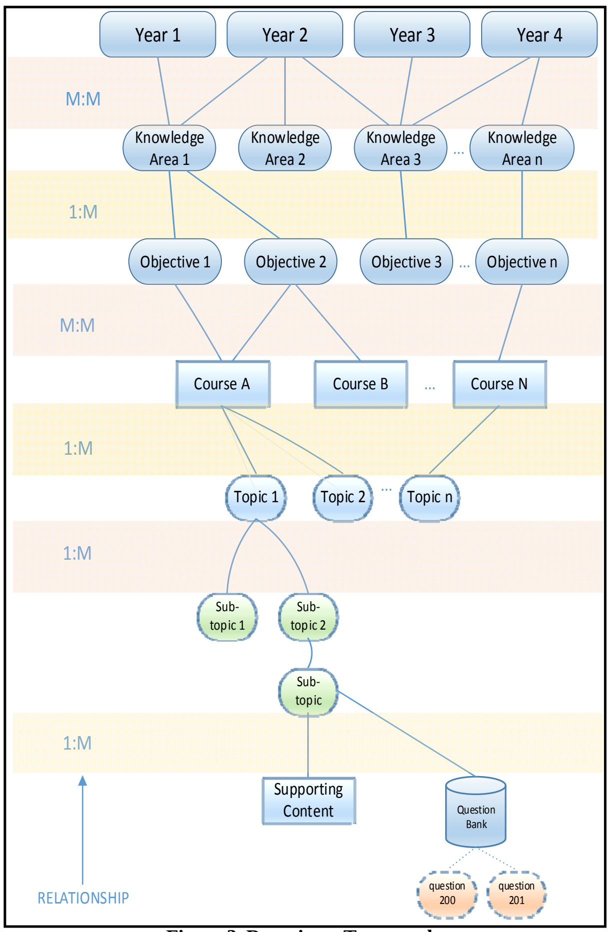

Figure 3. Repository Topography

The international collaborative teaching paradigm consist of a repository of teaching material that represents the commonalities and variabilities that are essential for the teaching of software engineering. The topography of this repository will be an ontologically structured set of teaching and assessment artifacts, based on a set of program/course outcomes. A key element of the ontologically structured repository of artifacts will be the use of cases around which the outcomes will be realized. Figure 3 illustrates the ontological relationship that will be implemented in the repository between the teaching and assessment artifacts. The supporting content at the bottom of the ontology map includes the case examples that are now identified as a fundamental artifact of the repository.

A goal of the research work is providing users with the facility to develop program/course curriculum within a technology framework. Figure 4 illustrates the hierarchical organization of the components of a curriculum. The body of knowledge that is deemed as appropriate for an undergraduate program in software engineering is designated in SEEK[6]. SEEK is organized into three levels. The top level is the Knowledge Area, which represents a particular sub-discipline of software engineering that is a significant part of SE knowledge that an undergraduate should know. The second level is the units where each knowledge area is broken down into smaller divisions or modules. Each unit is then subdivided to form the lowest level, which is a set of topics. The research introduces case examples that are linked to the topics; the case examples provide the connection between theory and practice.

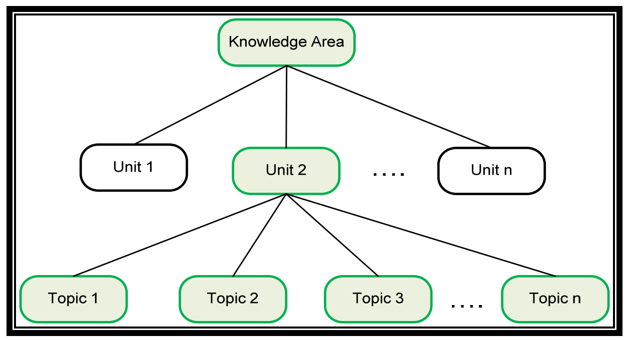

Figure 4. SEEK hierarchical organization of KAs, Units and Topics

Figure 5 provides an overview of the methodology. Phase 1 details the data capture for the repository. This builds on the work of Grant, et. al. in [12 13] in order to outline a path to the realization of the repository. Phase 2 implements the repository development and population, given the input from Phase 1. Finally, Phase 3 illustrates utilization of the repository for the delivery of the content and student assessments as well as automated feedback to the repository.

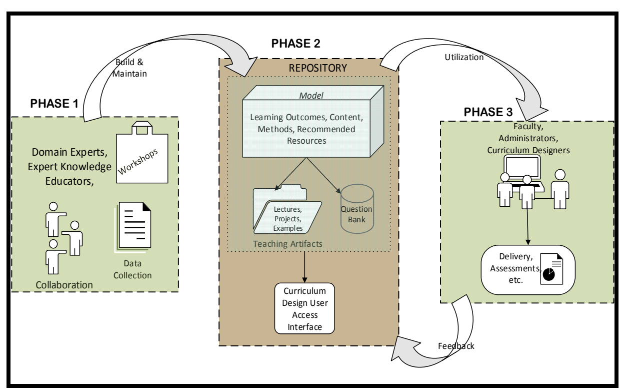

Figure 5. Overview of Three Phase Research Project

A. *Cloud Services*

The work outlined in this subsection is based on research conducted by Atif Mohammad towards his Ph.D. dissertation [9]. The service providers using Cloud can develop a large body of web, database, network, and PI services that may be distributed among services users with acceptable service level agreements (SLAs). The utilization

ISBN: 978-988-14047-3-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

of Cloud computing will be increasing with the passage of time. Cloud computing is a new discipline, introducing new fundamental concepts, where the needs of service designers and providers are on the increase.

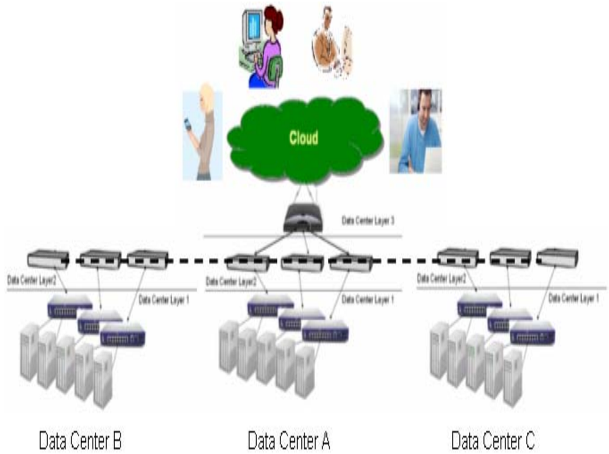

Figure 6. A combination of services for consumers

Figure 6 depicts the aforementioned as a set sequence of three stages:

1. Data center Layer 1 contains the data processing and storage servers, connected with each other for efficient and fast data transmission.
2. Layer 2 contains smart routers to balance the incoming request load [14], for layer 1 processing with firewalls to provide communication security.
3. Layer 3 contains request receivers and response broadcasters.

The use of Cloud computing provides a user (an organization) computing resources, which are utilized on-demand. With the increase or decrease of computing servers, network bandwidth and storage, these utilizations are subject to an SLA or service level agreement among user and the Cloud computing services provider. Cost varies with utilization. Figures 7 and 8 illustrate a Cloud internal service communication among several services.

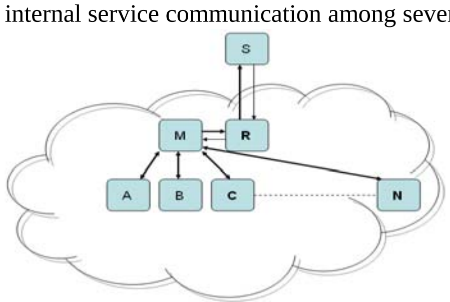

Figure 7. Services available in the Cloud

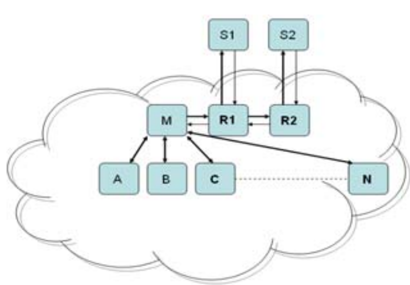

Figure 8. Utilization of two services S1 and S2

The service **S** executes a search request for a property for sale on a repository **R**. If the requested data is not available service, **R** generates a request as shown path 5 in Figure 9 to the Master search service **M**, which generates a query from registered service with it as data providing services. Data repository service **R** delivers the data with using return path r to property search service **S** the required data, if it is available from previous searched data. In other case, the Master Service **M** will find data and send it to Data repository service **R**.

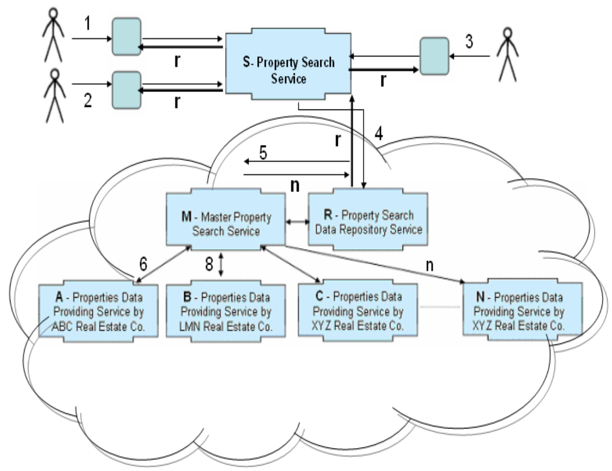

Figure 9. Requestors Servicing Output

Let us denote the diagrammatic notations as given in Figure 10.

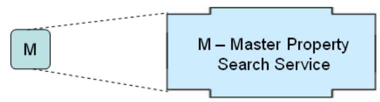

Figure 10. Denoting Service M as a start/end service Marker

In case, of a second user wanting to obtain information on properties available for Short Sale uses, Service **S2**, will find the data available by the Short Sale Data Repository of **R2**. The search transaction is able to query repository **R1** that might contain some properties. This might be on short sale, and searched by user **1** in combination of properties on sale by several property selling companies. In case **R1** does not have data, the request moves to service **M** for further search. This example illustrates that a Service can be a self-describing black-box function, which may be called by another service. This will provide resultant dataset in response to request by a service user or by another service.

V. RESULTS

The SWEET group has conducted workshops, as, as described in Phase I of Figure 5. These workshops produced vital information that will be used as input to Phase II activities, of repository developing. The earlier version of the ACM/IEEE-CS document [3] was used as a seed document to the activities of determining commonalities and variabilities of SE education.

The single input used for the breakout panel sessions was sections from the earlier version of [6]. The section used comprised introductory information on the IEEE-CS/ACM description of SEEK, along with tables of the SEEK areas, with the supplementary information. Samples of the SEEK

ISBN: 978-988-14047-3-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

areas used in the workshop, are presented in tables 2 and 3.

The first column of Table 2 lists the Knowledge Area/Knowledge Unit (KA/KU), coded for reference. The Title column lists the topic area, and column three (hrs) the recommended number of lecture hours for the topic.

TABLE 2: SAMPLE SEEK KNOWLEDGE AREA

KA/KU	Title	hrs
CMP	**Computing Essentials**	**172**
CMP.cf	Computer Science foundations	140
CMP.ct	Construction technologies	20
CMP.tl	Construction tools	4
CMP.fm	Formal construction methods	8

Table 3 presents the detail topics of the Computer Science Foundation section of the SEEK Computing Essentials area. Column 1 is coded identifier of the topic. Column 2 is the topic title, column 3 the Bloom's classification, column 4 the topic's relevance, column 5 the lecture hours, and column 6 notes the related topics.

TABLE 3:SAMPLE SEEK KNOWLEDGE AREA: DETAIL

Reference		kc,a	E,D,O	hrs	Related
CMP	**Computing Essentials**			**172**	
CMP.cf	*Computer Science foundations*			140	
CMP.cf.1	Programming Fundamentals (control & data, typing, recursion)	a	E		
CMP.cf.2	Algorithms, Data Structures/Representation (static & dynamic)	a	E		CMP.ct.1, CMPfm.5, MAA.cc.1
CMP.cf.3	Problem solving techniques	a	E		CMP.cf.1
…	…	…	…		…
CMP.cf.13	Semantics of programming languages		D		

TABLE 4:SAMPLE WORKSHOP OUTPUT TABLE

Professional Practice			
TOPIC	**YEARS**	**DEPTH**	**RATIONALE**
PRF.psy	1 to 4	15%	# 5 & #6 are change from K to C->"DEALING" is not merely KNOWING. Dealing needs to provide course of actions to certain issues e.g. forecasting, predicting, identifying, presenting alternative solutions. All items are ESSENTIAL
PRF.com	1 to 4	40%	All items are ESSENTIAL
PRF.pr	1 & 4	45%	All items are ESSENTIAL
Software Management			
TOPIC	**YEARS**	**DEPTH**	**RATIONALE**
MGT.con	2 to 4	10%	All items are ESSENTIAL
MGT.pp	2 to 4	31%	All items are ESSENTIAL
MGT.per	2 to 4	11%	All items are ESSENTIAL
MGT.ctl	2 to 4	21%	All items are ESSENTIAL #5 & #6 be change from O to E and classified as K->to achieve quality performance/output would require effective supervision/management
MGT.cm	2 to 4	27%	All items are ESSENTIAL #7 is change from D to E and classified as A->Security is an essential part of software configuration management. Topic #7 must be renamed as Distribution, Back-up, and Security. #4 be change from C to K->BUILDS deals more on conceptual/theoretical rather than actual application in the classroom set-up

Table 4 resulted from the deliberations on **Professional Practice** and **Software Management**. It is to be noted that the column **hrs** was changed to **Depth**, and assigned percentage of teaching time, instead of hours. Overall, the workshop groups were at a high percentage of agreement with the original SEEK topics. The areas of greatest divergence were with respect to the Bloom's taxonomy classification and percentage of teaching time for each topic.

VI. CONCLUSION

The work of this paper is an amalgamation of multiple research efforts that are geared toward the development of a repository of software engineering teaching and assessment artifacts. This report draws on research effort from [2, 3, 4, 6, 9, 12, 13]. The project has collect data from a series of workshop that will be the input to Phase 2 of the project. This phase will result in the design, development, and population of the repository. A series of workshops will follow for the purpose of training and use of the repository. The ultimate goal is for sustainability through evolution of the repository.

REFERENCES

[1] P. Thibodeau, "IT jobs will grow 22% through 2020, says U.S.," *Computerworld*, [Online].

[2] E. S. Grant, T. Stokke, H. Johnson, E. Song, V. Y. Sien, Y.W. Choong, T. Salta, J. Redoban, "International Seminar on Software Engineering and its Application in Education", GSTF Annual International Conference on Computer Science Education: Innovation and Technology (CSEIT 2010).

[3] E. S. Grant, "Technology-Driven Software Engineering Curriculum Development", Technology-Driven Software Engineering Curriculum Development, at the 27th Conference on Software Engineering Education and Training (CSEE&T 2013)

[4] E. S. Grant, "Case Method for Computing Education (CMCE) A Strategy for Teaching Software Engineering", First International Workshop on Case Method for Computing Education (CMCE) in conjunction with the 22nd Asia-Pacific Software Engineering Conference (APSEC 2015).

[5] P. Bourque and R. E. Fairly, *Guide to the Software Engineering Body of Knowledge (SWEBOK®)*: Version 3.0. Ed. IEEE Computer Society, New Jersey, USA, Jan. 2014.

[6] *Computer Science Curricula 2013, Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*, the Joint Task Force on Computing Curricula, ACM and IEEE-CS, New Jersey, Dec. 2013.

[7] S. M Williams, "Putting Case-based Instructions into Context: Examples from Legal and Medical Education," *Journal of Learning Sciences*, vol2, 4, 1992, pp.367–427.

[8] J. H. [2] Shulman, Happy Accidents: Cases as Opportunities for Teacher Learning. American Educational Research Association April 2002, New Orleans, USA.

[9] A. F. Mohammad and E. S. Grant "Cloud Computing, SaaS, and SOA 3.0: A New Frontier", Proceedings of the International Conference on Cloud Computing and Virtualization 2010, Singapore.

[10] T. Erl, *Service-Oriented Architecture (SOA): Concepts, Technology, and Design*, Prentice Hall, New Jersey USA, 2009.

[11] T. Erl, R. Puttini, and Z. Mahmood *Cloud Computing: Concepts, Technology & Architecture*, Prentice Hall, New Jersey USA, 2014.

[12] Grant, E. Online Collaborative Teaching of Software Engineering in an International Multi-Campus Environment. First HAU International Workshop on Enhancing Teaching & Learning of Software Engineering, Holy Angel University Angeles City, Philippines August 12, 2011.

[13] Grant, E., Stokke, T., Helps, W., Esquivel, J., Concepcion, W. Enhancing Teaching & Learning of Software Engineering in an International Environment. In Proceedings of the 2nd Annual International Conference on Computer Science Education: Innovation and Technology (CSEIT 2011), pages 36-41, 2011.

[14] C. Kopparapu. Load Balancing Servers, Firewalls, and Caches. John Wisely & Sons Inc., 2002.

ISBN: 978-988-14047-3-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)