Using GSwE2009 in the Creation and Modification of Graduate Software Engineering Programs and Related Curricula

Mark Ardis
Stevens Institute of Technology
Hoboken, NJ, USA
mark.ardis@stevens.edu

Shawn Bohner
Rose-Hulman Institute of Technology
Terre Haute, IN, USA
bohner@rose-hulman.edu

Lucia Camilloni
Diego Vallespir
Universidad de la República
Montevideo, Uruguay
lcamilloni@fing.edu.uy,
dvallesp@fing.edu.uy

Sylvia Ilieva
University of Sofia
Sofia, Bulgaria
sylvia@fmi.uni-sofia.bg

Abstract

The current ACM and IEEE Computer Society recommended guidelines for graduate software engineering programs (GSwE2009) were created in 2009 by an international group of experts from industry, government and academia. We report here on some of the early experiences using GSwE2009 to create a new curriculum in software assurance and to create and/or improve 4 different academic programs in 3 countries. All of these experiences have been positive, confirming the validity and usefulness of GSwE2009.

1. Introduction

In 2009 the Institute of Electrical and Electronics Engineers Computer Society (IEEE-CS) and the Association for Computing Machinery (ACM) accepted ownership of a new set of curriculum guidelines for graduate software engineering: the Curriculum Guidelines for Graduate Degree Programs in Software Engineering (GSwE2009) [1]. Since then several schools have used GSwE2009 to create or improve their programs. At least one other curriculum project also used GSwE2009 as a model and primary source. In this paper we describe some of those experiences.

Section 2 presents a brief overview of GSwE2009 and its relationship to earlier work in this area. Section 3 illustrates how GSwE2009 influenced the development of a new set of curriculum guidelines for software assurance programs. Sections 4-7 describe how GSwE2009 was used in the creation or modification of 4 different software engineering programs in 3 countries. Finally, section 8 summarizes these experiences and draws some conclusions.

2. Overview of GSwE2009

The first efforts to standardize software engineering curricula were undertaken in the late 1970s by the Subcommittee on Model Curricula in Software Engineering of the IEEE Computer Society [2]. Unfortunately, this work was never officially endorsed by the Computer Society. However, committee members helped start the first Master of Software

Engineering (MSwE) programs in the United States, and other pioneering schools also based their programs on the work of that committee.

In the winter of 1988 the Education Program at the Software Engineering Institute (SEI) at Carnegie Mellon University held a workshop of leading software engineering educators to design a recommended curriculum for graduate study. The results were first presented at the 1988 Conference on Software Engineering Education [3], and then later published by the SEI as a separate report [4]. Those recommendations were used by several software engineering programs over the years. Since then the nature and role of software has evolved considerably.

In 2007 the Integrated Software and Systems Engineering Curriculum (ISSEC) project, funded by the U.S. Department of Defense's Office of the Secretary of Defense, began the development of a new set of guidelines. Over two years more than 40 authors from industry, academia and government participated in the creation of GSwE2009.

GSwE2009 includes:

- Expected outcomes when a student graduates from a master's program
- Expected student background when entering the master's program
- Curriculum architecture
- Core body of knowledge (CBOK)

In addition, guidance for construction, maintenance and evolution of the recommendations are provided. The guidelines are targeted toward students pursuing a professional master's degree. That is, typical students are not interested in pursuing a research career, rather they are practicing professionals seeking to improve their knowledge and skills.

The core body of knowledge (CBOK) of GSwE2009 was influenced by several existing standards, especially the Software Engineering Body of Knowledge (SWEBOK) [5]. Indeed, most of the knowledge units in CBOK were derived from the knowledge areas of SWEBOK. Additional knowledge units were added where necessary, but future updates of SWEBOK are likely to include those same topics. For each knowledge area GSwE2009 recommends an appropriate level of achievement based on Bloom's taxonomy [6], a popular learning scale used in these types of curriculum recommendations.

The Institute of Electrical and Electronics Engineers Computer Society (IEEE-CS) and the Association for Computing Machinery (ACM) have both adopted GSwE2009 as part of their shared curriculum recommendations in computing. Future versions of GSwE2009 will be created and maintained by those two professional societies.

3. Using GSwE2009 on MSwA project

In 2009 the Software Assurance Curriculum (SwAC) project was established to "develop and present a core body of knowledge (BoK) from which to create a master's level degree program in software assurance, as a stand-alone offering and as a track within existing software engineering and computer science master's degree programs" [7]. The project was funded by the US Department of Homeland Security and led by Nancy Mead at the SEI. Three of the members of the SwAC team were also authors of GSwE2009.

A Master of Software Assurance (MSwA) program has many similarities to a Master of Software Engineering (MSwE) program:

- both programs assume that entering students have a bachelor's degree in computing or a related field
- both programs are designed for professionals working in the software industry
- both programs cover a broad range of topics across the entire software development lifecycle

Given these similarities the SwAC project decided to include a track on software assurance within a MSwE program as one of the possible implementation strategies.

The SwAC project adapted some of the same basic principles of curriculum development as had been used on previous curriculum efforts, including the GSwE2009 project. For example, the curriculum architecture for MSwA shares some of the same structure as that of GSwE2009, including the notions of a core set of courses and a recommended capstone experience. The expected prerequisite background of students entering a MSwA program are similar to those for a MSwE program.

4. Using GSwE2009 at Stevens Institute of Technology

In response to the publication of GSwE2009 Stevens Institute of Technology modified its MSwE degree program to better cover the recommended core body of knowledge, including more systems engineering material. We also investigated adding more application domain tracks. We were aided in this process by an effort to compare our curriculum to the GSwE2009 guidelines and to other programs.

Stevens participated in a comparison study of 12 graduate software engineering programs [8] sponsored by the ISSEC project that created GSwE2009. The study included data about:

- demographics of participating institutions
- entrance requirements
- required courses
- profiles for hypothetical "typical" student types
- comparisons of the "typical" students to the GSwE2009 outcomes at different points throughout the program

Through participation in the study we identified some areas of our program that we wanted to strengthen. We also confirmed that other parts of our program were healthy and needed no modification.

In 2010 we created two new tracks on software assurance: one for software developers who need to construct dependable software systems, and one for managers who need to acquire and manage trustworthy software systems [9]. Each track consists of four elective courses to be taken in addition to our six required core courses in the MSwE curriculum. We also updated our coverage of software assurance topics in our core courses to make sure that we included all the recommended material from the MSwA curriculum.

In addition to the two new tracks within the MSwE program we created two certificate programs in software assurance for students who already have graduate degrees, or who are not ready to commit to an entire graduate program. One certificate is designed for software developers and the other is for managers. Students participating in certificate programs can also apply all courses toward a graduate degree.

We found that the GSwE2009 recommendations were very helpful in our curriculum changes. First, they helped ensure that courses continued to be up to date with modern software engineering practices. Second, they provided enough flexibility to tailor our program to our particular strengths. The core material is covered in half of the coursework, allowing students to select specialty tracks that best suit their career plans. But even within the core courses we were able to emphasize dependability concepts that best suited our students.

5. Using GSwE2009 at Rose-Hulman Institute of Technology

Rose-Hulman Institute of Technology is a small (~2000 students), highly-ranked Engineering, Science, and Mathematics college with an undergraduate program in software engineering based largely on the Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering [10]. In 2009 the Rose-Hulman Computer Science and Software Engineering (CSSE) department created a new Master of Science in Software Engineering (MSSE) degree program based largely on GSwE2009. The primary motivations for the program are: 1) to respond to the growing demand for software engineering professionals in the metropolitan Indianapolis area, and 2) for Rose-Hulman to better serve the engineering and software communities at large. The MSSE degree complements the successful undergraduate software engineering program, the only ABET accredited program of its type in Indiana.

The design of the MSSE program was shaped largely by GSwE2009. The MSSE is a terminal master's degree program designed primarily for software professionals that have at least one year of experience working in public or private sector software organizations. The program meets the demands and constraints of both working professionals looking for part-time study and those who can devote one year for full-time study. We teach on a quarter system, a term per season (i.e., Fall, Winter, Spring, and Summer), and most students take one or two courses per term as part-time students.

The MSSE program offers a classroom-based approach that encourages networking with other technical professionals pursuing the degree. Class sizes are small and the style of teaching is largely very interactive – driving discussions to tease out critical analyses and ideas to support a project or paper each term. Interaction in the classroom is key, as the instructor orchestrates the topics and background knowledge while students offer up experiences and posit positions to defend in the discussions. This “learn by interaction and doing” is favored by our students over traditional methods of instruction due to the practical nature of the software engineering field.

The ideal MSSE candidate has an undergraduate degree in a computing related field (e.g., software engineering, computer science, computer engineering), but students from other fields may be provisionally accepted and take bridge courses to complete the program entrance requirements. Prerequisites to the program are: 1) one or more years of work experience in software development or maintenance; 2) evidence of technical communication skills, both oral and written; and 3) academic background in object-oriented concepts, data structures, analysis of algorithms, and discrete mathematics.

The fifty-one credit hour MSSE degree requirements include forty credit hours of software engineering technical and management course work, eight credit hours of capstone project work, and three credit hours of software engineering or engineering management seminars. As part of the forty credit hours, students must take six core software engineering courses, two software engineering electives and two specialization electives.

The program started in the fall of 2010 and is on track to graduate its first MSSE student(s) in the spring of 2013. Courses are taught off-campus in the evening largely at corporate locations where the Institute has strong ties and the concentration of software engineering firms is high. This has served as both a co-branding strategy and as a convenience for our students. The facilities have proven as good as what could be offered on campus.

The CSSE department previously tried to get a software engineering program approved in 2006, but was unsuccessful. GSwE2009 was particularly helpful in refining the original proposal for the MSSE program, especially its response to emerging needs of the software engineering community, and lending external credibility and confidence to get it approved.

The integration of systems engineering concepts into the curriculum has been seen as a significant strength by companies whose employees participate in the program. We have both a systems engineering course and modules in several other courses that serve to integrate key systems and software themes in the curriculum. In response to student requests software concepts and examples have become the center of both our systems engineering and project management courses.

GSwE2009 provided a strong foundation for the creation of our program educational objectives. The GSwE2009 expected outcomes for graduates served as the basis for the MSSE program educational objectives, which in turn, guided expected student outcomes and course learning objectives. In composing these objectives we were guided by the combined wisdom of the many authors and adopters of GSwE2009. Their advice was also helpful in some of the other challenges of starting a new program, such as marketing, acquiring needed resources and making effective use of adjunct faculty. Our experience at Rose-Hulman has been a good one in large part because of the GSwE2009 community.

6. Using GSwE2009 at University of Sofia

Sofia University (SU) is the largest and most prestigious research and teaching university in Bulgaria. Established in 1888, SU currently has around 22,000 students enrolled in 98 bachelor's (BSc) and over 200 master's (MSc) degree programs offered by 16 faculties. The Faculty of Mathematics and Informatics (FMI) of SU has developed into a leading institution in the field of Information and Communication Technologies (ICT) research and higher education with more than 150 researchers and more than 2500 students (BSc, MSc and PhD).

In 2005 an internal project of FMI was implemented in order to transform the informatics curricula to the current state of the art. Following the ACM/IEEE CC2005 series recommendations [11] FMI developed BSc programs in computer science, software engineering and information systems. The main aims were to introduce some well-established curriculum standards and new styles of teaching. The main challenge was to simultaneously satisfy recommendations of the ACM/IEEE Computing Curricula, the Bologna Declaration, and several European ICT curricula recommendations.

The Software Engineering Master program started in the 2004-2005 academic year as a result of a World Bank project "Market oriented master programs in Microelectronics, Software Engineering and Management." A number of strategic sources related to ICT education and particularly to software engineering were reviewed, including: the Bologna process [12] and the Lisbon strategy [13], Career Space Industry Consortium recommendations [14], ACM/IEEE recommendations [10][11][15][16], and advice from ICT companies (e.g., Intel, IBM, Microsoft, HP). Existing MSc programs in software engineering in Europe, the United States and elsewhere were examined and compared. The specific needs of the Bulgarian ICT industry were identified through a specially designed survey followed by a number of individual meetings and discussions. These efforts ultimately led to the establishment of a new Software Engineering department within FMI on March 1, 2007. Currently the department has 5 professors involved in master's program education.

FMI is using GSwE2009 to improve the curricula of its master's program in software engineering. Following the recommendations of GSwE2009 in 2011 we updated the core courses to better cover the 11 Knowledge Areas (KA) of the Core Body of Knowledge (CBOK). In addition we developed and offered a new track based on identified Bulgarian ICT industry needs and the research interests and competencies of software engineering department staff. The track covers technology entrepreneurship and innovation and consists of 4 courses: Technology Entrepreneurship and Innovation in IT, Organizational Behavior, Student Company, and an additional elective. We hope that this track will help the country to bridge the existing gap of ICT specialists and managers, and to stimulate new ideas and innovations.

In order to address the recommended outcomes of GSwE2009 we extended the topics related to professional ethics, legal aspects of software engineering, research methodology, organizational behavior, team spirit and responsibilities in agile methods. The software engineering master's program at FMI is based on teamwork on course projects, which are supposed to produce prototypes or useful tools expected to enrich the software engineering educational environment. The students are involved in development, maintenance and refinement of real software. This helps them to understand the software engineering process and technologies better, and it helps them adapt to a teamwork style. Some of the projects the students work on come from the research areas of the professors, which might eventually evolve into diploma theses or in rare cases even to a PhD thesis. Other projects come from current internal university needs.

All students in the FMI software engineering master's program complete both an internship and a master's thesis as capstone experiences. The internship is a form of practicum recommended by GSwE2009: a software development project done for a real external customer. Students apply for a position from a pool of internship openings in the Bulgarian software industry and are placed in positions appropriate to their skills and motivation. In some cases companies sponsor multiple teams working in parallel on the same task. Each student has two tutors – one from the University, and the other from the company. The internship ends with evaluation by the company tutor [17].

The master's thesis usually is individual work but in same cases could be teamwork with clear explanation of each team member's role and contribution. The thesis is based on a research project completed under the supervision of a faculty member, resulting in a technical report accepted by a three-member faculty committee. The report must be defended in an oral presentation.

The real evaluation of our efforts is shown in student enrolment figures over the last 5 years. A growing interest in software engineering education is observed in Bulgaria. For example, the number of applicants at the master's level has been increasing. Also, the highest scores of applicants on the entrance exam at FMI are for the software engineering specialty. For the last few academic years there were more then 80 applicants for the MSc degree program in software engineering, but only 25 could be accepted due to limited resources: teaching staff, lecture halls, computer labs, etc.

The GSwE2009 recommendations were really helpful for us in the process of improving the established master's program in software engineering. Currently we are working on course materials and pedagogical style in order to address the recommended Bloom Level competencies for software engineering suggested by GSwE2009.

7. Using GSwE2009 at UdelaR

The Universidad de la República (UdelaR) in Uruguay is the biggest university in the country and the only public one. The Informatics Undergraduate Degree in this University belongs to the School of Engineering. This degree has similarities with the curriculum guidelines for undergraduate degree programs in computer science [18] and software engineering [10] developed by the ACM and the IEEE Computer Society. About 550 students are enrolled in the program. Around 110 students get their degree annually, as it takes 5 years to complete the degree. Both the undergraduate degree and the master's degree at the School of Engineering are defined at two levels: (1) curriculum and (2) implementation of the curriculum. The curriculum is approved at the central level in the university whereas the implementation is approved at the School of Engineering without requiring the approval of the university.

The curriculum has as its main components the definition of the knowledge areas (called subjects) and the minimum credits needed in each subject in order to graduate. A credit is a measure of the effort an average student needs to make to pass a course, usually equal to 15 working hours. These hours are divided into: attendance in classes, individual study, laboratory work, etc. The curriculum also establishes the minimum number of credits a student needs to complete the degree. (The sum of the minimum number of credits per subject may be less than the total required).

The implementation of the curriculum is a set of restrictions on the curriculum. In a simple way the implementation can be understood as the definition of a group of courses the students must take in order to get the degree. There can be different implementations for the same curriculum. In that case each one is called a profile.

During 2011 we developed a curriculum for a master’s degree in software engineering to be taken at the School of Engineering. We focused on GSwE2009 to develop it. This master’s degree started in April 2012 with four small courses. With the completion of these four courses a student might obtain the Software Development Module Certificate of the SWEBOK Certificate Program [19]. At the moment, we do not have the complete implementation of the curriculum yet; that is to say, we have not yet defined all the compulsory courses.

We have defined the architecture of the curriculum in such a way that the subjects coincide with the 11 Knowledge Areas (KA) of the Core Body of knowledge (CBOK). GSwE2009 suggests covering all 11 KAs. Our curriculum does not require minimum credits in each subject, but students must get at least one credit in at least 6 subjects. The total number of credits necessary to get the degree is 70. As in GSwE2009, the intention of our master’s curriculum is that those who complete the degree can master all of the fundamental areas of software engineering. However, taking into account the number of professors of the Software Engineering Research Group (“Grupo de Ingeniería de Software”, referred to as "GrIS" below), only one credit is required in 6 subjects so as not to include restrictions in the curriculum that will be difficult to comply with when teaching the courses.

GSwE2009 specifies 10 expected outcomes for graduates of master's degree programs covering technical, ethical and learning aspects. Our curriculum specifies that students should comply with 9 out of these 10 outcomes. Mastering software engineering in one application domain (the Domain outcome in GSwE2009) is the outcome that is not included. The GrIS is not focused on the study of software engineering for a particular application domain. This implies that the courses to be taught in the master's degree in software engineering will be generic as regards to an application domain. Because of this, professionals will not be trained in a specific/particular application domain. Currently in Uruguay it is common for information technology professionals, especially those who are software engineers, to change jobs regularly. This means that these professionals must constantly learn new application domains. In this context it does not seem reasonable to complete a master's degree in which a particular application domain should be developed in depth, but just the opposite. It was decided, instead, to provide greater depth in some of the subjects of the curriculum.

GSwE2009 assumes that students entering a master's degree program have completed a computer science or an engineering undergraduate degree, have taken an introductory course in software engineering, and have at least 2 years of practical experience in some aspect of software engineering. The master's degree program of our School of Engineering has as its only entrance requirement to have an informatics undergraduate degree of at least 360 credits. Therefore, the entrance requirements are less than those expected in GSwE2009. However, though the entrance requirements are less, it is important to highlight that in Uruguay most students start working while they are completing their undergraduate degree. It is also expected that most students that start the master's degree will have completed their informatics undergraduate degree at our School of Engineering.

Our informatics undergraduate degree has 450 credits and two compulsory courses related to software engineering: Introduction to Software Engineering and Software Engineering Project. Introduction to Software Engineering has 10 credits. The aim of the course is to give the students an overview of the most relevant aspects of software engineering in order to facilitate their training in later courses. It has a strong theoretical component and practical work (on paper) to be done by the students in groups of about 8. In particular, practical work is done on: specification of requirements, specification of software architecture, and development of a testing plan. This course also focuses on introducing the students to the use of certain techniques and specific tools, so that the student can play different roles on a software engineering project.

The Software Engineering Project is a 15-credit course. Its aim is to reinforce and deepen knowledge of software engineering, contrast it with its practical application, and integrate it with aspects of other subjects. Students work on projects in groups of 10 to 15 for a real customer (local business companies). For this purpose, a process similar to the Rational Unified Process is followed, where each student plays a specific role. These two courses compensate for the entrance requirements suggested in GSwE2009. Given the entrance requirements of our curriculum, we cannot assure that all the students that enter took these courses. We assume, however, that the great majority will have completed both. All of the KAs proposed by GSwE2009 as preparation before starting a master's degree program are considered in the compulsory courses of our undergraduate degree. However, at present we cannot establish the Bloom level in each one as the GSwE2009 does.

GSwE2009 recommends a capstone experience that can be a thesis, a project or a practicum experience. It can be done either individually or in a team. Our university demands that all master’s degrees include a thesis done individually. Within this frame individual projects are admitted as long as the final product is a thesis. Through work on their thesis students are expected to deepen their knowledge in a specific area of knowledge; contributing strongly to complying with the Depth outcome set forth in GSwE2009.

GSwE2009 estimates that an average master's program consists of 33 to 36 American credit hours. One credit nominally equates to approximately 13 to 14 contact hours between faculty and student, plus homework (between 2 and 3 hours outside the class per week for each American credit hour). Thus the total number of hours required for the master's degree is between (13 + 13*2)*33 and (14 + 14*3)*36, or between 1287 and 2016 total hours. The total expected hourly load of our master’s degree is 1650 hours, within the range estimated by GSwE2009. Our program has 1050 hours devoted to courses and 600 hours for the final thesis. Note that our thesis hourly load is approximately twice as much as is estimated by GSWE2009 for capstone experiences.

We feel that our program is a successful adaptation of GSwE2009 to meet our local needs. Although we do not currently include all the KAs in CBOK in our implemented courses we expect to hire more faculty in the future to allow us to teach all the needed subjects.

8. Summary

GSwE2009 is a welcome update to the graduate software engineering guidelines created more than 20 years earlier at the SEI. Its structure provides flexibility for a variety of types of individual programs while maintaining a useful standard of excellence.

Schools that wish to establish new degree programs are given clear guidance in recommended subjects and pedagogy. This is seen in the way that the UdelaR program organized their curriculum around the 11 KAs of CBOK. They also made sure to include a capstone experience consistent with GSwE2009 recommendations. When they measured the expected student effort it was consistent both with GSwE2009 guidelines and with their own expectations for graduate students. Another new program, at Rose-Hulman Institute, made good use of the GSwE2009 expected student outcomes and the integration of systems engineering topics.

Schools that seek to refresh or strengthen their programs are also well served by its contents, especially the CBOK. The program at the University of Sofia extended their curriculum material in areas that needed refreshing, such as legal issues and agile methods. The program at Stevens Institute updated material in systems engineering and security. Existing programs should consider reviewing their curricula to see if they cover all of the CBOK of GSwE2009.

In addition, GSwE2009 serves as a model for other curriculum efforts. The MSwA project used the GSwE2009 curriculum architecture as a model for its own curriculum architecture. In particular, the ability to accommodate different starting points (expected student backgrounds) was helpful in matching the architecture to different academic programs.

We expect that many more schools will adopt or adapt GSwE2009 before it needs updating. Hopefully we won't have to wait another 20 years for that to occur.

References

[1] Pyster, A. (ed.), Graduate Software Engineering 2009 (GSwE2009) Curriculum Guidelines for Graduate Degree Programs in Software Engineering, Integrated Software & Systems Engineering Curriculum Project, Stevens Institute, September 30, 2009. http://www.gswe2009.org/curriculum/recommendations/document/

[2] Fairley, R.E. Toward Model Curricula in Software Engineering, Proceedings of the Ninth SIGCSE Technical Symposium on Computer Science Education, ACM SIGCSE Bulletin Volume 10 Issue 3, August 1978, pages 77-79.

[3] Ardis, M., The design of an MSE curriculum, in Proceedings of the SEI Conference on Software Engineering Education, (published as Springer Lecture Notes in Computer Science 327), April 1988.

[4] Ardis, M. & Ford, G. SEI Report on Graduate Software Engineering Education (CMU/SEI-89-TR-021). Software Engineering Institute, Carnegie Mellon University, 1989. http://www.sei.cmu.edu/library/abstracts/reports/89tr021.cfm

[5] SWEBOK, Guide to the Software Engineering Body of Knowledge, P. Bourque and R. Dupuis (Eds.). IEEE Computer Society Press, 2004, http://www.computer.org/portal/web/swebok

[6] Bloom, B.S. (Ed.), Taxonomy of educational objectives: The classification of educational goals: Handbook I, cognitive domain, Longmans, 1956.

[7] Mead, N. R., et al. (2010a). Master of software assurance reference curriculum (CMU/SEI-2010-TR-005/ESD-TR-2010-005). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. http://www.cert.org/archive/pdf/10tr005.pdf

[8] M. Ardis, D. Frailey and N. Hutchison (ed.s), Comparisons of GSwE2009 to Current Master's Programs in Software Engineering, Stevens Institute of Technology, November 2009.

[9] Software Assurance at Stevens Institute of Technology http://stevens.edu/softwareassurance

[10] ACM/IEEE, Software Engineering 2004, Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, ACM and the IEEE Computer Society, 2004.

[11] ACM/IEEE, Computing Curricula 2005 - The Overview Report, ACM and the IEEE Computer Society, 2005.

[12] The Bologna Process - Towards the European Higher Education Area http://ec.europa.eu/education/higher-education/bologna_en.htm, retrieved 1/24/2013.

[13] The Lisbon Treaty: a comprehensive guide, http://europa.eu/legislation_summaries/institutional_affairs/treaties/lisbon_treaty/index_en.htm, retrieved 1/24/2013.

[14] Career Space Curriculum Development Guidelines: New ICT Curricula for the 21st Century, Designing Tomorrow's Education, http://www.career-space.com/, 2001a.

[15] ACM/IEEE, Computing Curricula 2001, Computer Science, ACM and the IEEE Computer Society, 2001.

[16] ACM/AIS/AITP, Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems, Association for Computing Machinery (ACM), Association for Information Systems (AIS) & Association of Information Technology Professionals (AITP), 2002.

[17] Nikolov R., S. Ilieva, (2007) Education in Informatics at Sofia University - current status and future plans, Electronic Journal Innovation in Teaching And Learning in Information and Computer Sciences (ITALICS), Volume 6, Issue 3, June 2007, pp. 65-77, http://www.ics.heacademy.ac.uk/italics/vol6iss3.htm

[18] ACM/IEEE, Computing Curricula 2008, Computer Science, ACM and the IEEE Computer Society, 2008.

[19] SWEBOK Certificate Program, http://www.computer.org/portal/web/certification/scp, retrieved 1/24/2013.