AUTHOR(S): LARUSDOTTIR, M., DANIELS, M. and MCDERMOTT, R.

TITLE: Quality assurance using international curricula and employer feedback.

YEAR: 2015

Publisher citation: LARUSDOTTIR, M., DANIELS, M. and MCDERMOTT, R. 2015. Quality assurance using international curricula and employer feedback. In D'Souza, D. and Falkner, K. Proceedings of the 17th Australian computing education conference (ACE 2015), 160, 27 - 30 January 2015, Sydney, Australia. Sydney: ACS Inc [online], pages 19-27. Available from: http://crpit.com/confpapers/CRPITV160Larusdottir.pdf

OpenAIR citation: LARUSDOTTIR, M., DANIELS, M. and MCDERMOTT, R. 2015. Quality assurance using international curricula and employer feedback. In D'Souza, D. and Falkner, K. Proceedings of the 17th Australian computing education conference (ACE 2015), 160, 27 - 30 January 2015, Sydney, Australia. Sydney: ACS Inc, pages 19-27. Held on OpenAIR [online]. Available from: http://openair.rgu.ac.uk.

Quality Assurance using International Curricula and Employer Feedback

Marta Lárusdóttir
Reykjavik University
Reykjavik, Iceland
marta@ru.is

Mats Daniels
Uppsala University
Uppsala, Sweden
mats.daniels@it.uu.se

Roger McDermott
Robert Gordon University
Aberdeen, Scotland
roger.mcdermott@rgu.ac.uk

Abstract

The focus of this paper is the quality assurance process for the bachelor program in the School of Computer Science at Reykjavik University, which is a combination of outcome- and process-oriented quality assurance. Faculty members and employers of graduates provided information for the quality assessment. The results provide both detailed quantitative data and more qualitative information that give all stakeholders a variety of ways to interpret the status of the quality of education. This type of assessment has raised the awareness of the faculty members on how abstract topics and learning outcomes from an international standard can be used when revising the curricula of a particular course. A notable feature of this type of analysis is its use of employer-generated data to examine graduate knowledge and skills. The contribution of the paper is to provide an example of how a quality assurance process can be made more valuable to both faculty and degree stakeholders by combining outcome- and process-oriented quality assurance strategies.

Keywords: Quality assurance, Evaluation, Degree programs

1 Introduction

Quality assurance of education programs is a complex task and can serve several different functions such as helping to identify pedagogical strengths and weaknesses in a program, or, in extreme cases, providing evidence for its cessation. This complexity is compounded by the fact that the process itself can be conducted by different stakeholders, e.g. national agencies or individual departments within a particular institution. Moreover, the methodology used - specifically the focus of the quality assurance process and the type of assurance procedures used - may significantly affect the conclusions that are drawn. In most cases attention is directed to either the features of the educational experience (including curriculum content, course administration, delivery and assessment mechanisms...) or to an assessment of the abilities of the graduating students. In both these cases, fundamental questions arise about what precisely should be measured and which set of criteria should be used. These issues are even more problematic when attempting to assess areas for which there may be no obvious or well-established metrics, e.g. professional skills such as intercultural competence. Furthermore, consideration also needs to be given to whether the issues to be measured are known in advanced by those being evaluated, since this could potentially lead to "cosmetic" adjustments made to subvert the accuracy of the evaluation process.

The focus of this paper is the quality assurance process taking place in the computer science bachelor program at Reykjavik University, Iceland. The process was partly influenced by the Swedish national quality assurance process for computer science programs performed in 2012/2013. The Reykjavik process is of interest in that it combines an assessment of program content and delivery with evaluation of graduates' abilities. Rationale for choices, methods for conducting the quality assurance, some results as well as conclusions will be covered in this paper. We highlight two key features of the Reykjavik process. The first is the use of the ACM/IEEE computer science curricula 2013 (ACM/IEEE 2013) (henceforth referred to as the "ACM Curricula 2013") to bridge the gap between the typically fairly abstract national degree criteria and the more tangible aspects of course implementation, and to provide a rather concrete description for evaluating findings. The second is the use of employer responses to assess relevant graduate attributes.

2 Quality Assurance

As stated above, quality assurance is a complex endeavour in which the details of context are important. In this current work, the academic department is taken to be the main stakeholder and performs the quality assurance process in order to ascertain strengths and weaknesses so as to improve the program. There are several ways to ensure the validity of this kind of process. One is to base any review on the accreditation criteria for computer science programs (ABET 2010) devised by internationally recognised accreditation organisations such as ABET (formerly known as the Accreditation Board for Engineering and Technology). ABET conducts assessments, including site visits, outside the US and have also influenced national quality assurance programmes, e.g. in Estonia. Another effort to ensure validity is conducted by the European association for quality assurance in higher education (ENQA) (ENQA 2013), which is an association within the European Union evaluating quality assurance processes in its member countries.

This kind of quality benchmarking is useful, especially for so-called process-oriented quality assurance which focuses on what an education program contains and how it is delivered. An alternative strategy for conducting the quality assurance process is outcome-based assurance where the abilities of students after a course or a degree program are assessed, and this has recently become more popular. ABET changed their assessment strategy towards this at the turn of the century (Lattuca et al. 2006) and Sweden is at the end of four year national quality assurance cycle for all degree programs which mainly uses outcome-based procedures (HSV 2012). Process-oriented assurance focuses on the general process by which education is carried out and there are many readily available sources of information which may be used to feed in to this analysis. However, there is often a lack of attention to the experience of the learner. By contrast, outcome-based assurance tries to assess the quality of the program by determining if suitable outcomes have been achieved. This lends itself to a student-focused approach but assumes that there is agreement on what outcomes should be measured and what constitute the criteria for success. Since both alternatives have their strengths and concomitant weaknesses, there is current interest in looking at approaches which use positive aspects of both practices to evaluate the quality of a program. One such attempt is that of Reykjavik University Computer Science department.

3 The Reykjavik University Setting

3.1 The Computer Science Program

The bachelor program in computer science at Reykjavik University started in 1998 and the taught content was, at that time, strongly influenced by the 1991 version of the ACM/IEEE computer science curriculum (Tucker 1991). The program had an extensive review in 2008 based on the 2001 version for the computer science subfield (ACM/IEEE 2001). During this overhaul, the revision of the standard from 2008 was also taken into consideration (ACM/IEEE 2008). The program includes 17 mandatory course units in computer science and mathematics for a total of 102 ETCS (one ETCS is 1/60 of a "student year") and a mandatory final group work project that is 12 ECTS for each student. In addition, students can select between four "emphasis lines" which consist of 30 ECTS in courses related to their focus subject.

3.2 Quality Assurance Method

In 2013, the program was the subject of a quality assurance evaluation as part of an ongoing national cycle of Higher Education review based on the Quality Enhancement Framework (Rannis 2011). The main aim of this framework is to support the quality assurance efforts of Icelandic Higher Education institutions by providing guidance on the objectives, requirements and operational procedures for evaluating quality at both the institutional and departmental level. In terms of compliance with QA regulation, the main source of documentation, the Quality Enhancement Handbook for Icelandic Higher Education, specifies that "all institutions will be required to conduct regular internal reviews covering each of their subject areas" and the subject-level review was scheduled for the School of Computer Science within the 2013 calendar year.

An important question for such reviews is the basis on which the quality assurance process should progress. As mentioned in section 2 in this paper, there are two basic approaches generally termed process-oriented and outcomes-oriented. The former tends to examine the structural elements of the educational process (e.g. content, curriculum, learning objectives, teaching styles) and map it against some set of trans-institutional standards which act as a benchmark for best practice in the area. The second approach looks at the output of the educational process and tries to determine whether the students that have undergone the experience do indeed possess the knowledge, skills (and attitudes to learning) that the program seeks to deliver. A number of difficulties present themselves in this situation. For example, a choice needs to be made on what constitutes an appropriate criteria of success, how the assessment of these measures should take place, and who provides the data for making such a decision. One influential input to the discussions for the Reykjavik review was the recent (2012/2013) national quality assurance process for computer science in Sweden, which took a strongly outcome-based approach.

The obvious starting point for any examination of educational quality in an Icelandic degree program is the national degree criteria (Rannis 2011). Unfortunately, while providing a useful framework to discuss general aspects of learning at the subject level, these criteria were found to be too abstract to serve directly as the basis for constructing learning objectives for the various course units. Following historical precedent, therefore, it was decided to use the 2013 Ironman draft of the ACM Computer Science curriculum as a bridging document linking the high-level pedagogical objectives of the national criteria to specific learning objectives within particular course units (ACM/IEEE 2013).

An attempt was made to map the general objectives of the national degree criteria to the more specific statement of skills contained in chapter 3 of the ACM/IEEE curriculum document. For example, it was possible to map the statement from the national criteria that a student graduating from a bachelor of science program should be "capable of interpreting and presenting scientific issues and research findings", (Education ministry 2011) to the ACM Curriculum guidelines on communication and organizational skills: "Graduates should have the ability to make effective presentations to a range of audiences about technical problems and their solutions. This may involve face-to-face, written, or electronic communication. They should be prepared to work effectively as members of teams. Graduates should be able to manage their own learning and development, including managing time, priorities, and progress." [ACM/IEEE 2013, p.22].

The example given above illustrates two things. Firstly the ACM document articulated a description of the various knowledge and skills elements to be found within the generic computer science curriculum areas at a much finer level of granularity than the national document itself and this enabled clearer discussion of the criteria for success. Secondly the ACM document served a normative function by acting as a benchmark for comparing the disposition of knowledge and skill elements within the courses of the Reykjavik program with those that the ACM curriculum deemed to be necessary elements of a computer science bachelor program. This gives the process-oriented element of the quality assurance process but it does not address the problem of how to assess outcome-based criteria such as the ability to demonstrate appropriate capabilities in a graduate working environment. In order to evaluate this aspect of the program, information on the performance of newly-graduated students was sought from employers.

3.3 The ACM/IEEE Curricula

The ACM/IEEE document identifies two main pedagogical elements of the curriculum: *Knowledge areas* and *Characteristics of graduates*. The former specifies the content areas of the subject whereas the latter identifies the more general, interdisciplinary skills and competencies that a student should develop through engagement with the educational program.

3.3.1 Knowledge Areas

The knowledge areas are part of the “Body of knowledge” section of the curricular document (ACM/IEEE 2013) and define the topical areas of computer science as seen by ACM and IEEE. There are 18 knowledge areas in the 2013 standard, see table 1 in section 5.1, and two of them are new to this version, i.e. “Information assurance and security” and “Parallel and distributed computing”.

Each knowledge area is described by a list of sub-areas with associated topics and learning outcomes, and the document also specifies a number of “curricular hours” assigned to each sub-area. The sub-areas are identified as either “core” or “elective” and the core parts are in their turn subdivided into “tier-1” and “tier-2”, each with an associated number of “curricular hours”. This classification builds on a view that all computer science programs should ensure that all of the tier-1 and most (preferably 90-100%, but at least 80%) of the tier-2 is mastered by all their students. A complete computer science program should also offer a significant part of the elective material.

3.3.2 Characteristics of Computer Science Graduates

The characteristics of computer science graduates define the competencies these students should have at graduation. The idea behind these definitions is to capture overarching characteristics that typically span several of the knowledge areas and which are important for graduate success in the computer science profession. There are eleven characteristics identified in the ACM curricula 2013 (ACM/IEEE 2013), see table 2 in section 5.2. The expectation is that at least an elementary level of all should be achieved at graduation by all students.

4 Analysis of Educational Setting and Delivery

The analysis of the educational setting and delivery is done in two parts, a process-oriented part and an outcome oriented part. The process-oriented part of the quality assurance evaluation at University A was targeted on the educational setting, both on the course content and the course learning outcomes. The learning outcomes for course units are also relevant for the outcome-oriented part of the evaluation, but to a lesser degree since the evaluation of these are focused more on the competencies students have gained at the time of completing the course and not on the holistic competences graduates have when completing the bachelor degree.

The educational setting is analyzed by two separate methods. The first was to compare how many of the topics and learning outcomes suggested in the ACM curricula 2013 were situated in mandatory courses at Reykjavik University. All faculty member teaching mandatory courses took part in this evaluation (n=10), including 2 professors, 2 associate professors and 6 assistant professors. Each faculty members checked how many of the topics suggested in the ACM standard are covered in their course and the degree to which the learning outcomes articulated in the course unit documentation matched that found in the ACM document. This comparison was structured by the knowledge areas from the ACM curricula 2013.

The second method was to estimate how much focus was placed on each of the characteristics of computer science graduates in the mandatory courses. A guideline document was developed to assist the faculty members in conducting this comparison. There was an initial workshop for the faculty when the quality assurance evaluation was introduced and the chosen process explained. The process involved several stages in order to guide faculty members in how to conclude their part of the assessment and was concluded with a joint workshop analyzing the results from both methods.

4.1 Analysing Educational Setting - Coverage of Topics and Learning Outcomes

For the first method of analysing the educational setting, a spreadsheet with topics and learning outcomes for tier-1 and tier-2 of the knowledge areas was composed. The faculty were asked to fill in the coverage for each topic and learning outcome associated with the knowledge areas related to the courses they teach using the guideline document.

The spreadsheet contained the topics for each knowledge area in each course and the extent to which the topics were covered. The coverage of the learning outcomes for each knowledge area was captured in terms of the ACM/IEEE levels of achievement (termed familiarity, usage, or assessment) as well as describing assessment method, i.e. (written) exam, oral (exam), group (project), (individual) assignment, and other. All of this was also subdivided into tier-1 and tier-2.

In the analysis phase, the coverage for each knowledge area was computed as a percentage and the level of learning outcome was compared to the expected level in the ACM curricula 2013.

4.2 Analysing Educational Setting - Emphasis on Characteristics of Graduates

A spreadsheet with the mandatory courses and the specified characteristics of computer science graduates (with the exception of the first, which was assumed to be covered by the analysis of knowledge areas covered) was composed. Faculty were then asked to fill in the level at which each characteristic is supported. This was encoded as a "0", "1" or "2", i.e. not covered or only marginally mentioned (0), part of the course (1), and central to the course (2). In addition, faculty were asked to comment on their evaluation.

In the analysis phase, the number of 0's, 1's and 2's were computed for each mandatory course. The number of courses with 0's, 1's and 2's for each characteristic was also computed.

4.3 Analysing Educational Delivery – The Employers' Assessment of Graduate Skills

The main target of the second part of the evaluation was employer perception of graduate skills. Ten companies were chosen and semi-structured interviews were conducted, each lasting for about one hour. The interviewees work in different domains: two at big software companies (more than 100 employees), two at middle size companies (around 40 employees), two at web development companies, two at telecommunication companies, one at a software development department in a bank and one at a gaming company. Three of the interviewees were female and seven males. Typical roles of the interviewees were: Director of the company, director of IT department and chief development officer so they all had a managerial role and had been involved in hiring people for the last three to 13 years. All except one had hired graduates from SCS at RU, and the percentage of hirings from RU was typically 50-70% of all the hirings.

The interviews were all conducted at the workplace of the interviewees, typically in a meeting room. Two faculty members from SCS at RU attended each interview and, roughly speaking, one of them led the interview whereas the other one took notes. The interviews lasted from 45 minutes up to one hour. The interviews were semi-structured and the major topics covered in the interviews were background information about hirings and the company, their opinion of graduates from SCS at RU, their comparison of graduates from RU to graduates from other universities and their thoughts about possible new study programs or courses. Near the end of the interview we asked the interviewees if they had some general comments or questions. All interviews were audio recorded for further references. Interviewees were asked to fill in a web based questionnaire based on the characteristics of graduates described in the ACM Curricula 2013 (ACM/IEEE 2013) that was sent to them after the interview.

4.3.1 Interviews with Employers

The interviewees were asked about their background at the companies and if they had been involved in hiring graduates from Reykjavik University. They were also asked to provide numbers of hirings of BS graduates in Computer Science from Reykjavik University. The main focus of the interviews was to ask about the employees' opinion of the performance of the graduates from Reykjavik University, and especially to get their views of the strengths and weaknesses of the graduates' education. In addition, interviewees were asked if they thought that some knowledge or skill was missing, and whether there was a need for new courses, or lines of emphasis, which would satisfy their own need to recruit better qualified graduates.

All interviewees were willing to discuss these issues and gave good comments and feedback on these questions.

4.3.2 Questionnaire to Employers

A web-based questionnaire was constructed based on eleven characteristics of computer science graduates from the ACM Curricula 2013. Employers were then asked to rate how well graduates from Reykjavik University performed on each of these, based on a 5 point Likert scale, e.g. employers were asked to rate if they agreed that: "Graduates from Reykjavik University have good project experience skills". They were also asked to rate the importance of each of the characteristics (e.g. "Project experience skills are important for my company").

As the data sought by the questionnaire was much more detailed than that provided by the interviews, it was decided to send this afterwards in the expectation that this would maximize the quantity and quality of the data returns. Only seven interviewees concluded the questionnaire. One interviewee had not hired any graduates from RU, so this person was naturally dismissed concluding the survey, but despite several emails, the two missing responses were not forthcoming. The questionnaire was anonymised, so it was impossible to find out which people did not respond.

5 Findings

The results of the analysis of the educational setting are summarized below. Table 1 presents the knowledge area topics and the learning outcomes for those knowledge areas, and table 2 presents the characteristics of computer science graduates. The summary of results from the employer survey is given in table 3.

5.1 Coverage of Knowledge Areas

According to the ACM/IEEE curricula, all of tier-1 should be covered for all computer science programs. Analysis of table 1 regarding the coverage of knowledge areas (KAs) reveals that this is not the case for the mandatory courses at Reykjavik University, which is, perhaps, not surprising since the program was being compared to a cutting edge standard. Coverage of six KAs are fully covered or almost so, meaning that close to half of the tier-1 KAs are satisfied. However six are either not covered at all or only covered to a small extent and three are covered to some degree, which together with a total coverage of 65% of tier-1 indicates a need for change if striving to follow the ACM curricula 2013.

		Topics						Learning Outcomes					
		Tier 1	Covered	%	Tier 2	Covered	%	Tier 1	Covered	%	Tier 2	Covered	%
AL	Algorithms and Complexity	22	21	95%	15	11	73%	25	24	96%	14	10	71%
AR	Architecture and Organization	0	0		39	29	74%	0	0		36	27	75%
CN	Computational Science	4	0	0%	0	0		5	0	0%	0	0	
DS	Discrete Structures	35	30	86%	5	3	60%	30	25	83%	6	3	50%
GV	Graphics and Visualisation	4	2	50%	4	2	50%	4	2	50%	4	2	50%
HCI	Human-Computer Interaction	10	10	100%	8	7	88%	5	5	100%	3	3	100%
IAS	Information Assurance and Security	16	4	25%	24	6	25%	17	3	18%	25	5	20%
IM	Information Management	4	0	0%	16	11	69%	6	3	50%	23	14	61%
IS	Intelligent Systems	0	0		20	5	25%	0	0		19	3	16%
NC	Networking and Communication	10	10	100%	22	15	68%	7	7	100%	15	10	67%
OS	Operating Systems	12	11	92%	19	16	84%	12	9	75%	24	23	96%
PD	Parallell and Distributed Comp.	10	0	0%	21	3	14%	6	1	17%	22	3	14%
PL	Programming Languages	11	11	100%	20	19	95%	7	6	86%	15	15	100%
SDF	Software Development Fundamentals	24	24	100%	0	0		39	37	95%	0	0	
SE	Software Engineering	8	6	75%	36	25	69%	13	10	77%	50	32	64%
SF	Systems Fundamentals	24	9	38%	17	11	65%	29	9	31%	15	8	53%
SP	Social Issues and Prof. Practice	23	4	17%	14	3	21%	33	3	9%	14	0	0%
		217	142	65%	280	166	59%	238	144	61%	285	158	55%

Table 1: Coverage of knowledge areas - topics and learning outcomes

Looking at tier-2, which is recommended to be covered at above 80%, we see that 59% of this is covered and thus does not conform to the ACM benchmark. The KAs covered well at the tier-1 level are also catered for at tier-2 and a few of the KAs not covered at the tier-1 level are covered to a better degree at the tier-2 level. The problematic ones are those deemed to be covered at neither level.

The information assurance and security and the parallel and distributed computation KAs are among these, which is not surprising since these were only introduced in the 2013 version of the curriculum recommendation. The social issues and professional practice topic is the third KA not covered at the required level in either tier-1 or tier-2. This probably reflects the observation that faculty as well as program coordinators have a focus on the technical aspects of computer science. This assumption is further investigated in a forthcoming article (Daniels et al 2015). Two other KAs worth noting are intelligent systems and computational science, both of which are peripheral to the intentions of the program and consequently it is not unexpected that these scores are low.

Some of the areas are covered in elective courses, but this is deemed to not be of interest here, since the intent is to investigate the areas that all students should learn.

The data for learning outcomes show slightly worse results than the preceding investigation of topics covered. The KAs with poor coverage reappear when looking at the learning outcomes, which is perhaps not surprising. The two new areas are just slightly worse with regard to assessing learning objectives, but a significant low score is presented by the social issues and professional practice KA. This KA is barely covered at all when it comes to assessment, which is probably related to faculty being unsure about how to assess such competencies in general. Previous work on assessing professional competencies (Daniels 2011, Cajander et al. 2012) can provide support so as to improve this situation.

5.2 Coverage of Characteristics of Graduates

Investigation of table 2 regarding the focus on characteristics of computer science graduates, called competencies in the following, reveals that just over a third of the mandatory courses cover all of the competencies. However, a more interesting question is whether there are aspects of developing competencies that come up in few courses and at a superficial level, since those cases could indicate a lack of provision for allowing development of the competencies in question.

Characteristics of graduated	Introduction to CS	Software requirements and design	Programming	Practical project	Discrete math I	Discrete math II	Software Engineering	Computer Architecture	Algorithms	Problem solving	Calculus and Statistics	Data structures	Operating systems	Computer communications	Databases	Programming languages	Web programming	Final Project	**Not in the course**	**Covered**	**A core subject in the course**
Familiarity with common themes and principles	1	1	0	1	1	1	1	1	2	0	1	1	1	1	1	0	1	1	3	14	1
Appreciation of the interplay between theory and practice	0	1	0	2	1	1	1	1	2	1	1	1	1	1	1	1	1	1	2	14	2
System-level perspective	1	2	0	2	0	0	1	1	1	0	0	1	2	1	1	1	2	1	5	9	4
Problem solving skills	1	2	1	2	1	1	1	1	2	2	1	1	2	1	0	0	1	2	2	10	6
Project experience	1	1	0	2	0	0	2	0	0	0	0	0	0	0	1	0	2	2	11	3	4
Commitment to life-long learning	2	1	1	1	0	0	1	0	1	0	0	0	0	0	1	1	1	1	8	9	1
Commitment to professional responsibility	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	1	1	11	7	0
Communication and organizational skills	1	1	0	2	0	0	2	0	1	2	0	0	0	0	1	0	1	2	9	5	4
Awareness of the broad applicability of computing	2	1	1	0	0	0	1	1	0	0	0	0	1	1	1	0	1	1	8	9	1
Appreciation of domain-specific knowledge	1	1	1	1	0	0	1	1	0	0	0	0	2	1	0	0	0	2	9	7	2
Number of 0	1	0	5	1	7	7	0	4	4	7	7	6	4	4	3	7	1	0			
Number of 1	7	8	5	4	3	3	8	6	3	1	3	4	3	6	7	3	7	6			
Numer of 2	2	2	0	5	0	0	2	0	3	2	0	0	3	0	0	0	2	4			

Table 2: Coverage of characteristics of computer science graduates

None of the faculty members emphasise the "Commitment to professional responsibility" as a core competency in their course. During discussions about this result among faculty members, two alternatives were proposed for improvements. The first was to embed elements of this topic in a variety of course units within the program. The second alternative was to include this material in the course unit called "Introduction to computer science" in their first semester. While the "Project experience" competence may appear to be underrepresented within the program, being only covered in seven courses, in four of these it is the major pedagogical component. Many courses also include project work as a problem solving experience, so students are often developing this competence by working in groups to solve smaller projects.

This type of activity provides a process-oriented analysis of the Reykjavik program and illustrates the benefits that can be gained by comparing the current curriculum with an international standard. However, it does not address the question of how effective such a curriculum is for student-learning. For this, it is more natural to use an analysis which looks at output data, that is done in the next section.

5.3 Outcome Oriented Assessment

For an outcome-oriented assessment of curricular content/knowledge areas, one source of information are the standard, published output measures such as exam result data which can be correlated with a range of comparable programs in similar institutions. However, it is much more difficult to assess outcomes for the specified graduate characteristics in that way. In addition, exam result data and degree classifications do not necessarily give a complete picture of the range of skills and competencies developed by students throughout their period of study; this may only become apparent when they are asked to demonstrate such capabilities over a sustained period within a professional working environment. It is important therefore to examine the views of stakeholders such as employers who can provide a more contextualised analysis of such competencies.

In order to do this, employers were asked to estimate how well newly-qualified graduates from RU fulfilled these characteristics and how important each characteristic is to their company. Responses were given using a Likert scale from 1 to 5, see results in table 3.

Item	Applies to RU graduates	Important for company
Technical understanding	4.00	4.86
Familiarity with common themes and principles	3.71	4.86
Appreciation of the interplay between theory and practice	3.57	4.00
System-level perspective	3.86	4.43
Problem solving skills	3.86	4.86
Project experience	3.86	4.29
Commitment to life-long learning	3.71	4.86
Commitment to professional responsibility	3.43	4.57
Communication and organizational skills	3.86	4.29
Awareness of the broad applicability of computing	3.14	3.71
Appreciation of domain-specific knowledge	4.00	4.29

Table 3: Summary of the results from the employer survey

There were five characteristics that the employers rated as very important to their company, having an average above 4.5 in importance. These were: "Familiarity with common themes and principles", "Problem solving skills", "Commitment to life-long learning", "Commitment to professional responsibility" and "Technical understanding". For the first four of those the difference between the importance rating and how well that competence applies to RU graduates is 1.0 or more (marked in red in the table) indicating that these characteristics should be a particular focus for curriculum development when changing the program in the future.

It should be noted that "Commitment to professional responsibility" was not emphasised as a core subject in any of the mandatory courses in the curriculum, so that particular result is not unexpected. "Familiarity with common themes and principles" and "Commitment to life-long learning" are each only emphasised as a core subject within one course unit, so again, the difference between the needs of employers and graduate performance may not be surprising. However, problem solving skills are emphasised in six compulsory courses, so the difference between the two ratings is disappointing and indicates an important gap for that competence that needs to be addressed through curricular enhancement. In this particular case, further investigation suggested that the difference could be related to some respondents' perception of a recent, local decline in programming skills.

5.4 Further Results from the Employers

The feedback from the employer survey indicated that there were no major concerns about the levels of competence of the RU graduates and in general, the view was positive. Four interviewees mentioned that RU prepares graduates well for working in the industry after their studies, and that RU students were proficient with the tools and processes used in the industry, particularly the agile methodology. Three of the employers had groups working in parallel in other countries (Ukraine, Serbia and Britain), which allowed them to discuss the relative strengths of the RU graduates with those they have worked with from other countries. The respondent working with a team in Ukraine stated that in his/her opinion, the Ukrainian employees are better programmers and want to discuss methods, understand and have opinions on solutions. The respondent having a team in Serbia described that those team members have more theoretical education and not as much practical experience as graduates from RU. Finally, the respondent working with a team in Britain noted that it is harder to get a permanent job in Britain than in Iceland, so the British graduates are more focused and more concerned about doing a good job than employees here in Iceland in his opinion.

When asked about RU graduates weaknesses, there were various answers. Some employers mentioned that RU graduates should develop more professional behaviour and show better discipline in their work. Two respondents mentioned that the programming skills of RU graduates should be improved, and one informant mentioned that RU graduates could have better skills in designing from scratch using design patterns. Two informants mentioned that RU graduates could improve their testing skills and one mentioned in particular that automatic testing should be emphasised more in the RU programs. One respondent mentioned that their company has one tester per every four programmers and it has been hard to find good testers on the market.

When asked about, if there were some courses or topics missing in our curriculum, the answers were really spread, mentioning web programming, front end programming, testing and management of IT systems. The employers were asked specially about the structure of the studies. One employer mentioned that he would like us to have four lines: one for "hard core" programming; one for web programming; one system administration (system administrators are mostly not educated at a university level), and one testing line. Additionally one informant wanted to divide our studies in two lines; one programming line and one front-end programming line.

The employers in general want better work ethics, emphasis on testing and more commitment to quality. It is also important to keep in mind that the employers felt more individual differences between their employees, rather than thinking of them as RU graduates, graduates from other universities in Iceland or abroad. Therefore probably many of their comments can be interpreted as holding for CS graduates in general rather than only for the RU graduates. However, their comments are useful to improve the studies at RU in order to prepare RU graduates better for their future jobs in industry.

6 Discussion

In this section we will first discuss the validity of the findings and then summarise and discuss the lessons learned.

6.1 Validity of the Findings

The validity of the findings is subject for discussion. While some element of confirmation bias will be present due to the evaluation being done by faculty with a vested interest in a good outcome, the classification system for inclusion is fairly transparent and standard moderation practices would mitigate against this. There is also a question of consistency both in terms of how well the faculty entered numbers into the spreadsheets, and more importantly, their understanding of what the terms meant. However, faculty information events prepared academic staff for the process and this would also serve to reduce these kinds of errors.

The evaluation process itself did involve revisiting decisions on the allocation of scores and the concluding session, in which faculty discussed the data provided some degree of confidence in the robustness of decisions about scores and agreement on the meaning of the classification criteria. It should be stressed that the objective of the assessment was to see if there are extensive gaps in the coverage of the knowledge areas suggested in the ACM standard in the curriculum for computer science at Reykjavik University. Consequently the objective was not measure exactly the coverage, but to gather information on whether there were some knowledge areas where the curriculum differed greatly from the topics and learning outcomes suggested by the ACM standard. In the absence of a systematic error, this objective would be reached even if some faculty members were too positive/negative about the details of their own course units.

One further source of concern is the likelihood that the technical aspects of the curriculum are better understood by the faculty involved in the evaluation than those that relate to competencies. That results and the generally poorer outcome for the competencies in the process-oriented analysis, indicate that further work is required to establish a common understanding of what competencies are and how they can be developed and assessed. This point also applies to other stakeholders, such as employers, who appear to be even less accustomed to vocabulary related to competencies than faculty.

6.2 Lessons learned

It was generally felt by faculty that combining the process-oriented evaluation based on the ACM standard with outcome-oriented evaluation, based on interviews with employers of graduates, provided a good methodology for obtaining a more complete picture of the quality of the program. The process generated both detailed quantitative data and more qualitative information that gave stakeholders a good mixture of results to interpret the status of the quality of the education. This assessment has raised the awareness of the faculty members of what topics and learning outcomes should be included in their courses, when revising the curricula of the courses. Already a half a year after the exercise, some of the faculty members have used the results of the assessment to iterate their course content and learning outcomes for the course.

In future iterations of similar comparisons between the topics and learning outcomes in the RU curricula to the ACM standard, it would be beneficial to ask the faculty member responsible for each course to estimate how much of the course is used on topics and learning outcomes that are covered in the standard and then how much time is used on other topics and learning outcomes. This would help to estimate how much is taught beyond a given standard and will therefore give more holistic picture of the curricula. Another lesson is that faculty members were asked to mark how each learning outcome is tested, e.g., individual or group assignment, is it on the test, etc., but that data was not analysed, so that information is not needed in future comparisons.

Conducting the interviews with employers of graduates from RU was a positive experience. All the respondents appeared to be open minded and willing to give feedback, both on the skills of the RU graduates and, in more general terms, on how the CS education could be improved to better satisfy the needs of their company. We asked them to estimate how many employees they had hired from RU the last five years, but unfortunately did not manage to give adequate notice before requiring this information. Our experience was that they would have needed a longer time to answer that question properly. Asking them to fill in a questionnaire after the interviews was good, because the interviews dealt with general issues and so the response to the questionnaire was on a more detailed level. However, it was hard to obtain the data in which we were interested, so one alternative would be to ask the informants to fill in the questionnaire on paper during the interview. The downsides of this alternative are that filling in the questionnaire would take time from the interview itself and it might affect the interviewees' responses by observing them. Additionally filling in the questionnaire during the interview would probably change the focus of the interviewees to talking about the questions they had answered in the questionnaire.

7 Conclusions

Going through a quality assurance process can be quite frustrating and consume a great deal of time and energy. There were much controversy around the Swedish national process especially about the lack of feedback to the degree granting institutions about how to enhance their educational setting as a result of the experience. The Reykjavik process was, on the other hand, received quite positively after some initial complaints about having to go through with the work. It therefore provided an excellent opportunity to discuss the results and move towards improving the computer science program.

The ACM/IEEE computer science curricula 2013 [1] is an important contributor to the positive reaction in Reykjavik. It served well as a replacement for local learning objectives in the computer science program, since those were rather outdated and were instead a target for improvement after the quality assurance process. The good fit of the ACM curricula [1] with the national degree criteria in Iceland [13] was important for those responsible for reporting to the national project.

The Reykjavik quality assurance process illustrates how the ACM curricula 2013 [1] can be used to provide a well-founded base for further discussions about development of an education program. While we believe that there is no clear resolution to the question of how compliant a program should be with regard to the ACM tier-1 and tier-2 criteria or how much conscious deviation from the standard should be allowed, we nevertheless believe that it is of high value to bring it up to the table for discussion.

It is also a welcome finding that the ACM curricula could be used to capture traditionally abstract learning objectives regarding general competencies. The ACM curricula turned out to be an excellent base for conducting semi-structured interviews and constructing a survey in order to get information from employers of students from the education programme. Satisfaction of learning objectives regarding general competencies is in our opinion often quite questionable in computer science programs of today and we hope this work will encourage others to look seriously into how to achieve this.

8 Acknowledgements

We would like to thank professor Anna Ingolfsdottir and assistant professor Henning Ulfarsson for their contribution in the subject level review while planning, analyzing and reporting the findings described here. Furthermore we would like to thank all the faculty members that took part in the review for their contribution.

9 References

ABET (2010) Criteria for accrediting computing programs, ABET, Inc, Baltimore

ACM/IEEE (2013) Computer Science Curricula 2013, Ironman draft, February 2013, http://ai.stanford.edu/users/sahami/CS2013/ironman-draft/cs2013-ironman-v1.0.pdf, assessed December 19, 2013

ACM/IEEE (2001). Joint Task Force on Computing Curricula. Computing Curricula 2001 Computer Science. Journal of Educational Resources in Computing (JERIC), 1 (3es), Fall 2001. Retrieved from: http://www.acm.org/education/education/education/curric_vols/cc2001.pdf

ACM/IEEE (2008). Joint Task Force on Computing Curricula: Computer Science Curriculum 2008: An Interim Revision of CS 2001. Report from the Interim Review Task Force, Association for Computing Machinery, IEEE Computer Society. http://www.acm.org//education/curricula/ComputerScience2008.pdf

Cajander, Å., Daniels, M., and McDermott, R. (2012) "On Valuing Peers – Theories of Learning and Intercultural Competence", Computer Science Education, vol 22, pp 319-342

Daniels, M. (2011) Developing and Assessing Professional Competencies: a Pipe Dream? Experiences from an Open-Ended Group Project Learning Environment, Digital Comprehensive Summaries of Uppsala Dissertations from the Faculty of Science and Technology, nr 808, Acta Universitatis Upsaliensis, Uppsala.

Daniels, M., Cajander, Å., Clear, T., and McDermott, R. (20015) Collaborative Technologies in Global Engineering: New Competencies and Challenges, International Journal of Engineering Education (accepted for publication)

Education ministry (2011) National qualification framework for higher education, http://www.stjornartidindi.is/DocumentActions.aspx?ActionType=Open&documentID=afd35930-4c5a-4de4-bd7e-2134da404446, assessed August 28, 2013

ENQA (2013), Guidelines for external reviews of quality assurance agencies in the European higher education area, ENQA, Occasional papers, 19

HSV (2012) Högskoleverkets system för kvalitetsutvärdering 2011–2014, Rapport 2012:15 R, Högskoleverket, Stockholm, Sweden

Icelandic Center for Research - Rannis, (2011) "Quality Enhancement Handbook for Icelandic Higher Education", retrieved on the 10th of January 2014 at: http://www.rannis.is/files/handbook_complete_1558767620.pdf

Lattuca, L. R., Terenzini, P. T., & Volkwein, J. F. (2006). Engineering Change: A Study of the Impact of EC2000: Executive Summary. ABET, Incorporated.

Tucker, A. B. (1991). Computing curricula 1991. Communications of the ACM, 34(6), 68-84