

The University of Ottawa Undergraduate Software Engineering Program: Leading and Innovative

Timothy Lethbridge, Liam Peyton, Daniel Amyot, Stéphane Somé
 School of Electrical Engineering and Computer Science
 University of Ottawa
 Ottawa, ON K1N 6N5 Canada
 tcl, lpeyton, ssome, damyot@eecs.uottawa.ca

Abstract—We nominate the University of Ottawa Software Engineering Program for the 2017 CSEE&T Hall of Fame. This was one of the first three undergraduate software engineering programs in Canada. It was used to help guide the development of the IEEE/ACM SE2004 curriculum recommendations. It is accredited by both the Canadian Engineering Accreditation Board and the Computer Science Accreditation Council of Canada. It is a bilingual English-French program that has undergone continuous improvement over the years, and now is co-op only, meaning that all its students are required to have three work terms of job experience before graduating.

Keywords—software engineering undergraduate program

I. INTRODUCTION

The University of Ottawa (uOttawa) established its undergraduate Software Engineering (SE) program in 1997. Its first students graduated in 2001, at which time it was accredited by the Canadian Engineering Accreditation Board and also by the Computer Science Accreditation Council of Canada. The program was one of the first three in Canada, the others being at Western University and McMaster University. It is the only bilingual program in Canada.

The full program as it exists today appears on the uOttawa website [1] as does its list of courses [2]. Table 1 gives an outline of the program with courses designed especially for the SE program coloured orange, other computing courses in yellow, math courses blue, plus engineering and science in green. There is also a management and entrepreneurship option that is omitted due to space constraints (see <http://goo.gl/Rftekp> for details).

The program accepts about 75 new students each year who can study in English, French or both. Most courses are offered in either language, with a few (such as the capstone) being bilingual. Students are required to complete three co-op work terms in industry; most complete four. The first year Professional Communication and Responsibility course prepares them for this.

II. EARLY HISTORY OF THE PROGRAM

Prior to the mid 1990s, focused SE programs had mostly been at the Masters level, e.g. [3]. Undergraduate programs in computer science and computer engineering usually covered only parts of the growing field of SE, so many felt it was time

to expand the undergraduate SE offerings. CS programs would remain for those who want flexibility and coverage of the breadth of computing, while SE programs would focus on life-cycle development and management of large systems, with an engineering mind-set. The objective is to produce students who can competently lead development of high quality software.

Table 1: Overview of the uOttawa SE Program

First year	Second year
CSI, CS2	Computer Architecture
Digital Systems	Discrete Structures
Professional Communication and Responsibility	Data Structures / Algorithms
Calculus 1, 2	Databases I
Discrete Math	Intro to SE
Physics 1, 2	Software Construction
Technical Report Writing	Linear Algebra
3-5 Work Terms Start Summer of Year 1	Prob. & Statistics
	Chemistry
	Engineering elective
Third year	Fourth year
Communications and Networking	Capstone Project 1,2
Design & Analysis of Algorithms	Software Proj. management
Operating Systems	Real Time Systems
Professional Practice	Technical electives (2+)
Requirements Analysis	Other electives (up to 4)
Software Design & Architecture	
Software Quality Assurance	
User Interface Design & Analysis	
Circuit Theory and Electronics	
Engineering Economics	

Several universities initially started by creating SE options attached to other programs. This was the original path followed in 1996 by uOttawa [4]. The Rochester Institute of Technology [5] was one of the universities that took the lead and accepted its first students into a full undergraduate SE program in 1996.

By 1997, uOttawa had converted the Option into a degree program. It also merged its Computer Science department with Electrical and Computer Engineering to form the School of Information Technology and Engineering (later renamed Electrical Engineering and Computer Science). Many other

Canadian universities followed, creating SE programs with considerable similarity to the uOttawa program.

The program today retains many aspects of the program as established in 1997. Some persistent features include:

- A first year largely the same as that of both the Computer Science and Computer Engineering programs, facilitating inter-program transfer. The first years of the three programs contain classic CSI and CS2 courses, calculus and other math courses, technical report writing and some science.
- A core introductory sequence of two SE courses in second year. In 2001 the first author wrote a textbook to cover the first course of this sequence [6]. This first course is also taken by Computer Science and Computer Engineering, but after this, those programs diverge from the SE program.
- A full professionalism and ethics course (SEG2911).
- Four focused courses in third year covering, requirements, design, quality assurance, and user interfaces. The user interface course pro-actively addressed a knowledge gap among software practitioners, later confirmed by research [7].
- A full-year capstone course in fourth year, requiring students in groups of 3-5 to fully manage development of a significant system for real customers [8]. The customers can be in an existing company, or be people in the ‘open market’ who sign up to try out innovative software being developed by entrepreneurial students.

The lead author was invited to join the team developing the SE2004 ACM/IEEE curriculum guidelines [9], and became Curriculum Co-Chair. Many aspects of uOttawa’s program were incorporated into SE2004 model curriculum sequences. The knowledge flow was two-way: Once SE 2004 was established, the uOttawa program was adjusted to match it almost exactly, hence serving as a test case for the standard.

III. CONTINUAL IMPROVEMENT AND INNOVATION

The program has undergone many small modifications in response to feedback from students, accreditors and others. The program carefully tracks various performance metrics to find areas of potential improvement. We meet with students each year to obtain feedback; we also make changes in response to student surveys conducted by the university. The following is a small sample of changes made.

The capstone project has evolved to require agile methods and encourage entrepreneurship. It is taught using a Socratic method where the professor asks students about issues they are facing and dynamically adjusts classes to discuss these issues.

Although the program always had a co-op work experience option, it innovated by making this required in 2015. The university works with employers to find jobs, although some students find their own. This is now the largest co-op program at uOttawa, and the only compulsory co-op program.

The program also introduced a first-year course to train students in key job skills prior to their first work term, including interviewing, professional communication, teamwork, website management, version control with git, and a

“pitch” presentation (evaluated by local entrepreneurs). Other programs at uOttawa are now considering a similar course

An ‘Innovation Project’ elective was recently added as an elective for 3rd- and 4th-year students, allowing them to work individually on a project, in addition to their capstone.

There have also been some ‘failures’ resulting in lessons learned. At one point, we tried a ‘biomedical’ option, but it didn’t have enough distinct content. We have also refactored the science content numerous times: A large amount of choice makes scheduling difficult for students, but a fixed set of courses must be relevant. The current sequence provides a basis in circuits and electronics.

IV. CONCLUSIONS

We nominate the uOttawa Software Engineering program for the CSEE&T 2017 Hall of Fame because: a) It was one of the first undergraduate SE programs. b) It served to guide SE2004 and was in turn guided by SE2004. c) It is, as far as we know, the only English-French bilingual SE program. d) It is dual accredited by both Computer Science and Engineering accreditation agencies. e) It has a constellation of best-practice and innovative features such as requiring co-op, and having a full-semester capstone course with real customers. f) It has led the way for the establishment of other programs, especially in Canada. g) It undergoes continual improvement, incorporating feedback from students and accreditors

ACKNOWLEDGMENT

We thank curriculum committee members who have shaped the program over the years. We thank accreditation visitors, co-op employers and alumni, who have given us feedback. We thank the student association and the many students who have helped guide the program’s development.

REFERENCES

- [1] University of Ottawa, Undergraduate Programs: Software Engineering. <http://catalogue.uottawa.ca/en/undergrad/basc-software-engineering>, visited June 2017.
- [2] University of Ottawa, Programs and Courses: Software Engineering. <http://catalogue.uottawa.ca/en/courses/seg/>, visited June 2017.
- [3] J.E. Tomayko, “Carnegie Mellon software development studio: a five year retrospective”, CSEE&T 1996
- [4] T.C. Lethbridge, D. Ionescu, A. Mili, D. Gibbons, “An Undergraduate Option in Software Engineering: Analysis and Rationale”, 10th CSEE&T, pp. 120-129, 1997.
- [5] J. Fernando Naveda, M.J. Lutz, “Crafting a baccalaureate program in software engineering”, CSEE&T 1997
- [6] T.C. Lethbridge, and R. Laganière. *Object Oriented Software Engineering: Practical Software Development Using UML and Java*, First Edition, McGraw Hill: Maidenhead, UK, 2001
- [7] T.C. Lethbridge “What Knowledge is Important to a Software Professional?”, IEEE Computer, May, pp. 44-50, 2000.
- [8] T.C. Lethbridge, Software Engineering Capstone Project, <http://www.site.uottawa.ca/~tcl/seg4910-11/> Accessed June 2017
- [9] T.C. Lethbridge, R. LeBlanc, A. Sobel, T. Hilburn, J. Diaz-Herrera, “SE 2004: Recommendations for Undergraduate Software Engineering Curricula”, IEEE Software, Nov-Dec 2006, pp. 19-25.