Software Engineering Final Assessment Report & Implementation Plan

Faculty / Affiliated University College	Faculty of Engineering
Degrees Offered	Bachelor of Engineering Science (BESc) in Software Engineering
Modules Reviewed	Software Engineering
External Reviewers	Patrick Lam, Associate Professor, Associate Director of Software Engineering, University of Waterloo Stéphane S. Somé, Associate Professor SEG Co-op Coordinator School of Electrical Engineering and Computer Science, University of Ottawa
Internal Reviewer	Joan Finegan, Associate Dean Faculty of Social Science
Date of Site Visit	November 26, 2018
Evaluation	**Good Quality with Report in Two Years**
Approval Dates	SUPR-U: April 17, 2019 SCAPA: May 1, 2019 Senate (FYI only): June 7, 2019
Year of Next Review	2026-27

In accordance with Western's Institutional Quality Assurance Process (IQAP), the Final Assessment Report provides a summary of the cyclical review, internal responses and assessment and evaluation of the undergraduate Software Engineering module delivered by the Department of Electrical and Computer Engineering. This report considers the following documents: the program's self-study, the external consultants' report and the responses from the Department and Faculty. The Final Assessment Report identifies the strengths of the program, opportunities for program enhancement and improvement and details and prioritizes the recommendations of the external consultants and prioritizes those recommendations that are selected for implementation.

The Implementation Plan details the recommendations from the Final Assessment Report that are selected for implementation, identifies who is responsible for approving and acting on the recommendations, any action or follow-up that is required and the timeline for completion.

The Final Assessment Report and Implementation Plan is sent for approval through SUPR-U, SCAPA, Senate and the Ontario Universities' Council on Quality Assurance and is made available on a publicly accessible location on Western's IQAP website. The Final Assessment Report and Implementation Plan is the only document resulting from the undergraduate cyclical review process that is made public, all other documents are confidential to the Program, Faculty and SUPR-U.

Executive Summary

The Software Engineering program is one of the undergraduate engineering programs offered by the Department of Electrical and Computer Engineering. The program has been continuously accredited by the Canadian Engineering Accreditation Board (CEAB) since 2001. The most recent ICAP review was done at the same time as the CEAB review. The Reviewers were provided with the self-study which included: course descriptions, class sizes, module information, teaching evaluations, percentage of courses taught by full-time faculty, library resources, and the vita of faculty members

On the day of the review, reviewers met with the Vice Provost (Academic Programs), the Department Chair, the Associate Dean, and the Acting Dean of Engineering. Groups meetings were held separately with faculty members, students, administrative staff and technical staff. Overall, the reviewers' impression of the program was positive, and they were particularly impressed by the students.

They felt the curriculum satisfied the requirements for software engineering programs as expressed in the IEEE/ACM Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering. The program has adopted the CEAB graduate attributes model in the current round of accreditation. The 12 graduate attributes of the CEAB map appropriately to the seven Western Degree Outcomes. Indeed, the reviewers believe that amongst the four strategic priorities of Western, software engineering exceled in experiential learning particularly because of its project-focused curriculum and opportunities for internships and co-op. Another area that stood out was academic counselling. Students were effusive in their praise for their counsellors.

Significant Strengths of the Program

The following program strengths are identified in both the self-study and the External Consultants' Report:

- Strong program that meets CEAB accreditation
- A very strong experiential learning component. (For example, the third-year project is organized with the active collaboration of an industrial partner. Students are asked to develop a solution to a real-world problem submitted by the partner.)
- Helpful and supportive academic counsellors
- Excellent well-trained professors with established research records. Teaching excellence has been recognized by teaching awards.
- The virtual computer lab and IT resources
- Excellent, newly-built physical environment

Summary of the Reviewers' Key Recommendations and Department/Faculty Responses

1. Enrolment in software engineering is exploding. The reviewers expressed concerns about long-term viability of the program at current staffing levels. They suggested either hiring additional faculty or implementing smaller enrolment caps. Since the program is popular, enrolment caps are likely to be met with opposition. Thus, hiring one or two additional faculty was seen as the preferred solution. The Department concurs, and in the short-term, has asked for permission in 2019-2020 budget submission to hire one Limited-Term Faculty Member in the Software Engineering Area for three years. The Department hopes to hire eventually a regular faculty member.

2. Computer Science and Software Engineering share common interests, and although a previous IQAP report suggested there be more collaboration between the two groups, there is still room for improvement. At the time of the review, prerequisites and registration barriers made it difficult for Software Engineering students to take required computer science technical electives. The Department is aware of these problems and has introduced proposals to facilitate registration in courses and to allow cross flow so that computer science students can take software engineering courses. To facilitate collaboration, the reviewers recommended that a person from Computer Science be on their curriculum committee. The Department prefers regular meetings with the Computer Science Chair and Undergraduate Chair.

3. The reviewers were impressed by the students' enthusiasm for the program and their insights. Currently, the undergraduate committee includes a student representative in a non-voting role. However, the reviewers recommended that the Department publicize the role of the student on the curriculum committee and ensure regular input from students.

4. The reviewers felt that curriculum modernization efforts emphasizing agile processes should accelerate. "The current state-of-the-art in software development process employs the so-called "agile development model," which is iterative. Previously, software engineering academic curricula tended to teach the "waterfall model", which involved sequentially carrying out discrete steps in the development process; it was originally intended as a straw man and was never viable. Program faculty are aiming to teach the agile model, but the students perceive still too much waterfall-style content, especially in the context of the software requirements course. Specifically, faculty asserted that the design courses employ iterations, but the students do not agree. We believe that the development processes taught and applied in the design projects are still waterfall-type and oriented towards the production of exhaustive quantities of documentation. We strongly recommend that the faculty introduce newer iterative, incremental, customer-centered processes." The Department has examined the three project based-design courses to try to reduce the perception of the waterfall model.

5. Another suggestion to improve the curriculum was to add another course on algorithms. Currently there is one standard course (SE 2205) offered in second year. Given that algorithms is a fundamental computer-science topic and other programs include two courses, the reviewers suggested that the curriculum committee consider adding a second course on design and analysis of algorithms. The Department endorsed this recommendation.

EXHIBIT V, Appendix 26

Other Opportunities for Program Improvement and Enhancement

1. Alumni serve important roles. Their success is a measure of the quality of the program. Moreover, they are a valuable resource for students. Though there are good reasons to establish relationships with alumni, it has been difficult to do so. Faculty believe that contact with alumni must go through the Department of Alumni Relations & Development. It was recommended that they be given more ownership of relationships with alumni.

2. Career Services plays a vital role in helping students get internships and co-ops. Yet students felt that they did not get enough help and reported long wait times to meet with career counsellors. Thus, the reviewers recommended that additional resources be made available to the Career Services office. That said, the Department reports that to address the growth in students looking for placements, an Employer Relationship Specialist has been hired to work with software students and the companies that hire them. They plan to hire a second Career Services Office- Student Specialist. The department's records show that on average a student can have an appointment in a week.

EXHIBIT V, Appendix 26

Implementation Plan

The Implementation Plan provides a summary of the recommendations that require action and/or follow-up. The Department Chair, in consultation with the Dean of the Faculty will be responsible for monitoring the Implementation Plan. The details of progress made will be presented in the Deans' Annual Report and filed in the Office of the Vice-Provost (Academic).

Recommendation	Proposed Action and Follow-up
1. Ensure alignment of faculty resources with student enrolment	-consider an additional faculty appointment or implement a cap on student enrolment
2. Establish stronger ties with computer science	-include a member from computer science in undergraduate meetings -establish regular meetings
3. Solicit feedback from students	-ensure students know that they have representation on undergrad committee -consider whether they should be voting members
4. Emphasize 'agile process' in curriculum	-examine curriculum and course outlines with the view to modernizing it
5. Introduce a second course on algorithms	-prepare course outline and DAP to introduce new course