

Software Engineering
B.S. Program

Program Criteria for Software and Similarly Named Engineering Programs

Introduction

The software engineering B.S. program is housed in the Department of Computer Engineering, which also offers computer engineering B.S. and computer engineering B.S. (Turkish) programs, along with graduate programs. It was established in 2008 and is offered only in English. At the time of the visit, there were 10 faculty members, two technicians, and one administrative officer supporting the program. In fall 2016, the program enrolled 128 students. In the 2015-16 academic year, seven bachelor degrees were awarded.

Program Strength

1. The department chair provides exceptional program leadership, and both the faculty and students acknowledged his interpersonal and management skills. The chair exhibits exceptional attitude and commitment towards advancing the program, which bodes well for the future of the program.

Program Deficiency

1. Program Criteria The applicable program criteria require that the curriculum include computing fundamentals, software design and construction, requirements analysis, security, verification, and validation; software engineering processes and tools appropriate for the development of complex software systems; and discrete mathematics, probability, and statistics, with applications appropriate to software engineering. Review of the course syllabi, self-study, and student transcripts revealed that the topic of security was not addressed in the curriculum. This topic has been added in a required course in 2016-17, but students who graduated in the past were not exposed to this topic. Furthermore, while the curriculum includes a probability and statistics course, it does not appear to include applications appropriate to software engineering and there was no evidence of the application of probability and statistics in software engineering courses. Finally, insufficient evidence was found of the

use of appropriate tools for the development of complex software systems. For these reasons, the program is not in compliance with the applicable program criteria.

- 30-day due-process response: The EAC acknowledges receipt of documentation indicating that CMSE353, Security of Software Systems, was added to the curriculum in fall 2016 and was satisfactorily completed by those graduating at the end of fall 2016. The EAC also acknowledges receipt of information regarding the use of probability and statistics in CMSE473, Software Process and Management, along with evidence that MATH322, Probability and Statistical Methods, is now a prerequisite for CMSE473. The EAC further acknowledges receipt of evidence that appropriate tools are employed in CMSE473. With evidence that the application of probability and statistics and the use of appropriate tools are limited to CMSE473, there is the possibility that these topics may not be adequately covered on an on-going basis. Thus, future compliance with this criterion may be jeopardized.
- The deficiency is now cited as a concern.
- Post 30-day due-process information: The EAC acknowledges receipt of documentation after the 30-day due-process period providing evidence that MATH322, Probability and Statistical Methods, is now a prerequisite for CMSE346, Computer Networks, in addition to being a prerequisite for CMSE473, and that students were tested about application of probability and statistics in CMSE346 in the spring of 2017. Evidence was also provided that students use appropriate tools in CMSE322, Software Design, in addition to CMSE473.
- The concern is resolved.

Program Weakness

1. Criterion 4. Continuous Improvement This criterion requires that the program regularly use appropriate, documented processes for assessing and evaluating the extent to which the student outcomes are being attained. It further states that the results of these evaluations must be systematically utilized as input for the continuous improvement of the program. Review of the self-study and interviews with program faculty in charge of assessment and evaluation

revealed that the program has an assessment process in place, but several issues were identified. First, in two of six of the assessment tools, course grades are being utilized to judge the attainment of student outcomes. While course grades represent the extent to which a student has successfully met an individual faculty member's requirements and expectations for a unit or course, use of course grades alone, when two or more outcomes are associated with a course, is not sufficient to determine the extent to which each student outcome is being attained. Second, the program does not separate computer engineering students from software engineering students when collecting assessment data in common courses, making measurement of attainment of student outcomes for software engineering students difficult. Third, student outcomes (a) through (k) were assessed only in courses common with computer engineering, but were not assessed in the software engineering core courses. Finally, assessment results were reported for one semester only, so there was no indication that the results of assessment are systematically utilized for continuous improvement on an ongoing basis and no specific improvements were reported. For these reasons, the program lacks strength of compliance with this criterion.

- 30-day due-process response: The EAC acknowledges receipt of information regarding the discontinuation of the use grades for assessment of student outcomes. Beginning fall 2016, outcome (h) is being assessed in CMSE 406, one of two culminating major design courses, which includes a requirement for an essay on global, economic, environmental, and societal issues. The EAC also acknowledges receipt of documentation differentiating the collection of data between computer engineering and software engineering students in 2015-16 and continuing in 2016-17. The EAC further acknowledges receipt of an updated course to student outcome mapping, along with information concerning the assessment of student outcomes (a) through (k) in CMSE406. Finally, the EAC acknowledges receipt of evidence that changes made in CMPE371 and the newly added CMPE473 have improved the attainment of student outcome (a).
- The weakness is resolved.

Program Concern

1. Criterion 5. Curriculum This criterion states that the students must be prepared for engineering practice through a curriculum culminating in a major design experience based on the knowledge and skills acquired in an earlier course work and incorporating appropriate engineering standards and multiple realistic constraints. Review of major design project reports showed that students do participate in these major design projects by applying the knowledge from earlier coursework. Unfortunately, this knowledge appears to be applied in an inconsistent or ad-hoc manner in several areas including non-functional requirements specification, use case modeling, architecture and design, and verification and validation. Furthermore, the engineering standards used were not always the most appropriate ones, e.g. use of data flow diagrams in object-oriented design. Finally, the choice of the project in some cases did not allow consideration of realistic constraints, e.g., a hotel management system project and a dormitory management system project. With current practices in the major design experience, the program is in compliance with the criterion, though compliance is at marginal levels. Unless care is taken in the major design experience, future compliance with this criterion may be jeopardized.
 - 30-day due-process response: The EAC acknowledges receipt of documentation that CMSE405 and CMSE406, which comprise the culminating major design sequence, have been modified to require that students use new guidelines and templates insuring use of appropriate methodologies and standards. In addition, these two courses now have a dedicated coordinator who makes sure that realistic restraints are considered and that well-known software development methods are utilized. However, evidence that the new guidelines and template have been put into use has not been provided.
 - The concern remains unresolved.
 - Post 30-day due-process information: The EAC acknowledges receipt of post due-process documentation regarding the use of new guidelines and templates for the capstone design sequence. Although effort has been made to improve capstone projects, issues remain. These include non-functional requirements being specified without using metrics, use case narratives containing technical language rather than a user-oriented language, or non-specification of information used in use case narratives. Use case narratives (or detailed

descriptions) are considered as part of the design. Quality assurance techniques, such as reviews or inspections, are absent. Testing is done in an ad-hoc way without a test plan that covers key test types.

- The concern remains unresolved.

Program Observation

1. The processes used to assess and evaluate student outcomes appear to be complex and time consuming and may not be sustainable. The use of appropriate sampling strategies may make the process less burdensome and more effective.