Oregon Institute of Technology
Computer Systems Engineering Technology Department
Software Engineering Technology Program Assessment Plan
2011-2012

I. Introduction

The Software Engineering Technology (SET) program was implemented in 1984 and was initially accredited by TAC of ABET in 1991. The Portland program was established in Fall 1996 under the same accreditation. The Associate degree was accredited by TAC of ABET in 2009. The program has continuously evolved as industrial changes have warranted.

II. Mission, Objectives and Student Learning Outcomes

The program faculty reviewed and approved, with a few changes, the mission, objectives and student learning outcomes for the program during the first week of fall term. The mission statement, objectives and program outcomes for the baccalaureate program are located on the Oregon Tech website at www.oit.edu/provost/learningoutcomes/cset/swbs. The associate program's mission statement, objectives and program outcomes are located at www.oit.edu/provost/learningoutcomes/cset/swae.

Bachelor Program Mission

The mission of the Software Engineering Technology (SET) Bachelor's Degree program within the Computer Systems Engineering Technology (CSET) Department at Oregon Institute of Technology is to prepare our students for productive careers in industry and government by providing an excellent education incorporating industry-relevant, applied laboratory based instruction in both the theory and application of software engineering. The program is to serve a constituency consisting of our alumni, our employers, and our Industrial Advisory Board. Major components of the SET program's mission in the CSET Department are:

I. To educate a new generation of Software Engineering Technology students to meet current and future industrial challenges and emerging software trends.

II. To promote a sense of scholarship, leadership, and professional service among our graduates.

III. To enable our students to create, develop, apply, and disseminate knowledge within the software development environment.

IV. To expose our students to cross-disciplinary educational programs.

V. To provide government and high tech industry employers with graduates in software engineering and related professions.

Bachelor Program Educational Objectives

The Program Educational Objectives of Oregon Tech's Software Engineering Technology program are to produce graduates that:

A. Use their knowledge of engineering to creatively and innovatively solve difficult computer systems problems.

B. Regularly engage in exploring, learning and applying state-of-the-art hardware and software technologies to the solution of computer systems problems.

C. Will be an effective software development team member that contributes innovative software design solutions to the resolution of business, scientific or government computer systems problems.

D. Will communicate effectively and successfully, both individually and within multi-disciplinary teams.

Bachelor Program Student Learning Outcomes

Software Engineering Technology baccalaureate graduates will have demonstrated:

1. an ability to identify, formulate, and solve software engineering problems, including the specification, design, implementation, and testing of software systems that meet specification, performance, maintenance and quality requirements (Program Objective A, B, and C);

2. the ability to elicit, analyze and specify software requirements through a productive working relationship with various stakeholders of the project (Program Objective A and C);

3. an understanding of the core areas of software engineering. (data structures, theory of computation, operating systems, compilers, programming languages, computer architecture). (Program Objective A);

4. an ability to function effectively on teams (Program Objective A and C);

5. an understanding of professional, ethical and social responsibility (Program Objective C);

6. a recognition of the need for, and an ability to engage in life-long learning (Program Objective C);

7. knowledge of and ability to apply discrete math, probability and statistics (Program Objective B);

8. the successful design, development, deployment and maintenance of a major software based project (Program Objective A, B and C);

9. an ability to convey technical material through oral presentation and interaction with an audience (Program Objective A and C);

10. an ability to convey technical material through written reports which satisfy accepted standards for writing style (Program Objective A and C);

11. an ability to evaluate the impact of potential solutions to software engineering problems in a global society, using their knowledge of contemporary issues and emerging software engineering trends, models, tools, and techniques (Program Objective A and C);

Associate Program Mission

The mission of the Software Engineering Technology (SET) Associate Degree program within the Computer Systems Engineering Technology (CSET) Department at Oregon Institute of Technology is to prepare our students for entry level careers in the software industry and government by providing applied laboratory based instruction. The program is to serve a constituency consisting of our alumni, our employers, and our Industrial Advisory Board. Major components of the SET program's mission in the CSET Department are:

I. To provide a new generation of Software Engineering Technology students with a solid background in computer programming.

II. To enable our students to create, develop and apply knowledge within a technical software environment.

III. To provide government and high tech industry employers with entry level graduates in computer programming and related professions.

Associate Program Educational Objectives

The Program Educational Objectives of Oregon Tech's Software Engineering Technology program are to produce graduates that:

A. Assist in solving computer systems problems using their knowledge of computer programming.

B. Regularly engage in learning and applying state-of-the-art hardware and software technologies to the solution of computer systems problems

C. Will communicate effectively and successfully in the workplace.

Associate Program Outcomes

Software Engineering Technology associates graduates will have demonstrated:

1. an ability to identify, formulate, and solve computer programming problems, including the specification, design, implementation, and testing of programs that meet specification, performance, maintenance and quality requirements (Program Objective A, B, and C);

2. an understanding of the core areas of software engineering (data structures and programming languages). (Program Objective A);

3. an understanding of professional, ethical and social responsibility (Program Objective B);

4. a recognition of the need for, and an ability to engage in life-long learning (Program Objective B);

5. the successful design and development of a computer program (Program Objective A, B and C);

6. an ability to communicate through oral presentation and interaction with an audience (Program Objective B);

7. an ability to convey technical material through written reports which satisfy accepted standards for writing style (Program Objective B);

III. Three-Year Cycle for Assessment of Student Learning Outcomes

The department assesses the program educational objectives and student learning outcomes on a three-year cycle. During the six-year ABET cycle, the objectives and learning outcomes will thus be fully assessed twice.

All appropriate accreditation documents are housed on a SharePoint site maintained by the department. All department members have access to this site, but the documents are not viewable by the general public. The public can view the baccalaureate outcomes at www.oit.edu/provost/learningoutcomes/cset/swbs and the associate outcomes at www.oit.edu/provost/learningoutcomes/cset/swbs.

Bachelor Degree Assessment Cycle

Table 3-1: Baccalaureate Outcome Assessment Timeline

#	**Learning Outcomes**	**10-11**	**11-12**	**12-13**	**13-14**	**14-15**	**15-16**	**16-17**
1	an ability to identify, formulate, and solve software engineering problems, including the specification, design, implementation, and testing of software systems that meet specification, performance, maintenance and quality requirements			**X**				
2	the ability to elicit, analyze and specify software requirements through a productive working relationship with various stakeholders of the project		**X**			**X**		
3	an understanding of the core areas of software engineering		**X**			**X**		
4	an ability to function effectively on teams			**X (I)**				
5	an understanding of professional, ethical and social responsibility			**X(I)**				
6	a recognition of the need for, and an ability to engage in life-long learning	**X**			**X**			**X(I)**
7	knowledge of and ability to apply discrete math, probability and statistics		**X(I)**			**X**		
8	the successful design, development, deployment and maintenance of a major software based project			**X**				
9	an ability to convey technical material through oral presentation and interaction with an audience	**X**			**X**			**X(I)**
10	an ability to convey technical material through written reports which satisfy accepted standards for writing style	**X**			**X**			**X(I)**
11	an ability to evaluate the impact of potential solutions to software engineering problems in a global society, using their knowledge of contemporary issues and emerging software engineering trends, models, tools, and techniques	**X**			**X**		**X(I)**	

Associate Degree Assessment Cycle

Table 3-2: Associate Outcome Assessment Timeline

#	**Learning Outcomes**	**10-11**	**11-12**	**12-13**
1	an ability to identify, formulate, and solve computer programming problems, including the specification, design, implementation, and testing of programs that meet specification, performance, maintenance and quality requirements			**X**
2	an understanding of the core areas of software engineering		**X**	
3	an understanding of professional, ethical and social responsibility			**X**
4	a recognition of the need for, and an ability to engage in life-long learning	**X**		
5	the successful design and development of a computer program		**X**	
6	an ability to communicate through oral presentation and interaction with an audience	**X**		
7	an ability to convey technical material through written reports which satisfy accepted standards for writing style	**X**		

IV. Summary of 2011-12 Assessment Activities

Though faculty planned to assess several outcomes in the program in the 2011-12 academic year, due to faculty load issues within the department it was decided to focus assessment efforts on the BS outcome #7 which was in alignment with the institutional assessment activities for 2011-12.

BS #7 Knowledge of and ability to apply discrete math, probability and statistics

Direct Assessment #1

The faculty assessed this outcome in MATH 327 Discrete Mathematics spring term 2012. The assessment consisted of a paper exploring various discrete mathematical theories scored by a rubric. Results for the 16 students who participated in this assessment are shown in Table 4-1.

Table 4-1: Direct Assessment BS #7, MATH 327, Spring 2012

Performance Criteria	Assessment Method	Measurement Scale	Minimum Acceptable Performance	Results
Concepts	Rubric-scored paper	1-4 proficiency scale	75% score 3 or 4	56%
Application	Rubric-scored paper	1-4 proficiency scale	75% score 3 or 4	75%
Evaluation	Rubric-scored paper	1-4 proficiency scale	75% score 3 or 4	50%

Though this assessment was administered in a course not taught by Software Engineering Technology faculty, papers were scored by faculty within the program. Upon review, program faculty felt student performance was as expected for a math course at this level. As shown in the higher result for the application criteria, faculty expect that student performance will improve once they have the opportunity to apply these theories within the context of the discipline.

Direct Assessment #2

The faculty assessed this outcome in CST 315 Embedded Sensor Interfacing and I/O and CST 417 Embedded Networking fall term 2011. This outcome was also assessed in CST 466 Embedded System Security spring term 2012. In each course students were given a problem related to a topic in the course involving mathematical reasoning related to probability, calculus or discrete mathematics. The problems were then graded with a rubric by the program director and results were reported for institutional assessment purposes. Students met faculty expectations on this assessment.

Indirect Assessment

The program faculty administered an exit survey to graduating seniors spring term 2012. Nineteen Software Engineering Technology students responded to the survey with 84% reporting that they were prepared by the program in knowledge of and ability to apply discrete math, probability and statistics. Program faculty are satisfied with the results of the indirect assessment of this outcome.

V. Summary of Student Learning Outcomes

BS #7 Knowledge of and ability to apply discrete math, probability and statistics

Program faculty are pleased with student's ability to apply discrete math, probability and statistics when related to a topic within the discipline. Result show appropriate growth for this outcome. No further actions are necessary at this time.

Appendix A
Course Mapping Matric

Table A-1: Bachelor Curriculum Course Mappings to Outcomes

BS SET Courses	1	2	3	4	5	6	7	8	9	10	11
CST 102 – Introduction to Computer Systems	●			●	●					●	
CST 116 – C++ Programming I	●	●	●					●		●	
CST 123 – Topics in Computer Science											
CST 126 – C++ Programming II	●	●	●					●		●	
CST 130 – Computer Organization			●								
CST 131 – Computer Architecture			●								
CST 136 – Object Oriented Programming with C++	●	●	●			●		●		●	
CST 162 – Introduction to Digital Logic			●								
CST 211 – Data Structures	●	●	●			●		●		●	
CST 213 – Programming Languages I			●								
CST 223 Programming Languages II			●					●			
CST 229 – Grammars			●				●	●			
CST 238 – GUI Programming	●	●	●			●		●	●	●	
CST 240 – Unix	●	●	●					●		●	
CST 250 – Assembly Language Programming	●	●	●					●		●	
CST 316 – Software Process Management	●	●		●		●		●	●	●	
CST 320 – Compiler Methods	●	●	●			●	●	●		●	
CST 324 – Database Systems and Design	●	●	●			●		●			
CST 326 – Software Design and Implementation I	●	●		●		●		●	●	●	
CST 328 – Computer Graphics			●			●					
CST 334 – Project Proposal	●	●						●		●	
CST 336 – Software Design and Implementation II	●	●		●		●		●	●	●	
CST 352 – Operating Systems	●		●			●					
CST 346 - .NET Programming in C#			●			●					
CST 412 – Senior Development Project	●	●				●		●	●	●	
CST 415 – Computer Networks	●		●			●					
CST 422 – Senior Development Project	●	●				●		●	●	●	
CST 432 – Senior Development Project	●	●				●		●	●	●	
CST 465 – Web development with ASP.NET	●	●	●			●		●			
MATH 111 – College Algebra								●			
MATH 112 –Trigonometry								●			

BS SET Courses	1	2	3	4	5	6	7	8	9	10	11
MATH 251 – Differential Calculus								●			
MATH 252 – Integral Calculus								●			
MATH 254N – Vector Calculus I								●			
MATH 327 – Discrete Mathematics							●	●			
MATH 465 - Mathematical Statistics							●	●			
PHY 221 – General Physics with Calculus							●	●			
PHY 222 – General Physics with Calculus							●	●			
PHY 223 – General Physics with Calculus							●	●			
WRI 121 – English Composition										●	
WRI 122 – English Composition										●	
WRI 227 – Technical Report Writing										●	
WRI 350 – Documentation Development										●	
SPE 111 – Fundamentals of Speech									●		
SPE 321 – Small Group and Team Communication									●		
PSY 201 – Psychology				●							●
PSY 347 – Organizational Behavior					●						
BUS 304 – Engineering Management				●	●					●	●
IMGT 345 – Industrial Economics							●				