2018-2019 SET Assessment Report

1 Program Mission and Educational Objectives

The mission of the Software Engineering Technology (SET) Bachelor's Degree Program within 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 graduates, our employers and our Industrial Advisory Board. Major components of the SET Program's mission in the CSET Department are:

1. To educate a new generation of Software Engineering Technology students to meet current and future industrial challenges and emerging software trends;
2. To promote a sense of scholarship, leadership and professional service among our graduates;
3. To enable our students to create, develop, apply and disseminate knowledge within the field of software engineering;
4. To expose our students to cross-disciplinary educational programs;
5. To provide employers with graduates in software engineering and related professions.

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

1. Use their knowledge of engineering to creatively and innovatively solve difficult computer systems problems;
2. Regularly engage in exploring, learning and applying state-of-the-art hardware and software technologies to the solution of computer systems problems;
3. Will be an effective team member that contributes to innovative software design solutions to the resolution of real world problems;
4. Will communicate effectively both as an individual and within multi-disciplinary teams.

2 Program Description and History

The Software Engineering Technology (SET) program was implemented in Klamath Falls in 1984 and was initially accredited by TAC of ABET in 1991. The Portland program was established in Fall 1996 under the same accreditation and is currently located on the Wilsonville campus.

Enrollment

Campus	**Fall 2014**	**Fall 2015**	**Fall 2016**	**Fall 2017**	**Fall 2018**
Klamath Falls	173	177	147	157	159
Wilsonville	116	128	136	116	111
Totals	289	305	283	273	270

Employment

Employed full time	89
Continuing education	1
Looking for employment	8
Not looking for employment	2
Median Salary	$67,000

3 Program Student Learning Outcomes

Our Program Student Learning Outcomes are taken from ABET ETAC. We did not change these this year.

Software Engineering Technology baccalaureate graduates will have demonstrated:

A. an ability to select and apply the knowledge, techniques, skills, and modern tools of the discipline to broadly-defined engineering technology activities;

B. an ability to select and apply a knowledge of mathematics, science, engineering, and technology to engineering technology problems that require the application of principles and applied procedures or methodologies;

C. an ability to conduct standard tests and measurements; to conduct, analyze, and interpret experiments; and to apply experimental results to improve processes;

D. an ability to design systems, components, or processes for broadly-defined engineering technology problems appropriate to program educational objectives;

E. an ability to function effectively as a member or leader on a technical team;

F. an ability to identify, analyze, and solve broadly-defined engineering technology problems;

G. an ability to apply written, oral, and graphical communication in both technical and non-technical environments; and an ability to identify and use appropriate technical literature;

H. an understanding of the need for and an ability to engage in self-directed continuing professional development;

I. an understanding of and a commitment to address professional and ethical responsibilities including a respect for diversity;

J. a knowledge of the impact of engineering technology solutions in a societal and global context; and

K. a commitment to quality, timeliness, and continuous improvement.

4 Curriculum Map

The Bachelor of Science in Software Engineering Technology degree requires 187 credit hours as prescribed by the curriculum outline.

Curriculum

Required courses and recommended terms during which they should be taken:

Freshman Year Fall

- CST 116 - C++ Programming I Credit Hours: 4
- CST 162 - Digital Logic I Credit Hours: 4
- MATH 111 - College Algebra Credit Hours: 4
- WRI 121 - English Composition Credit Hours: 3

Total: 15 Credit Hours

Winter

- CST 126 - C++ Programming II Credit Hours: 4
- CST 130 - Computer Organization Credit Hours: 3
- MATH 112 - Trigonometry Credit Hours: 4
- SPE 111 - Public Speaking Credit Hours: 3
- WRI 122 - Argumentative Writing Credit Hours: 3

Total: 17 Credit Hours

Spring

- CST 120 - Embedded C Credit Hours: 4
- CST 131 - Computer Architecture Credit Hours: 3
- CST 136 - Object-Oriented Programming with C++ Credit Hours: 4
- MATH 251 - Differential Calculus Credit Hours: 4

Total: 15 Credit Hours

Sophomore Year Fall

- CST 250 - Computer Assembly Language Credit Hours: 4
- CST 276 - Software Design Patterns Credit Hours: 4
- MATH 252 - Integral Calculus Credit Hours: 4
- WRI 227 - Technical Report Writing Credit Hours: 3

Total: 15 Credit Hours

Winter

- CST 211 - Data Structures Credit Hours: 4
- CST 240 - Linux Programming Credit Hours: 4
- MATH 254 - Vector Calculus I Credit Hours: 4
- PSY 201 - Psychology Credit Hours: 3

Total: 15 Credit Hours

Spring

- CST 223 - Concepts of Programming Languages Credit Hours: 3
- CST 236 - Engineering for Quality Software Credit Hours: 4
- CST 238 - Graphical User Interface Programming Credit Hours: 4
- MATH 327 - Discrete Mathematics Credit Hours: 4

Total: 15 Credit Hours

Junior Year Fall

- CST 229 - Introduction to Grammars Credit Hours: 3
- CST 316 - Junior Team-Based Project Development I Credit Hours: 4
- CST 324 - Database Systems and Design Credit Hours: 4
- PHY 221 - General Physics with Calculus Credit Hours: 4
- SPE 321 - Small Group and Team Communication Credit Hours: 3

Total: 18 Credit Hours

Winter

- CST 320 - Compiler Methods Credit Hours: 4
- CST 326 - Junior Team-Based Project Development II Credit Hours: 4
- PHY 222 - General Physics with Calculus Credit Hours: 4
- WRI 350 - Documentation Development Credit Hours: 3

Total: 15 Credit Hours

Spring

- CST 334 - Project Proposal Credit Hours: 1
- CST 336 - Junior Team-Based Project Development III Credit Hours: 4
- CST 352 - Operating Systems Credit Hours: 4
- PHY 223 - General Physics with Calculus Credit Hours: 4
- Social Science Elective Credit Hours: 3

Total16 Credit Hours

Senior Year Fall

- BUS 304 - Engineering Management Credit Hours: 3
- CST 412 - Senior Development Project Credit Hours:
- CST 415 - Computer Networks Credit Hours: 4
- Humanities Elective Credit Hours: 3
- Technical Elective Credit Hours: 3 [a]

Total: 16 Credit Hours

Winter

- CST 422 - Senior Development Project Credit Hours: 3
- MATH 465 - Mathematical Statistics Credit Hours: 4
- Humanities Elective Credit Hours: 3
- Social Science Elective Credit Hours: 3
- Technical Elective Credit Hours: 3 [a]

Total: 16 Credit Hours

Spring

- ANTH 452 - Globalization Credit Hours: 3
- CST 432 - Senior Development Project Credit Hours: 2
- MGT 345 - Engineering Economy Credit Hours: 3
- Humanities Elective Credit Hours: 3
- Technical Elective Credit Hours: 3 [a]

Total: 14 Credit Hours

Total for a B.S. in Software Engineering Technology: 187 Credit Hours

[a] Three additional CST upper division courses. One CST upper division elective course may be exchanged for an upper division MATH course

Mapping of courses to PSLOs

Course	Title	ESLO										
		A	B	C	D	E	F	G	H	I	J	K
ANTH 452	Globalization									X	X	
BUS 304	Engineering Management										X	
CST 116	C++ Programming I	X										
CST 120	Embedded C	X										

CST 126	C++ Programming II	X										
CST 130	Computer Organization	X										
CST 131	Computer Architecture	X										
CST 136	Object-Oriented Programming with C++	X										
CST 162	Digital Logic I	X										
CST 211	Data Structures	X		X								
CST 223	Concepts of Programming Languages	X										
CST 229	Introduction to Grammars		X									
CST 236	Engineering for Quality Software	X		X	X							
CST 238	Graphical User Interface programming	X										
CST 240	Linux Programming	X							X			
CST 250	Computer Assembly Language	X										
CST 276	Software Design Patterns	X										
CST 316	Junior Team-Based Project Development I	X				X	X	X	X	X		X
CST 320	Compiler Methods	X										
CST 324	Database Systems and Design	X										
CST 326	Junior Team-Based Project Development II	X				X	X	X	X	X		X
CST 334	Project Proposal				X		X	X	X			X
CST 336	Junior Team-Based Project Development III	X				X	X	X	X	X		X
CST 352	Operating Systems	X										
CST 412	Senior Development Project	X					X	X	X			X
CST 415	Computer Networks	X										
CST 422	Senior Development Project						X	X	X			X
CST 432	Senior Development Project						X	X	X			X
Humanities elective												
Humanities elective												
Humanities Elective												
MATH 111	College Algebra		X									
MATH 112	Trigonometry		X									
MATH 251	Differential Calculus		X									
MATH 252	Integral Calculus		X									
MATH 254N	Vector Calculus I		X									

MATH 327	Discrete Mathematics		X									
MATH 465	Mathematical Statistics		X									
MGT 345	Engineering Economy											
PHY 221	General Physics with Calculus		X									
PHY 222	General Physics with Calculus		X									
PHY 223	General Physics with Calculus		X									
PSY 201	General Psychology											
Social Science elective												
Social Science elective												
SPE 111	Public Speaking					X		X				
SPE 321	Small Group and Team Communication					X		X				
Technical Elective												
Technical Elective												
Technical Elective												
Total 187												
WRI 121	English Composition							X				
WRI 122	Argumentative Writing							X				
WRI 227	Technical Report Writing							X				
WRI350	Documentation Development							X				

5 Assessment Cycle

PSLO	2018-2019	2019-2020	2020-2021
A. an ability to select and apply the knowledge, techniques, skills, and modern tools of the discipline to broadly-defined engineering technology activities;	X		
B. an ability to select and apply a knowledge of mathematics, science, engineering, and technology to engineering technology problems that require the application of principles and applied procedures or methodologies;		X	
C. an ability to conduct standard tests and measurements; to conduct, analyze, and interpret experiments; and to apply experimental results to improve processes;			X
D. an ability to design systems, components, or processes for broadly-defined engineering technology problems appropriate to program educational objectives;	X		
E. an ability to function effectively as a member or leader on a technical team;		X	
F. an ability to identify, analyze, and solve broadly-defined engineering technology problems;			X
G. an ability to apply written, oral, and graphical communication in both technical and non-technical environments; and an ability to identify and use appropriate technical literature;		X	
H. an understanding of the need for and an ability to engage in self-directed continuing professional development;		X	
I. an understanding of and a commitment to address professional and ethical responsibilities including a respect for diversity;	X		
J. a knowledge of the impact of engineering technology solutions in a societal and global context;			X
K. a commitment to quality, timeliness, and continuous improvement.	X		

6 Assessment Activities

6.1 PSLO A: an ability to select and apply the knowledge, techniques, skills, and modern tools of the discipline to broadly-defined engineering technology activities

6.1.1 Assessment activities:

1. Junior Project (CST 316-336); Evaluate documentation developed winter quarter
2. Senior Project (CST 412-432); Evaluate the code submitted as part of the final deliverable spring quarter
3. Indirect: An exit survey was given to graduating seniors. As part of the survey, students were asked to rate their proficiency on each of our PSLO's.

6.1.2 Rubric

The following rubric was used for both direct measurements.

Category: A	4 Highly Proficient	3 Proficient	2 Some Proficiency	1 Limited or no Proficiency	Score
Applies the knowledge, techniques, skills of Software Engineering Technology to broadly-defined engineering technology activities	Works independently to find and implement good solutions to technical problems	Can solve many technical problems, but their solutions are not always of highest quality	Has difficulty finding solutions to technical problems	Unable to solve many technical problems	
Selects modern tools of Software Engineering Technology broadly-defined engineering technology activities	Were able to identify and use appropriate tools on their own	Required assistance in choosing tools but were able to learn and use them on their own	Required some assistance in both choosing and learning tools.	Highly dependent on others for tool choice and use	

6.1.3 Klamath Falls Results

For Junior Project, 87% of students scored a 3 or above on both criteria. The other 13% scored a 2.

For Senior Project, 65% of students scored a 3 or above on the first criterion, and 70% scored 3 or above on the second criterion. For both criteria, there was at least one student that scored a 1.

Indirect: 96% of students reported that the were proficient or highly proficient on this PSLO

6.1.4 Portland-Metro Results

For Junior Project, 86% of students scored a 3 or above on both criteria. The other 14% scored a 2. This data is consistent with the Klamath Falls data.

For Senior Project, 76% or students scored a 3 or above on both criteria. For both criteria, there were two students who scored a 1. While the percent of students meeting the criteria was higher than in Klamath Falls, there were also more students who scored a 1.

6.1.5 Discussion

Both campuses had seniors scoring below juniors. The instructors from both campuses reported that the low scores had more to do with motivation than with technical ability. It is unclear whether this is simply a cohort problem. Given that the student populations are distinct, it seems unlikely that both locations would experience a cohort problem in the same year.

Another possible explanation for the drop in senior scores is that students get burnt out by senior year. This in turn could just be senioritis, or it could be an indication that there is something in our program that causes this.

This year's data isn't sufficient to determine the cause of the drop. We will monitor next years seniors to see if the problem repeats itself. If so, we will need to look for systemic problems in our program.

6.2 PSLO D: an ability to design systems, components, or processes for broadly-defined engineering technology problems appropriate to program educational objectives

6.2.1 Assessment activities:

1. Junior Project (CST 316-336); Evaluate design documentation
2. Senior Project (CST 412-432); Evaluate the Use Case, Object Model, and Dynamic Model documents
3. Indirect: An exit survey was given to graduating seniors. As part of the survey, students were asked to rate their proficiency on each of our PSLO's.

6.2.2 Rubric

The following rubric was used for both the direct measurements

Performance Criteria	**High Proficiency (4)**	**Proficiency (3)**	**Developing Proficiency (2)**	**Limited/No Proficiency (1)**
Identify critical elements of the design	Identified at least 85% of the critical design elements.	Identified at least 75% of the critical design elements.	Identified at least 60% of the critical design elements.	Identified less than 60% of the critical design elements.
Create a detailed design specification addressing each of the identified critical design elements	The document is sufficiently complete and clear so that another developer could pick it up and complete the project.	Some aspects of the document need additional clarification.	Major portions of the design are not sufficiently documented.	The design is poorly documented.
Generate a implementable solution for each of the identified critical design elements	Student has a reasonable chance of implementing the entire design within the project timeline with minimal changes to the design.	There are some aspects of the design that may need to be reworked or re-scoped for the project to be completed.	Project design requires significant rework in order to be implementable.	Project can't be implemented as designed.

6.2.3 Klamath Falls Results

For JP: 80% of students scored a 3 or better for the first and third criteria, but only 40% scored a 3 or better on the second criterion.

For SP: 70% or students scored a 3 or better for the first and third criteria, and 65% scored a 3 or better on the second criterion.

Indirect: 92% of students reported that they were either proficient or highly proficient at this PSLO

6.2.4 Portland-Metro Results

For JP: 86% of students scored a 3 or better on all criteria, with the majority of those scoring a 4.

For SP: 71% of students scored a 3 or better on the first criterion, but only 48% scored a three or better on the second two criteria. Several students scored a 1 on all three criteria.

6.2.5 Discussion

Similar to PSLO A, there is insufficient data to determine if the drop in scores for seniors is merely a cohort problem or if it is indicative of a systemic problem. We will monitor next year's seniors to see if we see a recurrence of this pattern.

6.3 PSLO I: an understanding of and a commitment to address professional and ethical responsibilities including a respect for diversity

6.3.1 Assessment Activities:

1. CST 238: evaluate student's commitment to globalization of code. Several questions were included in the final exam that attempted to measure how important the student thought globalization of a project was and what steps they would take to globalize a project. This is a measure of their respect for cultural and linguistic diversity of users.
2. CST 415: The students were presented with a variety of scenarios and asked if the actions taken in the scenario were ethical. Responses were both numeric (very ethical to very unethical) and by short answer in which they gave their reasons for why the actions were/were not ethical.
3. Indirect: An exit survey was given to graduating seniors. As part of the survey, students were asked to rate their proficiency on each of our PSLO's.

6.3.2 Results

For CST 238: 72% of students indicated that globalization was important to them, but only 53% showed a commitment to do something about it.

For CST 415: For most of the scenarios, 100% of the students were able to identify the ethical nature of the scenario. 83% of the responses showed that a logical process was followed in evaluating the scenarios, and 80% of the responses demonstrated that their evaluation (judgement) was based on sound reasoning.

Indirect: 92% of students reported that they were either proficient or highly proficient at this PSLO

6.4 PSLO K: a commitment to quality, timeliness, and continuous improvement

6.4.1 Assessment activities:

1. Junior Project (CST 316-336); Evaluate Gant charts or other scheduling documentation
2. Senior Project (CST 412-432); Evaluate Project Plans developed and maintained throughout the

year
3. Indirect: An exit survey was given to graduating seniors. As part of the survey, students were asked to rate their proficiency on each of our PSLO's.

6.4.2 Rubric

The following rubric was used for both the direct measurements:

Category: K	4 Highly Proficient	3 Proficient	2 Some Proficiency	1 Limited or no Proficiency	Score
a commitment to quality	Self motivated to only submit work of highest quality	Self motivated to only submit their best work, even if their best is not of highest quality	Student submits low quality work, but wants to improve	Doesn't seem bothered by submitting low quality work	
a commitment to timeliness	Consistently meets deadlines	Meets most deadlines and works hard even if they can't meet a specific deadline	Consistently misses deadlines but knows they need to do better	Consistently misses deadlines and isn't bothered by that.	

6.4.3 Klamath Falls Results

For JP: 82% of students scored a 3 or better on the first criterion, and 70% scored 3 or better on the second criterion.

For SP: 75% of students scored a 3 or better on the first criterion, and 70% scored a 3 or better on the second criterion.

Indirect: 96% of students reported that they were either proficient or high proficient at this PSLO

6.4.4 Portland-Metro Results

For JP: 81% of students scored a 3 or better on both criteria. There was a single 1 on the first criteria.

For SP: 76% of students scored a 3 or better on both criteria. Two students scored a 1 on both criteria.

6.4.5 Discussion

Both campuses align reasonably well for this PSLO. The Klamath Falls campus was a little weaker on the timeliness criteria. The last time we evaluated timeliness we noticed this weakness. Scores have improved, but we need to continue to focus on this.

7 Data-driven Action Plans: Changes Resulting from Assessment

7.1 PSLO A: an ability to select and apply the knowledge, techniques, skills, and modern tools of the discipline to broadly-defined engineering technology activities

Scores for seniors were significantly lower than for juniors. The instructor for senior project felt that the issue was not that the students weren't able to perform at a higher level, simply that they did not care to do so. During the current academic year, we will have conversations as a program to determine possible causes for this. Possibilities include (but are not limited to)

1. Our program is difficult enough that students are burning out by the end of their senior year
2. Students have already found employment early enough in their senior year that they've lost some motivation to work hard at finishing their schooling.
3. This was a one-year blip the was reflective of the particular students in this cohort, but it does not reflect a problem in our program.

7.2 PSLO D: an ability to design systems, components, or processes for broadly-defined engineering technology problems appropriate to program educational objectives

The data showed a significant problem writing for our junior class. The previous iteration of assessing this PSLO did not show as significant a problem with writing. We evaluate this year's juniors to see if the problem is systemic or if it is a cohort problem. We will also look for other courses where we can give students an opportunity to write design specifications so they have practice before getting to junior project.

7.3 PSLO I: an understanding of and a commitment to address professional and ethical responsibilities including a respect for diversity

Based on the data we collected, students did a good job evaluating ethical situations. We will look at reformulating the assessment assignment to make it easier to evaluate using the Ethical Reasoning rubric.

Students showed an awareness of the need for reaching a global audience, but were less willing to invest the time up front to facilitate this. We will continue to emphasize the value of this in our GUI class.

7.4 PSLO K: a commitment to quality, timeliness, and continuous improvement

Based on the data we collected, students are doing OK on this outcome. However, particularly on the Klamath Falls campus, we need to continue to work on instilling in our students a commitment to timeliness.

8 Closing the Loop: Evidence of Improvement in Student Learning

8.1 2017-2018 Collect data for Criterion J: A knowledge of the impact of engineering technology solutions in a societal and global context.

The data we reported for Criterion I relates to this criterion as well. Students showed an awareness for reaching a global audience, but were less willing to invest time up front to facilitate this. Now that we have this data, we know we need to continue to emphasize the importance and value of reaching customers who are different from us.

8.2 Teach some sections of CST 116 in the Linux command line environment instead of Visual Studio

This was completed this year. The goal was to determine if removing some of the automatic help (syntax highlighting, auto-completion, etc.) would cause students to have to think for themselves instead of relying on the tool to do their thinking for them. We were hoping that this would improve students' problem solving skills and debugging skills. This is related to Criterion C.

There was some concern that students would have trouble adapting to the Linux command line environment instead of using the more familiar Windows environment. Our experiment revealed this to not be the case. Very little time was needed to get students sufficiently proficient in the Linux environment so that they could create, compile, and run programs.

There was also some concern as to whether the students who started in Linux would be able to adapt to Visual Studio for CST 126. In the Linux classes, we switched to Visual Studio for the last few weeks of the term. With that approach, the instructor in CST 126 reported no noticeable difference in the Visual Studio skills between those who started in Linux as compared with those who started in Visual Studio.

Anecdotal evidence suggests the Linux command line environment did help students to understand the difference between a source file and an executable and the process by which you turn one into the other. However, more work is needed to determine if the effect is big enough to switch all our intro classes to the Linux environment.

8.3 Teach some sections of CST 116 using C style IO instead of C++ style IO.

We had observed some confusion in students when they encountered file IO. They thought "`cout`" was a statement, so when they encountered file IO where there was an operation that was similar to what they did with `cout`, but didn't include that "statement", they were confused. We were hoping that a functional approach to IO instead of an operator overloading approach would relieve the confusion.

The classes that were done with the C style IO approach did seem to go better. However, another section done in parallel with the C++ approach, where additional time was spent trying to avoid the confusion also went well. As a result, we don't know if the different approach is any better than just a more careful treatment of the material.