

**Proposal for a New Academic Program**

**Institution:** Oregon State University - Cascades

**College/School:** College of Engineering

**Department/Program Name:** Electrical Engineering & Computer Science

**Degree and Program Title:** Bachelor of Science, Software Engineering

1. **Program Description**
2. Proposed Classification of Instructional Programs (CIP) number.

14.0903, Computer Software Engineering

1. Brief overview (1-2 paragraphs) of the proposed program, including its disciplinary foundations and connections; program objectives; programmatic focus; degree, certificate, minor, and concentrations offered.

People tend to equate computer science with “learning to code” or computer programming: the practice of writing software that solves problems in different domains. Indeed, students wishing to learn to write software arrive at an undergraduate institution to find that their intellectual interest fits closest with computer science, the only major with “computer” in its title. But the academic discipline of computer science is not about writing software.

Computer science is \_, . Other common degree programs include computer engineering, which emphasizes \_ , and management information systems, which is \_.

The academic discipline of computer science is broad. TODO.

The software engineering program differs from computer science in that it emphasizes the communication skills, methodologies, tools, technologies, teamwork, professional practice, design, and architectures critical for building scalable, long-lasting software systems. The program emphasizes these software engineering principles over, but not at the exclusion of, computer science principles. The curriculum includes high-impact, team- and project-based courses early in the curriculum, followed by computer science courses that explore deeper topics.

During year one, students engage in a team-based project in which they create a “full stack” application in which they

Incorporates computer science courses at the end of the curriculum.

Incorporates experiential learning and professional practice throughout all four years of the program.

Lifelong learning (Economist, 2017): How to study, think, metacognition. On the job training is shrinking, leaving more people responsible for their own skill building.

1. Course of study – proposed curriculum, including course numbers, titles, and credit hours.

The program of study follows the degree standards at Oregon State University, incorporating both existing courses and new courses. The “SE” prefix indicates new courses, which are also followed by the CAT II proposal numbers (99XXX).

|  |  |
| --- | --- |
| Freshman Year | Sophomore Year |
| Fall **15cr**  SE 101: Programming I (2) (99XXX)  SE 111: Software Development I (6) (99XXX)  SE 107: Professional Seminar (1) (99XXX)  WR 121#\*: English Composition (3)  HHS 231#\*: Lifetime Fitness for Health (2)  PAC XXX#\*: Various Physical Activity Courses (1) | Fall **15 cr**  SE 211: Data Science Engineering I (6) (99XXX)  SE 210: Apprenticeship II (1) (99XXX)  ST 351: Intro to Statistical Methods (4)  BACC Core#: Biology with Lab (4) |
| Winter **13 cr**  SE 102: Programming II (2) (99XXX)  SE 112: Software Development II (6) (99XXX)  SE 110: Apprenticeship I (1) (99XXX)  MTH 112#: Elementary Functions (4) | Winter **15 cr**  SE 212: Data Science Engineering II (6) (99XXX)  SE 210: Apprenticeship II (1)  ST 352: Intro to Statistical Methods (4)  BACC Core#: Physical Science with Lab (4) |
| Spring **15cr**  SE 103: Programming III (2) (99XXX)  SE 113: Software Development III (6) (99XXX)  SE 110: Apprenticeship I (1)  COMM 111#\*: Public Speaking (3)  WR 327#\*: Technical Writing (3) | Spring **15 cr**  SE 213: Data Science Engineering III (6) (99XXX)  SE 210: Apprenticeship II (1)  MTH 231: Elements of Discrete Mathematics (4)  BACC Core#: Bio or Physical Science with Lab (4) |
| Junior Year | Senior Year |
| Fall **16 cr**  SE 301: Elements of Computing Systems I (2) (99XXX)  SE 311: Scalability, Infrastructure and Security I (6) (99XXX)  SE 310: Apprenticeship III (1) (99XXX)  CS 261: Data Structures (4)  BACC Core#: Western Culture or American History (3) | Fall **16 cr**  SE 411: Business of Software I (4, WIC) (99XXX)  SE 410: Apprenticeship IV (1) (99XXX)  XX XXX: Upper-Division Elective (4)  CS 325: Analysis of Algorithms (4)  BACC Core#: Difference, Power & Discrimination (3) |
| Winter **16 cr**  SE 302: Elements of Computing Systems II (2) (99XXX)  SE 312: Scalability, Infrastructure and Security II (6) (99XXX)  SE 310: Apprenticeship III (1)  CS 271: Computer Architecture & Assembly Language (4)  BACC Core#: Cultural Diversity (3) | Winter **16 cr**  SE 412: Business of Software II (4, WIC) (99XXX)  SE 410: Apprenticeship IV (1)  XX XXX: Upper-Division Elective (4)  CS 344: Operating Systems I (4)  BACC Core#: Synthesis, Contemporary Global Issues (3) |
| Spring **16 cr**  SE 303: Elements of Computing Systems III (2) (99XXX)  SE 313: Scalability, Infrastructure and Security III (6) (99XXX)  SE 321: Apprenticeship III (1)  CS 381: Programming Language Fundamentals (4)  BACC Core#: Literature & Arts (3) | Spring **15 cr**  SE 413: Business of Software III (4, WIC) (99XXX)  SE 410: Apprenticeship IV (1)  CS 391#: Social and Ethical Issues in Computer Science (3) (Bacc Core Synthesis: Science, Technology, Society)  CS XXX: Upper-Division CS Elective (4)  BACC Core#: Social Processes & Institutions (3) |

**\*Freshman Skill Courses (16 credits) #BACC Core (48 cr) 99XXX- CAT II course proposal ~Change in Location only**

1. Manner in which the program will be delivered, including program location (if offered outside of the main campus), course scheduling, and the use of technology (for both on-campus and off-campus delivery).

The OSU-Cascades campus will deliver the Software Engineering program as a face-to-face, on-campus program in Bend, while housed within the OSU College of Engineering. Course scheduling

The first two years of the program include pre-professional courses unique to the program that, for the time being, must be taken at OSU-Cascades. The second two years are professional school courses that require admission to the College of Engineering professional school. Grade point average in select pre-professional courses dictates admission. The pre-professional school courses used for admittance into the professional school will follow the same model as those used by many of the other schools in the College of Engineering.

1. Adequacy and quality of faculty delivering the program.
2. Adequacy of faculty resources – full-time, part-time, adjunct.
3. Other staff.
4. Adequacy of facilities, library, and other resources.
5. Anticipated start date.
6. **Relationship to Mission and Goals**
7. Manner in which the proposed program supports the institution’s mission, signature areas of focus, and strategic priorities.

*As a land grant institution committed to teaching, research and outreach and engagement, Oregon State University promotes economic, social, cultural and environmental progress for the people of Oregon, the nation and the world* (OSU, 2017).

The Software Engineering program incorporates effective, research-supported pedagogy within a hands-on classroom and curriculum design; organizes opportunities for student and faculty outreach into the community of Central Oregon; and integrates economic, social, cultural and environmental issues through world-relevant coursework and professional practice.

*This mission is achieved by producing graduates competitive in the global economy, supporting a continuous search for new knowledge and solutions and maintaining a rigorous focus on academic excellence… (OSU, 2017)*

The Software Engineering program aspires to become a world-class educational experience that transforms its students into emerging leaders who can design and deliver large, scalable, high-performing, and maintainable software systems, which are in high demand across the global economy. The cross-disciplinary, project-based curriculum design, integration with professional practice, synthesis of computer science topics, and incorporation of continuously evolving technologies supports the achievement of the OSU mission.

*…particularly in the three Signature Areas: Advancing the Science of Sustainable Earth Ecosystems, Improving Human Health and Wellness, and Promoting Economic Growth and Social Progress (OSU, 2017).*

The Software Engineering program is inherently cross-disciplinary, and positions itself across OSU’s three Signature Areas. For example, students may engineer a software system that transmits, aggregates, mines and visualizes sensor data for river restoration; they may engage with a local medical device company to improve software reliability and security, or build applications that assist the disabled; and, they may integrate new systems, techniques and solutions that they bring to market through their own startup business, creating new jobs and economic growth.

Strategically, one priority for the OSU-Cascades campus is to distinguish itself with innovative, world-class, “destination degree programs” not offered at most institutions. The Software Engineering program exemplifies such a program, following the success of the existing Energy Systems Engineering program, and co-existing with other innovative new programs, such as Outdoor Products.

1. Manner in which the proposed program contributes to institutional and statewide goals for student access and diversity, quality learning, research, knowledge creation and innovation, and economic and cultural support of Oregon and its communities.

The Software Engineering program approaches student access and diversity as a continuum, beginning with K-12 outreach (Bush & Miller, 2017; Wilson et al, 2010) and bringing K-12 students onto campus to observe Software Engineering classes and students (Engle et al, 2006). The program ensures an approachable introductory course sequence for students with diverse backgrounds (Cohoon & Tychonievich, 2011), implements an authentic, hands-on, and world-relevant active learning experience (Eisenberg, 2017), cultivates a community of supportive peers through specific learning methods (Nagappan, 2013), and integrates professional exposure early in the curriculum, to support student career-readiness upon graduation.

The Software Engineering program has, as its advantage, the ability to implement effective teaching and learning techniques, and experiences for knowledge creation, based on decades of prior research in engineering and computer science education. Furthermore, the engagement of real-world problems, elicited from local, regional and statewide businesses directly contributes to innovation, economic and cultural support of Oregon.

1. Manner in which the program meets regional or statewide needs and enhances the state’s capacity to:
2. improve educational attainment in the region and state;
3. respond effectively to social, economic, and environmental challenges and opportunities; and
4. address civic and cultural demands of citizenship.

Central Oregon is an education desert or a “community where students have few postsecondary options from which they can choose.” Nationally, students travel less than 50 miles to attend a 4-year public university. OSU-Cascades is a critical player in the social and economic development of Central Oregon and cannot serve the needs of the students and employers of the region without further expansion of academic programs and degrees. In addition, Central Oregon Community College’s role cannot be understated in Central Oregon, where over 86% of all students in higher education are enrolled. The Software Engineering program capitalizes on the lower division preparation provided by COCC to remote communities to recruit diverse students. By working in partnership with COCC, we have created a pathway to the undergraduate degree that can be started at either institution.



“The zip code that a child is born into oftentimes determines their life chances.” (Hillman & Weichman, 2016)

1. **Accreditation**
2. Accrediting body or professional society that has established standards in the area in which the program lies, if applicable.

The Accreditation Board for Engineering and Technology (ABET) provides accreditation to engineering programs in the United States. The Computing Sciences Accreditation Board (CSAB) leads the ABET Engineering Accreditation Commission. Members include the Association of Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers (IEEE).

1. Ability of the program to meet professional accreditation standards. If the program does not or cannot meet those standards, the proposal should identify the area(s) in which it is deficient and indicate steps needed to qualify the program for accreditation and date by which it would be expected to be fully accredited.

The Software Engineering program outcomes and quality assessment (Section 5) specifically include the outcomes and standards set forth by the Criteria for Accrediting Engineering Programs (ABET, 2017b). The courses and course learning outcomes cover the “breadth and depth of engineering and computer science topics” in the Program Criteria for Software and Similarly Named Engineering Programs (ABET, 2017b). Identical to the accreditation criteria, the Software Engineering curriculum includes 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.

1. If the proposed program is a graduate program in which the institution offers an undergraduate program, proposal should identify whether or not the undergraduate program is accredited and, if not, what would be required to qualify it for accreditation.

N/A

1. If accreditation is a goal, the proposal should identify the steps being taken to achieve accreditation. If the program is not seeking accreditation, the proposal should indicate why it is not.

Following ABET guidelines, we will record data for the required Self-Study Report during the first three years of the Software Engineering program. We will submit a Request for Evaluation in the third year of the program, and plan to obtain accreditation by the program’s fourth or fifth year of operation (ABET, 2017a).

1. **Need**
2. Anticipated fall term headcount and FTE enrollment over each of the next five years.
3. Expected degrees/certificates produced over the next five years.
4. Characteristics of students to be served (resident/nonresident/international; traditional/ nontraditional; full-time/part-time, etc.).
5. Evidence of market demand.
6. If the program’s location is shared with another similar Oregon public university program, the proposal should provide externally validated evidence of need (e.g., surveys, focus groups, documented requests, occupational/employment statistics and forecasts).
7. Estimate the prospects for success of program graduates (employment or graduate school) and consideration of licensure, if appropriate. What are the expected career paths for students in this program?
8. **Outcomes and Quality Assessment**
9. Expected learning outcomes of the program.
10. Methods by which the learning outcomes will be assessed and used to improve curriculum and instruction.
11. Nature and level of research and/or scholarly work expected of program faculty; indicators of success in those areas.
12. **Program Integration and Collaboration**
13. Closely related programs in this or other Oregon colleges and universities.
14. Ways in which the program complements other similar programs in other Oregon institutions and other related programs at this institution. Proposal should identify the potential for collaboration.
15. If applicable, proposal should state why this program may not be collaborating with existing similar programs.
16. Potential impacts on other programs.

**7. External Review**

If the proposed program is a graduate level program, follow the guidelines provided in *External Review of New Graduate Level Academic Programs* in addition to completing all of the above information.

*Revised May 2016*

**References**

ABET (2017a). Accreditation Policy and Procedure Manual (APPM), 2017 – 2018. Retrieved from<http://www.abet.org/accreditation/accreditation-criteria/accreditation-policy-and-procedure-manual-appm-2017-2018/>

ABET (2017b). Criteria for Accrediting Engineering Programs, 2017 – 2018. Retrieved from<http://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-engineering-programs-2017-2018/>

Bush, J. & Miller, S. (2017). Analysis of Associations between Motivation and Previous Computer Science Experience, Gender, Ethnicity and Privilege as Observed in a Large Scale Survey of Middle School Students. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (SIGCSE '17). ACM, New York, NY, USA, 705-705. doi:<https://doi.org/10.1145/3017680.3022441>

Cohoon, J. & Tychonievich, L. (2011). Analysis of a CS1 approach for attracting diverse and inexperienced students to computing majors. In *Proceedings of the 42nd ACM technical symposium on Computer science education* (SIGCSE '11). ACM, New York, NY, USA, 165-170. doi:<http://dx.doi.org/10.1145/1953163.1953217>

Eisenberg M. (2017) Approaching Computer Science Education Through Making. In: Fee S., Holland-Minkley A., Lombardi T. (eds) *New Directions for Computing Education*. Springer, Cham. doi:<https://doi.org/10.1007/978-3-319-54226-3_3>

Engle, J., Bermeo, A., O’Brien, C. (2006). Straight from the Source: What Works for First-Generation College Students. Washington, DC: Pell Institute for the Study of Opportunity in Higher Education. Retrieved from<http://www.pellinstitute.org/downloads/publications-Straight_from_the_Source.pdf>

Hillman, N. & Weichman, T. (2016). Education Deserts: The Continued Significance of “Place” in the Twenty-First Century. *Viewpoints: Voices from the Field*. Washington, DC: American Council on Education. Retrieved from<http://www.acenet.edu/news-room/Documents/Education-Deserts-The-Continued-Significance-of-Place-in-the-Twenty-First-Century.pdf>

Nagappan, N., Williams, L., Ferzli, M., Wiebe, E., Yang, K., Miller, C., & Balik, S. (2003). Improving the CS1 experience with pair programming. In *Proceedings of the 34th SIGCSE technical symposium on Computer science education* (SIGCSE '03). ACM, New York, NY, USA, 359-362. DOI=http://dx.doi.org/10.1145/611892.612006

Oregon State University (2017). Oregon State University Mission Statement. Retrieved from<http://leadership.oregonstate.edu/trustees/oregon-state-university-mission-statement>

Wilson, C., Sudol, L. A., Stephenson, C. & Stehlik, M. 2010. Running on Empty: The Failure to Teach K-12 Computer Science in the Digital Age. The Association for Computing Machinery & The Computer Science Teachers Association. Retrieved from http://runningonempty.acm.org/fullreport2.pdf