THE UNIVERSITY OF ARIZONA

NEW ACADEMIC PROGRAM - MAJOR
Preliminary Proposal Form

I. Program Details

a. **Name (and Degree Type) of Proposed Academic Program:**

BS Software Engineering

b. **Academic Unit(s)/College(s):**

College of Engineering / Electrical and Computer Engineering Dept and
College of Engineering / Systems and Industrial Engineering Dept

c. **Campus/Location:** Main Campus

d. **Primary Instructional Modality:** In Person

e. **First Admission Term:** Fall 2021

II. Brief Program Description:

The Bachelor of Science in Software Engineering synergistically integrates proven engineering techniques and discipline with software development best practices that encompass all aspects of the software development lifecycle (SDLC). The curriculum includes core principles from systems engineering, electrical and computer engineering, and software engineering. The curriculum is based on a solid foundation of mathematics, including calculus, physics, and discrete math. The courses include topics related to software requirements analysis, design, code, integration, verification testing, and software project management.

The Software Engineering curriculum is designed to prepare students to meet the ever-growing demands within the commercial, industrial, and federal government job sectors. Relevant software methodologies, such as Agile development, automated testing using continuous integration, and SW DevOps to increase the velocity of software application and service delivery, are also integral to the curriculum. Using these types of agile and adaptive approaches, students will be well suited for the many diverse opportunities in a rapidly growing and ever evolving career in software engineering.

The program is built on a firm engineering foundation, discovery-based education and 'learn-by-doing' approach. Students complete projects in every semester of the program to provide emphasis in communication, teamwork, critical thinking,

and professionalism. Students have flexibility in designing their course of study and can select technical electives from a pool of courses in different software and computer engineering domains such as web and mobile applications, embedded systems, and other interdisciplinary areas.

Software engineering students will acquire multiple software development skills including: experience with multiple programming languages, data structure constructs, algorithm implementation, databases, operating systems, networking, embedded systems, cloud computing, configuration management, software assurance and the use of open source software libraries. Additionally, students in the Software Engineering degree program develop skills required to work with other engineering disciplines to design and develop creative software solutions that are integral to a variety of large-scale, complex systems. Software systems are typically multi-faceted and often include millions of lines of code with diverse origins or pedigrees. Graduates of the bachelor's degree program in Software Engineering learn to use structured and well-defined engineering approaches to develop, evaluate and maintain software centric systems.

III. Projected Enrollment for the First Three Years

The table below summarizes the full-time projected enrollments in the Software Engineering degree program extrapolated out over the first 5 years, after which enrollment is anticipated to remain at a steady state . These numbers were estimated based on actual enrollments in similar software engineering degree programs as reported by the American Society for Engineering Education (ASEE).

Year 1 (2021 / 2022)	Year 2 (2022 / 2023)	Year 3 (2023 / 2024)	Year 4 (2024 / 2025)	Year 5 (2025 / 2026)
60	**120**	**220**	**350**	**350**

Table III.1 Projected Enrollments Over 5 Years

As evidence of student interest in this degree program, enrollment statistics over the years 2016 – 2019 were evaluated at other universities across the country. Table III.2 contains a summary of the full-time student enrollments in Software Engineering at the universities shown, as reported by ASEE. *It should be noted that these are only a subset of total enrollments reported by ASEE in software engineering programs in North America, including Canada and Puerto Rico.*

University	Degree Name	2016 Full Time	2017 Full Time	2018 Full Time	2019 Full Time
Arizona State University	Software Engineering (B.S.)	**528**	**512**	**552**	**563**
Auburn University	Software Engineering (BSWE)	**304**	**374**	**376**	**356**
Brigham Young University - Idaho	Software Engineering	**187**	**254**	**302**	**244**
California Polytechnic State University, San Luis Obispo	Software Engineering (B.S.)	**245**	**254**	**253**	**253**
Iowa State University	Software Engineering (B.S.)	**536**	**348**	**374**	**393**
Mississippi State University	Software Engineering (B.S.)	**129**	**124**	**127**	**145**
San Jose State University	Software Engineering (B.S.)	**203**	**500**	**525**	**518**
The University of Texas at Arlington	Software Engineering (B.S.)	**145**	**151**	**182**	**216**
The University of Texas at Dallas	Software Engineering (BSSE)	**281**	**315**	**397**	**392**
University of California, Irvine	Software Engineering (B.S.)	**189**	**176**	**190**	**194**
University of Nebraska, Lincoln	Software Engineering (BSSE)	**38**	**95**	**162**	**181**
University of Wisconsin, Platteville	Software Engineering (B.S.)	**198**	**184**	**217**	**194**

Table III.2 Subset of Recorded ASEE Enrollments in Software Engineering Degree Programs at Comparable State Universities

IV. Evidence of Market Demand

The Bureau of Labor Statistics reports the total number of Software Engineering jobs in 2010 to be 1.85M, and projects to over 2.4M in 2020; a 23% increase over 10 years. U.S New & World Report ranked software developer/engineer as the best technology job in America.

In the past 12 months, Burning Glass reports that there was a total of 43,563 job postings for *Computer Software Engineering* (CIP = 14.0903) from 08/01/19 – 07/31/2020. As shown in Table IV.1 below, the number of jobs is expected to have a relatively high growth rate over the next 10 years.

GROWTH BY GEOGRAPHY

Geography	Selected Occupations	Total Labor Market	Relative Growth
Nationwide	15.46 %	4.24 %	High

Table IV.1. National Growth Projected in Software Engineering

The career outlook for software engineers has continued an upward trajectory for the past 6 years, as depicted in the graph below:

	2014	2015	2016	2017	2018	2019
Employment (BLS)	31,420	34,020	35,160	38,610	38,280	38,792

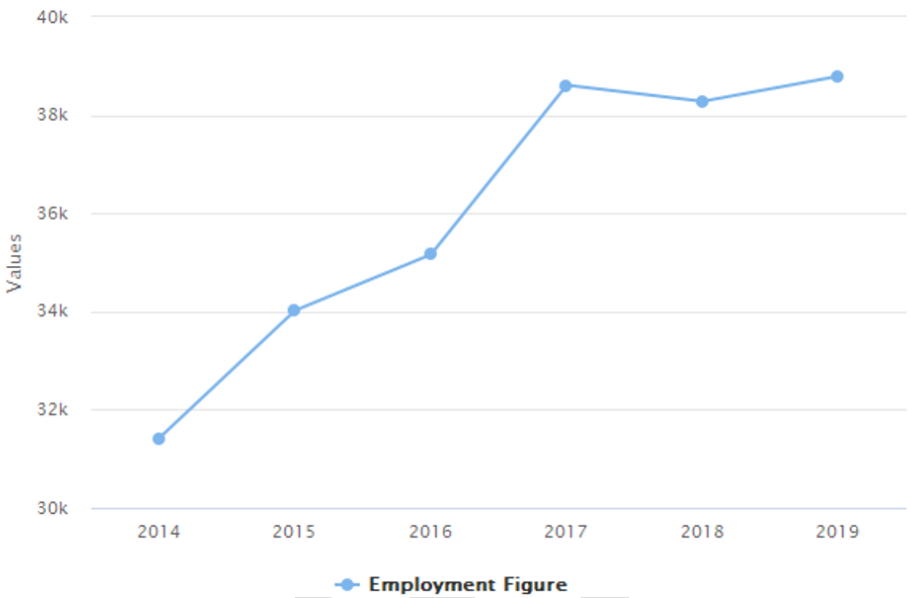

Figure IV.2. Career Outlook for Software Engineering over the Past 6 Years

Nationwide, the top locations posting requisitions for Software Engineers is shown in the table below:

Location	Postings
California	309,439
Texas	173,408
Virginia	114,302
New York	98,786
Florida	80,669
North Carolina	80,604
Illinois	68,747
Massachusetts	63,683
Georgia	62,505
Colorado	59,274

Table IV.3. Software Engineering Posting by State

Within the state of Arizona, the top employers hiring software engineers include the following:

TOP EMPLOYERS HIRING

Experience Level: All Experience

Employer	Postings	Market Share (%)
Raytheon	931	4.00%
USAA	765	3.29%
Wells Fargo	564	2.42%
General Dynamics	310	1.33%
Honeywell	304	1.31%
Northrop Grumman	299	1.28%
American Express	297	1.28%
Deloitte	288	1.24%
The Boeing Company	176	0.76%
Mitsubishi	169	0.73%
Anthem Blue Cross	154	0.66%
CVS Health	145	0.62%
IBM	132	0.57%
Viasat	126	0.54%
Amazon	120	0.52%

Table IV.4. Arizona Top Employers with Software Engineering Needs

The top 15 specialized software skills required by employers include a variety of programming languages, software engineering, Linux operating system, software project management, software DevOps, Scrum (a particular form of Agile), quality assurance, and systems engineering. The number of job postings and the anticipated growth in that particular skill category specifically requested in these different software engineering skills is summarized in Table IV.5, as shown below:

Skill	Postings	Projected Growth	Salary Premium	Competitive Advantage
Java	14112 (32%)	-13.18%	No	No
Software Development	11197 (26%)	5.78%	No	No
SQL	10788 (25%)	-13.3%	No	No
Software Engineering	7641 (17%)	7.27%	No	No
JavaScript	7611 (17%)	6.81%	No	No
Python	6560 (15%)	61.12%	No	No
Linux	5650 (13%)	-12.57%	No	No
Microsoft C#	5130 (12%)	-25.69%	No	No
DevOps	5048 (12%)	107.85%	Yes	No
Project Management	4706 (11%)	-19.74%	No	No
Oracle	4620 (11%)	-16.26%	No	No
Quality Assurance	4600 (11%)	39.46%	No	No
Git	4239 (10%)	59.81%	No	No
Scrum	4156 (10%)	39.96%	No	No
Systems Engineering	3740 (9%)	-4.85%	No	No

Table IV.5. Postings for the Top 15 Software Engineering Specialized Skills

The top 15 software engineering skill clusters are shown in the table below:

Skill	Postings
Software Development Principles	19331 (44%)
Java	14291 (33%)
System Design and Implementation	13878 (32%)
SQL Databases and Programming	12479 (29%)
JavaScript and jQuery	10579 (24%)
Software Quality Assurance	8917 (20%)
Operating Systems	8518 (19%)
Scripting Languages	7644 (17%)
Web Development	7380 (17%)
Software Development Methodologies	7316 (17%)
Business Process and Analysis	7146 (16%)
Database Administration	7136 (16%)
Microsoft Development Tools	7056 (16%)
Cybersecurity	6963 (16%)
Programming Principles	6954 (16%)

Table IV.6. Postings for the Top 15 Software Engineering Skill Clusters

V. Similar Programs Offered at Arizona Public Universities

There are a total of 37 ABET accredited bachelor's level programs in Software Engineering, with 30 of these located in the United States. In Arizona, Arizona State University (ASU) has both an ABET accredited On-campus and an Online Software Engineering Bachelor's degree program. In 2019, ASU had 1,322 students enrolled in their program, of which, ASEE classified 563 as full-time

degree seeking students. In 2019, 143 BS Software Engineering degrees were conferred at ASU.

Embry-Riddle Aeronautical University in Prescott, AZ had 63 full-time students in their Software Engineering program in 2019, and conferred 6 students with Software Engineering degrees.

The University of Arizona (UArizona) also offers a BS in Computer Science degree located in the College of Science. In 2019, there were 1626 students enrolled in the Computer Science program. While there are some similarities in the curriculum between a Computer Science program and a Software Engineering degree program, there are also major differences. Software engineers are trained in all aspects of the software development lifecycle (SDLC) which includes requirements specification and analysis, software architecture / design, test planning and integration, verification / validation, and maintenance/support. The curriculum also includes key attributes of the overall engineering discipline that includes mathematics through calculus, physics, chemistry, product design, configuration management, quality assurance, safety, reliability, as well as the cost, schedule, and delivery pipeline of software. Software engineers develop complex software that is often part of a large scale, complex and multi-disciplinary engineering product / system.

Other synergistic degree programs at UArizona include those offered by the School of Information (iSchool), Eller College of Management, and the College of Applied Science & Technology (CAST). The iSchool offers two related BS degrees in Information Science and Technology, and Game Design and Development. Eller College of Management offers a BS in Management Information Systems. CAST offers BS degrees in Applied Computing, Intelligence and Information Operations, and Computer Science. While there may be some classes within these programs that could potentially be electives within the Software Engineering degree program, there are significant differences between the curriculum and skillsets developed in all these programs.

VI. New Resources Required

a. ***Library Acquisitions –***

There are no anticipated additional library acquisition needs with the Software Engineering degree program.

b. ***Equipment/Physical Facilities –***

At this time, there are no additional equipment / physical facility needs identified. Many of the tools and lab facilities used by this degree program already exist for other courses that are offered within the ECE and SIE departments. Many of the software development tools traditionally used for software engineering courses are “open source” and readily available

for faculty and students to use at little to no charge. If there are licenses that become necessary for different software engineering courses in the future, those costs could be included in course fees.

c. ***Additional Faculty/Staff***

The additional faculty and staff for the Software Engineering degree program has been broken down into multiple categories, as is depicted in Table VI.1. As shown, the needs have been extrapolated out over 5 years. It is currently envisioned that most of the curriculum for the first year will be closely aligned to many of the other engineering degree programs. As such, for the first year of the degree offering, new faculty can focus on developing the curriculum for the 2nd and 3rd years of the program. However, because a few of the courses that are part of the new curriculum are already very heavily populated courses, such as ECE 175 and Engr 102, additional faculty, TA and lab assistant resources will be required to help mitigate the increased demand.

Resource Type	Year 1	Year 2	Year 3	Year 4	Year 5	Total
Tenured Track Faculty	**2**	**2**	**2**	**2**	**2**	***10***
Professor of Practice	**1**	**1**	**1**	**0**	**0**	***3***
Adjunct	**1**	**1**	**0**	**0**	**0**	***2***
Teaching Assistants	**4**	**4**	**6**	**6**	**6**	***26***
UG Advisor	**1**	**0**	**0**	**0**	**0**	***1***
Other Staff (IT etc)	**0**	**1**	**0**	**0**	**0**	***1***
Lab Assistants	**4**	**6**	**8**	**8**	**8**	***34***

Table VI.1. Projected Additional Faculty / Staff Resources

d. ***New Courses to be Created for the Degree*** –

It is currently anticipated that 7 new "required courses" will be developed for this degree. There are several existing Math, Physics, ECE, and SIE courses that will be used to construct the curriculum. There are also several existing SIE, ECE, ESOC, ISTA, and CS courses that could be utilized as Software Engineering degree electives. Furthermore, it is anticipated that future faculty will likely develop new courses aligned with their own research interests and offer them as additional software

engineering electives.

e. ***How New Resources Will Be Funded –***

New resources will be funded through Responsible Center Management (RCM) funds that are allocated by the College of Engineering.

THE UNIVERSITY OF ARIZONA

New Academic Program Workflow Form

General

Proposed Name: Software Engineering

Transaction Nbr: 00000000000086

Plan Type: Major

Academic Career: Undergraduate

Degree Offered: Bachelor of Science

Do you want to offer a minor? Y

Anticipated 1st Admission Term: Fall 2021

Details

Department(s):

ENGR

DEPTMNT ID	DEPARTMENT NAME	HOST
2302	Systems & industrial Engineering	N
2303	Electrical & Computer Engr	Y

Campus(es):

MAIN

LOCATION	DESCRIPTION
TUCSON	Tucson

Admission application terms for this plan: Spring: Y Summer: Y Fall: Y

Plan admission types:

Freshman: Y Transfer: Y Readmit: Y Graduate: N

Non Degree Certificate (UCRT only): N

Other (For Community Campus specifics): N

Plan Taxonomy: 14.0903, Computer Software Engineering.

Program Length Type: Program Length Value: 0.00

Report as NSC Program:

SULA Special Program:

Print Option:

Diploma: Y Bachelor of Science Software Engineering

Transcript: Y Bachelor of Science Software Engineering

Conditions for Admission/Declaration for this Major:

All students are enrolled as Engineering, No Major Selected until they have completed the following:
- Calculus I with a grade of C or better
- 12 or more UA credits of coursework within the Engineering curriculum (shown above)
- Admissions GPA of 2.0 or higher

Requirements for Accreditation:

Accreditation Board for Engineering and Technology, ABET

Program Comparisons

University Appropriateness

The University of Arizona College of Engineering currently does not offer an engineering degree related explicitly to Software Engineering. UA engineering students that have historically leaned more toward software engineering careers have typically obtained Electrical and Computer Engineering degrees, with an emphasis towards Computer Engineering, and take various computer programming courses as electives. Alternatively, these students obtain a Computer Science degree that lead to future software career opportunities but lack the specific engineering discipline and emphasis offered with the Software Engineering degree curriculum. Since Software Engineering is growing as a `in high-demand¿ engineering degree (as shown by the Burning Glass analysis), it is likely that students are selecting other universities since UArizona does not offer Software Engineering. It is believed that offering an innovative Software Engineering degree will attract new students. As we have seen over the past decade, software has become an integral element/component within the systems, products, and technologies that are part of the 4th industrial revolution.

Given the ever-increasing demand in the industry for software engineers, there is an equally increasing opportunity for the CoE to diversify their degree offerings by offering this new degree. This will attract more students to the University of Arizona.

Additionally, two of the colleges¿ strategic pillars are:

1) Driving student success for a rapidly changing world, and
2) Tackling critical problems at the edges of human endeavor.

The new Software Engineering degree plays a critical role in both pillars. The students graduating with the degree in Software Engineering will be better positioned to develop the skills and mindsets to be leaders in the areas of space, natural and built environments, ever-increasing automation and connectivity, human and intelligent systems, data, computing, and network sciences.

By offering a competitive, relevant, and experiential-based learning Software Engineering program to prospective students, it increases not only the net enrollment in the college, but also the ability to grow research programs that are attractive to forthcoming undergraduate and graduate students. All of which contribute to higher recruitment numbers and bringing additional revenue to the College and University. Hence, we can recruit the faculty that can significantly impact their areas of research and education. These faculty will pursue research grants to advance the state-of-the-art in software engineering and integrate their research into the curriculum. The broader impact of these faculty will ultimately drive the program's national ranking higher.

While less obvious, another goal for the program is to increase the number of female and other underrepresented students in the College of Engineering by offering the Software Engineering degree. Additional features and programs that contribute to enhancing student success and increasing diversity and inclusion will also be included in the support infrastructure for the degree.

Arizona University System

NBR	PROGRAM	DEGREE	#STDNTS	LOCATION	ACCRDT
1	Software Engineering	BS	1322	Arizona State University	Y
2	Computer Science	BS	1387	University of Arizona	Y

Peer Comparison

See attached file for a detailed analysis of the proposed Software Engineering degree compared to the Software Engineering degree programs at Arizona State University, Iowa State University and also UArizona's Computer Science degree program.

Faculty & Resources

Faculty

Current Faculty:

INSTR ID	NAME	DEPT	RANK	DEGREE	FCLTY/%
00747035	Richard Scholes	2303	Adj. Assoc. Prof	Master of Science	.20
00971536	Donald Bruyere	2302	Lecturer	Master of Science	.25
01840754	Sherilyn Keaton	2302	Lecturer	Master of Bus Admin	.50
02600592	Kenneth Head	2302	Professor	Doctor of Philosophy	.25
03308095	Jerzy Rozenblit	2303	Professor	Doctor of Philosophy	.25
15304542	Ali Akoglu	2303	Assoc. Prof	Doctor of Philosophy	.25
16207802	Jonathan Sprinkle	2303	Assoc. Prof	Doctor of Philosophy	.25
16600630	Loukas Lazos	2303	Professor	Doctor of Philosophy	.50
22050348	Benjamin Dicken	0412	Lecturer	Master of Science	.50
22053534	Ricardo Valerdi	2302	Professor	Doctor of Philosophy	.25
22060179	Ratchaneekorn Thamvichai	2303	Prof. Pract.	Doctor of Philosophy	.80
22063694	Samuel Peffers	2302	Lecturer	Doctor of Philosophy	.25
22067613	Gregory Ditzler	2303	Assit. Prof	Doctor of Philosophy	.25
22071369	Vignesh Subbian	2302	Assit. Prof	Doctor of Philosophy	.25
22072066	Sharon O'Neal	2302	Adj. Lect.	Master of Science	.50
22084246	Gregory Heileman	2303	Professor	Doctor of Philosophy	.25
22088449	Melanie Lotz	0412	Lecturer	Master of Science	.50
23393839	Robert McBride	2303	Adj. Lect.	Doctor of Philosophy	.35

Additional Faculty:

The additional required faculty for the Software Engineering degree program have been broken down into multiple categories, as shown below. These numbers reflect the academic year that the additional faculty will be recruited / hired over 5 years. It is envisioned that the majority of the curriculum for the first year will be closely aligned to many of the other engineering degree programs. Hence for the first year of the degree offering, new faculty can focus on developing the curriculum for the 2nd and 3rd years of the program. However, because a few of the curriculum courses are already very heavily populated

courses, such as ECE 175 and Engr 102, additional faculty, TA and lab assistant resources will be required to help mitigate the increased demand.

Year 1 (2021-2022):
- Tenured Track Faculty = 2
- Professor of Practice = 1
- Adjunct Faculty = 1

Year 2 (2022-2023):
- Tenured Track Faculty = 2
- Professor of Practice = 1
- Adjunct Faculty = 0

Year 3 (2023-2024):
- Tenured Track Faculty = 2
- Professor of Practice = 0
- Adjunct Faculty = 0

Year 4 (2024-2025):
- Tenured Track Faculty = 1
- Professor of Practice = 0
- Adjunct Faculty = 0

Year 5 (2025-2026):
- Tenured Track Faculty = 1
- Professor of Practice = 0
- Adjunct Faculty = 0

Total additional faculty over 5 years:
- Tenured Track Faculty = 8
- Professor of Practice = 2
- Adjunct Faculty = 1

Current Student & Faculty FTE

DEPARTMENT	UGRD HEAD COUNT	GRAD HEAD COUNT	FACULTY FTE
0412	1387	72	27.00
2302	530	138	24.35
2303	393	172	34.00

Projected Student & Faculty FTE

	UGRD HEAD COUNT			GRAD HEAD COUNT			FACULTY FTE		
DEPT	YR 1	YR 2	YR 3	YR 1	YR 2	YR 3	YR 1	YR 2	YR 3
2302	30	60	110	0	0	0	1.25	2.25	4.25
2303	30	60	110	0	0	0	2.00	4.00	4.00

Library

Acquisitions Needed:

There are no anticipated additional library acquisition needs with the Software Engineering degree program.

Physical Facilities & Equipment

Existing Physical Facilities:

Students and faculty for the proposed Software Engineering degree program will utilize the existing classrooms, laboratories, computer facilities, physical equipment that is currently available to all Engineering majors; in particular the Electrical and Computer Engineering Department and the Systems and Industrial Engineering Department.

Additional Facilities Required & Anticipated:

At this time, there are no additional equipment / physical facility needs identified. Many of the tools and lab facilities used by this degree program already exist for other courses offered within the ECE and SIE departments. Many software development tools traditionally used for software engineering courses are ¿open source¿ and readily available for faculty and students to use at little to no charge. Cloud based resources, such as those offered through Amazon Web Services (AWS) or Microsoft Azure, can host integrated development environments for courses that require more comprehensive integrations of multiple tool types of toolsets to support student projects. If there are licenses or cloud-based resource usage fees that become necessary for different software engineering courses in the future, those costs could be included in course fees.

Other Support

Other Support Currently Available:

because the Software Engineering degree will be a collaborative program offered between the ECE and SIE Departments, other support staff from those two departments will be available to meet the needs of the program. However, as noted below, there are additional support roles required as the program reaches its steady state enrollment which is expected within 4 years.

Other Support Needed over the Next Three Years:

The additional staff that will be required to support the Software Engineering Degree program over the next 3 years is shown below:

Undergraduate Advisor: 1
IT Support: 1
Finance Support: 1

The following additional resources will be required as semester hires only over the next 3 years:

	Fall 2021 / Spring 2022	Fall 2022 / Spring 2023	Fall 2023 / Spring 2024
Teaching Assistants:	4	8	12
Graders:	1	3	4
Laboratory Assistants:	4	20	0

Comments During Approval Process

12/22/2020 2:45 PM

SHARONONEAL

Comments
For this proposal, we projected our additional resources and financial projections out 5 years, as opposed to the required 3.

12/23/2020 12:18 AM

ROZANNEC

Comments
Approved.

12/23/2020 1:56 AM

SON

Comments
Approved.

12/23/2020 1:57 AM

SON

Comments
Approved.

THE UNIVERSITY OF ARIZONA

New Academic Program - Undergraduate Major
CURRICULAR INFORMATION

I. MAJOR DESCRIPTION: *Provide a marketing/promotional description for the proposed program (recommend working with your college marketing team). The description will be displayed on the advisement report(s), Degree Search, and should match departmental and college websites, handouts, promotional materials, etc.*

The Bachelor of Science in Software Engineering synergistically integrates proven engineering techniques and discipline with software development best practices that encompass all aspects of the software development lifecycle (SDLC). The curriculum includes core principles from systems engineering, electrical and computer engineering, and software engineering. The curriculum is based on a solid foundation of mathematics, including calculus, physics, and discrete math. The courses include software requirements analysis, design, code, integration, verification testing, and software project management. While there are some similarities between the Software Engineering degree program and UArizona's Computer Science degree program offered in the College of Science, there are also major differences. In Computer Science, students focus more on the programming fundamentals and theoretical applications of developing software. Software Engineering students, on the other hand, focus more on solving complex, multi-faceted/multi-disciplined engineering problems and product development.

The Software Engineering curriculum prepares students to meet the ever-growing demands of commercial, industrial, and federal government job sectors. Relevant software methodologies, such as Agile development, automated testing using continuous integration, and SW DevOps (DevOps combines software development (Dev) and IT operations (Ops)) to increase software application velocity and service delivery, are also integral to the curriculum. Using these types of agile and adaptive approaches, students will be well suited for the many diverse opportunities in a rapidly growing and ever-evolving career in software engineering.

The program has a firm engineering foundation, discovery-based education, and an experiential learning approach. As a part of the curriculum, students work on and complete projects in every semester of the program that emphasize communication, teamwork, critical thinking, and professionalism. This program's flexibility allows students to design their course of study and select technical electives from a diverse pool of courses in software and computer engineering domains such as web and mobile applications, embedded systems, and other interdisciplinary areas.

Software engineering students will acquire considerable software development skills, including: experience with multiple programming languages, data structure constructs, algorithm implementation, databases, operating systems, networking,

embedded systems, cloud computing, configuration management, software assurance, and the use of open-source software libraries/programs. Additionally, students in the Software Engineering degree program develop skills required to work on teams and with other engineering disciplines. These skills include the ability to design and develop creative software solutions that are integral to a variety of large-scale and complex systems. Software systems are typically multi-faceted and often include millions of lines of code with diverse origins or pedigrees developed by teams of software engineers. Graduates of the bachelor's degree program in Software Engineering learn to use structured and well-defined engineering approaches to develop, evaluate, and maintain software-centric systems.

II. JUSTIFICATION/NEED FOR THE MAJOR: *Describe the purpose and need for the proposed major, and how the major fulfills the needs of students, as well as the local, national, and/or global community. Provide market analysis data or other tangible evidence of the need for and interest in the proposed major (and emphases, if applicable). This might include results from surveys of current students, alumni, or employers, or reference to enrollments in peer programs. Include an assessment of the employment opportunities for graduates of the program for the next three years. Curricular Affairs can provide a job posting/demand report by CIP code of the proposed major, upon request.*

The Bureau of Labor Statistics reports the total number of Software Engineering jobs in 2010 to be 1.85M, and projects to over 2.4M in 2020; a 23% increase over ten years. U.S News & World Report ranked software developer/engineer as the best technology job in America.

In the past 12 months, Burning Glass reports that there was a total of 43,563 job postings for *Computer Software Engineering* (CIP = 14.0903) from 08/01/19 – 07/31/2020. As shown in Table II.1 below, the number of jobs is expected to have a relatively high growth rate over the next 10 years.

GROWTH BY GEOGRAPHY

Geography	Selected Occupations	Total Labor Market	Relative Growth
Nationwide	15.46 %	4.24 %	High

Table II.1. National Growth Projected in Software Engineering

The career outlook for software engineers has continued an upward trajectory for the past 6 years, as depicted in the Figure II.1 below:

	2014	2015	2016	2017	2018	2019
Employment (BLS)	31,420	34,020	35,160	38,610	38,280	38,792

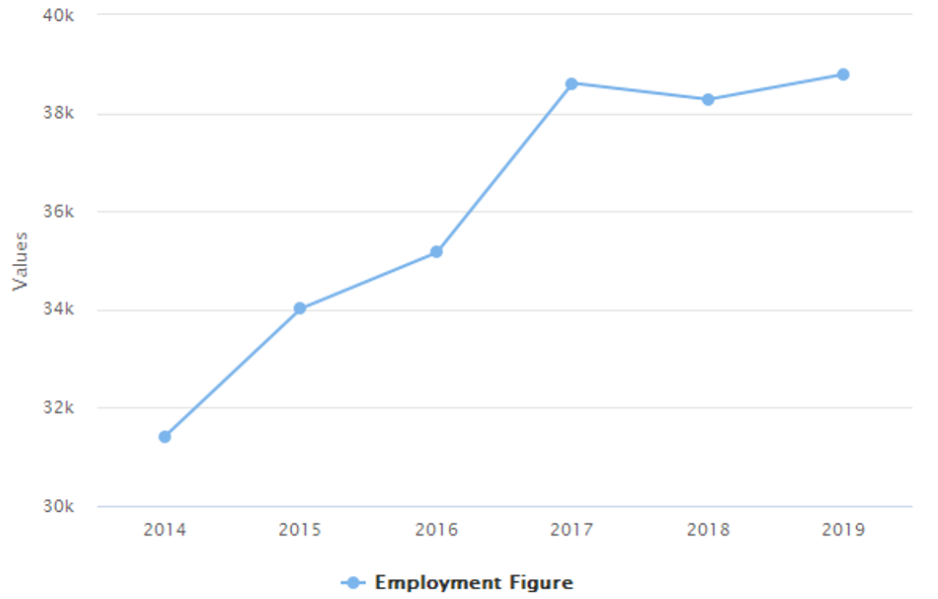

Figure II.1 Career Outlook for Software Engineering over the Past 6 Years

Nationwide, the top locations posting requisitions for Software Engineers is shown in Table II.2 below:

Location	Postings
California	309,439
Texas	173,408
Virginia	114,302
New York	98,786
Florida	80,669
North Carolina	80,604
Illinois	68,747
Massachusetts	63,683
Georgia	62,505
Colorado	59,274

Table II.2. Software Engineering Posting by State

Within the state of Arizona, the top employers hiring software engineers include the following (see Table II.3):

TOP EMPLOYERS HIRING

Experience Level: All Experience

Employer	Postings	Market Share (%)
Raytheon	931	4.00%
USAA	765	3.29%
Wells Fargo	564	2.42%
General Dynamics	310	1.33%
Honeywell	304	1.31%
Northrop Grumman	299	1.28%
American Express	297	1.28%
Deloitte	288	1.24%
The Boeing Company	176	0.76%
Mitsubishi	169	0.73%
Anthem Blue Cross	154	0.66%
CVS Health	145	0.62%
IBM	132	0.57%
Viasat	126	0.54%
Amazon	120	0.52%

Table II.3. Arizona Top Employers with Software Engineering Needs

The top 15 specialized software skills required by employers include a variety of programming languages, software engineering, Linux operating system, software project management, software DevOps, Scrum (a particular form of Agile), quality assurance, and systems engineering. The number of job postings and the anticipated growth in that specific skill category specifically requested in these different software engineering skills is summarized in Table II.4, as shown below:

Skill	*Postings*	*Projected Growth*	*Salary Premium*	*Competitive Advantage*
Java	14112 (32%)	-13.18%	No	No
Software Development	11197 (26%)	5.78%	No	No
SQL	10788 (25%)	-13.3%	No	No
Software Engineering	7641 (17%)	7.27%	No	No
JavaScript	7611 (17%)	6.81%	No	No
Python	6560 (15%)	61.12%	No	No
Linux	5650 (13%)	-12.57%	No	No
Microsoft C#	5130 (12%)	-25.69%	No	No
DevOps	5048 (12%)	107.85%	Yes	No

Project Management	4706 (11%)	-19.74%	No	No
Oracle	4620 (11%)	-16.26%	No	No
Quality Assurance	4600 (11%)	39.46%	No	No
Git	4239 (10%)	59.81%	No	No
Scrum	4156 (10%)	39.96%	No	No
Systems Engineering	3740 (9%)	-4.85%	No	No

Table II.4. Postings for the Top 15 Software Engineering Specialized Skills

The top 15 software engineering skill clusters are shown in Table II.5 below:

Skill	*Postings*
Software Development Principles	19331 (44%)
Java	14291 (33%)
System Design and Implementation	13878 (32%)
SQL Databases and Programming	12479 (29%)
JavaScript and jQuery	10579 (24%)
Software Quality Assurance	8917 (20%)
Operating Systems	8518 (19%)
Scripting Languages	7644 (17%)
Web Development	7380 (17%)

Software Development Methodologies	7316 (17%)
Business Process and Analysis	7146 (16%)
Database Administration	7136 (16%)
Microsoft Development Tools	7056 (16%)
Cybersecurity	6963 (16%)
Programming Principles	6954 (16%)

Table II.5. Postings for the Top 15 Software Engineering Skill Clusters

III. MAJOR REQUIREMENTS: Complete the table below. Note: information in this section must be consistent throughout the proposal documents (comparison charts, four year plan, curricular/assessment map, etc.) and will be used to build the Academic Advisement Report (ADVIP). Please include letters of support for any courses not offered by the proposing department (see Workflow Input form). Complete the table in Appendix A if requesting a corresponding minor.

Total units required to complete the degree	120 units
Upper-division units required to complete the degree	46
Foundation courses	
Second language	None required
Math	• MATH 125 – Calculus I • MATH 129 – Calculus II

	• Math 243 – Discrete Math • MATH / PHYS / STATS Elective
General education requirements	3 courses/ 9 units –Tier I 3 courses/ 9 units – Tier II 2 courses/ 6 units – English Composition I & II 2 courses/ 8 units – Basic Science (in place of Tier I Natural Science requirement)
Pre-major? (Yes/No). If yes, provide requirements. Provide email(s)/letter(s) of support from home department head(s) for courses not owned by your department.	Yes. Completion of 12 or more UA credits of coursework within the Engineering curricula may that include: • MATH 125 – Calculus I • MATH 129 – Calculus II • PHYS 141 – Introductory Mechanics • ENGL 102 – English Composition II • ENGR 102 – Intro to Engineering • ECE 175 – Programming I
List any special requirements to declare or gain admission to this major (completion of specific coursework, minimum GPA, interview, application, etc.)	All students are enrolled as *Engineering*, *No Major Selected* until they have completed the following: • Calculus I with a grade of C or better • 12 or more UA credits of coursework within the Engineering curriculum (shown above) • Admissions GPA of 2.0 or higher
Major requirements	
Minimum # of units required in the major (units counting towards major units and major GPA)	56 units in major + 12 units technical electives
Minimum # of upper-division units required in the major (upper division units counting towards major GPA)	37 upper division units in major + 9 upper division technical electives
Minimum # of residency units to be completed in the major	30
Required supporting coursework (courses that do not count towards major units and major GPA, but are required for the major). Courses listed must include prefix, number, units, and title. Include any limits/restrictions needed (house number limit, etc.). Provide email(s)/letter(s) of support from home	MATH 125 Calculus I (3 units) MATH 129 Calculus II (3 units) MATH 243 Discrete Math (3 units) PHYS 141 Introductory Mechanics (4 units) PHYS 241 Introductory Electricity and Magnetism (4 units) MATH / PHYS / STAT Elective (3 units)

department head(s) for courses not owned by your department.	
Major requirements. List all major requirements including core and electives. If applicable, list the emphasis requirements for each proposed emphasis*. Courses listed count towards major units and major GPA. Courses listed must include prefix, number, units, and title. Mark new coursework (New). Include any limits/restrictions needed (house number limit, etc.). Provide email(s)/letter(s) of support from home department head(s) for courses not owned by your department.	ENGR 102 A/B - Introduction to Engineering (3 units) (New) SFWE 101 – Introduction to Software Engineering (3 units) ECE 175 – Programming I (3 units) (New) SFWE 201 – Sophomore Colloquium (1 unit) ECE 274A – Digital Logic (3 units) ECE 275 – Programming II (3 units) SIE 277 Object Oriented Modeling and Design (3 units) SIE 305 – Engineering Probability and Statistics (3 units) (New) SFWE 301 – Software Requirements Analysis and Test (3 units) ECE 311 – Engineering Ethics (1 unit) (New) SFWE 304 Software Preceptor I (1 unit) ECE 369A' – Computer Organization (3 units w/o lab) (New) SFWE 302 – Software Design Process (3 units) CSC 345' - Algorithms (3 units, CoE majors) (New) SFWE 401 – Software Assurance and Security (3 units) (New) SFWE 403– Software Project Management (3 units) SIE 464 – Cost Estimation (3 units) ENGR 498A – Interdisciplinary Capstone (3 units) (New) SFWE 402 – Software DevOps (3 units) ENGR 498B – Interdisciplinary Capstone (3 units) (New) SFWE 404 – Software Preceptor II (3 units) Select 12 units of technical electives from other Engineering courses (i.e. ECE, SIE, or other applicable engineering courses), CSC or ISOC. See major advisor for course approval.
Internship, practicum, applied course requirements (Yes/No). If yes, provide description.	Yes. Complete 6 units: ENGR 498 A and ENGR 498B (Interdisciplinary Capstone in Senior year)
Senior thesis or senior project required (Yes/No). If yes, provide description.	Yes. Interdisciplinary Design project that is part of ENGR 498 A/B.

Additional requirements (provide description)	None
Minor (specify if optional or required)	Optional
Any double-dipping restrictions (Yes/No)? If yes, provide description.	No

**Emphases are officially recognized sub-specializations within the discipline. ABOR Policy 2-221 c. Academic Degree Programs Subspecializations requires all undergraduate emphases within a major to share at least 40% curricular commonality across emphases (known as "major core"). Total units required for each emphasis must be equal. Proposed emphases having similar curriculum with other plans (within department, college, or university) may require completion of an additional comparison chart. Complete the table found in Appendix B to indicate if emphases should be printed on student transcripts and diplomas.*

IV. NEW COURSES NEEDED: *If new courses are required for the proposed degree, UA Course Add forms must be submitted simultaneously with this proposal [link to course process/additional instruction]. List all course additions in progress in the table below. Add rows as needed.*

Is a new prefix needed? If so, provide the subject description so Curricular Affairs can generate proposed prefix options.

Course prefix and number (include cross-listings)	Units	Title	Course Description	Pre-Requisites	Modes of delivery (online, in-person, hybrid)	Course Fee? (Y/N) [Link to process]	Course Form transact-ion number	Anticipated first term offered	Typically Offered (F, W, Sp, Su)	Use in the major (required /elective)	Faculty members available to teach the courses
SFWE 101	3	Introduction to Software Engineering	This course introduces students to the different software development lifecycle (SDLC) phases used in developing, delivering, and maintaining software products. Fundamental software development tools will be used in	Engr 102	In Person	No	TBD	Spring 2022	Sp	Required	Sharon ONeal

			projects to familiarize students to the basic tasks involved in modifying, building, and testing software. The course will also lay the foundation for achieving academic and career success in Software Engineering.								
SFWE 201	1	Sophomore Colloquium	A colloquium designed to help students understand what software engineers do in diverse industry and research settings. Students will interact and network with professional, academic, and other industry leaders in software technology arenas, and take tours to local companies. The course helps students select course electives relevant to their career objectives, and chart a path to explore specific software applications that they are interested in.	None	In Person	No	TBD	Fall 2022	F	Required	TBR (New Faculty)
SFWE 301	3	Software Requirements Analysis and Test	This course teaches students how to derive and develop software requirements that are measurable, testable and lead to a compliant software design and implementation. Using industry best practices and tools, students will	SFWE 101, Advanced Standing	In Person	No	TBD	Fall 2023	F	Required	TBR (New Faculty)

			learn how to elicit, analyze, specify, and validate software requirements. Students will develop Software Requirements Specifications and Test Plans and Procedures used in requirement verification and test. In addition to developing and analyzing software requirements, students will also learn how to establish and maintain a software requirement configuration baseline for subsequent changes, updates, and enhancements to the software product as it evolves over time.								
SFWE 302	3	Software Design Process	This course explores current processes, methods, and tools used in modeling and designing software systems. A fundamental and experiential component of this course provides students the opportunity to work in teams on a semester long software design project that utilizes industry best practices, common software design patterns, and modeling tools. They will analyze various software quality attributes such as	SIE 277, Advanced Standing	In Person	No	TBD	Spring 2024	Sp	Required	TBR (New Faculty)

			Interoperability, Reliability, Extensibility and Portability in design decision making. Traceability between software requirements and the resulting software designs, and ultimately code implementation, will also be part of the semester project.								
SFWE 304	1	Software Preceptor I	This course provides the opportunity for junior software engineering majors to work in software development labs for entry level students taking ENGR 102, ECE 175, ECE 275 and other similar courses. Students will help 1st and 2nd year students set up their software development environments and tools, and provide guidance and help with writing, testing, and debugging code.	ECE 275, Advanced Standing	In Person	No	TBD	Fall 2023	F/Sp	Required	TBR (New Faculty)
SFWE 401	3	Software Assurance and Security	In this course, students will learn to develop and utilize secure software coding standards and defensive programming techniques to ensure that the software functions in its intended manner; free from coding errors and inadvertent bugs that may have been	ECE 275, Advanced Standing	In Person	No	TBD	Fall 2024	F	Required	TBR (New Faculty)

			introduced during development. This course promotes software integrity, security, quality, and reliability in the software as it is developed and tested. Using automated tools such as static code analyzers, students will evaluate both their own and other students' code to ensure it conforms to secure coding standards, is free from potential vulnerabilities, and executes as intended.								
SFWE 402	3	Software DevOps	In this course, students will develop a software product using a Software DevOps workflow approach with common tools used in industry. Students will develop a Continuous Integration capability that enables automated testing of the software via development of test scripts that are automatically executed and used to analyze and evaluate test results. DevOps also enables students to develop, test, and deliver software products faster and more efficiently, while at the same time providing a development pipeline of new capabilities and features	ECE 275, SFWE 301, SFWE 302	In Person	No	TBD	Spring 2025	Sp	Required	TBR (New Faculty)

			to consumers. Using DevOps workflows and continuous integration / continuous delivery approaches, students will be able to assess if the quality and reliability of the software is increased.								
SFWE 403	3	Software Project Management	In this course, students are introduced to common software project management methodologies used to manage a large, multi-faceted software development project. Using commercially available and open-source tools, students will plan and track the progress of a software development project, and use traditional software metrics to evaluate and monitor team performance, predict cost and schedule execution, develop and assess software risks/mitigation plans, and measure the quality and maturity/volatility of the software product itself.	Advanced Standing, SWFE 101	In Person	No	TBD	Fall 2024	F	Required	TBR (New Faculty)
SFWE 404	1	Software Preceptor II	This course provides the opportunity for senior software engineering majors to work in software development labs for entry level students taking ENGR	SFWE 304	In Person	No	TBD	Fall 2024	F/Sp	Required	TBR (New Faculty)

			102, ECE 175, ECE 275 and other similar courses. Students will help 1st and 2nd year students set up their software development environments and tools, and provide guidance and help with writing, testing, and debugging code.								

Subject description for new prefix (if requested). Include your requested/preferred prefix, if any:

We are requesting that the new Software Engineering courses be given a prefix of “SFWE”. These courses are specific to the BS in Software Engineering or a Minor in Software Engineering.

V. FOUR-YEAR PLAN: provide a sample four-year degree plan that includes all requirements to graduate with this major and takes into consideration course offerings and sequencing. Use generic title/placeholder for requirements with more than one course option (e.g. Upper Division Major Elective, Minor Course, Second Language, GE Tier 1, GE Tier 2). Add rows as needed.

Semester 1		Semester 2		Semester 3		Semester 4	
Course prefix and number	Units	Course prefix and number	Units	Course prefix and number	Units	Course prefix and number	Units
MATH 125	3	SFWE 101	3	SFWE 201	1	SIE 277 (or ECE 373)	3
ENGR 102 A/B	3	MATH 129	3	Basic Science / Lab	4	SFWE Elective	3
ENGL 101	3	GE Tier 1	3	PHYS 141	4	GE Tier 1	3
Basic Science / Lab	4	ENGL 102	3	ECE 274A	3	PHYS 241	4
GE Tier 1	3	ECE 175 (or CSC 110)	3	ECE 275	3	Math 243	3
Total	16	Total	15	Total	15	Total	16

Semester 5		Semester 6		Semester 7		Semester 8	
Course prefix and number	Units	Course prefix and number	Units	Course prefix and number	Units	Course prefix and number	Units
SIE 305	3	ECE 369A' (or CSC 252)	3	SFWE 401	3	SFWE 402	4
SFWE 301	3	GE Tier 2	3	SFWE 403	3	SFWE Elective	3
ECE 311	1	SFWE 302	3	SIE 464	3	SFWE Elective	3
MATH/PHYS/STAT Elective	3	CSC 345'	3	ENGR 498A	3	ENGR 498B	3
SFWE Elective	3	GE Tier 2	3	GE Tier 2	3	SFWE 404	1
SFWE 304	1						
Total	14	Total	15	Total	15	Total	14

Appendix A. Minor Requirements. Complete if requesting a corresponding minor, with the same title as the major.

Minimum total units required	19
Minimum upper-division units required	13
Total transfer units that may apply to the minor	6
List any special requirements to declare/admission to this minor (completion of specific coursework, minimum GPA, interview, application, etc.)	• Meet with academic advisor and obtain permission to declare minor • Complete all pre-requisite coursework

Minor requirements. List all minor requirements including core and electives. Courses listed must include course prefix, number, units, and title. Mark new coursework (New). Include any limits/restrictions needed (house number limit, etc.). Provide email(s)/letter(s) of support from home department head(s) for courses not owned by your department.	• ECE 175 Programming I
	• ECE 275 Programming II
	• (New) SFWE 301 Software Requirements Analysis and Test
	• CSC 345' Analysis of Discrete Structures and Algorithms (for CoE majors)
	• (New) SFWE 403 Software DevOps
	• Choose 1 SFWE Elective from the following courses:
	  ○ (New) SFWE 302 Software Design Process
	  ○ (New) SFWE 401 Software Assurance and Security
	  ○ (New) SFWE 403 Software Project Management

Internship, practicum, applied course requirements (Yes/No). If yes, provide description.	None
Additional requirements (provide description)	None
Any double-dipping restrictions (Yes/No)? If yes, provide description.	No

Learning Outcomes and Assessment Plan:

The use and reporting of program assessment data to improve the quality of academic programs for students is required of all degree programs. Assessment should focus foremost on collecting data that will inform you about student learning that matters and should be viewed as a continuous source of knowledge for program and institutional improvement. A member of the Assessment Team in the Office of Instruction and Assessment must review and approve the student learning outcomes, curriculum maps, and process of assessment descriptions for all new programs.

Process of Assessment

The Process of Assessment section is for describing how faculty and staff will be involved in the development, implementation, and use of student learning outcomes. This section should include sufficient detail so that when new personnel are in place, the assessment can continue without interruption.

Your Process of Assessment:

The proposed Software Engineering degree program has well defined student learning outcomes with a documented and effective process for periodic review and continuous improvement. The learning outcomes include both well-defined activities or problems that are either practical and narrowly focused on terms of scope, or broadly defined activities that are relatively complex, broad in scope, and involve a variety of resources (such as processes, materials, or techniques) used in innovative ways. The assessment plan regularly evaluates the extent to which the Student Learning Outcomes (SLO) are being attained. The results of these evaluations are systematically used as input for the program's continuous improvement actions.

For the Software Engineering degree program, we plan to:

1. Have an Academic Programs Assessment Committee to implement, maintain, and improve assessment tools for continuous improvement and to analyze assessment data and formulate recommendations for improving the program based on these analyses.
2. Review all educational objectives annually with the Software Engineering Advisory Council and the Student / Faculty Advisory Council.
3. Charge the ECE and the SIE Undergraduate Program Committees with ensuring that the curriculum provides students with the experience necessary to meet the Software Engineering Educational Objectives and Program Outcomes.
4. Create an interdisciplinary committee with faculty from ENGR, CS, MIS, and the iSchool, to investigate opportunities for collaboration and course/project development that will strengthen the interdisciplinary nature of software and information-based systems
5. Review all educational objectives annually at the ECE and SIE Faculty retreats held before the Fall semester.

Figure 1 shows the overall flow diagram for the Software Engineering degree program's continuous improvement process, which follows a well-established process used by the Systems and Industrial Engineering Department.

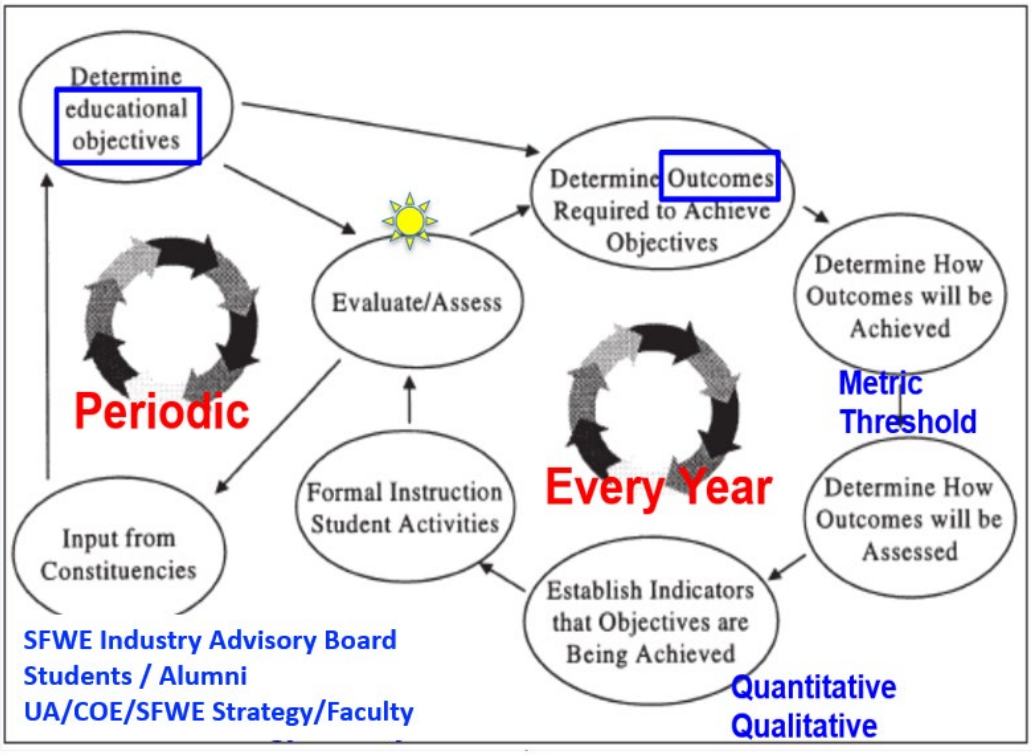

Figure 1. Software Engineering Degree Outcome Assessment and Continuous Improvement Process

As the student progresses through their academic program, there are three primary procedures in place to ensure that the student is making appropriate progress and attaining the required course grades for success:

1. SAAR (Student Academic Advising Report) (Student Advisement Report), an online record keeping database for which both student and advisor have access.

2. UAccess/Analytics Student Center, the academic course review conducted each semester by College and Department to determine performance (GPA of 2.0 or greater). At the end of each semester, students that are not meeting those requirements are considered “at-risk students” and an intervention is conducted. The Academic Program Coordinator contacts all at-risk students to set up meetings with the student, the Academic Program Coordinator, and appropriate SW Engineering faculty to create academic agreements; and

3. Senior Degree Check, required a semester prior to graduation.

Additionally, the College of Engineering conducts a CoE Senior Exit Survey for all engineering graduates. A SWFE Senior Exit Survey will be requested from all Software Engineering graduates which is specific to the SW Engineering courses and projects, and current and future career goals for each senior. The information from the survey will be an integral part of the annual SIE and ECE Engineering Undergraduate Committee (UGC) assessment and will be used to provide input and guidance to the continuous improvement activities for the Software Engineering degree.

In consultation with the Office of Instruction & Assessment, we plan to evaluate the overall Software Engineering program after the first graduating class has completed the program, and every year thereafter. Using well-established processes for developing program outcomes, we will evaluate:

1. The extent to which the program outcomes meet the needs of our students at the time of graduation.

2. Assess how the degree and program outcomes meets the needs of industry.
3. Evaluate how the outcomes meet our program objectives for alumni shortly after their graduation.

Learning Outcomes

Program student learning outcomes (SLOs) are clear, concise statements that describe how students can demonstrate their mastery of program goals. These statements identify the knowledge, skills, or attitudes that students will be able to demonstrate, represent, or produce upon successful completion of the program. Please list your learning outcomes:

Your Learning Outcomes (please number them):

Learning Outcome #	Learning Outcome Statement
1	*An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics.*
2	*An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors.*
3	*An ability to communicate effectively with a range of audiences.*
4	*An ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts.*
5	*An ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives.*
6	*An ability to develop and conduct appropriate experimentation, analyze, and interpret data, and use engineering judgment to draw conclusions.*
7	*An ability to acquire and apply new knowledge as needed, using appropriate learning strategies.*

These Learning Outcomes match the ABET Learning Outcomes that are required for ABET accreditation.

Assessment Activities and Measures (Assessment Plan) and Curriculum Map

Assessment activities or measures should align with learning outcomes and clearly answer the question: How can we check that this learning outcome is being met by our students? Start by defining one direct and one indirect measure for each outcome. You do NOT need a separate assessment for each outcome; some assessments may measure more than one learning outcome. Once you have your measures identified for each outcome, contact the Assessment Team in the Office of Instruction and Assessment for the creation of your map in Taskstream. The team member will also review the rest of the assessment plan at this time.

The Software Engineering Curriculum map depicted in Figure 2 shows the required courses mapped to the seven ABET compliant Student Learning Outcomes (SLO). For each outcome, the figure indicates when each outcome is *Introduced*, *Practiced* and *Assessed* for the different courses that are part of the curriculum. As reflected in the figure, at least three different courses are mapped to each outcome. Further, there are four or more courses that satisfy the outcome for many SLOs. Existing SIE or ECE courses have well-defined assessment measures and assessment protocols in place. The assessment measures and practices for SFWE courses will be established and evolve as new courses are developed, offered, and assessed.

University of Arizona AMS » Sandboxes
Ingrid Novodvorsky Playspace

BS Software Engineering Curriculum Map

Courses and Activities Mapped to BS Software Engineering

	Outcome						
	ABET Outcome #1 An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics.	ABET Outcome #2 An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors.	ABET Outcome #3 An ability to communicate effectively with a range of audiences.	ABET Outcome #4 An ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts.	ABET Outcome #5 An ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives.	ABET Outcome #6 An ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions.	ABET Outcome #7 An ability to acquire and apply new knowledge as needed, using appropriate learning strategies.
Courses and Learning Activities							
MATH 129 Calculus II	I						
ENGL 102 Composition II			P				
ECE 175/CSC 110 Computer Programming	IPA	IPA			IPA		
SFWE 201 Sophomore Colloquium				I/A			I
Basic Science Lab	I/P						
PHYS 141 Intro Mechanics	I						
ECE 274A Digital Logic	IPA	IPA			IPA		
ECE 275 Object Oriented Software		P/A					

Figure 2. Bachelor of Science in Software Engineering Curriculum Mapped to Student Learning Outcomes (1 of 2)

SIE 277 Object Oriented Design		P/A	P/A		P/A		
PHYS 241 Intro E & M	I						
MATH 243 Discrete Mathematics	I						
SIE 305 Probability & Statistics	I/A					I/A	
SFWE 301 Software Reqmnts. Analysis & Testing	IPA	IPA	I/P				
ECE 311 Engineering Ethics		I/A	I/A	I/A			
SFWE 304 Software Engineering Preceptor					P/A		
SFWE 302 Software Design Process		IPA	I/P			I/P	
CSC 345 Algorithms	IPA						
SFWE 401 Software Assurance & Security	IPA			I/A			
SFWE 403 Software Project & Process Mgt.		IPA	IPA	I/A	P/A		
ENGR 498A/B Senior Capstone		P/A	P/A		P/A	P/A	P/A
SFWE 402 Software Dev. Ops.					P/A	P/A	
SFWE 404 Software Engineering Preceptor					P/A		
Exit Survey Indirect Measure	A	A	A	A	A	A	A

Legend : I Introduced | P Practiced | A Assessed | I/P Introduced/Practices

Last Modified: 12/07/2020 11:20:50 AM

Figure 2 (Continued). Bachelor of Science in Software Engineering Curriculum Mapped to Student Learning Outcomes (2 of 2)

A rubric will be created for each new SFWE course that identifies Criteria, Measures of Assessment, and an Achievement level rating for the different criteria/categories evaluated. The Achievement levels will include: “Exemplary”, “Satisfactory”, “Developing”, and “Unsatisfactory”. For new courses developed in the curriculum, a plan for identifying the courses of evidence and assessment measures will be developed as the courses are developed.

An example rubric that could be adapted for the specific SFWE courses is shown in Figure 3 *(Note: this only exemplifies the intent of the rubrics to be developed, not the actual rubric)*. At the end of every semester, a team comprised of the course instructor and the SIE and ECE UGCs, will score the rubric using the measures of assessment identified for the course. A Root Cause and Corrective action plan will be developed for any course that scores “Developing” or below. Assessment results are documented and formally maintained in a controlled location at the end of each semester and will be published at the UArizona Assessment website. The scores will be tracked over time to facilitate the Continuous Improvement and corrective action plans remain effective from semester to semester, year to year.

PI	Outcome 1	An ability to identify, formulate and solve complex engineering problems by applying principles of engineering, science and mathematics.			
	Performance Indicators	UNSATISFACTORY	DEVELOPING	SATISFACTORY	EXEMPLARY
1a	Understanding of fundamental principles, concepts, and theory (e.g., governing equations, word equations)	Major lack in either the understanding of fundamental conceptual notions and/or theory in the topic at hand.	Grasps some concepts and an understanding of the theory. The student struggles to remember the main steps required in the derivation of important equations from basic principles.	Good grasp of most concepts and an understanding of the theory. The student understands the main steps required in the derivation of important equations from basic principles.	Mastery of the concepts and theory of the topic at hand. Ability to write precisely all the steps required in the derivations of fundamental equations.
1b	Mathematical formulation and model simplification of engineering problems	Inability to identify the basic principles involved and corresponding assumptions. The student can't write the model equations and/or its variables.	Lists some of the principles involved in a problem and corresponding assumptions. The student struggles to write the model equations and correctly identify the problem variables.	List of all principles involved in a problem and corresponding assumptions. The student is able to write the model equations and correctly identify the problem variables.	List of all principles involved in a problem and corresponding assumptions. Mathematical formulation is clean with clear explanation and identification of its variables (scalar, vector, matrix) and their relation to physical representation.
1c	Solution techniques (analytical and/or numerical)	Inability to identify an appropriate solution approach. Derivations and results contain unit inconsistencies. The unit inconsistencies are not explicitly identified.	Identifies but struggles to implement a proper solution technique. Some of the derivations are correct. Some of the results are presented with the correct units or recognized unit error. Struggles to identify unrealistic solutions.	Identification and implementation of proper solution technique. Most of the derivations are correct. Results are presented with the correct units or recognized unit error. Identification of unrealistic solutions.	Identification and implementation of proper solution technique. In the case of analytical problems, derivations are rigorous and symbolic. Numerical results are presented with an appropriate number of digits and correct units.
	Outcome 2	An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety and welfare, as well as global, cultural, social, environmental and economic factors.			
	Performance Indicators	UNSATISFACTORY	DEVELOPING	SATISFACTORY	EXEMPLARY
2a	Ability to formulate a design problem for specified needs and technical requirements.	Fails to understand the design requirements and/or integrate them in the design problem.	Struggles with formulating a design problem that meets the specified needs and technical requirements.	Formulates the design problem based on understood requirements leading to a feasible design solution.	Ability to foresee potential challenges and identify multiple design formulations that will lead to feasible design solutions.
2b	Ability to identify disciplines, techniques, and tools for the solution of the design problem.	Does not know the basic relevant disciplines, techniques, and tools necessary as part of the design process.	Capable of identifying some of the disciplinary skills, techniques, and tools for solving the problem.	Broadly identifies disciplinary skills, techniques, and tools needed to obtain a design satisfying some of the requirements. Ability to identify and distribute tasks.	Clearly identifies specific disciplinary skills, techniques, and tools to obtain a design satisfying all of the requirements. Ability to identify and distribute tasks.
2c	Ability to account for non-technical factors (e.g., cost) and formulate or quantify societal impacts (e.g., pollution, safety)	Cannot foresee non-technical societal implications or integrate them in the formulation of the design problem.	Identifies some non-technical factors associated with the problem.	The student is able to formulate given societal impacts and non-technical consequences as constraints or objectives of the design problem.	Can formulate his/her own design problem combining technical and societal objectives and/or constraints.
	Outcome 3	An ability to communicate effectively with a range of audiences.			
	Performance Indicators	UNSATISFACTORY	DEVELOPING	SATISFACTORY	EXEMPLARY

Figure 3. Example Rubric for Assessing Student Learning Outcomes (1 of 3)

3a	Oral communication. Ability to inform, persuade, and present results and conclusions.	Provides an unclear and unstructured and/or poorly formatted presentation. Does not respond effectively to questions.	Presents facts but lacks organization. Presentation materials are okay. Struggles to present pursuading arguments.	Ability to present relevant technical information in a concise and organized way. Quality presentation materials. Ability to understand and answer questions (including acknowledgment of limitations) . Able to convey a message to instructors and other students.	Clearly presents technical content in a concise and pedagocial way. High quaity presentation materials. Ability to answer questions in a clear and concise way. Able to convey a message at a level outside the classroom (e.g., at technical conferences, industrial customers, general audience).
3b	Written communication. Ability to structure a technical document in a logical manner. Ability to produce professionally formatted documents.	Provides a disorganized and/or poorly formatted document.	Produces a document containing several formatting and grammatical errors.	Ability to present relevant technical information in an organized way. Presents in a proper format, the objectives, the hypotheses, the methods, results, and conclusions.	Ability to present relevant technical information in a structured narrative. Presents with a format of the highest quality, the objectives, the hypotheses, the methods, results, and conclusions.
3c	Ability to communicate through design iterations in response to client feedback.	Misunderstanding of the feedback and/or poor expression of the required design changes.	Responds to some of the feedback.	Clear understanding of the feedback and expression of the required design changes.	Clear understanding of the feedback and concise expression of the required design changes along with adequate justification.
	Outcome 4	**An ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental and societal contexts.**			
	Performance Indicators	**UNSATISFACTORY**	**DEVELOPING**	**SATISFACTORY**	**EXEMPLARY**
4a	Understanding of responsibilities to stakeholders, teammates, and professional societies	Insufficient contributions.	Becoming aware of the stakeholders.	Fulfills the required tasks. Participates actively and make contribution on the team.	Clear understanding of responsibilities and highly proactive in fulfilling them.
4b	Ability to differentiate unethical and ethical behaviors.	Misses obvious ethical failures.	Starting to identify unethical behavior.	Reasonable explanation of an ethical behavior and actions.	Ability to identify specific actions addressing ethical concerns.
4c	Ability to make informed judgements considering consequences and liability.	Failure to identify obvious broader consequences.	Beginning to consider the consequences of their decisions.	Based on available information and data, ability to identify broader consequences.	Ability to identify broader consequences and provide recommendations to mitigate adverse consequences.
	Outcome 5	**An ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks and meet objectives.**			
	Performance Indicators	**UNSATISFACTORY**	**DEVELOPING**	**SATISFACTORY**	**EXEMPLARY**
5a	Interaction with the group. Collegiality.	Uncooperative	Beginning to listen to others, but challenged to work together.	Ability to listen, respond, and compromise for the good of the project.	Proven interpersonal skills improving the cohesion of the group for improved efficacy and efficiency.
5b	Dependability	Unreliable	Completes some of the assigned tasks, but lacks consistency.	Carries out requested tasks fully and in a timely manner. Communicates progress on regular basis. Warns the group of potential delays and problems.	In addition to satisfactory criteria, demonstrates a willingness to assist other team members.

Figure 3 (Continued). Example Rubric for Assessing Student Learning Outcomes (2 of 3)

5c	Contributions to the team	Unengaged	Shares ideas with team, but struggles with engaging fully with the team.	Provides intellectual and leadership contributions as an active member of the team.	Provided intellectual and leadership contributions which markedly benefited the project.
	Outcome 6	An ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions.			
	Performance Indicators	**UNSATISFACTORY**	**DEVELOPING**	**SATISFACTORY**	**EXEMPLARY**
6a	For given technical objectives, ability to identify a set of input and output quantities of interest and appropriate experiment(s).	Inability to identify appropriate tools/techniques (experimental or computational) and select parameters to test a technical objective such as a design requirement. Tends to attempt impractically complex solutions.	Struggles with identifying the tools and/or techniques needed to develop and conduct the appropriate experiments.	Ability to identify tools and/or techniques (experimental or computational) and select appropriate parameters to test a technical objective such as a design requirement.	Ability to test technical objectives (e.g., design requirements) in an efficient manner (e.g., avoids unecessary complexity).
6b	Effectively carries out the appropriate experiment(s).	Unable to perform the relevant experiment without major assistance.	Developing an ability to identify the proper experiments.	With minor guidance, given the materials and equipment, ability to set up the experiments.	Full autonomy setting up the experiments given materials and equipment.
6c	Basic interpretation of the raw data and physical interpretation based on engineering judgment.	Does not provide error metrics and/or cannot relate the measurements to physical quantities. Does not identify meaningless results.	Beginning to apply statistical strategies to raw data. Starting to question inaccurate data.	Ability to process raw data, generate appropriate statistics, and identify outlier behaviors. Ability to relate statistics to physically-meaningful quantities and/or phenomena.	Ability to identify outlier behaviors and propose revisions to the experiment(s). Ability to relate statistics to physically-meaningful quantities and/or phenomena and understand their engineering implications.
	Outcome 7	An ability to acquire and apply new knowledge as needed, using appropriate learning strategies.			
	Performance Indicators	**UNSATISFACTORY**	**DEVELOPING**	**SATISFACTORY**	**EXEMPLARY**
7a	Ability to identify needed additional knowledge.	Failure to identify obvious limitations of in-class knowledge in the context of a particular problem (e.g., using linear theory in a nonlinear range).	Beginning to consider learning beyond the classroom.	For a given problem, successfully identify some needed additional knowledge to solve a problem beyond what is covered in the classroom (e.g., understand the underlying hypotheses and limitations of in-class techniques).	For a given problem, successfully identify *all* relevant additional knowledge to solve a problem beyond what is covered in the classroom.
7b	Ability to obtain appropriate information and tools beyond the classroom.	Failure to find approriate information.	Developing a sense of when the classroom tools are lacking.	For a given problem, find required missing data. Also identify possible theories and tools, not covered in the classroom, that would enable a correct solution.	For a given problem, find required missing data. Thorough search for theories and tools, not covered in the classroom, and use judgement to assess their relevance.
7c	Ability to understand and apply knowledge obtained beyond the classroom.	Unable to understand and/or apply any new theories and/or tools.	Beginning to learn beyond the classroom.	Correctly use the newly found data. Understand and successfully apply at least one of the newly found theories and/or tools to a given problem with some guidance (e.g, through step by step approach).	Correctly use the newly found data. Thorough understanding and efficient application of the newly found theories and/or tools.

Figure 3 (Continued). Example Rubric for Assessing Student Learning Outcomes (3 of 3)

Table 1 delineates the sources of evidence, assessment measurements, and data collection points for each planned learning outcome. Since existing SIE and ECE courses already have well-established measures and assessment plans in place, those measures are specifically outlined in the table. For new courses developed in

the curriculum, a plan for identifying the sources of evidence and assessment measures is provided in the table.

Table 1. Learning Outcome 1 Assessment Measures and Data Collection Points

Learning Outcome 1:	***An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics***	
Source of Evidence	**Assessment Measures**	**Data Collection points**
For each course that contributes to this outcome, specific student artifacts will be evaluated and assessed. The sources of evidence can include: - Class assignments - Exams - Course projects - Course reports - other forms of student work For existing courses in the SW Engineering program, the evidence used to measure the effectiveness of the student outcome has been already been defined and will be followed. For new courses, the specific evidence used will be defined as the course is developed and re-evaluated as part of the continuous improvement activities for the new course.	**ECE 175 and 274A:** A rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. **SIE 305:** Homework Assignment #10 for which 75% of the students must score 80% or higher. For **SFWE 301 and SFWE 401**, a rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. Student self-assessment survey (used for indirect measures of the outcome).	• End of each course • Assessment is planned annually, at a minimum

Table 1 (Continued). Learning Outcome 2 Assessment Measures and Data Collection Points

Learning Outcome 2:	***An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors***	
Source of Evidence	**Assessment Measures**	**Data Collection points**
For each course that contributes to this outcome, specific student artifacts will be evaluated and assessed. The sources of evidence can include: - Class assignments - Exams - Course projects - Course reports - other forms of student work For existing courses in the SW Engineering program, the evidence used to measure the effectiveness of the student outcome has been already been defined and will be followed. For new courses, the specific evidence used will be defined as the course is developed and re-evaluated as part of the continuous improvement activities for the new course.	**SIE 277:** Semester Design Project for which 75% of the students must score 70% or higher. **ENGR 498 A/B:** 90% of the students must score 80% or higher (Average of SRD, PDR, & CDR) . **ECE 175, 274A, 275, and ECE 311:** A rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. For **SFWE 101, SFWE 301, SFWE 302, and SFWE 403**, a rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. Student self-assessment survey (used for indirect measures of the outcome).	• End of each course • Assessment is planned annually, at a minimum

Table 1 (Continued). Learning Outcome 3 Assessment Measures and Data Collection Points

Learning Outcome 3:	***An ability to communicate effectively with a range of audiences***	
Source of Evidence	**Assessment Measures**	**Data Collection points**
For each course that contributes to this outcome, specific student artifacts will be evaluated and assessed. The sources of evidence can include: - Class assignments - Exams - Course projects - Course reports - other forms of student work For existing courses in the SW Engineering program, the evidence used to measure the effectiveness of the student outcome has been already been defined and will be followed. For new courses, the specific evidence used will be defined as the course is developed and re-evaluated as part of the continuous improvement activities for the new course.	**SIE 277:** Semester Design Project for which 75% of the students must score 70% or higher. **ECE 311:** A rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. **ENGR 498 A/B:** 90% of the students must score 80% or higher (Average of CDR and Final Report). For **SFWE 403**, a rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. Student self-assessment survey (used for indirect measures of the outcome).	• End of each course • Assessment is planned annually, at a minimum

Table 1 (Continued). Learning Outcome 4 Assessment Measures and Data Collection Points

Learning Outcome 4:	***An ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts***	
Source of Evidence	**Assessment Measures**	**Data Collection points**
For each course that contributes to this outcome, specific student artifacts will be evaluated and assessed. The sources of evidence can include: - Class assignments - Exams - Course projects - Course reports - other forms of student work For existing courses in the SW Engineering program, the evidence used to measure the effectiveness of the student outcome has been already been defined and will be followed. For new courses, the specific evidence used will be defined as the course is developed and re-evaluated as part of the continuous improvement activities for the new course.	**ECE 311:** A rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. For **SFWE 101, SFWE 201, SFWE 401, SFWE 403**, a rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. Student self-assessment survey (used for indirect measures of the outcome).	• End of each course • Assessment is planned annually, at a minimum

Table 1 (Continued). Learning Outcome 5 Assessment Measures and Data Collection Points

Learning Outcome 5:	***An ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives***	
Source of Evidence	**Assessment Measures**	**Data Collection points**
For each course that contributes to this outcome, specific student artifacts will be evaluated and assessed. The sources of evidence can include: - Class assignments - Exams - Course projects - Course reports - other forms of student work For existing courses in the SW Engineering program, the evidence used to measure the effectiveness of the student outcome has been already been defined and will be followed. For new courses, the specific evidence used will be defined as the course is developed and re-evaluated as part of the continuous improvement activities for the new course.	**SIE 277:** Semester Design Project for which 75% of the students must score 70% or higher. **ECE 175, and ECE 274A:** a rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. **ENGR 498 A/B:** 90% of the students must score 80% or higher (Average of TE (Fall) and TE (Spring)). For **SFWE 101, SFWE 304, SFWE 402, SFWE 403 and SFWE 404**, a rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. Student self-assessment survey (used for indirect measures of the outcome).	• End of each course • Assessment is planned annually, at a minimum

Table 1 (Continued). Learning Outcome 6 Assessment Measures and Data Collection Points

Learning Outcome 6:	***An ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions***	
Source of Evidence	**Assessment Measures**	**Data Collection points**
For each course that contributes to this outcome, specific student artifacts will be evaluated and assessed. The sources of evidence can include: - Class assignments - Exams - Course projects - Course reports - other forms of student work For existing courses in the SW Engineering program, the evidence used to measure the effectiveness of the student outcome has been already been defined and will be followed. For new courses, the specific evidence used will be defined as the course is developed and re-evaluated as part of the continuous improvement activities for the new course.	**SIE 305:** Homework Assignment #10 with 75% of the students must score 80% or higher **ENGR 498 A/B:** 90% of the students must score 80% or higher (Average of PDR and FAR). For **SFWE 402**, a rubric will be used to assess this student outcome. Specific sources of evidence will be identified in each course and used in the rubric based assessment. For each rubric category, a rating will be given commensurate with the rubric criteria rating scheme. Student self-assessment survey (used for indirect measures of the outcome).	• End of each course • Assessment is planned annually, at a minimum

Table 1 (Continued). Learning Outcome 7 Assessment Measures and Data Collection Points

Learning Outcome 7:	***An ability to acquire and apply new knowledge as needed, using appropriate learning strategies***	
Source of Evidence	**Assessment Measures**	**Data Collection points**
For each course that contributes to this outcome, specific student artifacts will be evaluated and assessed. The sources of evidence can include: - Class assignments - Exams - Course projects - Course reports - other forms of student work For existing courses in the SW Engineering program, the evidence used to measure the effectiveness of the student outcome has been already been defined and will be followed. For new courses, the specific evidence used will be defined as the course is developed and re-evaluated as part of the continuous improvement activities for the new course.	**ENGR 498 A/B:** 90% of the students must score 80% or higher (Average of PDR, CDR, and Final Report). Student self-assessment survey (used for indirect measures of the outcome).	• End of each course • Assessment is planned annually, at a minimum

THE UNIVERSITY OF ARIZONA

New Academic Program - Undergraduate Major
PEER COMPARISON

Select three peers (if possible/applicable) for completing the comparison chart from ABOR-approved institutions, AAU members, and/or other relevant institutions recognized in the field. The comparison programs are not required to have the same degree type and/or title as the proposed UA program. Information for the proposed UA program must be consistent throughout the proposal documents.

Program name, degree, and institution	Proposed UA Program	Arizona State University BS Software Engineering	Iowa State University BS Software Engineering	University of Arizona BS Computer Science
Current number of students enrolled		• 1322 (Total) • 563 (Full time) *(Source: ASEE 2019)*	• 416 (Total) • 393 (Full time) *(Source: ASEE 2019)*	• 1387 Total *(Source: UArizona 2019)*
Program Description	The Bachelor of Science in Software Engineering synergistically integrates proven engineering techniques and discipline with software development best practices that encompass all aspects of the software development lifecycle (SDLC). The curriculum includes core principles from systems engineering, electrical and computer engineering, and software engineering. The curriculum is based on a solid foundation of mathematics, including calculus, physics, and discrete math. The courses include topics related to software requirements analysis, design, code, integration, verification testing and software	The BS program in Software Engineering blends engineering, computing, project leadership and software construction. Students learn how to make creative software solutions to today's problems. Software systems are complex, often including millions of lines of code. Graduates of the bachelor's degree program in Software Engineering possess the knowledge and skills of a defined engineering approach to complex systems analysis, planning, design, and construction.	The Bachelor of Science degree in Software Engineering is jointly administered by the College of Engineering and the College of Liberal Arts and Sciences. The Software Engineering program provides undergraduate students with the opportunity to learn software engineering fundamentals, to study applications of state-of-the art software technologies and to prepare for the practice of software engineering. The student-faculty interaction necessary to realize this opportunity occurs within an environment motivated by the principle that excellence in undergraduate education is	Computer science emerges from the interaction of two powerful kinds of machines: computers and the human brain. Computer scientists are inventive, innovative, collaborative thinkers creating software solutions and synergies on the cutting edge of technology. The Information Age is here; computer scientists are shaping it. All undergraduate students begin the program as Pre-Computer Science. Students are admitted to the BA or BS in Computer Science program once pre-major courses are completed, and admission criteria are met.

	project management. While there are some similarities between the Software Engineering degree program and UArizona's Computer Science degree program offered in the College of Science, there are also major differences. In Computer Science, students focus more on the programming fundamentals and theoretical applications of developing software. Software Engineering students, on the other hand, focus more on solving complex, multi-faceted/multi-disciplined engineering problems and product development. The Software Engineering curriculum is designed to prepare students to meet the ever-growing demands within the commercial, industrial, and federal government job sectors. Relevant software methodologies, such as Agile development, automated testing using continuous integration, and SW DevOps to increase the velocity of software application and service delivery, are also integral to the curriculum. Using these types of agile and adaptive approaches, students will be well suited for the many diverse opportunities in a	The program has a unique, project-driven curriculum, establishing a new model for software engineering education. The program is built around the concepts of engaged learning, discovery-based education and learn-by-doing. Students complete projects in every semester of the program to provide emphasis in communication, teamwork, critical thinking, and professionalism. Students have flexibility in designing their course of study; they select technical electives from a pool of courses in different software engineering application areas such as Web and Mobile applications, Embedded systems, and other interdisciplinary areas.	enhanced by an integrated commitment to successful, long-term research and outreach programs. The software engineering curriculum offers many elective choices in software engineering. Students may also take elective courses in computer engineering and computer science.	

	rapidly growing and ever evolving career in software engineering.			

Target Careers	• Software developer
	• Software Engineer for variety of application areas:
	  ○ Web
	  ○ Mobile
	  ○ Embedded systems
	  ○ Avionics
	  ○ Robotics
	  ○ Other software related fields
	• Software Quality Assurance
	• Software Leadership
		• Software developer
		• Software Engineer for variety of application areas:
		  ○ Web
		  ○ Mobile
		  ○ Embedded systems
		  ○ Avionics
		  ○ Robotics
		  ○ Other software related fields
		• Software Quality Assurance
		• Software Leadership
			• Software developer
			• Software Engineer for variety of application areas:
			  ○ Web
			  ○ Mobile
			  ○ Embedded systems
			  ○ Avionics
			  ○ Robotics
			  ○ Other software related fields
			• Software Quality Assurance
			• Software Leadership
				• Computer programmer
				• Software developer
				  ○ Artificial Intelligence programmer
				  ○ Machine Learning programmer
				• Database specialist
				• Mobile application developer
				• Web developer

Emphases? (Yes/No) List, if applicable	No	No	No	No
Minimum # of units required in the major	56 units in major + 12 units technical electives *(120 units for degree)*	56 units in major + 12 units technical electives *(120 units for degree)*	7 units basic Engineering + 46 units core major + 6 units technical electives *(125 units for degree)*	15 units Pre-Major + 21 units Major + 9 units technical electives *(120 units for degree)*
Level of Math required	**Substantial** Includes 23 total units of Calculus I & II, Discrete Math, Statistics and Physics (2 courses)	**Substantial** Includes 22 total units of Calculus I, II, III, Discrete Math, Linear Algebra, Statistics, Physics (1 course)	**Substantial** Includes 23 total units of Calculus I, II, Differential Equations, Statistics and Physics (1 course)	**Significant** Includes 12 units of Pre-Calculus, Calculus I, and Calculus II or Linear Algebra
Level of Second Language required	None	None	2nd semester proficiency	2nd semester proficiency
Pre-Major? (Yes/No) If yes, provide requirements.	Yes • *Admitted as "Engineering - No Major Selected"* • Completion of 12 or more UA credits of coursework within the Engineering curricula may that include: • MATH 125 – Calculus I	No	No	Yes • CSC 110 - Intro to Computer Programming 1 • CSC 120 - Intro to Computer Programming 2 • CSC 210 - Software Development

	• MATH 129 – Calculus II • PHYS 141 – Introductory Mechanics • ENGL 102 – English Composition II • ENGR 102 – Intro to Engineering • ECE 175 – Programming I			• CSC 245 - Intro to Discrete Structures
Special requirements to declare/gain admission to major? (i.e. pre-requisites, GPA, application, etc.)	All students are enrolled as *Engineering*, *No Major Selected* until they have completed the following: • Calculus I with a grade of C or better • 12 or more UA credits of coursework within the Engineering curriculum (shown above) • Admissions GPA of 2.0 or higher	• Minimum 1210 SAT combined evidence-based reading and writing • Math score of a minimum 24 ACT combined score or 3.00 minimum ABOR GPA or class ranking in top 25% of high school class • Transfer students minimum transfer GPA of 2.75.	In addition to the high school course requirements (4 years English, 3 years Math, 3 years Science, and 2 years Social Studies), students applying for admission to the College of Engineering must complete two years of a single world foreign language in high school, or two semesters of college in a single world language. Students applying for admission to the College of Liberal Arts and Sciences must complete a third year of social studies and two years of a single foreign language.	• Completion of the Pre-Major Courses (shown above) • GPA of 3.0 or higher in CSC 230, 210, and 245 • Cumulative GPA of 2.4 or higher • GPA of 2.0 or higher in all attempts at UA CSC courses • Complete at least 2 programming courses at UA
Internship, practicum, or applied/experiential requirements? If yes, describe.	Yes • Senior Interdisciplinary Capstone (ENGR 498A and ENGR 498B)	Yes • Computing Capstone Project I and II (SER 401 and 402)	Yes • Optional 1 semester Co-Op or 10-week Internship • Senior Design I and II (SE 491 and SE 492)	Optional Students may earn up to 6 units of internship credit.

Additional questions:

1. *How does the proposed program align with peer programs? Briefly summarize the similarities between the proposed program and peers, which could include curriculum, overall themes, faculty expertise, intended audience, etc.*

The UA Software Engineering degree program is like all three of the peer programs that require and build strong foundational skills in Math and/or Physics as part of their curriculums. The Math courses are similar and deviate only after Calculus II. All programs also offer several introductory programming courses using a variety of computer programming languages. Additional programming skill development/experience is an integral part of the experiential course work and projects in other required courses. All programs offer Object-Oriented SW Analysis and Design, and a variety of technical electives to match student special interest areas. The 3 Software Engineering programs (UA, ASU, and Iowa State) all offer full-semester courses in software requirements analysis and test, software architecture and design, software security, computer organization, and some form of project and/or process management. Additionally, the software engineering degree programs also offer a 2-semester capstone course. Students in all the programs will be able to pursue software development careers in a variety of diverse and expansive applications areas including: Web-based development, mobile application development, embedded systems, robotics, machine learning, artificial intelligence, and other software-related fields.

For the three Software Engineering degree programs, the student learning outcomes (SLOs) comply with the seven ABET criteria for Engineering majors, and also comply with the following 2 additional ABET requirements specific to a Software Engineering degree program:

1. Curriculum
The curriculum must provide both breadth and depth across the range of engineering and computer science topics implied by the title and objectives of the program.
The curriculum must 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.

2. Faculty
The program must demonstrate that faculty members teaching core software engineering topics have an understanding of professional practice in software engineering and maintain currency in their areas of professional or scholarly specialization.

2. *How does the proposed program stand out or differ from peer programs? Briefly summarize the differences between the proposed program and peers, which could include curriculum, overall themes, faculty expertise, intended audience, etc.*

While there are many similarities in the Software Engineering degree programs at UArizona, ASU and Iowa State, there are also key differences that are very attractive to prospective UArizona students. First and foremost, because the proposed degree program at UA is offered out of the College of Engineering, one of its major strengths is the multi-disciplinary influences provided by the Systems and Industrial Engineering, and Electrical and Computer Engineering Departments. The elective options available to students are very diverse and can include courses that give students a broad-based experience in not only software engineering, but also Electrical Engineering and/or Systems Engineering specialties. The intersections of the ECE, SE, and the SW Engr degree programs, foster the ability to tackle interdisciplinary engineering problems to meet the evolving technological changes and requirements to meet society's needs. This manifests itself in software engineering students being an integral part of the highly successful Interdisciplinary Capstone course (ENGR 498A/B) where students work on multi-disciplined teams to develop products for a diverse set of industry and/or academia sponsors.

The new UArizona Software Engineering courses will be developed using relevant and industry-focused technology solutions, tools, languages, and methodologies in a diverse portfolio of applications. Wherever possible, the software development tools and platforms used in the coursework will consist of widely available open-source integrated development environments (IDEs), operating systems (OS), and cloud-based infrastructures. The Software Assurance and Security course will follow best practice secure coding standards/methodologies and use commercially available static code analyzers widely used in the industry to ensure compliance to secure coding practices. The Software DevOps course uses a state-of-the-art software DevOps workflow approach with common tools used in the industry. Software DevOps enables students to develop, test, and deliver software products faster and more efficiently, while at the same time providing a development pipeline of new capabilities and features to consumers. Using DevOps workflows and continuous integration / continuous delivery (CI/CD) approaches, students will be able to plan, develop, and deliver software features to meet customer's ever-evolving needs. Students will also learn to track and evaluate how the software's quality and reliability is increased using the SW DevOps approaches.

While all three Software Engineering programs have some form of Software Project Management course, the UArizona course will also cover different common SW development methodologies. These methodologies include the traditional waterfall development approach, as well multiple Agile methodologies (Scrum, Peer Programming, KanBan, etc). Further, students will learn how to use different measures and metrics to track and predict progress in meeting delivery milestones. UA's Software

Project Management course will also cover software configuration control (SWCM), how to make use software reliability predictive measures, and software quality assurance (SWQA). The Cost Estimation course in the UA program provides students with exposure to different methodologies and tools used to estimate software development costs for different types of products.

As is often asked, what are the differences between the UArizona BS Software Engineering degree and the UArizona Computer Science degree? To begin with, the ABET criteria for the two degrees are different. The Software Engineering degree complies with the ABET criteria for Engineering degrees and the two additional software engineering specific criteria discussed above (Curriculum and Faculty). The Computer Science degree is compliant with the ABET criteria for Computer Science and Computing. The SW Engineering degree is comprised of more than 25% of courses in math, basic science, and physics, while the Computer Science has only 12 units of math. Physics courses are not required in the Computer Science program. The Venn diagram shown in Figure 1 below highlights some of the other major differences between the programs. While there are some intersections in both programs between the topics and types of classes in each respective degree programs, the focus of each program is very different. In Computer Science, students focus more on the programming fundamentals and theoretical applications of developing software. Software Engineering students, on the other hand, focus more on solving complex, multi-faceted/multi-disciplined engineering problems and product development.

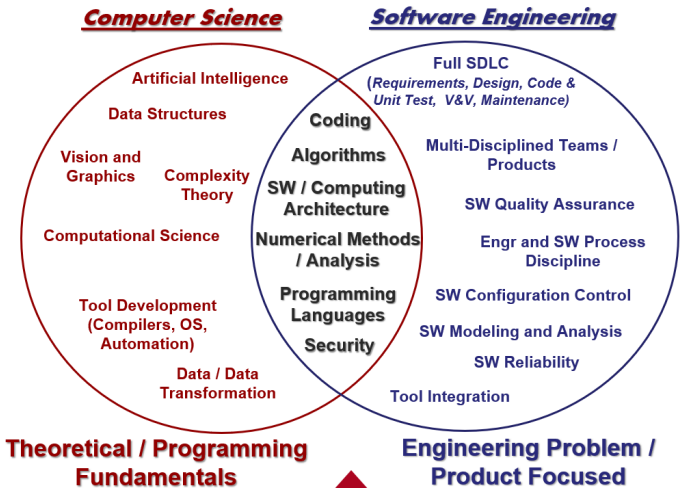

Figure 1. Comparison of Computer Science and Software Engineering Programs

3. *How do these differences make this program more applicable to the target student population and/or a better fit for the University of Arizona?*

The University of Arizona College of Engineering currently does not offer an engineering degree related explicitly to Software Engineering. UA engineering students that have historically leaned more toward software engineering careers have typically obtained Electrical and Computer Engineering degrees, with an emphasis towards Computer Engineering, and take various computer programming courses as electives. Alternatively, these students obtain a Computer Science degree that lead to future software career opportunities but lack the specific engineering discipline and emphasis offered with the Software Engineering degree curriculum. Since Software Engineering is growing as a '*in high-demand'* engineering degree (as shown by the Burning Glass analysis), it is likely that students are selecting other universities since UArizona does not offer Software Engineering. It is believed that offering an innovative Software Engineering degree will attract new students. As we have seen over the past decade, software has become an integral element/component within the systems, products, and technologies that are part of the 4th industrial revolution.

Given the ever-increasing demand in the industry for software engineers, there is an equally increasing opportunity for the CoE to diversify their degree offerings by offering this new degree. This will attract more students to the University of Arizona.

Additionally, two of the colleges' strategic pillars are:

1) Driving student success for a rapidly changing world, and
2) Tackling critical problems at the edges of human endeavor.

The new Software Engineering degree plays a critical role in both pillars. The students graduating with the degree in Software Engineering will be better positioned to develop the skills and mindsets to be ***leaders*** in the areas of space, natural and built environments, ever-increasing automation and connectivity, human and intelligent systems, data, computing, and network sciences.

By offering a competitive, relevant, and experiential-based learning Software Engineering program to prospective students, it increases not only the net enrollment in the college, but also the ability to grow research programs that are attractive to

forthcoming undergraduate and graduate students. All of which contribute to higher recruitment numbers and bringing additional revenue to the College and University. Hence, we can recruit the faculty that can significantly impact their areas of research and education. These faculty will pursue research grants to advance the state-of-the-art in software engineering and integrate their research into the curriculum. The broader impact of these faculty will ultimately drive the program's national ranking higher.

While less obvious, another goal for the program is to increase the number of female and other underrepresented students in the College of Engineering by offering the Software Engineering degree. Additional features and programs that contribute to enhancing student success and increasing diversity and inclusion will also be included in the support infrastructure for the degree.

THE UNIVERSITY OF ARIZONA®

BUDGET PROJECTION FORM ON CAMPUS

Name of Proposed Program or Unit: Software Engineering

Budget Contact Person:	Projected		
	1st Year 2021 - 2022	2nd Year 2022 - 2023	3rd Year 2023 - 2024
METRICS			
Net increase in annual college enrollment UG	60	120	220
Net increase in college SCH UG	540	1,320	2,880
Net increase in annual college enrollment Grad	-	-	-
Net increase in college SCH Grad	-	-	-
Number of enrollments being charged a Program Fee (differential tuition) Lower division	60	120	160
Number of enrollments being charged a Program Fee (differential tuition) Upper division	-	-	60
New Sponsored Activity (MTDC)			
Number of Faculty FTE			
FUNDING SOURCES			
Continuing Sources			
UG RCM Revenue (net of cost allocation)	134,591.27	293,990.47	603,717.01
Enrollment (calculation only - included in UG RCM Revenue)	*42,444.70*	*80,447.47*	*145,743.44*
SCH (calculation only - included inUG RCM Revenue)	*92,146.57*	*213,543.00*	*457,973.57*
Grad RCM Revenue (net of cost allocation)	15,193	32,556	49,919
Program Fee RCM Revenue (net of cost allocation) Lower division	39,427.54	78,312.31	104,059.96
Program Fee RCM Revenue (net of cost allocation) Upper division			78,855.07
F and A Revenues (net of cost allocations)		122,154	244,309
UA Online Revenues			
Distance Learning Revenues			
Reallocation from existing College funds (attach description)			
Other Items (attach description)			
Total Continuing	**$ 189,211**	**$ 527,013**	**$ 1,080,859**
One-time Sources			
College fund balances			
Institutional Strategic Investment			
Gift Funding			
Other Items (attach description)			
Total One-time	**$ -**	**$ -**	**$ -**
TOTAL SOURCES	**$ 189,211**	**$ 527,013**	**$ 1,080,859**
EXPENDITURE ITEMS			
Continuing Expenditures			
Faculty	180,000	360,000	540,000
Other Personnel	192,100	342,660	306,180
Employee Related Expense	112,586	205,107	260,104
Graduate Assistantships	66,761	133,521	200,282
Other Graduate Aid			
Operations (materials, supplies, phones, etc.)	140,500	140,500	140,500
Additional Space Cost			
Other Items (attach description)			
Total Continuing	**$ 691,947**	**$ 1,181,788**	**$ 1,447,066**
One-time Expenditures			
Construction or Renovation			
Start-up Equipment			
Replace Equipment			
Library Resources			
Other Items (attach description)	800,000	800,000	800,000
Total One-time	**$ 800,000**	**$ 800,000**	**$ 800,000**
TOTAL EXPENDITURES	**$ 1,491,947**	**$ 1,981,788**	**$ 2,247,066**
Net Projected Fiscal Effect	**$ (1,302,735)**	**$ (1,454,775)**	**$ (1,166,207)**

Note: This worksheet contains the information for a combined On Campus and Online offerings of the
The first year reflects the projections for On Campus only. The current plan is to develop the Online pro
begin offering an Online variant in the 2nd year, 2022-2023.

THE UNIVERSITY OF ARIZONA.

BUDGET PROJECTION FORM ON CAMPUS and Online

Name of Proposed Program or Unit: Software Engineering

Budget Contact Person:

METRICS	1st Year 2021 - 2022
Net increase in annual college enrollment UG	60
Net increase in college SCH UG	540
Net increase in annual college enrollment Grad	-
Net increase in college SCH Grad	-
Online SCH	270
Number of enrollments being charged a Program Fee (differential tuition) Lower division	60
Number of enrollments being charged a Program Fee (differential tuition) Upper division	-
New Sponsored Activity (MTDC)	
Number of Faculty FTE	

FUNDING SOURCES

Continuing Sources		
UG RCM Revenue (net of cost allocation)		134,591.27
Enrollment (calculation only - included in UG RCM Revenue)		*42,444.70*
SCH (calculation only - included in UG RCM Revenue)		*92,146.57*
Grad RCM Revenue (net of cost allocation)		15,193
Program Fee RCM Revenue (net of cost allocation) Lower division		39,427.54
Program Fee RCM Revenue (net of cost allocation) Upper division		
F and A Revenues (net of cost allocations)		
UA Online Revenues		103,680
Distance Learning Revenues		
Reallocation from existing College funds (attach description)		
Other Items (attach description)		
Total Continuing	**$**	**292,891**

One-time Sources
College fund balances
Institutional Strategic Investment

Gift Funding		
Other Items (attach description)		
Total One-time	$	-
TOTAL SOURCES	$	292,891
EXPENDITURE ITEMS		
Continuing Expenditures		
Faculty		180,000
Other Personnel		192,100
Employee Related Expense		112,586
Graduate Assistantships		66,761
Other Graduate Aid		
Operations (materials, supplies, phones, etc.)		140,500
Additional Space Cost		
Other Items (attach description)		
Total Continuing	$	691,947
One-time Expenditures		
Construction or Renovation		
Start-up Equipment		
Replace Equipment		
Library Resources		
Other Items (faculty start up costs)		800,000
Total One-time	$	800,000
TOTAL EXPENDITURES	$	1,491,947
Net Projected Fiscal Effect	$	(1,199,055)

Software Engineering degree program, extrapolated over the first 5 years.
ogram offering during the first year the On campus program is offered, and

Projected

2nd Year 2022 - 2023	3rd Year 2023 - 2024	4th Year 2024 - 2025	5th Year 2025 - 2026
120	220	350	350
1,320	2,880	5,230	5,790
-	-	-	-
-	-	-	-
660	1,440	2,615	2,895
120	160	230	190
-	60	120	160

293,990.47	603,717.01	1,041,155.92	1,127,068.13
80,447.47	*145,743.44*	*227,277.96*	*226,998.73*
213,543.00	*457,973.57*	*813,877.96*	*900,069.40*
62,941	115,030	180,141	206,186
78,312.31	104,059.96	149,058.28	122,711.83
	78,855.07	156,624.61	208,119.93
122,154	244,309	366,463	427,540
253,440	552,960	1,004,160	1,111,680
$ 810,838	$ 1,698,931	$ 2,897,603	$ 3,203,306

$ -	$ -	$ -	$ -
$ 810,838	$ 1,698,931	$ 2,897,603	$ 3,203,306
360,000	540,000	630,000	720,000
359,160	471,180	618,540	616,620
210,222	311,254	380,412	408,270
250,352	450,634	700,986	801,127
140,500	140,500	133,000	133,000
$ 1,320,234	$ 1,913,568	$ 2,462,938	$ 2,679,017
800,000	800,000	400,000	400,000
$ 800,000	$ 800,000	$ 400,000	$ 400,000
$ 2,120,234	$ 2,713,568	$ 2,862,938	$ 3,079,017
$ (1,309,396)	$ (1,014,637)	$ 34,665	$ 124,289

Added other personnel for online
Added ERE for online
Added TA's for online

These tables represent the additional resources for the On Campus Pr

Projected Additional Resource Acquisition Plan (by Year) (On Campus					
Resource Type	**2021-2022**	**2022-2023**	**2023-2024**	**2024-2025**	**2025-2026**
Tenured Track Faculty	2	2	2	1	1
PoP (On Campus)	1	1	0	0	0
Adjunct	1	0	0	0	0

Projected Additional Salaries by Year (On Campus Only)					
Resource Type	**2021-2022**	**2022-2023**	**2023-2024**	**2024-2025**	**2025-2026**
Tenured Track Faculty	2	4	6	7	8
Professor of Practice	1	2	2	2	2
Adjunct	1	1	1	1	1
UG Advisor	1	1	1	1	1
Other Staff	1	2	2	2	2

Other Resources Required (semester hires only) (On Campus Only)					
Resource - additional in pe	**2021 - 2022**	**2022 - 2023**	**2023 - 2024**	**2024 - 2025**	**2025 - 2026**
TAs	4	8	12	20	25
Graders	1	3	4	12	11
Lab Assts	4	20	0	0	0

rogram ONLY

Only)
Total New Instructors Acquired Over 5 Years
8
2
1

Note: The table to the left shows the year and number of additional resources that we intend to ACQUIRE in that year. Should be used to account for start-up costs only.

Note: The table to the left shows the total number of additional salaried employees required each year. The numbers in this table are cumulative and extend out over each year.

*Note: The table to the left shows the total number of semester hires required each year. For example, in year 2025-2026, there are a total of 25 TA positions for the entire school year above and beyond what we currently have in 2020-2021. These numbers should be multiplied * the 1 semester TA costs (~16K), NOT a full year TA costs.*

These tables represent the additional resources for both the combined On

Projected Additional Resource Acquisition Plan (by Year) (On Campus + O					
Resource Type	**2021-2022**	**2022-2023**	**2023-2024**	**2024-2025**	**2025-2026**
Tenured Track Faculty	2	2	2	1	1
PoP (On Campus)	1	1	0	0	0
Professor of Practice (Online)	0	0	2	2	0
Adjunct (On Campus)	1	0	0	0	0
Adjunct (Online)	0	1	1	0	0

Projected Additional Salaries by Year (On Campus + Online)					
Resource Type	**2021-2022**	**2022-2023**	**2023-2024**	**2024-2025**	**2025-2026**
Tenured Track Faculty	2	4	6	7	8
PoP (On Campus)	1	2	2	2	2
PoP (Online)	0	0	2	4	4
Adjunct (On Campus)	1	1	1	1	1
Adjunct (Online)	0	1	2	2	2
UG Advisor	1	1	1	1	1
Other Staff	1	2	2	2	2

Other Resources Required (semester hires only) (On Campus + Online)					
Resource Type	**2021 - 2022**	**2022 - 2023**	**2023 - 2024**	**2024 - 2025**	**2025 - 2026**
TAs (On Campus)	4	8	12	20	25
TAs (Online)	0	7	15	22	23
Graders (On Campus)	1	3	4	12	11
Graders (Online)	0	0	0	0	0
Lab Assts (On Campus)	4	20	0	0	0
Lab Assts (Online)	0	0	0	0	0

Campus Program and the Online Program

nline)
Total New Instructors Acquired Over 5 Years
8
2
4
1
2

Note: The table to the left shows the year and number of additional resources that we intend to ACQUIRE in that year. Should be used to account for start-up costs only.

Note: The table to the left shows the total number of additional salaried employees required each year. The numbers in this table are cumulative and extend out over each year.

Note: The table to the left shows the total number of semester hires required each year. For example, in year 2025-2026, there are a total of 25 TA positions for the entire school year above and beyond what we currently have in 2020-2021. These numbers should be multiplied * the 1 semester TA costs (~16K), NOT a full year TA costs.

Operating Expenses

Category	FY Cost	Explanatory Notes
Dept. Travel	$25,500	Dept travel, student travel awards
Events/Conferences	$15,000	Dept events, conferences NACADA,
Office Supplies	$5,000	
Other Operating Expens	$75,000	Lab support, visa support, graduation support, phones, equipme
Recruitment	$15,000	Faculty recruitment
Student group support	$5,000	Seminar and Department Visitor
Total	$140,500	

	FY21	FY22	FY23
SCH Calulation			
UG	261	250	247
UG tax rate	31.96%	31.96%	31.96%
Strategic Investment tax rate	2.66%	3.33%	3.66%
Total	170.64	161.78	159.02
Removed the 25.25% tax charged by COE per Larry			
Head Count			
Enrollment /HC	1082	1036	1029
UG tax rate	31.96%	31.96%	31.96%
Strategic Investment tax rate	2.66%	3.33%	3.66%
Total	707	670	662
Lower Division Differential Tuition			
DT	450	450	450
Fin Aid 14%*	63	63	63
	387	387	387
Admin and Instituitonal Cost Diff/Program Fee 12.78%	49	49	49
	338	338	338
Strategic Investment tax rate	9	11	12
TOTAL (per semester)	329	326	325
Upper Division Differential Tuition			
DT	900	900	900
Fin Aid 14%*	126	126	126
	774	774	774
Admin and Instituitonal Cost Diff/Program Fee 12.78%	99	99	99
	675	675	675
Strategic Investment tax rate 2.66%	18	22	25
TOTAL (per semester)	657	653	650
Data pulled from the RCM projection			
Online SCH Revenue	600		
Retained at the University (36%)	216		
Balance to COE (64%)	384		

FY24	FY25
243	244
31.96%	31.96%
4.00%	4.33%
155.62	155.45
1014	1018
31.96%	31.96%
4.00%	4.33%
649	649
450	450
63	63
387	387
49	49
338	338
14	15
324	323
900	900
126	126
774	774
99	99
675	675
27	29
648	646

Graduate Tuition Modeling Sheet

Instructions: Enter data in cells with this peach fill

Student	Residency	One Semester Tuition	RC Waiver	Tuition Paid	Tuition Less Financial Aid	% SCH in ENGR	RA FTE
A	Resident	5,969	5,858	-	-	100%	none
B	Resident	5,969		5,969	5,133	100%	0.5
C	Resident	5,969	-	5,969	5,133	100%	0.25 or 0.33

	Enrollment						
0.75	75% Allocation of Tuition Paid - Enrollment	RC Waiver Add Back	Allocated Net (Less Fin Aid)	RC Waiver Expense	SCER (12.78%)	SI (2.66%)	Net Revenue to Dept
Student A	-	4,394	4,394	(5,858)	187	39	(1,238)
Student B	3,850.01	-	3,850	-	(492)	(102)	3,256
Student C	3,850.01	-	3,850	-	(492)	(102)	3,256

	SCH					
0.25	25% Allocation of Tuition Paid - SCH in ENGR	RC Waiver Add Back	Allocated Net (Less Fin Aid)	SCER (12.78%)	SI (2.66%)	Net Revenue to Dept
Student A	-	1,465	1,465	(187)	(39)	1,238
Student B	1,283	-	1,283	(164)	(34)	1,085
Student C	1,283	-	1,283	(164)	(34)	1,085

Assessments

12.78%	Support Center Expense Recovery (SCER)
2.66%	Stragetic Investment Tax (SI)

Total Net Revenue		Percentage of Tuition received by Dept
Student A	-	0%
Student B	**4,341**	73%
Student C	**4,341**	73%
Total	**8,682**	

This worksheet contains information required to compute the On Campus program offering

	Year 1	Year 2	Year 3	Year 4	Year 5
Tenured Track Faculty	180000	180000	180000	90000	90000
ERE	55800	55800	55800	27900	27900
		180000	180000	180000	90000
		55800	55800	55800	27900
			180000	180000	180000
			55800	55800	55800
				180000	180000
				55800	55800
					180000
					55800
Total Salary	**180000**	**360000**	**540000**	**630000**	**720000**
Total ERE	**55800**	**111600**	**167400**	**195300**	**223200**
Professor of practice	66000	66000	66000	66000	66000
ERE	20460	20460	20460	20460	20460
		66000	66000	66000	66000
		20460	20460	20460	20460
Total Salary	**66000**	**132000**	**132000**	**132000**	**132000**
Total ERE	**20460**	**40920**	**40920**	**40920**	**40920**
Adjunct (2 courses)	16500	16500	16500	16500	16500
ERE	5115	5115	5115	5115	5115
Total Salary	**16500**	**16500**	**16500**	**16500**	**16500**
Total ERE	**5115**	**5115**	**5115**	**5115**	**5115**
UG Advisor	50000	50000	50000	50000	50000
ERE	15500	15500	15500	15500	15500
Total Salary	**50000**	**50000**	**50000**	**50000**	**50000**
Total ERE	**15500**	**15500**	**15500**	**15500**	**15500**
Staff	50000	50000	50000	50000	50000
ERE	15500	15500	15500	15500	15500
		50000	50000	50000	50000
		15500	15500	15500	15500
Total Salary	**50000**	**100000**	**100000**	**100000**	**100000**
Total ERE	**15500**	**31000**	**31000**	**31000**	**31000**

Total Graders	**1920**	**5760**	**7680**	**23040**	**21120**
Total ERE	**42.24**	**126.72**	**168.96**	**506.88**	**464.64**
Total Lab Assistants	**7680**	**38400**			
Total ERE	**168.96**	**844.8**			
Overall	192100	342660	306180	321540	319620
ERE	112586	205107	260104	288342	316200
Total	551447	1041288	1306566	1573685	1773073

requirements for additional personnel.

	Year 1	Year 2	Year 3	Year 4
Teaching Assistant	38600	77200	115800	193000
ERE	4284.6	8569.2	12853.8	21423
Tuition	23876	47752	71628	119380
Total	**66760.6**	**133521.2**	**200281.8**	**333803**
	4	8	12	20

	Year 2	Year 3	Year 4	Year 5
Num Fac	2	4	6	7
F&A		$ 139,414	$ 278,827	$ 418,241
Research $	$ 400,000	$ 800,000	$ 1,200,000	$ 1,400,000
	$ 260,586	$ 521,173	$ 781,759	$ 912,052
	$ 139,414	$ 278,827	$ 418,241	$ 487,948

$12/hr, 10 hr, 16 wks

Year 5

	241250
	26778.75
	149225
	417253.75
	25

Year 6

	8
$	487,948
$	1,600,000
$	1,042,345
$	557,655

This worksheet contains information required to compute the Online program offering requirements for a

New

	Year 1	Year 2	Year 3	Year 4	Year 5
Tenured Track Faculty	180000	180000	180000	90000	90000
ERE	55800	55800	55800	27900	27900
		180000	180000	180000	90000
		55800	55800	55800	27900
			180000	180000	180000
			55800	55800	55800
				180000	180000
				55800	55800
					180000
					55800
Total Salary	**180000**	**360000**	**540000**	**630000**	**720000**
Total ERE	**55800**	**111600**	**167400**	**195300**	**223200**
Professor of practice	66000	66000	66000	66000	66000
ERE	20460	20460	20460	20460	20460
		66000	66000	66000	66000
		20460	20460	20460	20460
Total Salary	**66000**	**132000**	**132000**	**132000**	**132000**
Total ERE	**20460**	**40920**	**40920**	**40920**	**40920**
Adjunct (2 courses)	16500	16500	16500	16500	16500
ERE	5115	5115	5115	5115	5115
Total Salary	**16500**	**16500**	**16500**	**16500**	**16500**
Total ERE	**5115**	**5115**	**5115**	**5115**	**5115**
UG Advisor	50000	50000	50000	50000	50000
ERE	15500	15500	15500	15500	15500
Total Salary	**50000**	**50000**	**50000**	**50000**	**50000**
Total ERE	**15500**	**15500**	**15500**	**15500**	**15500**
Staff	50000	50000	50000	50000	50000
ERE	15500	15500	15500	15500	15500
		50000	50000	50000	50000
		15500	15500	15500	15500
Total Salary	**50000**	**100000**	**100000**	**100000**	**100000**

Total ERE	**15500**	**31000**	**31000**	**31000**	**31000**	
Total Graders	**1920**	**5760**	**7680**	**23040**	**21120**	$12/hr, 10 hr,
Total ERE	**42.24**	**126.72**	**168.96**	**506.88**	**464.64**	
Total Lab Assistants	**7680**	**38400**				
Total ERE	**168.96**	**844.8**				
Overall	192100	342660	306180	321540	319620	
ERE	112586	205107	260104	288342	316200	
Total	551447	1041288	1306566	1573685	1773073	

additional personnel.

	Year 1	Year 2	Year 3	Year 4	Year 5
Teaching Assis	38600	77200	115800	193000	241250
ERE	4284.6	8569.2	12853.8	21423	26778.75
Tuition	23876	47752	71628	119380	149225
Total	**66760.6**	**133521.2**	**200281.8**	**333803**	**417253.75**
	4	8	12	20	25

	Year 2	Year 3	Year 4	Year 5	Year 6
Num Fac	2	4	6	7	8
F&A		$ 139,414	$ 278,827	$ 418,241	$ 487,948
Research $	$ 400,000	$ 800,000	$ 1,200,000	$ 1,400,000	$ 1,600,000
	$ 260,586	$ 521,173	$ 781,759	$ 912,052	$ 1,042,345
	$ 139,414	$ 278,827	$ 418,241	$ 487,948	$ 557,655

Online additional personnel

	Year 1	Year 2	Year 3	Year 4	Year 5
Professor of practice			132000	264000	264000
ERE			40920	81840	81840
Total Salary	**0**	**0**	**132000**	**264000**	**264000**
Total ERE	**0**	**0**	**40920**	**81840**	**81840**
Adjunct (2 courses)		16500	33000	33000	33000
ERE		5115	10230	10230	10230
Total Salary	**0**	**16500**	**33000**	**33000**	**33000**
Total ERE	**0**	**5115**	**10230**	**10230**	**10230**
Overall Online	0	16500	165000	297000	297000
Online ERE	0	5115	51150	92070	92070
Teaching Assistant		67550	144750	212300	221950
ERE		7498.05	16067.25	23565.3	24636.45
Tuition		41783	89535	131318	137287
Total	**0**	**116831.05**	**250352.25**	**367183.3**	**383873.45**

Letter of Support from CAST (UA South):

From: Wagner, Paul E - (paulewagner) <paulewagner@arizona.edu>
Sent: Wednesday, December 9, 2020 9:44 AM
To: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Cc: Denno, Linda Lee - (ldenno) <ldenno@arizona.edu>; Xu, Li - (lxu) <lxu@arizona.edu>
Subject: Re: Request for Confirmation of Support for New Software Engineering Degree

Sharon,

I apologize. I thought I had responded to this. Li and I spoke about this program and we do not believe there are any issues.

One thing that did come up was if we could have a discussion in the spring on how we may be able to offer some of the courses to our students. We have done that with other programs and there are some classes that would certainly add additional benefit.

Li,

Please feel free to add anything.

Have a great day.

THE UNIVERSITY OF ARIZONA

Paul E Wagner, MS, MBA
Department Head, Applied Technology
Assistant Professor of Practice
College of Applied Science & Technology
THE UNIVERSITY OF ARIZONA

A Building, 125
1140 Colombo Ave | Sierra Vista, AZ 85635
Office: 520-458-8278 | Cell: 513-255-0435
paulewagner@arizona.edu
https://azcast.arizona.edu/

From: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Sent: Wednesday, December 9, 2020 9:40 AM
To: Wagner, Paul E - (paulewagner) <paulewagner@arizona.edu>
Cc: Denno, Linda Lee - (ldenno) <ldenno@arizona.edu>; Xu, Li - (lxu) <lxu@arizona.edu>
Subject: RE: Request for Confirmation of Support for New Software Engineering Degree

Paul and Professor Xu,

Just wanted to follow-up with you to send you the formal proposal that we have circulating thru the College of Engineering for peer review before we submit for formal approval at the end of the month.

Please let us know if we have your support for this new degree.

Again, I do not believe that there should be any impact to your program, courses, or resource load since we have not included any CAST courses in the curriculum. But because of some of the similarities in the programs, we do need to have your support to proceed.

If you have any questions or concerns, I would be happy to set up a Zoom meeting so that we can speak virtually.

Thank you in advance for your support,

Sharon ONeal

From: ONeal, Sharon L - (sharononeal)
Sent: Monday, November 23, 2020 9:36 AM
To: Wagner, Paul E - (paulewagner) <paulewagner@arizona.edu>
Cc: Denno, Linda Lee - (ldenno) <ldenno@arizona.edu>; Xu, Li - (lxu) <lxu@arizona.edu>
Subject: RE: Request for Confirmation of Support for New Software Engineering Degree

Paul and Professor Xu,

I am attaching the preliminary proposal that we submitted at the end of October, as well as our current course map, new course descriptions, and Student Learning Outcomes assessment plan.

We have been working with the Computer Science Dept and the College of Science Dean and they are both supportive of the program. While we do not require that the SW Engineering students take CSC courses, there are 3 that are duplicative of courses that we already have in the CoE. We have a meeting with the CSC Interim Dept Head, Dave Lowenthal, tomorrow to finalize some of the pre-requisite details if students opt to take their offerings of the courses in lieu of the ECE and/or SFWE courses.

We are pulling all the different artifacts together for the formal proposal over the next couple of days and will have that completed by the end of the week.

These files should give you the information you need to evaluate the existing and new courses that are part of the program, and understand the driving factors for the new degree.

I will be happy to send you the formal proposal when it is consolidated into a single document by the end of the week.

Thank you very much for your prompt response.

Sharon

From: Wagner, Paul E - (paulewagner) <paulewagner@arizona.edu>
Sent: Sunday, November 22, 2020 5:40 PM
To: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Cc: Denno, Linda Lee - (ldenno) <ldenno@arizona.edu>; Xu, Li - (lxu) <lxu@arizona.edu>
Subject: Re: Request for Confirmation of Support for New Software Engineering Degree

Sharon,

Thank you for reaching out regarding this new degree plan. Would you mind sending over the proposal with the list of courses that will be included in the proposal. Professor Li Xu is the program director for our Computer Science and Applied Computing Programs. Within Applied Computing is a software development track. I would like for her to verify that there aren't any conflicts with the proposal.

Thank you and I look forward to discussing this further.

Have a great day.

THE UNIVERSITY OF ARIZONA

Paul E Wagner, MS, MBA
Department Head, Applied Technology
Assistant Professor of Practice
College of Applied Science & Technology
THE UNIVERSITY OF ARIZONA

A Building, 125
1140 Colombo Ave | Sierra Vista, AZ 85635
Office: 520-458-8278 | Cell: 513-255-0435
paulewagner@arizona.edu
https://azcast.arizona.edu/

From: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Sent: Sunday, November 22, 2020 4:54 PM
To: Wagner, Paul E - (paulewagner) <paulewagner@arizona.edu>
Cc: Denno, Linda Lee - (ldenno) <ldenno@arizona.edu>
Subject: Request for Confirmation of Support for New Software Engineering Degree

Good Afternoon Professor Wagner,

The College of Engineering's Systems and Industrial Engineering and Electrical and Computer Engineering Departments are proposing a new BS undergraduate degree in Software Engineering. The

BS in Software Engineering synergistically integrates proven engineering techniques and discipline with software development best practices that encompass all aspects of the software development lifecycle (SDLC). The curriculum includes core principles from systems engineering, electrical and computer engineering, and software development. The curriculum is based on a solid foundation of mathematics, including Calculus, Discrete Math, and Physics. The courses include topics related to software requirements analysis, design, code, integration, and verification testing. We are planning to develop 7 new courses as part of the curriculum. We have obtained a very enthusiastic endorsement from Provost Folks, Dean Hahn (CoE) and Interim Dean Cheu (CoS) for this new degree.

The Software Engineering curriculum is designed to prepare students to meet the ever-growing demands within the commercial, industrial, and federal government job sectors.

The table below summarizes the full-time projected enrollments in the Software Engineering degree program extrapolated out over the first 5 years, at which we believe we will achieve a steady enrollment number. These numbers were estimated based on actual enrollments in similar software engineering degree programs as reported by the American Society for Engineering Education (ASEE).

Year 1 (2021 / 2022)	Year 2 (2022 / 2023)	Year 3 (2023 / 2024)	Year 4 (2024 / 2025)	Year 5 (2025 / 2026)
60	120	220	350	350

There are no specific courses from CAST that are part of the required curriculum for this degree at this time. However, there may future interest by some of our prospective students in registering for a few of your related courses as electives, if feasible.

I'm writing to obtain your support for our plan to require these courses in our supporting coursework. Kindly respond with your acknowledgement and support for this new degree program, so that it can be incorporated in the proposal that we are finalizing to submit for ABOR approval in early 2021. If you have any questions, please feel free to reach out either via email or by cell at (520) 822-4040.

The College of Engineering looks forward to our continuing collaborations and mutual student development.

Sharon ONeal
Software Engineering Degree Proposal Coordinator
520-822-4040
Systems and Industrial Engineering Dept

Letter of Support from Physics Dept:

Hi Sharon,

I am supportive of this plan. I am also copying Rebecca Gomez and Kelly Grimm so that they are in the loop.

Best regards,

Elliott Cheu, Ph.D.
Interim Dean, College of Science
Distinguished Professor of Physics
University of Arizona
(520) 621-4092

From: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Sent: Tuesday, December 1, 2020 12:35 PM
To: Cheu, Elliott C - (echeu) <echeu@arizona.edu>
Cc: Lowenthal, David K - (dkl1) <dkl1@arizona.edu>; Sprinkle, Jonathan - (sprinkjm) <sprinkjm@arizona.edu>
Subject: Formal Request - CoS Support for the CoE Software Engineering Degree / Proposal

Good Afternoon Dr. Cheu,

As we have discussed in a few meetings over the past few weeks, the Systems and Industrial Engineering and Electrical and Computer Engineering Departments are proposing a new BS undergraduate degree in Software Engineering. The BS in Software Engineering synergistically integrates proven engineering techniques and discipline with software development best practices that encompass all aspects of the software development lifecycle (SDLC). The curriculum includes core principles from systems engineering, electrical and computer engineering, and software development. The curriculum is based on a solid foundation of mathematics, including Calculus, Discrete Math, and Physics. The courses include topics related to software requirements analysis, design, code, integration, and verification testing. The Software Engineering curriculum is designed to prepare students to meet the ever-growing demands within the commercial, industrial, and federal government job sectors. We are planning to develop 7 new courses as part of the curriculum. As you are aware, over the course of the past several weeks, we have obtained a very enthusiastic endorsement from Provost Folks and Dean Hahn (CoE) for this new degree.

The table below summarizes the full-time projected enrollments in the Software Engineering degree program extrapolated out over the first 5 years, at which we believe we will achieve a steady enrollment number. These numbers were estimated based on actual enrollments in similar software engineering degree programs as reported by the American Society for Engineering Education (ASEE).

Year 1 (2021 / 2022)	Year 2 (2022 / 2023)	Year 3 (2023 / 2024)	Year 4 (2024 / 2025)	Year 5 (2025 / 2026)

60	**120**	**220**	**350**	**350**

Last week, Dr David Lowenthal (CS Acting Dept Head), and Dr Jonathan Sprinkle (ECE), and myself met to discuss the CSC courses that should be used as part of the curriculum for this new degree. We all agreed that the following two courses should be part of the program.

- CSC 110, Introduction to Programming, is an optional course in semester #2 (students will have the option of either taking ECE 175 or CSC 110)
- CSC 345, Analysis of Discrete Structures and Algorithms, is a required course in semester #6

Because CSC 345 requires all students to be a Computer Science major as a prerequisite, a special offering of this course will be made available for SW Engineering students. For the time being, we are referring to the SW Engineering student offering as "CSC 345 Prime" (or CSC 345'). We have included adequate additional resources in our proposal projections to cover the CSC faculty that will be required to teach these courses. We will continue to work with you and Dr Lowenthal to allocate additional faculty resource lines so that you will be able to accommodate the new software engineering students taking the courses mentioned above.

I'm writing to obtain your support for our plan to require these courses in our supporting coursework. Kindly respond with your acknowledgement and support for this new degree program, so that it can be incorporated in the proposal that we are finalizing to submit for ABOR approval in early 2021. If you have any questions, please feel free to reach out either via email or by cell at (520) 822-4040.

The College of Engineering looks forward to our continuing collaborations and mutual student development.

Sharon ONeal
Software Engineering Degree Proposal Coordinator
520-822-4040
Systems and Industrial Engineering Dept

From: Brooks, Catherine F - (cfbrooks) <cfbrooks@arizona.edu>
Sent: Wednesday, December 9, 2020 7:38 PM
To: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Subject: Re: Request for Confirmation of Support for New Software Engineering Degree

I apologize for my delay - we are in full support of this program proposal in the iSchool. If a memo beyond this email is needed certainly let me know. The iSchool is happy to provide courses and additional support when needed, we find a lot of synergies with the College of Engineering and look forward to collaborating on this and other new programs over time.

Sincerely, Catherine Brooks

Catherine F. Brooks, PhD, Director and Associate Professor
School of Information at UA, https://ischool.arizona.edu/
Faculty, Cognitive Science, Social, Cultural, and Critical Theory, and SLAT Graduate Interdisciplinary Programs

---

From: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Sent: Wednesday, December 9, 2020 9:46 AM
To: Brooks, Catherine F - (cfbrooks) <cfbrooks@arizona.edu>
Subject: RE: Request for Confirmation of Support for New Software Engineering Degree

Dr Brooks,

Just wanted to follow-up with you to send you the formal proposal that we have circulating thru the College of Engineering for peer review before we submit for formal approval at the end of the month. Please let us know if we have your support for this new degree, by responding to this email.

Again, I do not believe that there should be any impact to your program, courses, or resource load since we have not included any iSchool courses in the curriculum (although students may opt to choose some of your courses as electives as we previously discussed). But because of some of the similarities in the programs, we do need to have your written support to proceed. If you have any questions or concerns, I would be happy to set up a Zoom meeting so that we can speak virtually.

Thank you in advance for your support,

Sharon ONeal

From: ONeal, Sharon L - (sharononeal)
Sent: Sunday, November 22, 2020 4:29 PM
To: Brooks, Catherine F - (cfbrooks) <cfbrooks@arizona.edu>
Subject: Request for Confirmation of Support for New Software Engineering Degree

Good Afternoon Dr. Brooks,

As we previously discussed, the Systems and Industrial Engineering and Electrical and Computer Engineering Departments are proposing a new BS undergraduate degree in Software Engineering. The BS in Software Engineering synergistically integrates proven engineering techniques and discipline with software development best practices that encompass all aspects of the software development lifecycle (SDLC). The curriculum includes core principles from systems engineering, electrical and computer engineering, and software development. The curriculum is based on a solid foundation of mathematics, including Calculus, Discrete Math, and Physics. The courses include topics related to software requirements analysis, design, code, integration, and verification testing. We are planning to develop 7 new courses as part of the curriculum. We have obtained a very enthusiastic endorsement from Provost Folks, Dean Hahn (CoE) and Interim Dean Cheu (CoS) for this new degree.

The Software Engineering curriculum is designed to prepare students to meet the ever-growing demands within the commercial, industrial, and federal government job sectors.

The table below summarizes the full-time projected enrollments in the Software Engineering degree program extrapolated out over the first 5 years, at which we believe we will achieve a steady enrollment number. These numbers were estimated based on actual enrollments in similar software engineering degree programs as reported by the American Society for Engineering Education (ASEE).

Year 1 (2021 / 2022)	Year 2 (2022 / 2023)	Year 3 (2023 / 2024)	Year 4 (2024 / 2025)	Year 5 (2025 / 2026)
60	120	220	350	350

There are no specific courses from the iSchool that are part of the required curriculum for this degree. However, there may future interest by some of our prospective students in registering for a few of your related courses as electives. (Students will have an option of taking 12 units of electives as part of their degree).

I'm writing to obtain your support for our plan to require these courses in our supporting coursework. Kindly respond with your acknowledgement and support for this new degree program, so that it can be incorporated in the proposal that we are finalizing to submit for ABOR approval in early 2021. If you have any questions, please feel free to reach out either via email or by cell at (520) 822-4040.

The College of Engineering looks forward to our continuing collaborations and mutual student development.

Sharon ONeal
Software Engineering Degree Proposal Coordinator
520-822-4040
Systems and Industrial Engineering Dept

From: ONeal, Sharon L - (sharononeal)
Sent: Monday, November 23, 2020 3:19 PM
To: Doug Ulmer <ulmer@math.arizona.edu>
Cc: Kevin K. Lin <klin@math.arizona.edu>; Tina Deemer <deemer@math.arizona.edu>; Robert Indik <indik@math.arizona.edu>
Subject: RE: Request for Confirmation of Support for New Software Engineering Degree

Thank you for your prompt response and support of this new degree.

We will keep you posted on status when the degree is approved by ABOR and after we have started our recruitment activities.

Sharon Oneal

From: Doug Ulmer <ulmer@math.arizona.edu>
Sent: Monday, November 23, 2020 10:14 AM
To: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Cc: Kevin K. Lin <klin@math.arizona.edu>; Tina Deemer <deemer@math.arizona.edu>; Robert Indik <indik@math.arizona.edu>
Subject: Re: Request for Confirmation of Support for New Software Engineering Degree

Dear Sharon,

The Department of Mathematics supports your new degree proposal and will endeavor to offer enough space in Math 125, 129, and 243 to accommodate students in this major.

Best wishes,

Doug Ulmer

==========
Douglas Ulmer
Professor and Head
Department of Mathematics
University of Arizona
Tucson, AZ 85721
==========

On Nov 22, 2020, at 4:40 PM, ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu> wrote:

Good Afternoon Dr. Ulmer,

The Systems and Industrial Engineering and Electrical and Computer Engineering Departments are proposing a new BS undergraduate degree in Software Engineering. The BS in Software Engineering synergistically integrates proven engineering techniques and discipline with software development best practices that encompass all aspects of the software development lifecycle (SDLC). The curriculum includes core principles from systems engineering, electrical and computer engineering, and software development. The curriculum is based on a solid foundation of mathematics, including Calculus, Discrete Math, and Physics. The courses include topics related to software requirements analysis, design, code, integration, and verification testing. We are planning to develop 7 new courses as part of the curriculum. We have obtained a very enthusiastic endorsement from Provost Folks, Dean Hahn (CoE) and Interim Dean Cheu (CoS) for this new degree.

The Software Engineering curriculum is designed to prepare students to meet the ever-growing demands within the commercial, industrial, and federal government job sectors.
The table below summarizes the full-time projected enrollments in the Software Engineering degree program extrapolated out over the first 5 years, at which we believe we will achieve a steady enrollment number. These numbers were estimated based on actual enrollments in similar software engineering degree programs as reported by the American Society for Engineering Education (ASEE).

Year 1 (2021 / 2022)	Year 2 (2022 / 2023)	Year 3 (2023 / 2024)	Year 4 (2024 / 2025)	Year 5 (2025 / 2026)
60	120	220	350	350

As part of the curriculum, the following course(s) from your Dept will be required for the degree:

MATH 125 Calculus I in the 1st semester
MATH 129 Calculus II in the 2nd semester
MATH 243 Discrete Math in the 4th semester

I'm writing to obtain your support for our plan to require these courses in our supporting coursework. Kindly respond with your acknowledgement and support for this new degree program, so that it can be incorporated in the proposal that we are finalizing to submit for ABOR approval in early 2021. If you have any questions, please feel free to reach out either via email or by cell at (520) 822-4040.

The College of Engineering looks forward to our continuing collaborations and mutual student development.

Sharon ONeal
Software Engineering Degree Proposal Coordinator
520-822-4040
Systems and Industrial Engineering Dept

From: Brown, Susan A - (suebrown) <suebrown@arizona.edu>
Sent: Wednesday, December 9, 2020 9:51 AM
To: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Subject: RE: Request for Confirmation of Support for New Software Engineering Degree

Hi – I'm so sorry for not getting back to you.

I have no problem with this undergraduate degree program. You are correct that it does not have any direct impact on our undergraduate program. I fully support this and believe it will be an interesting option for UAZ undergrads.

Best,

Sue.

From: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Sent: Wednesday, December 9, 2020 9:44 AM
To: Brown, Susan A - (suebrown) <suebrown@arizona.edu>
Subject: RE: Request for Confirmation of Support for New Software Engineering Degree
Importance: High

Susan,

Just wanted to follow-up with you to send you the formal proposal that we have circulating thru the College of Engineering for peer review before we submit for formal approval at the end of the month. Please let us know if we have your support for this new degree.

Again, I do not believe that there should be any impact to your program, courses, or resource load since we have not included any MIS courses in the curriculum. But because of some of the similarities in the programs, we do need to have your support to proceed.

If you have any questions or concerns, I would be happy to set up a Zoom meeting so that we can speak virtually.

Thank you in advance for your support,

Sharon ONeal

From: ONeal, Sharon L - (sharononeal)
Sent: Sunday, November 22, 2020 4:32 PM

To: Brown, Susan A - (suebrown) <suebrown@arizona.edu>
Subject: Request for Confirmation of Support for New Software Engineering Degree

Good Afternoon Dr. Brown,

As I mentioned in prior correspondence, the Systems and Industrial Engineering and Electrical and Computer Engineering Departments are proposing a new BS undergraduate degree in Software Engineering. The BS in Software Engineering synergistically integrates proven engineering techniques and discipline with software development best practices that encompass all aspects of the software development lifecycle (SDLC). The curriculum includes core principles from systems engineering, electrical and computer engineering, and software development. The curriculum is based on a solid foundation of mathematics, including Calculus, Discrete Math, and Physics. The courses include topics related to software requirements analysis, design, code, integration, and verification testing. We are planning to develop 7 new courses as part of the curriculum. We have obtained a very enthusiastic endorsement from Provost Folks, Dean Hahn (CoE) and Interim Dean Cheu (CoS) for this new degree.

The Software Engineering curriculum is designed to prepare students to meet the ever-growing demands within the commercial, industrial, and federal government job sectors.

The table below summarizes the full-time projected enrollments in the Software Engineering degree program extrapolated out over the first 5 years, at which we believe we will achieve a steady enrollment number. These numbers were estimated based on actual enrollments in similar software engineering degree programs as reported by the American Society for Engineering Education (ASEE).

Year 1 (2021 / 2022)	Year 2 (2022 / 2023)	Year 3 (2023 / 2024)	Year 4 (2024 / 2025)	Year 5 (2025 / 2026)
60	120	220	350	350

There are no specific courses from the Management Information Systems (MIS) Department that are part of the required curriculum for this degree at this time. However, there may future interest by some of our prospective students in registering for a few of your related courses as electives, if feasible.

I'm writing to obtain your support for our plan to require these courses in our supporting coursework. Kindly respond with your acknowledgement and support for this new degree program, so that it can be incorporated in the proposal that we are finalizing to submit for ABOR approval in early 2021. If you have any questions, please feel free to reach out either via email or by cell at (520) 822-4040.

The College of Engineering looks forward to our continuing collaborations and mutual student development.

Sharon ONeal
Software Engineering Degree Proposal Coordinator
520-822-4040
Systems and Industrial Engineering Dept

From: Sumit Mazumdar <smtmazumdar28@gmail.com>
Sent: Thursday, December 10, 2020 11:14 AM
To: ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu>
Cc: Mazumdar, Sumitendra - (mazumdar) <mazumdar@arizona.edu>; Cheu, Elliott C - (echeu) <echeu@arizona.edu>; Wolgemuth, Charles - (wolg) <wolg@arizona.edu>
Subject: [EXT]Re: Request for Confirmation of Support for New Software Engineering Degree

External Email

Dear Sharon:

The Department of Physics supports the Consolidated Software Engineering Degree Proposal and will offer enough space in PHYS 141 and 241 to accommodate students in this major.
With regards,

Sumit Mazumdar
Professor and Head
Department of Physics

On Wed, Dec 9, 2020 at 9:50 AM ONeal, Sharon L - (sharononeal) <sharononeal@arizona.edu> wrote:

Dr Mazumdar,

Just wanted to follow-up with you to send you the formal proposal that we have circulating thru the College of Engineering for peer review before we submit for formal approval at the end of the month. Please let us know if we have your support for this new degree by responding to this email. We have already obtained Dr Cheu's letter of support with regard to the Computer Science Dept. But we wanted to make sure that the Physics Dept also has an opportunity to respond with their own affirmation.

The courses that we have included from the Physics Dept are shown in the email below.

If you have any questions or concerns, I would be happy to set up a Zoom meeting so that we can speak virtually.

Thank you in advance for your support,

Sharon ONeal

From: ONeal, Sharon L - (sharononeal)
Sent: Sunday, November 22, 2020 4:36 PM
To: mazumdar@email.arizona.edu
Cc: Cheu, Elliott C - (echeu) <echeu@arizona.edu>
Subject: Request for Confirmation of Support for New Software Engineering Degree

Good Afternoon Dr. Mazumdar,

The Systems and Industrial Engineering and Electrical and Computer Engineering Departments are proposing a new BS undergraduate degree in Software Engineering. The BS in Software Engineering synergistically integrates proven engineering techniques and discipline with software development best practices that encompass all aspects of the software development lifecycle (SDLC). The curriculum includes core principles from systems engineering, electrical and computer engineering, and software development. The curriculum is based on a solid foundation of mathematics, including Calculus, Discrete Math, and Physics. The courses include topics related to software requirements analysis, design, code, integration, and verification testing. We are planning to develop 7 new courses as part of the curriculum. We have obtained a very enthusiastic endorsement from Provost Folks, Dean Hahn (CoE) and Interim Dean Cheu (CoS) for this new degree.

The Software Engineering curriculum is designed to prepare students to meet the ever-growing demands within the commercial, industrial, and federal government job sectors.

The table below summarizes the full-time projected enrollments in the Software Engineering degree program extrapolated out over the first 5 years, at which we believe we will achieve a steady enrollment number. These numbers were estimated based on actual enrollments in similar software engineering degree programs as reported by the American Society for Engineering Education (ASEE).

Year 1 (2021 / 2022)	Year 2 (2022 / 2023)	Year 3 (2023 / 2024)	Year 4 (2024 / 2025)	Year 5 (2025 / 2026)
60	120	220	350	350

As part of the curriculum, the following course(s) from your Dept will be required for the degree:

PHYS 141 Introductory Mechanics in the 3rd semester

PHYS 241 Introductory Electricity and Magnetism in the 4th semester

I'm writing to obtain your support for our plan to require these courses in our supporting coursework. Kindly respond with your acknowledgement and support for this new degree program, so that it can be incorporated in the proposal that we are finalizing to submit for ABOR approval in early 2021. If you have any questions, please feel free to reach out either via email or by cell at (520) 822-4040.

The College of Engineering looks forward to our continuing collaborations and mutual student development.

Sharon ONeal
Software Engineering Degree Proposal Coordinator
520-822-4040
Systems and Industrial Engineering Dept

ARIZONA BOARD OF REGENTS
ASU ◆ NAU ◆ UA

New Academic Program Request

University of Arizona

Name of Proposed Academic Program: Bachelor of Science Software Engineering
Academic Department: 2303 - Electrical and Computer Engineering and 2302 - Systems and Industrial Engineering
Geographic Site: Main Campus (Tucson)
Instructional Modality: In Person
Total Credit Hours: 120 units
Proposed Inception Term: Fall 2021
Brief Program Description: The Bachelor of Science in Software Engineering synergistically integrates proven engineering techniques and discipline with software development best practices that encompass all aspects of the software development lifecycle (SDLC). The curriculum includes core principles from systems engineering, electrical and computer engineering, and software engineering. The curriculum is based on a solid foundation of mathematics, including calculus, physics, and discrete math. The courses include software requirements analysis, design, code, integration, verification testing, and software project management. While there are some similarities between the Software Engineering degree program and UArizona's Computer Science degree program offered in the College of Science, there are also major differences. In Computer Science, students predominately focus on the programming fundamentals and theoretical applications of developing software. Software Engineering students, on the other hand, focus more on solving complex, multi-faceted/multi-disciplined engineering problems and product development. The Software Engineering curriculum prepares students to meet the ever-growing demands of commercial, industrial, and federal government job sectors. Relevant software methodologies, such as Agile development, automated testing using continuous integration, and SW DevOps (DevOps combines software development (Dev) and IT operations (Ops)) to increase software application velocity and service delivery, are also integral to the curriculum. Using these types of agile and adaptive approaches, students will be well suited for the many diverse opportunities in a rapidly growing and ever-evolving career in software engineering. The program has a firm engineering foundation, discovery-based education, and an experiential learning approach. As a part of the curriculum, students work on and complete projects in every semester of the program that emphasizes communication, teamwork, critical thinking, and professionalism. This program's flexibility allows students to design their course of study and select technical electives from a diverse pool of courses in software and computer engineering domains such as web and mobile applications, embedded systems, and other interdisciplinary areas. Software engineering students will acquire considerable software development skills, including: experience with multiple programming languages, data structure constructs, algorithm implementation, databases, operating systems, networking, embedded

systems, cloud computing, configuration management, software assurance, and the use of open-source software libraries/programs. Additionally, students in the Software Engineering degree program develop skills required to work on teams and with other engineering disciplines. These skills include the ability to design and develop creative software solutions that are integral to a variety of large-scale and complex systems. Software systems are typically multi-faceted and often include millions of lines of code with diverse origins or pedigrees developed by teams of software engineers. Graduates of the Bachelor's degree program in Software Engineering learn to use structured and well-defined engineering approaches to develop, evaluate, and maintain software-centric systems.

Learning Outcomes and Assessment Plan:

The proposed Software Engineering degree program has well defined student learning outcomes with a documented and effective process for periodic review and continuous improvement. The learning outcomes include both well-defined activities or problems that are either practical and have narrowly focused terms of scope, or broadly defined activities that are relatively complex, broad in scope, and involve a variety of resources (such as processes, materials, or techniques) used in innovative ways. The assessment plan regularly evaluates the extent to which the Student Learning Outcomes (SLO) are being attained. The results of these evaluations are systematically used as input for the program's continuous improvement actions.

The Software Engineering Curriculum map depicted in Figure 1 shows the required courses mapped to the seven ABET compliant SLOs. For each outcome, the figure indicates when each outcome is *Introduced*, *Practiced* and *Assessed* for the different courses that are part of the curriculum. As reflected in the figure, at least three different courses are mapped to each outcome. Furthermore, there are four or more courses that satisfy the outcome for many SLOs. Specific assessment measures and practices for SFWE courses will be established and evolve as new courses are developed, offered, and assessed.

Curriculum Map - Courses and Activities Mapped to BS Software Engineering

University of Arizona AMS » Sandboxes
Ingrid Novodvorsky Playspace

BS Software Engineering Curriculum Map

Courses and Activities Mapped to BS Software Engineering

Courses and Learning Activities	Outcome						
	ABET Outcome #1 An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics.	ABET Outcome #2 An ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors.	ABET Outcome #3 An ability to communicate effectively with a range of audiences.	ABET Outcome #4 An ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts.	ABET Outcome #5 An ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives.	ABET Outcome #6 An ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions.	ABET Outcome #7 An ability to acquire and apply new knowledge as needed, using appropriate learning strategies.
MATH 129 Calculus II	I						
ENGL 102 Composition II			P				
ECE 175/CSC 110 Computer Programming	IPA	IPA			IPA		
SFWE 201 Sophomore Colloquium				I/A			I
Basic Science Lab	I/P						
PHYS 141 Intro Mechanics	I						
ECE 274A Digital Logic	IPA	IPA			IPA		
ECE 275 Object Oriented Software		P/A					

Figure 1. Bachelor of Science in Software Engineering Curriculum Mapped to Student Learning Outcomes (1 of 2)

SIE 277 Object Oriented Design		P/A	P/A		P/A		
PHYS 241 Intro E & M	I						
MATH 243 Discrete Mathematics	I						
SIE 305 Probability & Statistics	I/A					I/A	
SFWE 301 Software Reqmnts. Analysis & Testing	IPA	IPA	I/P				
ECE 311 Engineering Ethics		I/A	I/A	I/A			
SFWE 304 Software Engineering Preceptor					P/A		
SFWE 302 Software Design Process		IPA	I/P			I/P	
CSC 345 Algorithms	IPA						
SFWE 401 Software Assurance & Security	IPA			I/A			
SFWE 403 Software Project & Process Mgt.		IPA	IPA	I/A	P/A		
ENGR 498A/B Senior Capstone		P/A	P/A		P/A	P/A	P/A
SFWE 402 Software Dev. Ops.					P/A	P/A	
SFWE 404 Software Engineering Preceptor					P/A		
Exit Survey Indirect Measure	A	A	A	A	A	A	A

Legend : I Introduced | P Practiced | A Assessed | I/P Introduced/Practices

Last Modified: 12/07/2020 11:20:50 AM

Figure 1. Bachelor of Science in Software Engineering Curriculum Mapped to Student Learning Outcomes (2 of 2)

Existing SIE and ECE courses have well-established measures and assessment plans in place. Hence, those measures will be adhered to for the SWFE curriculum (refer the detailed Learning Outcomes Map and Assessment Plan section of this proposal).

A rubric will be created for each new SFWE course that identifies Criteria, Measures of Assessment, and an Achievement level rating for the different criteria/categories evaluated.

The Achievement levels include: “Exemplary,” “Satisfactory,” “Developing,” and “Unsatisfactory”. For new courses developed in the curriculum, a plan for identifying the sources of evidence and assessment measures will be developed as the courses are developed.

At the end of every semester, a team comprised of the SWFE instructor, and the SIE and ECE Undergraduate Studies Committees will score the rubric using the measures of assessment identified for the course. A Root Cause and Corrective action plan (RCCA) will be developed for any course that scores “Developing” or below. Assessment results are documented and formally maintained in a controlled location at the end of each semester and will be published on the UArizona Assessment website. The scores will be tracked over time to facilitate continuous improvement and ensure the corrective action plans remain effective from semester to semester, year to year.

In consultation with the Office of Instruction & Assessment, we plan to evaluate the overarching Software Engineering program after the first graduating class has completed the program, and every year thereafter. Using well-established processes for developing program outcomes, we will evaluate:

1. The extent to which the program outcomes meet the needs of our students at the time of graduation.
2. Assess how the degree and program outcomes meet the needs of industry.
3. Evaluate how the outcomes meet our program objectives for alumni shortly after their graduation.

We plan to:

1. Have an Academic Programs Assessment Committee to implement, maintain, and improve assessment tools for continuous improvement and to analyze assessment data and formulate recommendations for improving the program based on these analyses.
2. Review all educational objectives annually with the Software Engineering Advisory Council and the Student / Faculty Advisory Council.
3. Charge the ECE and the SIE Undergraduate Program Committees with ensuring that the curriculum provides students with the experience necessary to meet the Software Engineering Educational Objectives and Program Outcomes.
4. Review all educational objectives annually at the ECE and SIE Faculty retreats held before the Fall semester.

Figure 2 shows the overall flow diagram for the Software Engineering degree program's continuous improvement process, which follows a well-established process used by the Systems and Industrial Engineering Department.

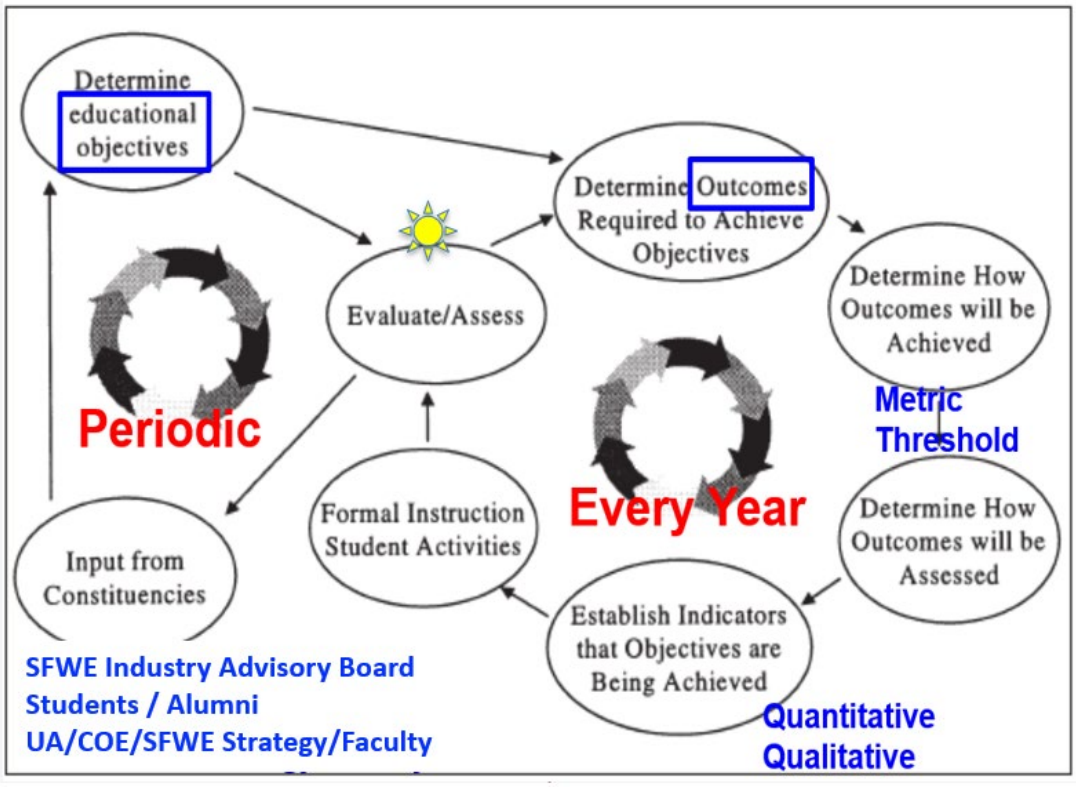

Figure 2. Software Engineering Degree Outcome Assessment and Continuous Improvement Process

Projected Enrollment for the First Three Years:

Table 1 below summarizes the full-time projected on campus enrollments in the Software Engineering degree program extrapolated out over the first 5 years, after which enrollment is anticipated to remain at a steady state. *(Note: An online variant of the program is also planned to be offered beginning in Year 2, however the numbers reflected in Table 1 below only reflect ON-campus enrollments.)*

Year 1	Year 2	Year 3	Year 4	Year 5
60	120	220	350	350

Table 1. Software Engineering Enrollment Projections

Evidence used to determine projected enrollment:

These numbers were estimated based on actual enrollments in similar software engineering degree programs at public universities as reported by the American Society for Engineering Education (ASEE) over the past five years as shown in Table 2 below:

University	Degree Name	2016 Full Time	2017 Full Time	2018 Full Time	2019 Full Time
Arizona State University	Software Engineering (B.S.)	528	512	552	563
Auburn University	Software Engineering (BSWE)	304	374	376	356
Brigham Young University - Idaho	Software Engineering	187	254	302	244
California Polytechnic State University, San Luis Obispo	Software Engineering (B.S.)	245	254	253	253
Iowa State University	Software Engineering (B.S.)	536	348	374	393
Mississippi State University	Software Engineering (B.S.)	129	124	127	145
San Jose State University	Software Engineering (B.S.)	203	500	525	518
The University of Texas at Arlington	Software Engineering (B.S.)	145	151	182	216
The University of Texas at Dallas	Software Engineering (BSSE)	281	315	397	392
University of California, Irvine	Software Engineering (B.S.)	189	176	190	194
University of Nebraska, Lincoln	Software Engineering (BSSE)	38	95	162	181
University of Wisconsin, Platteville	Software Engineering (B.S.)	198	184	217	194

Table 2. Software Engineering Enrollments at Comparative Universities

Evidence of Market Demand:

The Bureau of Labor Statistics reports the total number of Software Engineering jobs in 2010 to be 1.85M, and projects to over 2.4M in 2020; a 23% increase over ten years. U.S News & World Report ranked software developer/engineer as the best technology job in America.

In the past 12 months, Burning Glass reports that there was a total of 43,563 job postings for *Computer Software Engineering* (CIP = 14.0903) from 08/01/19 – 07/31/2020. As shown in Table 3 below, the number of jobs is expected to have a relatively high growth rate over the next 10 years.

GROWTH BY GEOGRAPHY

Geography	Selected Occupations	Total Labor Market	Relative Growth
Nationwide	15.46 %	4.24 %	High

Table 3. National Growth Projected in Software Engineering

The career outlook for software engineers has continued an upward trajectory for the past six years, as depicted in Figure 3 below:

	2014	2015	2016	2017	2018	2019
Employment (BLS)	31,420	34,020	35,160	38,610	38,280	38,792

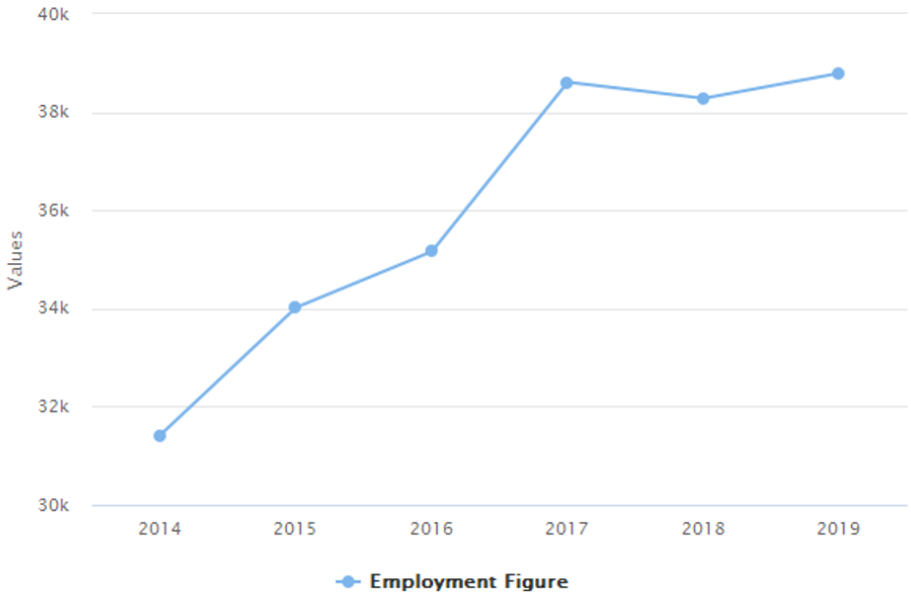

Figure 3. Career Outlook for Software Engineering over the Past 6 Years

Nationwide, the top locations posting requisitions for Software Engineers is shown in the Table 4 below:

Location	Postings
California	309,439
Texas	173,408
Virginia	114,302
New York	98,786
Florida	80,669
North Carolina	80,604
Illinois	68,747
Massachusetts	63,683
Georgia	62,505
Colorado	59,274

Table 4. Software Engineering Posting by State

Within the state of Arizona, the top employers hiring software engineers include the following (see Table 5):

TOP EMPLOYERS HIRING

Experience Level: All Experience

Employer	Postings	Market Share (%)
Raytheon	931	4.00%
USAA	765	3.29%
Wells Fargo	564	2.42%
General Dynamics	310	1.33%
Honeywell	304	1.31%
Northrop Grumman	299	1.28%
American Express	297	1.28%
Deloitte	288	1.24%
The Boeing Company	176	0.76%
Mitsubishi	169	0.73%
Anthem Blue Cross	154	0.66%
CVS Health	145	0.62%
IBM	132	0.57%
Viasat	126	0.54%
Amazon	120	0.52%

Table 5. Arizona Top Employers with Software Engineering Needs

The top 15 specialized software skills required by employers include a variety of programming languages, software engineering, Linux operating system, software project management, software DevOps, Scrum (a particular form of Agile), quality assurance, and systems engineering. The number of job postings and the anticipated growth in specific Software Engineering skill categories is summarized in Table 6, as shown below:

Skill	*Postings*	*Projected Growth*	*Salary Premium*	*Competitive Advantage*
Java	14112 (32%)	-13.18%	No	No
Software Development	11197 (26%)	5.78%	No	No
SQL	10788 (25%)	-13.3%	No	No
Software Engineering	7641 (17%)	7.27%	No	No
JavaScript	7611 (17%)	6.81%	No	No
Python	6560 (15%)	61.12%	No	No
Linux	5650 (13%)	-12.57%	No	No
Microsoft C#	5130 (12%)	-25.69%	No	No
DevOps	5048 (12%)	107.85%	Yes	No
Project Management	4706 (11%)	-19.74%	No	No
Oracle	4620 (11%)	-16.26%	No	No
Quality Assurance	4600 (11%)	39.46%	No	No
Git	4239 (10%)	59.81%	No	No
Scrum	4156 (10%)	39.96%	No	No
Systems Engineering	3740 (9%)	-4.85%	No	No

Table 6. Postings for the Top 15 Software Engineering Specialized Skills

The top 15 software engineering skill clusters are shown in Table 7 below:

Skill	Postings
Software Development Principles	19331 (44%)
Java	14291 (33%)
System Design and Implementation	13878 (32%)
SQL Databases and Programming	12479 (29%)
JavaScript and jQuery	10579 (24%)
Software Quality Assurance	8917 (20%)
Operating Systems	8518 (19%)
Scripting Languages	7644 (17%)
Web Development	7380 (17%)
Software Development Methodologies	7316 (17%)
Business Process and Analysis	7146 (16%)
Database Administration	7136 (16%)
Microsoft Development Tools	7056 (16%)
Cybersecurity	6963 (16%)
Programming Principles	6954 (16%)

Table 7. Postings for the Top 15 Software Engineering Skill Clusters

Similar Programs Offered at Arizona Public Universities:

There are 37 ABET accredited bachelor's level programs in Software Engineering, with 30 of these located in the United States. Arizona State University (ASU) has both an ABET accredited On-campus and an Online Software Engineering Bachelor's degree program in Arizona. In 2019, ASU had more than 1300 students enrolled in their program, of which ASEE classified 563 as full-time degree-seeking students. In 2019, ASU conferred 143 BS Software Engineering degrees.

In 2019, Embry-Riddle Aeronautical University in Prescott, AZ had 63 full-time students in their Software Engineering program and conferred six students with Software Engineering degrees.

The University of Arizona (UArizona) also offers a BS in Computer Science degree located in the College of Science. In 2019, there were 1626 students enrolled in the Computer Science program. While there are some similarities in the curriculum between a Computer Science program and a Software Engineering degree program, there are also significant differences. Software engineers are trained in all aspects of the software development lifecycle (SDLC), including requirements specification and analysis, software architecture / design, test planning and integration, verification / validation, and maintenance/support. The curriculum also includes key attributes of the overall engineering discipline that includes mathematics through calculus, physics, chemistry, product development, configuration management, quality assurance, safety, reliability, as well as the cost, schedule, and delivery pipeline of software. Software engineers develop software that is often part of a large, complex, and multi-disciplinary engineering product / system.

Other synergistic degree programs at UArizona include those offered by the School of Information (iSchool), Eller College of Management, and the College of Applied Science & Technology (CAST). The iSchool offers two related BS degrees in Information Science and Technology and Game Design and Development. The Eller College of Management offers a BS in Management Information Systems. CAST offers BS degrees in Applied Computing, Intelligence and Information Operations, and Computer Science. While some classes within these programs can potentially be electives within the Software Engineering degree program, there are significant differences between the curriculum and skillsets developed in all these programs.

New Resources Required? (i.e. faculty and administrative positions; infrastructure, etc.):

a. ***Library Acquisitions –***

There are no anticipated additional library acquisition needs with the Software Engineering degree program.

b. ***Equipment/Physical Facilities –***

At this time, there are no additional equipment / physical facility needs identified. Many of the tools and lab facilities used by this degree program already exist for other courses offered within the ECE and SIE departments. Many software development tools traditionally used for software engineering courses are “open source” and readily available for faculty and students to use at little to no charge. Cloud based resources, such as those offered through Amazon Web Services (AWS) or Microsoft Azure, can host integrated development environments for courses that require more comprehensive integrations of multiple tool types of toolsets to support student projects. If there are licenses or cloud-based resource usage fees that become necessary for different software engineering courses in the future, those costs could be included in course fees.

c. ***Additional Faculty/Staff***

The additional required faculty for the Software Engineering degree program have been broken down into multiple categories, as is depicted in Table 8 and 9. Table 8 shows the academic year that the additional faculty will be recruited / hired over 5 years.

Projected Additional Resource Acquisition Plan (by Year) (On Campus Only)						
Resource Type	2021-2022	2022-2023	2023-2024	2024-2025	2025-2026	Total New Instructors Acquired Over 5 Years
Tenured Track Faculty	2	2	2	1	1	8
Professor of Practice	1	1	0	0	0	2
Adjunct	1	0	0	0	0	1

Table 8. Projected Additional Faculty Resources Acquisition Plan

Table 9 shows the accrued total of new faculty and other staff over 5 years. It is envisioned that the majority of the curriculum for the first year will be closely aligned to many of the other engineering degree programs. Hence for the first year of the degree offering, new faculty can focus on developing the curriculum for the 2nd and 3rd years of the program. However, because a few of the curriculum courses are already very heavily populated courses, such as ECE 175 and Engr 102, additional faculty, TA and lab assistant resources will be required to help mitigate the increased demand.

Projected Additional Faculty and Staff Accrued over 5 Years					
Resource Type	2021-2022	2022-2023	2023-2024	2024-2025	2025-2026
Tenured Track Faculty	2	4	6	7	8
Professor of Practice	1	2	2	2	2
Adjunct	1	1	1	1	1
UG Advisor	1	1	1	1	1
Other Staff	1	2	2	2	2

Table 9. Projected Additional Faculty and Staff Resources Accrued Over 5 Years

The required number of additional teaching assistants (TAs), graders, and lab assistants (LA) for the Software Engineering degree program have been broken down as shown in Table 10. Because these resources are hired on a semester-by-semester basis only, the numbers shown in Table 9 reflect the total combined number of additional TA positions required over both semesters for the academic year *(i.e. Each table entry = Fall TA positions + Spring TA positions combined).*

Other Resources Required (semester hires only) (On Campus Only)					
Resource Type	2021 - 2022	2022 - 2023	2023 - 2024	2024 - 2025	2025 - 2026
TAs	4	8	12	20	25
Graders	1	3	4	12	11
Lab Assts	4	20	0	0	0

Table 10. Projected Additional Teaching Assistant, Graders, and Lab Assistant Positions for Each Academic Year *(numbers reflect the total for both semesters combined)*

Program Fee/Differentiated Tuition Required? YES ☐ NO ■
Estimated Amount: N/A

Program Fee Justification: N/A

Specialized Accreditation? YES ■ NO ☐

ARIZONA BOARD OF REGENTS
ASU ◆ NAU ◆ UA

Accreditor: Accreditation Board for Engineering and Technology, ABET